Learner Perceptions of Successful Engagement in Undergraduate Computer Science Education: A Grounded Theory

Meriel Olwen Huggard

School of Computer Science and Statistics

University of Dublin, Trinity College

A thesis submitted for the degree of

Doctor of Philosophy

May 2013

Declaration

I declare that this thesis has not been submitted as an exercise for a degree at this or any other university and it is entirely my own work.

I agree to deposit this thesis in the University's open access institutional repository or allow the library to do so on my behalf, subject to Irish Copyright Legislation and Trinity College Library conditions of use and acknowledgement.

Meriel Huggard.

Meriel Olwen Huggard
21 May 2013

Abstract

Computer science degree programs have proliferated due to the demands of industry and the pressures placed on higher education institutions by national governments. In tandem with this, the technological familiarity presumed in most undergraduate degree programs has also increased significantly, particularly in the Engineering and Science based disciplines. The computer science skills that these students are assumed to have, or are expected to acquire and develop, span a very broad spectrum of computer-based engagement. Many studies have highlighted the significant percentages of students who start degree programs in these areas and then fail to progress beyond the first year. Universities have been required to engage with, and address, the fundamental challenges facing these students and to support their successful progression throughout their undergraduate careers.

In light of these concerns, much work has focussed on identifying factors which influence student success on undergraduate degree programs in computer science and computer engineering. These factors range from quantifiable measures of mathematical understanding and past programming experience, through to concepts from psychology such as self-efficacy and computer anxiety. One concept that often appears in such studies is that of computer experience. This concept arose from quantitative studies in the humanities and social sciences, and was not designed to encompass the more complex computing abilities and skills expected of students on computer science programs. This thesis was motivated by this observation and its primary objective was to develop a theory that captures undergraduate computer

science students' perspectives of the computer related experiences that contribute to their perception of success.

The qualitative grounded theory methodology was chosen for this work based on its suitability for inductive theory generation. The outcome of this study was a theory of successful engagement that is grounded in the worldview of first year students of computer science in an Irish university. This theory encompasses four key abstracted categories: experience accumulation, empowering experimentation, perception of competence and language. This theory provides a valuable insight into the students' perspective of the situations, contexts and settings that influence their perceived success.

I dedicate this thesis to my late parents,
Olwen Annie and Henry Stephen Huggard.

Acknowledgements

There are many people who have guided and supported me along the rather scenic route I have taken to the completion of this work.

I am deeply indebted to my initial supervisor, the late Prof. John T. Lewis of the Dublin Institute of Advanced Studies, who was one of life's true gentlemen and scholars. The research environment he created for the Dublin Applied Probability Group (DAPG) was permeated by his boundless enthusiasm and energy. It is a very deep privilege and an honour to say that I was one of his research students and I can only hope that he would forgive me for taking so long to complete this work.

I greatly respect and admire the professionalism and patience of my current supervisor, Dr. Ciarán Mc Goldrick. He has known when to cajole me to continue and when to give me space to deal with my other commitments in life. I could not have a better supervisor, colleague and friend.

I would like to thank the School of Computer Science and Statistics in Trinity College Dublin, and in particular its former head, Dr. David Abrahamson, for granting me special leave to enable me to complete this work.

Contents

List of Figures

Chapter 1

Introduction

The past decade has seen a rapid growth in the level and nature of the technological competencies required from graduates across a wide range of scientific and engineering degree programs. Many such undergraduate programs incorporate core modules in computer programming whilst also requiring students to engage in significant computer based simulation and data analysis. It is essential that students on these courses acquire the necessary technological skills as quickly as possible. However, many struggle with the development of these skills and this has a negative impact on their performance on the modules in question and potentially throughout their undergraduate career.

There has been much discussion, both within Ireland [159] and internationally [5; 13; 100; 198], about the high levels of attrition observed on courses with a strong technological focus. The rate of attrition has been reported as being as high as 30 to 40% [13] and sometimes even exceeding 50% [233]. In a 2010 report on retention and progression, the Irish Higher Education Authority claim that the "non-presence" rate for students of computer science is 16% on university degree courses, and this jumps to 36% for courses offered in the Institute of Technology sector. These concerns have led to much attention being focused on the quantitative identification of factors which influence students success on technology based courses such as Computer Science and Computer Engineering.

As the author of this thesis observed in a previous work [112] "... there is very little continuity between subjects studied during the secondary education senior cycle and those encountered during the first year in university. Courses in areas such as computer programming and computer networking require a style of logical thinking that many students have not experienced prior to university entry. Moreover, many students are experiencing university life for the first time and haven't developed an appropriate sense of their own responsibility for, or ownership of, their learning".

In Ireland almost all students entering university direct from second-level have taken the School Leaving Certificate Examination conducted by the Irish State Examinations commission on behalf of the Department of Education and Skills. School leavers have usually been taught basic computer skills involving word and data processing, but it is most likely that they have not been introduced to computer programming in a formal school setting. Less than 1.2%, have taken examinations in Technology, the only Leaving Certificate subject with a syllabus that includes elements of computer science. This means that their first encounter with these topics is when they meet them as undergraduates; hence it is hardly surprising that the "non-presence rate" of 16% for students in computer science compares so unfavourably with the average "non-presence" rate of 9% [157].

Within Ireland exploration of the factors influencing student success on technology based degree programs have concentrated on the challenges faced by those in learning how to program. Computer programming grades are characterised by a bimodal distribution with unusually heavy tails i.e. with higher than usual rates of both failing and excelling grades [176]. Most investigations into the factors influencing those attaining high grades on programming courses are quantitative in nature e.g. [15; 16; 65; 114], where participants have filled in questionnaires or engaged in programming tasks which are then graded. More recently there have been two in-depth qualitative studies focused on learning computer programming: Stamouli and Huggard [193; 194] used phenomenography [146] to explore learning to program from the students' perspective, while Dunican [74] sought to develop a grounded theory [93] of the learning experiences of novice programmers.

This study is much broader in nature as it is not limited to student performance in introductory programming modules. Rather, its starting point was to explore the experiences that undergraduate computer science students believe lead them to be successful in their studies.

1.1 Aims and Objectives of this Research

The context of this study arose from the concept of computer experience, as articulated by Smith et al.[187] and Prince [168]. As they define it, the concept captures the computer experiences they have identified as being of most benefit to undergraduate students. However, their work focused on disciplines in the humanities and social sciences, so they did not consider the more complex requirements of students on courses that involve significant technological content. This study had its genesis in this observation and the primary objective of this work was to develop a theory that captures the undergraduate computer science students' perspective of the computer related experiences that contribute to their success.

The qualitative grounded theory methodology was used to develop a substantive theory of the experiences that influence success. However, as a grounded theory it makes no claims to be a formal theory with a range of application beyond the context of the study. As befits the grounded theory methodology, this research question was deliberately open and non-prescriptive, allowing the researcher scope to explore any experiences that the students felt were relevant, important or influential.

The research questions that motivated this work were:

- What experiences do students feel lead them, or others, to be successful in their early undergraduate studies?
- How can these experiences be categorised and used in the generation of a substantive theory?
- How can this theory be developed and presented in a way that will enhance understanding of the factors that students perceive influence success at undergraduate level?

1.2 Key Contributions

The key contribution of this thesis is the realisation of a "theory of successful engagement" that is grounded in the worldview of first year students of computer science in an Irish university. Moreover, this study is a self-contained piece of work within the grounded theory tradition. The emergent theory and its associated categories provide a deeper understanding of the interactions between the factors that students perceive influence success at the undergraduate level. The research contributes to the teaching and learning in technology based degree programs, as it employs an established research methodology to derive a theory of successful engagement and therefore other researchers can look to generalise it to their educational settings.

1.3 The Scope of this Work

When developing a grounded theory, the emphasis is on determining the validity and reliability of the findings in relation to the data collected and not on the extension or generalisation of the substantive theory to a wider setting. While it is important to set the findings in context, this context is of most significance to others who wish to draw on the emergent theory in future work.

The data collected for this work was gathered in 2006 and 2007 and thus the resultant theory must be viewed in the context of the computing technologies and applications available at that time.

Most participants in this study engaged with social media services such as Facebook (www.facebook.com) and bebo (www.bebo.com), however these were in their infancy in 2006. In particular, Facebook had not achieved the pervasive, dominant position in the market that it enjoys today. While participants in this study were extremely active on social networking sites, they had not grown up with these and they were aware of their novelty and emergent nature.

In 2006 the deployment of third generation mobile phone networks had just spawned a new generation of smart phones with enhanced data capabilities, for example the Apple iPhone was launched in 2007. Some of the participants were early adopters of these devices, but as before they had not formed part of their early childhood experience of computers and so had not been a part of their pre-university interactions with computers.

The study was conducted before the advent of tablet computers such as the Samsung Galaxy Tab and the Apple iPad. These touch screen devices provide a highly accessible and interactive media platform that facilitates easy access to technology and learning for very young children [48]. These devices are rapidly revolutionising the classroom and will play a key role in shaping the computing experiences of the next generation of undergraduate students.

All of these technological innovations mean that the contribution of this work needs to be viewed through the lens of the era in which the study was conducted. This does not mean that the emergent theory is not relevant today, rather it means that if the study were replicated today not all of the abstracted categories found may emerge from the data collected, while other, new categories that capture the wider range of technological experiences available to today's students may emerge in their place.

1.4 Overview of this Thesis

This chapter sets out the motivation and scope of this thesis in broad terms. It articulates the research questions, objectives and key contributions of the study.

Chapter 2 considers the field of computer science education wherein the contribution of this work lies. In particular the chapter focuses on the nature of work within this field, providing a context and motivation for the study presented in the subsequent chapters. Relevant qualitative and quantitative work on factors influencing student success is reviewed, with particular attention paid to qualitative projects as these are directly relevant to the methodology adopted in this study. This chapter also provides the reader with an appreciation and awareness of the settings in which the outcomes of this work are to be considered.

A detailed exposition of the grounded theory methodology, which lies at the heart of this work, follows in Chapter 3. The chapter begins with an exploration of what constitutes a theory and then considers the philosophical paradigms that shape the well known variants of the methodology. The principles and practices that ensure the trustworthiness of such qualitative work are then discussed.

Chapter 4 considers the design of the study and the collection of data. The process of data collection, including the gathering of initial data as well as subsequent data collection using theoretical sampling, are discussed. The logistical constraints within which the process of data collection had to take place, and the practices adopted to ensure the trustworthiness of the data, are then presented. The chapter concludes with a discussion on field notes and memoing.

Chapter 5 explores the processes used in the analysis of the gathered data. The techniques used to obtain the initial coding and articulation of concepts from the raw data are described. The emergence of a theory through the development of categories is then considered.

In Chapter 6 the analysis of the collected data is expounded upon. The coding processes applied to the data are illustrated, and the abstracted categories obtained presented. The emergent "theory of successful engagement" is then articulated and detailed in the context of the abstracted categories. A significant articulation of the principles and practices that ensure the trustworthiness of the theory obtained is then provided in Chapter 7.

Finally, the thesis concludes with a review of the findings of this study in the context of the field, a discussion of the contributions of this work, and suggestions for possible directions for future study.

1.5 Publication Record

The author of this thesis has been active in the field of Computer Science and Mathematics Education for over a decade. She has supervised one Ph.D. thesis in the field [193] and a summary of publications that are related to, and inform, this work is given below.

- Meriel Huggard, Ciarán Mc Goldrick, "Lessons Learnt from a Decade of Structured Support for Novice Programmers", International Conference on Engineering Education, Belfast, 22nd - 26th August 2011.

- Meriel Huggard and Ciarán Mc Goldrick, "Formalising Research Methods for Graduate Programs in Computer Science", Proceedings 37th Annual IEEE Frontiers in Education Conference, Milwaukee, Wisconsin, USA, 10th - 13th October 2007, ppS4G-16 - S4G-20.

- Ioanna Stamouli, Meriel Huggard, "Phenomenography as a tool for understanding our students", International Symposium for Engineering Education, Dublin City University, 17th - 19th September 2007, pp181 - 186.

- Meriel Huggard, "Computer Experience and the Computer Science Student", Poster Presentation, ACM SIGCSE 2006, Houston, Texas, 1st - 5th March 2006.
- Ioanna Stamouli and Meriel Huggard, "Learning Object Oriented Programming from the Student's Perspective", EARLI Jure 2006, Tartu, Estonia, 30th June - 4th July 2006.
- Meriel Huggard and Ciarán Mc Goldrick, "Incentivising students to pursue Computer Science Programmes", Proceedings 36th Annual IEEE Frontiers in Education Conference, San Diego, California, 28th - 31st October 2006, ppS4C-3 - S4C-8.
- Meriel Huggard and Ciarán Mc Goldrick, "Computer Experience - Enhancing Engineering Education", International Conference on Engineering Education, Puerto Rico, USA, 23 - 28 July 2006, ppT4C-21 - T4C-25.
- Ioanna Stamouli and Meriel Huggard, "Object Oriented Programming and Program Correctness: The Students Perspective", Proceedings ACM ICER, Kent, September 2006.
- Eileen Doyle, Ioanna Stamouli and Meriel Huggard, "Computer anxiety, Self-efficacy, Computer Experience: An Investigation Throughout a Computer Science Degree", Proceedings 35th Annual IEEE Frontiers in Education Conference, Indianapolis, Indiana, 19th - 22nd October 2005, ppS2H-3 - S2H-7.
- Meriel Huggard, "Programming Trauma: Can it be avoided?", British Computer Society Grand Challenges in Computing:Education, Newcastle, England, March, 2004, pp50 - 51.
- Ioanna Stamouli, Eileen Doyle and Meriel Huggard, "Establishing Structured Support for Programming Students", Proceedings 34th Annual IEEE Frontiers in Education Conference, Savannah, Georgia, 20th - 23rd October 2004, ppF2G-5 - F2G-9.

Chapter 2

Computer Science Education and Computer Experience

This chapter explores the research domain within which this work is situated. In particular it considers the field of computer science education and details how it seeks to bridge the gap between the academic in the classroom and the more formal field of educational research. Within the computer science education domain there is a rich literature on the challenges faced by freshman students and the factors influencing undergraduate student success. This chapter reviews this literature, placing the spotlight on the concept of computer experience as it is the focus of the research detailed in this thesis.

The field of computer science education is extremely broad and captures a wide range of approaches, from individuals reporting on their experience in the classroom to those exploring specific approaches or techniques with a view to their generalisation to a wider setting. The research methodologies employed differ significantly from those of computer science itself. One cannot study student behaviour using the concept of formal proof or through the creation of mathematical algorithms; rather the methodologies adopted are often more qualitative in nature and drawn from the fields of education, sociology, psychology and anthropology. Regardless of the research methodology adopted, the main goal of any study in the field of computer science education is to improve the quality of teaching and student learning and to contribute to the development of computer science as a whole.

2.1 Computer Science Education

For over forty years computer science education research has been has been the subject of much discussion and interest: in 2013 the ACM Technical Symposium on Computer Science Education (SIGCSE) will take place for the 44th time and the IEEE CS-sponsored Frontiers in Education (FIE) conference will enter its 43rd year, while the more targeted ACM Conference on International Computing Education Research (ICER) enters its sixth year. Fincher and Petre [78] state that computer science education is "an emergent area and is still giving rise to a literature" [78, p.1]. Fincher argues that it is an emergent field because the "status of participants is not always obvious" as "there is hardly anyone for whom this is their only research activity, or even their primary one" [50, p.337]. This is not to say that it is not a valid research domain, rather that it draws on methodologies developed in other fields. As Guzdial puts it: "The real challenge in computer education is to avoid the temptation to re-invent the wheel" [2, p.191]. He cautions against ignoring "the hundred of years of education, cognitive science and learning sciences research that have gone before us" [2, p.191]. Thus when considering the field we need to consider not only the range of subjects studied but also the theoretical foundations and methodologies upon which they are based.

For computer science education to emerge fully as a clearly identified, acknowledged field of research, it must be capable of articulating what its characteristics are and where it fits in the wider research environment. To this end much work has focused on examining the literature in the field with a view to identifying the sub-fields contained within computer science education research and the nature of the research methods employed. In particular the community has begun to reflect upon what qualifies as rigorous research within the field and how good research practice may be identified [50; 78; 144]. Much effort has focused on the nature of the existing body of research and on the development of classification schemes that can help improve its credibility as a research domain. An overview of these classifications is now presented.

2.1.1 Classifying studies in Computer Science Education

One of the first classifications of Computer Science Education research appears in [50], where Fincher suggests four broad subject headings under which individual studies may be grouped:

- Small-scale investigations of a single aspect of the discipline or practice,
- Investigations of specific mental and conceptual skills,
- Investigations based within the educational tradition,
- Investigations motivated by the use of tools in CS teaching and learning. [50, p.338-339]

While this classification goes some way towards capturing the differences between studies, the suggested headings are not distinct and mutually exclusive; rather they combine elements of the subject area being considered, research methodologies employed and expected research outcomes. This may make it difficult for the researcher to determine how best to classify any study that they are undertaking using these groupings. This has led to the development of classifications that focus on more narrow aspects of the field. For example, in a later work Fincher and Petre focused on capturing the range of interests that motivate people to carry out research in the field [78]:

- Student Understanding,
- Animation, visualisation and simulation,
- Teaching methods,
- Assessment,
- Educational Technology,
- Transferring professional practice to the classroom,
- Incorporating new developments and technologies,
- Transferring from face-to-face to distance education,

- Recruitment and retention (including diversity and gender),
- Construction of the discipline.

Fincher and Petre do not restrict themselves to these headings, but also comment on the nature and scope of each study that they consider. However, they do not provide a detailed classification scheme that would assist those seeking to articulate how their work fits within the field as a whole.

A more complete classification scheme has been articulated by Malmi et al. [144]. It captures seven different dimensions of research within the field based on a preliminary analysis of the literature:

- Prior work upon which the research is based i.e. theories, models, frameworks and instruments (e.g. Bloom's taxonomy [27], Bigg's SOLO taxonomy [23], Bandura's self-efficacy [11], established questionnaires),
- Technologies or tools used, not including those developed as part of the study (e.g. specific program development environments),
- Reference Discipline (e.g. Computer Science Education, Mathematics Education, Sociology),
- Research Purpose, classed as descriptive, evaluative or formulative,
- Research Framework i.e. the orientation or approach that guides the research,
- Data source, including both the nature of the data and how it is gathered,
- Analysis method i.e. qualitative, quantitative, or mixed methods (e.g. argumentation, statistical analysis, interpretative qualitative analysis).

In the following paragraphs Malmi et al.'s framework is used to describe how the work described in subsequent chapters fits into the broader field of computer science education.

The study detailed in this thesis arose from a consideration of the bi-dimensional view of the construct of computer experience articulated over a decade ago by Smith et al.[187]. While much work has focused on finding suitable methodologies to assess what Smith et al. termed "objective computer experience", little has been done to develop a measure of a construct they called "subjective computer experience" [189]. Their work focused on psychology students and it was hypothesised that these measures did not easily translate to settings involving students taking courses containing a significant technological component e.g. Computer Science, Engineering and Science. The concept of computer experience will be discussed in greater detail in section 2.5 below.

There were no technologies or tools whose use was significant for the purpose of this work. While a computer was used during the analysis of the data collected, this was not expected to influence the outcomes of the study. Thus the use of a computer for this purpose was not explicitly relevant to the research and did not fall into the definition of a technology or tool provided by Malmi et al. [144].

This work draws on a number of reference disciplines including Computer Science Education Research, for predictors of student success in Computer Science; Psychology, for Smith et al.'s definition of Subjective Computer Experience [189]; Sociology, from which the grounded theory methodology originated [93] and Education, for an understanding of existing theories of learning. To a lesser extent it draws on traditions as diverse as nursing and information systems, where grounded theory is widely used as a research framework.

The purpose of the research was formulative as it was concerned with the development of a theory of the concept of computer experience. The research framework adopted was the grounded theory methodology. The source of the qualitative data used in the generation of this theory was mainly research specific, as it was gathered from volunteer participants through the use of semi-structured interviews. The data collection process was guided by the grounded theory principles of theoretical sampling and theoretical saturation, which will be described in detail in Chapter 3 below. This data was subjected to interpretive qualitative analysis in order to generate a substantive, grounded theory of the construct of computer experience.

2.2 Philosophical Paradigms in Education Research

When beginning to look at research, one must start at the philosophical level with the paradigms that guide the study. These paradigms capture the world-view that informs the methodology and methods adopted [205] and they are valid across all research domains, irrespective of subject area. It is common to divide these into epistemological groupings e.g. positivist, anti-positivist, constructivist, post-modern etc. Some make a broader division of these into just two groups: positivism and "something which denies positivism" [160]. The later captures the subtlely different notions of anti-positivism [54], non-positivism [18] and post-positivism [81], while also encompassing the constructivist and critical methodologies [63]. Broadly speaking, the natural sciences tend to be positivist in nature, while the social sciences and humanities are more likely to use paradigms that deny positivism. However, these boundaries are not absolute and many research studies incorporate elements of both approaches [223].

Positivists argue that "reality is "out there", waiting to be discovered and that this reality is reflected in universal laws that may be discovered by the application of objective, replicable and "scientific" research methods " [83]. Their knowledge is verified by hypotheses that lead to the acceptance of this knowledge as established facts or laws. Methodologies employed are usually quantitative and technical in nature and give rise to substantive theories. While post-positivists approach research with the same focus on providing explanations based on establishing facts and laws, they view knowledge as being a set of probable facts or laws that arise from non falsified hypotheses [138]. Like positivists their focus is also on the development of substantive theory, however they allow for the use of qualitative methods alongside the more quantitative, technical ones adopted by positivists [138]. In contrast the constructivist's world view is subjective in nature and they consider reality as socially-constructed [138]. Each individual's experiences and their interpretations of these experiences are what give meaning to phenomena; with the individual's shared reality leading to a consensus on what constitutes knowledge. The "post-modern turn" [20] places an emphasis on the "social construction of social reality, fluid as opposed to fixed identities of the self, and the partiality of all truths" [138, p.178]. From the post-modern perspective the researcher is an "acknowledged participant" in the production of an always incomplete knowledge [137].

These approaches can seem very different, however they often reinforce each other in education research where qualitative and quantitative data is often readily available. Such complementary data streams can prove useful as they help to increase the trustworthiness of the results obtained through the process of triangulation (this will be discussed further in section 3.6).

Being able to identify the education tradition within which a work is situated allows for the development of research communities that share a common language and terminology. This greatly assists in the dissemination and validation of results as it provides researchers with a shared framework for the critical evaluation and judgement of each others work. It also enables them to determine the applicability and generalisability of any work that they encounter.

The researcher's philosophical paradigm will determine the nature of the research questions to be addressed and hence it will influence the nature of the research framework adopted [18]. The distinction between quantitative and qualitative research methods is important; as Gall et al. put it "... qualitative research is best used to discover themes and relationships at the case level, while quantitative research is best used to validate those themes and relationships in samples and populations. In this view qualitative research plays a discovery role, while quantitative research plays a confirmatory role" [81].

A researcher's philosophical paradigm will have a direct influence on how they conduct research. The term research framework is adopted by Malmi et al. [144] as it is an inclusive definition that goes beyond individual methods or techniques to capture "the overall orientation or approach that guides or describes the research". It includes well known qualitative methods (e.g. ethnography [226], grounded theory [93], phenomenography [145]) as well as qualitative experimental research methods including statistical methods and survey research. The research frameworks in computer science education identified by Malmi et al. [144] are given in table 2.1 below.

The nature of the study detailed in this thesis, and of the research questions that guided it, meant that it was not appropriate to adopt a positivist paradigm.

Action Research (AR)	A self-reflective systematic inquiry undertaken by participants to improve practice. Typically conducted as an iterative cycle of planning, action, change, reflection.
Case Study (CS)	In-depth, descriptive examination conducted in situ, usually of a small number of cases/examples .
Constructive Research (CR)	Research that aims to demonstrate and/or evaluate the feasibility of a proposed idea (concept implementation; proof-of-concept research). Revolves around the development of, e.g., software, technology, a teaching approach, or an evaluation instrument.
Delphi	Seeking consensus by showing a group of raters a summary of their ratings, with justifications, then iteratively inviting them to reconsider their ratings in the light of what the others have said.
Ethnography (Eth)	A branch of anthropology that deals with the scientific description of individual cultures.
Experimental Research (Exp)	Quantitative research based on manipulating some variables while varying and measuring others. This requires formation of control and experimental groups of participants with random assignment of participants or use of naturally formed groups.
Grounded Theory (GT)	Qualitative, data-driven research in the tradition of Glaser and/or Strauss which aims to formulate theories or hypotheses based on data.
Phenomenography (PhG)	Investigation of the significant variations in the ways that people experience a phenomenon (2nd order perspective).
Phenomenology (PhL)	Investigation of the richness and essence of a phenomenon by studying ones own or others experiences of it (1st order perspective).
Survey Research (Survey)	Quantitative research based on exploring the incidence, distribution and/or relationships of variables in non-experimental settings.

Table 2.1: Malmi et al.'s Research Framework dimension (source: [144])

Possible methodologies to be considered included grounded theory, phenomenology, phenomenography and general ethnographic methods. Grounded theory is an inductive qualitative research method that is data driven as the researcher allows the theory to emerge from the data collected [93]. This is in marked contrast to a positivist approach where the researcher formulates an hypothesis and then sets out appropriate ways to test it. Phenomenology [215] is a research methodology that draws on personal descriptions and accounts of a lived experience. As van Manen puts it:"The essence of phenomenon is universal which can be described through a study of the structure that governs the instances or particular manifestations of the essence of that phenomenon"[215]. Phenomenology is often described as being from a first order perspective as the research is studying their own, or others, experiences of a phenomenon. In contrast Phenomenography [145] is a second order research approach that seeks to identify the variations in the ways that others experience a phenomenon. While the phenomenographic approach encourages participants to voice their perceptions of a given phenomenon, the analysis is conducted in a manner that cuts across both individuals and contexts [97].

The research questions that motivated this study are such that grounded theory is the most suitable research approach to address them. As the aim was to discover the experiences that students found beneficial to them during their studies, an inductive methodology was the most appropriate one to adopt. The existing literature on computer experience, and on factors influencing student success, has been called into question as it is sometimes contradictory in nature. Hence the focus of this study was on developing a theory that emerged from, and was grounded in, the students' experience.

The grounded theory methodology is considered in greater depth in Chapter 3 while its underpinning philosophical paradigms are discussed in more detail in section 3.5.

2.3 Reviewing the Literature in a Grounded Theory Study

The originators of grounded theory, Glaser and Strauss, had a fundamental disagreement on how research using grounded theory should be conducted. This led to the development of two distinct schools of Grounded Theory: the Glaserian School and the Straussian School [197]. These difference are discussed in more detail in section 3.4 below. However, one of these differences is pertinent to the discussion contained in this chapter as it relates to the process of reviewing the literature and, in particular, to the position of a literature review within the research cycle. In their original work Glaser and Strauss did not strictly forbid the conduct of a literature review as they viewed their methodology as "...using various qualitative sources, alone and in combination, to generate theory effectively through comparative analysis" [93, p.163]. However, they did caution that the conduct of a literature review "... can actually restrict the development of a researcher's theorizing" [93, p168]. In his later works, Glaser espoused a rather different viewpoint stating: "There is a need not to review any of the literature in the substantive area under study" [87, p.31]. In short, Glaser believed that the literature should only be consulted once the research process is well under way and substantive data analysis has taken place, which contrasts with the Straussian view that the literature should be consulted prior to the collection of data.

Two of the key purposes of a literature review within the context of a grounded theory study are as follows: Firstly, it provides the researcher with the language and terminology needed to communicate their findings to others in the community. Moreover, through their immersion in the literature the researcher develops an appropriate level of theoretical sensitivity to the nuances contained within the data. This concept of theoretical sensitivity is an essential part of the open coding process (see section 3.3.1 below) within the grounded theory methodology. In the language of Strauss and Corbin this allows the researcher to be "open minded" but not "empty headed" [201]. Secondly, an awareness of the literature allows the researcher to synthesise their findings and situate them within the wider body of existing research. This provides the researcher with a context within which they can discuss the external validity and transferability of their work. These concepts will be explored further within the trustworthiness section of this work, section 3.6.

Glaser's argument against the conduct of a literature review in advance of the collection of data is that "the theory will not be preconceived by pre-empting concepts" [94, p.31]. However, he later accepted that it may be best for the researcher to engage with a "modicum of literature" [87, p.32] and he reminded the reader that the literature "... is always there. It does not go away!" [87, p.33]. He believed that the researcher would have ample opportunity to integrate the literature into their emerging theory during the sorting and writing processes. In particular he saw it as an essential part of the constant comparison process that lies at the heart of the Glaserian version of grounded theory.

As the grounded theory methodology is aimed at the development of an emergent theory grounded in the data, Glaser was of the opinion that initially the researcher should focus on the creation of an emergent theory. This will then guide the way they engage with the literature. In particular, the emergent theory will allow the researcher to be discriminating in terms of the literature they engage with, thus providing a greater breadth of coverage of the literature due to the more focused nature of the review process. Others who follow the Glaserian school of thought allow for an initial, general reading of the literature, provided "the reading is not too extensive" [190, p.20].

The Straussian school of thought differs in that it holds that a literature review is critical to the discovery of the extent of previous knowledge in the area, as this will allow the researcher to determine whether grounded theory is an appropriate research methodology to adopt. It is also essential for the development of theoretical sensitivity. However, they urge caution, as they warn that "familiarity with relevant literature can enhance sensitivity to subtle nuances in the data just as it can block creativity" [204, p.49].

The literature can also stimulate initial questions, providing the researcher with a "stepping off point" at which to commence the initial data collection process. This data will in turn generate questions that will guide the collection of data at later stages in the research cycle. The literature may also guide the researcher in the conduct of theoretical sampling [149]. For example, if the data collected appears to be at odds with the literature, then the researcher may seek to gather subsequent data in a way that helps to determine the nature of the discrepancy and the conditions that might give rise to it. The literature can also guide the researcher towards variations and refinements in concepts that may not be distinct in the initial stages of theory development.

In more recent times it has been necessary to satisfy ethics committees of the validity of any research proposed prior to the commencement of any data collection or analysis [204]. This requires the researcher to set out the general focus and specific methods to be adopted in their work, and to indicate the gaps in the existing knowledge base that they intend to address [149]. This precludes the adoption of a Glaserian approach to the literature. The ethical framework adopted for the study detailed in this work is discussed further in section 2.6.

Once a theory has been developed then both the Glaserian and Straussian schools of thought are in agreement on how the literature should then be used: it may confirm findings or highlight places where the theory developed differs from, or is at variance with, the existing literature.

2.3.1 Reflexivity and Reflexive Bracketing

In computer science education research, many come to the field through the small-scale investigations of their own work in the classroom or laboratory. This leads them further into the field as they seek ways to carry out more in-depth investigations that can be generalised to wider settings. These researchers are typically active in other fields of endeavour and for many computer science education research is not their primary field of interest [50]. Therefore most researchers possess *a priori* knowledge of the field that stems from their professional experience as well as from the literature. They must acknowledge the influence of this prior experience on their perspective and the way they interact with the data. This is done through the process of reflexivity, which Robson considers as "... an awareness of the ways in which the researcher as an individual with a particular social identity and background has an impact on the research process" [177, p.172].

McGhee et al. refer to reflexivity as a "turning back on the original action similar to the knee jerk reflex" [149, p335]. They go on to suggest: "Researchers should be aware of the impact of their previous life experience, including previous reading, and 'turn back' on these to appraise their effect". Thus, reflexivity allows the researcher to identify areas of potential bias and then try to eliminate these from their work in the collection and analysis of data as well as in the generation of theory. However, reflexivity can only be partial at the best of times because it is not possible for a researcher to set aside things they are unaware of [177]. In this work the process of reflexive bracketing was adopted and this is discussed further in subsection 3.3.1.

This process can lead to a tension within the researcher as they seek to achieve a balance between their existing knowledge of the field and the need to set this to one side in order to develop a theory that is truly grounded in the data. On the one hand the researcher's familiarity with the literature and the field may cause them to fail to identify new phenomena within the data, while on the other hand the researcher may become over-reflexive and as a consequence fail to produce a substantive theory that is fully grounded in the data.

Like many others in computer science education the author of this work has been a professional academic in the field for over a decade. She has been an active researcher in computer science and applied mathematics, as well as in the field of computer science education, see section 1.5. She has successfully competed for grants in both these areas [110; 111; 113] and has supervised Ph.D. students to completion in both disciplines (e.g. [25; 193]). She has been involved in the education of both undergraduate and postgraduate students, and has gained a considerable knowledge of the computer science education specific literature during this period. Given this background, in terms of professional experience and literature awareness, it was necessary for the author to constantly seek to be self-aware so that she could bracket this prior knowledge and experience during the course of this study.

As the author was familiar with the literature and the field as a whole, it was pragmatic to adopt a Straussian view of the literature: using it as a guide in the gathering of initial data and as a source of theoretical sensitivity throughout the conduct of the study, while simultaneously taking care to bracket out prior knowledge and resist the temptation to use it to force the emergent theory. In the following section a review of the existing literature on factors that influence success on undergraduate computer science courses is presented. In particular we consider the notions of objective and subjective computer experience, as defined by Smith et al.[187], which formed the initial stepping off point for the study detailed in this thesis.

2.4 Predicting Success in Computer Science

Many students find it difficult to make the transition from second-level into degree courses with a substantial computing component. The nature of a computer science degree programme is such that there is very little continuity between subjects studied during the secondary education senior cycle and those encountered during the freshman years in university [112]. Courses in areas such as computer programming and computer networking require a style of logical thinking that many students have not experienced prior to university entry. Moreover, many students are experiencing university life for the first time and haven't developed an appropriate sense of their own responsibility for, or ownership of, their learning. The challenges of student recruitment, retention and success in computer science at the undergraduate level have been of concern to both faculty members and university authorities for many decades [13; 41; 42; 53; 85; 133; 228] but still remains a source of much discussion and debate in the literature [5; 98; 183; 198].

A wide range of factors have been considered in the effort to determine predictors of success; these factors include age, gender, mathematical ability, second level performance, previous computer experience [187], computer anxiety [106], computer attitude [141] and computer self-efficacy [11].

Almost all of this work has been in the quantitative domain where a numerical scale is used to rate the students ability or performance and some form of statistical analysis (e.g. factor analysis or principal component analysis) is then performed to determine the most significant factors influencing success.

Many students experiencing difficulties during their first year at university cite the challenges encountered while learning to program as one of the major causes of their difficulties. Indeed many engineering and computer science student claim to "hate programming" and feel unable to grasp even the most basic skills [208]. Students who struggle with programming courses either fail and drop out of the course; or manage to continue but assiduously avoid future programming projects and ultimately choose a career that does not involve programming [192]. Psychological studies of programming date back over thirty years [185; 230], but appear to have had little impact on the culture of learning in a programming class, where the literature indicates that today's students have little or no confidence in their programming abilities [5; 198]. Many have viewed this as key to overall success on computer science and technology based degree programs and so they have concentrated their efforts on predicting student success on introductory programming courses (e.g. [16; 105; 228]). This is often known as the CS1 problem as this is the name given to the introductory programming modules offered in many north American higher education institutions. One of the challenges for those in computing education is to design an effective learning environment which results in a significant improvement in students' attitude to, and attainment in, programming. Indeed, computer programming is a key element of the computing curriculum [206; 207] published jointly by the ACM and the IEEE Computer Society. This curriculum provides recommendations for the structure and content of undergraduate programs in computer science and is often used by higher level education institutions to shape the design and content of their degree programs.

2.4.1 The Nature of Factors Influencing Success

If we are interested in gaining a deeper insight into the previous learning activities that impact on a student's ability to succeed, then we must look at the nature of the learning activities they have engaged in. In higher education one of the most commonly used educational frameworks for doing this is the taxonomy of educational objectives devised by Bloom et al. [27] in the 1950's. This classifies the nature of the learning activities that students engage in into three separate domains:

- Cognitive: mental skills (Knowledge),
- Affective: feeling and emotions (Attitude),
- Psychomotor: manual or physical skills (Skills) [27].

In Bloom et al.'s taxonomy the cognitive and affective domains are broken into a series of hierarchical levels. While Bloom et al. did not provide a classification for the psychomotor domain, the one most commonly used is that articulated by Simpson [186]. It should be noted that within the past decade a re-evaluation of Bloom et al.'s classification of the cognitive domain by Anderson et al. [4] argued that the top two or three levels (Analysis, Synthesis and Evaluation) should be considered as on the same level within the hierarchical scale. These hierarchical classifications of the three domains articulated by Bloom are given in table 2.2 below.

Cognitive Domain source: [27]	Affective Domain source: [27]	Psychomotor Domain source: [186]
Knowledge (recall of facts etc.) Comprehension Application Analysis Synthesis Evaluation	Receiving Responding Valuing Organising Characterising	Perception Set Guided Response Mechanism Complex Overt Response Adaption Origination

Table 2.2: Hierarchical Classifications within Bloom's Taxonomy

Another taxonomy that is commonly used in higher education is the Structure of the Observed Learning Outcomes (SOLO) taxonomy of Biggs and Collis [23]. This taxonomy aims to capture the increasing complexity of a student's understanding: this begins in a quantitative manner where the student accumulates facts and information and in later stages it moves to a qualitative phase where the student integrates this detail into their structural framework of understanding [120]. Both these taxonomies have been used in higher education to help the educator strike a balance between rote learning of content and higher level skills of synthesis and evaluation [120].

It has been remarked [217] that most studies in higher education focus on cognitive strategies and motivation and this applies to those considering factors that influence success in computer science also. Vermunt [217] discerned three types of learning activity: cognitive, affective and metacognitive or regulative (see table 2.3 below). Cognitive processing activities are ones that people use to process learning content. Affective learning activities relate to how people deal with the feelings that arise during learning. These may engender an emotional state that may positively, neutrally or negatively affect the progression of a learning process. Metacognitive regulation activities are those that regulate the cognitive and affective learning activities; hence they indirectly lead to the achievement of learning outcomes.

Types	Categories
Cognitive	Relating, structuring, analyzing, concretizing, applying, memorizing, critical processing, selecting
Affective	Attributing, motivating, concentrating, judging oneself, appraising, exerting effort, generating emotions, expecting.
Regulative	Orienting, planning, monitoring, testing, diagnosing, adjusting, evaluating, reflecting.

Table 2.3: Learning activities: types and categories (source: [217])

By far the most frequently mentioned factor for success in computer science is previous programming experience [16; 17; 29; 37; 40; 67; 73; 74; 77; 99; 133; 171; 222; 228]. These studies utilise a wide range of very different methodologies from qualitative to quantitative and mixed methods. They also draw their participants from very different populations, however they all are broadly in agreement that prior experience of programming has a positive influence on the success of those taking introductory programming courses at university level [225]. Another factor that had been considered is prior mathematical experience [16; 40; 192; 228]. One initiative at the University of Leeds attempted to stream students based on the results of an aptitude test and concluded that no relationship existed between the final result in programming and measured aptitude [66]. Others have looked at demographic factors and concluded that they are not suitable predictors of success [77]. While many of these factors fall into the cognitive domain, others have focused more on the affective domain by considering factors such as motivation and attitude.

Student motivation is a key issue in the learning process as students who are motivated to learn use higher order cognitive processes [62]. However, determining what motivates an individual is not trivial. A study of introductory science courses concluded that much of the observed student attrition is due to the failure of introductory courses to motivate students [210].

Biggs categorised motivation into four classes [22]:

- Extrinsic motivation: focuses on the outcome of the effort (i.e. a good future career),
- Social motivation: relates to what other people value (i.e. friends and family),
- Achievement motivation: the opportunity provided for ego-enhancement (e.g. students boost their egos by competing),
- Intrinsic motivation: the process of doing is central (e.g. a genuine interest in the subject).

Jenkins observed that in introductory computer science courses an individual's primary motivation appears to be a significant influence on course completion [119]. When looking at computing students who were struggling and likely to fail, Sheard and Hagan [184] found that these students were more likely to be extrinsically motivated, while their more successful colleagues usually fell into the other motivational classes. When looking at the difficulties of learning to program Jenkins found that students generally retain some form of motivation throughout their programming course, even if this is based on a negative factor such as a fear of failure [118].

Motivation may be considered as a product of two distinct concepts (see, for example, [22]): expectancy and value. A student must both expect to succeed and value that success. If the student does not value success or if they do not expect to succeed then they will not be motivated. Jenkins used this notion of motivation to comment that "if a student does not expect to pass a programming module, then, no matter how they value success, they will not be motivated, and will not engage in tasks that the teacher devises"[117]. Moreover, the criteria students use to measure success can vary widely: some students may be content to pass their examinations and progress from one year to the next, while others may seek to obtain results that place them towards the top of the class.

In the affective domain, there has also been considerable interest in peoples' attitudes towards computers. This has been due not only to the growing proliferation in computing devices in all aspects of modern life, but also because the changing nature of computing technologies means that attitudes and experiences are constantly evolving [82]. Many students are reluctant to use computers and consider working with them a daunting prospect [221]. Self-efficacy [11], computer anxiety [3], and computer experience [82; 187] have been shown to affect how students approach the use of computers during the course of their studies. These factors have been widely investigated, both individually and collectively, in relation to students of social science, health and business studies [3; 49; 143]. However, very few studies exist in the field of computer science [172]. The author of this thesis carried out a joint study with two of her research students to investigate whether computer science students are affected by the same factors as their counterparts in other disciplines [72]. The outcomes of this study are summarised in section 2.5.2 below.

2.4.2 Computer Anxiety, Self-efficacy and Computer Experience in Computer Science

Computer anxiety has been explored in great depth as a factor in many research studies [3; 32; 49]. It can be defined in terms of a psychological response, e.g. computer phobia [172], or in terms of a cognitive reaction, e.g. apprehension of computer technology [121]. The definition that seems to capture all of these characteristics is that of Bozionelos: Computer Anxiety is the "negative emotions and cognitions evoked in actual or imaginary interaction with computer-based technology" [32]. Individuals who suffer from computer anxiety usually display negative sentiments and physiological reactions to computers. These behaviour reactions include (1) avoidance of computer usage; (2) negative comments about computers; (3) employing extreme care with computers and (4) using the computer for the minimal amount of time [32]. Some of the physiological reactions include sweaty palms, shortness of breath and dizziness [14]. Computer anxiety affects general usage and the performance of tasks involving the computer.

Self-efficacy theory has been widely used in educational research, for example computer self-efficacy has been defined as an individuals judgement of their capability to use a computer [56]. Bandura defined self-efficacy as a judgement by an individual of how capable they are of successfully carrying out a course of action to attain a certain outcome [10; 11; 12]. Self-efficacy is not a measure of an individuals skills and past experience, but rather it seeks to look at what an individual believes they can achieve with the skills and knowledge they possess [12]. Bandura argues that individuals with high self-efficacy believe they will be successful at completing challenging tasks. This motivates them to attempt such tasks and to apply their best efforts to succeed. Those with a strong belief in their efficacy view difficult tasks as a challenge that they can master, rather than as an obstacle that must be avoided. Such individuals are more likely to persist when they encounter difficulties and seek coping strategies to overcome any problems they may encounter. They are also likely to be persistent, despite competing demands for their time and efforts. By contrast those with weak judgement of their own efficacy tend to avoid difficult tasks as they view them as a threat.

Studies that investigate the relationship between self-efficacy and computer anxiety, from fields as diverse as health science [227] and education [121], have revealed that there is a significant correlation between these two factors. Computer anxiety research reports that students who perform badly in course assessments or exams have a higher level of computer anxiety than those who succeed [3]. This finding is complemented by strong evidence of an association between high computer anxiety and low self-efficacy [34; 121]. This is not an unexpected outcome since as an individual becomes more confident in their ability to complete computer tasks their level of anxiety about doing those tasks decreases. This boost in confidence may be due to an increase in computer experience, for example that gained through education and training [102]. Therefore, it may be hypothesised that, for computer science students, low levels of self-efficacy and experience, and high levels of computer anxiety are predictors of a decreased level of performance.

As with Computer Anxiety, there is little agreement in the literature on a precise definition of computer experience [82]. Some researchers define it by the number of years of computer use, while others state that it the number of hours usage per week. One of the more comprehensive definitions states that computer experience consists of three components: amount of computer use, opportunities to use computers and diversity of experience [122]. This definition captures a greater breath of computer experience than most other measures.

Different interpretations exist in relation to computer anxiety and computer experience [34; 121]. Some researchers argue that increased experience leads to a decrease in computer anxiety while others state that experience exacerbates an individuals level of anxiety [32]. Thus, based on these studies it is evident that relationships exist between computer anxiety, self-efficacy and computer experience. However, the precise nature of these relationships differs depending on the student groups under consideration, e.g. health science, education, and so the results can not be generalised to other student populations.

2.5 Computer Experience

One of the constructs commonly used to predict how challenging undergraduate students will find courses involving the use of computers is that of computer experience. This concept has not been well defined in the literature [126] and many different instruments and metrics have been proposed for its measurement (the reader is referred to [187] for a detailed review of these). Potosky and Bobko consider an experienced computer user to be someone who "understands enough about computers in order to use them, more or less independent of specific software packages, reasons for use, and computer hardware features" [167, p.338]. While, at its broadest, computer experience can be defined as "the amount and type of computer skills a person acquires over time" [187] or "the degree to which a person understands how to use a computer" [167]. These definitions cover a very wide range of experiences and skills and many have struggled with the creation of a set of quantitative computer experience metrics that capture the essence of the construct. Moreover the nature of the experiences and skills considered may depend on the context e.g. van Braak [214] focused on assessing differences amongst university students and validated his study using data gathered from students of androgyny, psychology and education science, Smith et al. [189] gathered data from first year psychology students in their study of subjective computer experience, while Potosky and Bobko [167] collected data from students in vocational colleges and universities, as well as those in full-time employment as computer programmers.

Potosky and Bobko created a Computer Understanding and Experience Scale consisting of twelve items, self-assessed using a 5-point Likert scale [135]. For completeness these items are listed in table 2.4 below. It can be observed that these questions are reflective of their era (the late 1990s) in terms of the language used (e.g. "mainframe computer system", "PC") and the skills considered (e.g. writing computer programs and e-mail). There is a notable absence of the use of web-based skills e.g. browsing and searching (which would of have been well-established at the time) and the use of social media (which has only come to prominence in the past decade). However, this scale provides evidence of difficulties that persist in the quantitative definition of a scale for measuring computer experience. In particular, the competencies listed are self-assessed and are very dependent on the range of variation of individual understanding.

1.	I frequently read computer magazines or other sources of information that describe new computer technology
2.	I know how to recover deleted or lost data on a computer or PC.
3.	I know what a LAN is.
4.	I know what an operating system is.
5.	I know how to write computer programs.
6.	I know how to install software on a personal computer.
7.	I know what e-mail is.
8.	I know what a database is.
9.	I am computer literate.
10.	I regularly use a PC for word processing.
11.	I often use a mainframe computer system.
12.	I am good at using computers.

Table 2.4: Computer Understanding and Experience Scale (source: [167])

An essential requirement for observing and understanding something is an awareness of variations within it. As Marton puts it "by experiencing variation, people discern certain aspects of their environment; we could perhaps say that they become "sensitised" to those aspects" [147, p.11]. Marton et al. give four patterns of variation within a learning environment that capture the difference between elements that stay invariant and those that do not [147]:

1. Contrast: A point of reference with which you can compare something else e.g. if we are looking at the variation in colours then in order to know about the colour red you need to know about other colours e.g. blue and green also.

2. Generalisation: Variations in values of the aspect under consideration e.g. the variations within the colour red such as strawberry red and cherry red.

3. Separation: the ways in which some aspects of a phenomenon can vary while others remain unchanged.

4. Fusion: The experience of a phenomenon, and all its aspects, in its entirety.

We can easily see how an individual who has little interest in computing may rank themselves as having an extremely good understanding of what a local area network (LAN) is if they are aware that it is a network of computers and devices e.g. in a home or work environment. They may not be aware of the concept and all the possible variations with in it e.g. how it contrasts with personal, metropolitan or wide area networks in terms of geographical area covered, data rates offered etc., or the wide range of technologies available to create such networks. Indeed such an awareness may lead those with a much greater understanding of what a LAN is to give themselves a lower ranking on the Likert scale as they are aware of the variations within the concept about which their knowledge is sketchy or incomplete. Similarly one person may spend many hours producing documents using a common template in a word processing package and another may spend the time producing more complex document involving the use of many of the features of the word processing package. Both may report themselves as experienced users of the package, but their awareness of the variations within the word processing package vary greatly.

Another aspect of such quantitative scales is that self-assessment can often be unreliable as people often under or overrate their abilities. Boud and Falchikov found that high achievers tend to underrate their abilities and low achievers are prone to overrate their skills [31]. A more recent study by Ballantine et al. reported a statistically significant over-estimation of computer competence by first-year undergraduate students on a business related course, regardless of their achievement level [9]. Thus the validity of any self-assessed measurements of computer experience may be open to question.

2.5.1 Objective and Subjective Computer Experience

In an effort to provide a simple measure of computer experience some definitions equate it with the amount of computer use a person has engaged in [8; 43]. This single component formulation is seen by many as an over-simplification of the construct and so more sophisticated, multi-faceted metrics have been developed [122; 167; 182; 214]. In an effort to move the debate forward Smith et al. [187] postulated that the concept of computer experience should be viewed as one that is bi-dimensional in nature, with two distinct constituents: objective and subjective computer experience.

The concept of objective computer experience (OCE) was defined as "the totality of externally observable, direct and/or indirect human-computer interactions which transpire across time" [187, p.229]. Smith et al. built on the work of Jones and Clarke [122] to define direct OCE as the "amount of computer use, opportunity to use computers and diversity of experience" [187, p.229], with indirect OCE defined as "the medium through which information or knowledge about computers is acquired" [187, p.229].

The "amount of computer use" and "opportunity to use computers" are factors that reflect the amount of time spend interacting with computing technology and how often these technological resources are available to the individual (e.g. at home, work, school or on the move). The focus is on the quantity of time and resources the person has available to them, rather than the quality of their engagement and interaction with the resources. The "diversity of experience" seeks to capture the range and nature of the interactions with computers that has taken place e.g. through playing games, word processing, computer programming, development of information systems etc. Capturing the diversity of an individual's experience may prove challenging as the nature of the person's engagement with computing tasks may vary greatly and such variation is not easily captured using quantitative methods. The "sources of information" metric seeks to capture how computer related information has been acquired e.g. through the internet, peers, parents, teachers or through the observation of another's computing experiences [187]. It is considered indirect OCE as the person may not have to engage in any activities involving a computer in order to gain this experience or knowledge.

Subjective computer experience is defined as "a private psychological state, reflecting the thoughts and feelings a person ascribes to some existing computing event" [187, p.230]. As SCE is a "private psychological state", this means that it is something that cannot be observed or measured directly; rather it is a process that exists in an individual's mind. An existing computing event may refer to any experience that directly or indirectly involved computers. A direct experience involves actual interaction with, or manipulation of, the computer while indirect experiences are those gained from observing others, through reading or discussion. So in order to gain an understanding of SCE the researcher must first determine whether the experience is of a direct or indirect nature, they must then look at the cognitive and affective processes associated with that experience [187]. The cognitive factors used by Smith et al. were "perceived competency, control and perceived usefulness"; while the affective factors considered were "perceived enjoyment, anxiety, relaxed and more general feelings (e.g. eager, excited, overwhelmed, foolish)" [188, p.233].

In a later work, [189], Smith et al. discuss a methodology for the measurement of SCE and refer the reader to an unpublished work, [174], for more information on their Subjective Computer Experience Scale (SCES). The development of the SCES involved interviews with "two small focus groups, each consisting of four psychology undergraduate students varying considerably in levels of computer experience and expertise" [189, p.130]. These interviews were transcribed and analysed to identify possible themes for inclusion. These themes were "benefits of computer use, challenge and enjoyment associated with computer use, accessibility of technical support and assistance, frustration and anxiety relating to computer use" [189, p.130].

Using factor analysis, Smith et al. reduced their SCES to 25 items and five factors [189]. There are a number of major criticisms and concerns that may be raised about their work. Firstly the focus groups used to draw up their SCES were representative of a very specific group of individuals (psychology students); in addition these focus groups were extremely small. Thus it would be very difficult to argue that the outcomes of this qualitative study are in anyway trustworthy. Moreover, the SCES was validated using a sample of predominantly female first year undergraduate psychology students [189]. This would make if very difficult to generalise the findings to other groups and settings, a fact that is acknowledged by Smith et al. [189, p.140]. Smith et al. also raise concerns about the validity of their results as the surveys were not administered under the supervision of the researcher. Smith et al. make no claims to have carried out content validation checks (e.g. by connecting their work to the literature or through the use of expert panels) or construct validation (e.g. through a confirmatory factor analysis) [199].

In a more recent work, Prince [168] re-cast Smith et al.'s earlier work to bring it into alignment with the construct of a "value account" [21]. This hypothetical structure stores the accumulated positive and negative responses evoked by a person's interaction with stimuli (in this case the stimuli are the interaction with a computer and the thoughts and feelings associated with this interaction). Prince used the value account construct to re-defined objective computer experience as the "totality of interactions with the computer" and subjective computer experience as "the thoughts and feelings associated with the interactions".

In the creation of his version of SCE Prince sought to differentiate it from other constructs based on assessment of attitude: computer anxiety [106], computer attitude [141] and computer self-efficacy [11]. From Prince's perspective computer experience is the past objective and subjective stimuli that create these attitudes, and so any instrument that seeks to capture computer experience should measure the value account that led the person to develop these attitudes, rather than the attitudes themselves [168].

Prince drew on the work of Jones and Clarke [122], Smith et al. [187] and Hasan [103], to create a survey instrument to measure OCE. Weil et al. [222] used an adjective check-list to study attitudes to a single computing event and Prince re-purposed this list as a means of capturing SCE [168]. Prince validated his OCE and SCE survey instrument via a pilot study, which led to changes and refinements to the survey. For completeness this survey is presented in appendix E. This survey instrument was validated using data collected from 130 undergraduate business students [168].

A number of the concerns that can be raised about the work of Smith et al. [187; 189] are also relevant to the work of Prince [168]. While Prince sought to draw on a review group to assist in the validation of his survey instrument, this group was drawn from a narrow set of his colleagues i.e. those who develop or use psychometric Management Information Systems (MIS) constructs. This would limit the trustworthiness of the study to those engaged in this sphere. The final survey instrument created was validated using a sample of undergraduate business students. As before, this would raise concerns about the generalisability of this work. Prince is aware of this difficulty and cites it as an area for future work [168]. While Smith et al. drew participants from a cohort of 185 first year undergraduate psychology students with a mean age of 20 (std. deviation 5.26) [189, p.130], Prince drew his students for across a range of business majors with a mean age of 23 [168]. Prince comments that "most of the students use the computer a comparable amount of time and the same applications whether they want to or not" [168]. He suggests that this may be because their studies require them to complete similar tasks using a computer. While Smith et al. have captured aspects of their students experience prior to commencing their studies, it would appear that by focusing on students in the later stages of their undergraduate studies Prince has obtained results that focus more on the computer experiences of students during the course of their undergraduate studies.

In both the work of Smith et al. [187; 189] and Prince [168] the focus is on the development of a positivist, quantitative survey instrument to capture objective and subjective computer experience. This contrasts with the work presented in this thesis which adopts a post-positivist approach to the development of a theory of computer experience grounded in the experience of first year undergraduate students engaged on a course with a significant technological component, namely computer science.

2.5.2 An Exploration of Computer Anxiety and Computer Experience amongst Computer Science Students

The researcher and two of her graduate students undertook a study [72] motivated by the following questions:

- Do computer science students suffer from computer anxiety and, if so, does this persist across different years of study?

- Is there a relationship between computer anxiety and self-efficacy?
- Does computer experience affect the level of computer anxiety and self-efficacy?

The statistical sample used consisted of 163 participants, comprising 32 female and 131 male students drawn from all four years of a computer science degree program offered in an Irish University. Each student's participation was purely voluntary and they were assured of their anonymity. A detailed questionnaire was administered to the students within the first week of the first term of the academic year. It was felt that it was essential to capture the students initial conceptions and beliefs prior to their involvement in the learning process. This was done to insure that their responses where not influenced by any transient difficulties they may have been experiencing with individual course modules.

The students level of computer and software experience, computer anxiety and self-efficacy were assessed using a multi-part questionnaire. The first section of the questionnaire measured computer anxiety and was adapted from the Computer Anxiety Rating Scale (CARS) [123]. The CARS contains 10 statements which are anxiety laden while the remaining 9 statements are non-anxiety laden statements. Non anxious items are reverse scored before calculating the level of computer anxiety. It was necessary to review and change some of the questions to adjust their relevance for computer science students. Individual scores lie in a range between 19 (low anxiety level) and 95 (high anxiety level). The self-efficacy questionnaire [172] used was initially designed for the C++ programming language. Students rated their current self-efficacy on a Likert-type scale. Due to the fact that the student population for this study were taught Java as their main programming language, the original questions were adapted to reflect self-efficacy in this language. The survey consists of 32 questions and the measured level of self-efficacy for an individual can lie between 32 (low self-efficacy level) and 224 (high self-efficacy level). The final section of the questionnaire was related to computer experience. Three existing surveys [103; 115; 167] were combined to form this section. These surveys were not designed specifically for computer science students therefore it was necessary to adjust some of the questions to make them

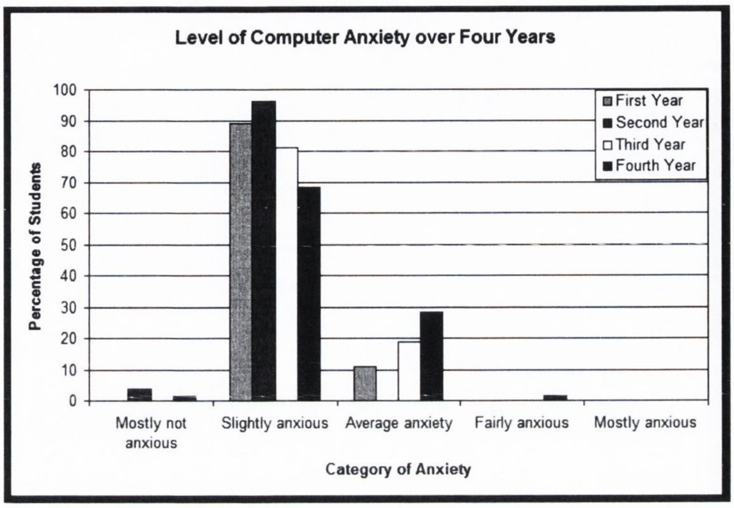

Figure 2.1: Level of Computer Anxiety over Four Years

more relevant to the sample population. An additional question was included in this section to determine the amount of time the students use a computer each week.

The data was analysed to look at the factors both individually and in combination. It was anticipated that computer science students would experience a low level of computer anxiety, if any, and that first year students would suffer from the greatest level of anxiety. In order to address this, computer anxiety was measured across the four years of the population. In figure 2.1 it can be observed that most students, independent of their year of study, fall under the category slightly anxious which indicates that their level of computer anxiety is relatively low.

The interesting observations on figure 2.1 relate to the results for first and fourth year students. The majority of first year students rate themselves as only "slightly anxious"; this is better than anticipated prior to administration of the questionnaire. The first year students have a high degree of confidence in their computer abilities. This may be due to the fact that during their second level education their main computer experience is limited to the use of office-type applications. The fourth year students display a wider range in their level of anxiety. This was not as expected: it was predicted that fourth year students

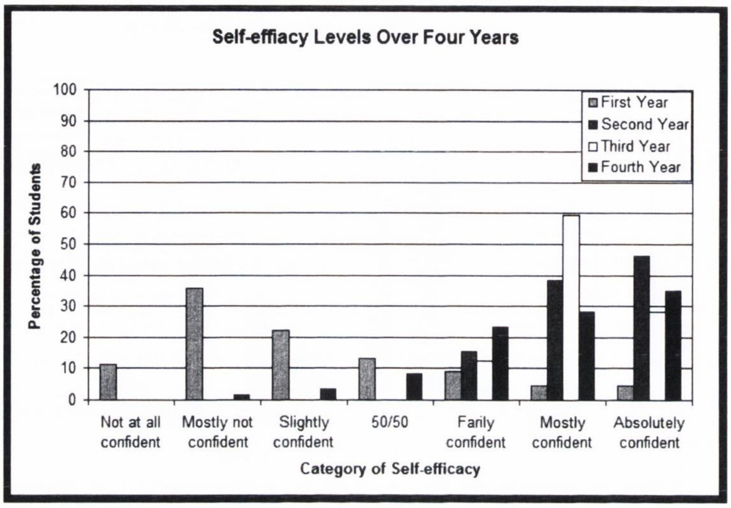

Figure 2.2: Level of Self-Efficacy over Four Years

would fall into the "mostly not anxious" or "slightly anxious" level due to the fact that they had been working closely with computers for the previous three years. One possible explanation is that the fourth year students are just commencing their final year at university. As a consequence, they are likely to feel anxious about doing well and achieving a good final grade for their degree. Another explanation may be that the fourth year students know more about computer science and therefore are more aware of the limitations of their knowledge.

The next sets of results are for self-efficacy and computer anxiety. Firstly, the level of self-efficacy across the four years of study is examined. It was expected that the first year students have a lower belief in their programming abilities. This is partially due to the fact that programming is not compulsory for second level students in Ireland, so most first year students would have little or no programming experience. Figure 2.2 presents the distribution across the seven levels of self-efficacy for each year. The first year students have lower levels of self-efficacy; this is as initially expected. A closer examination of Figure 2.2 reveals that the fourth year students have a wider variation in their self-efficacy levels than those in their second or third years. This may be attributed to the fact that they are about to commence their first major individual programming project and hence they may be experiencing some self-doubt.

To further investigate the relationship between computer anxiety and self-efficacy, scatter plots were generated and Pearsons correlation coefficients calculated. The scatterplots in figures 2.3,2.4,2.5 and 2.6 show that first year students have a medium level of computer anxiety with low self-efficacy. The corresponding Pearson correlation coefficients, given in Table 2.5, do not show a statistically significant relationship between the two factors for first year students. The correlation coefficients for the other three years indicate that there is a significant negative relationship between the factors. Moreover, this relationship increases in strength with the year of study. Therefore, as the level of self-efficacy increases the level of computer anxiety decreases. In the computer usage and software experience section of the questionnaire, the majority of students from each year were observed to have "average experience" except for the second year students who were more confident and rated their ability as "very experienced". The fact that both first year and fourth year students rated themselves similarly might be explained by the different meanings that students were giving to the term "experience". The fourth year students are close to completing their computer science degree and so should rate their experience more highly. Given that this is the last year of their degree course they may be more aware of the deficiencies in their knowledge and capabilities.

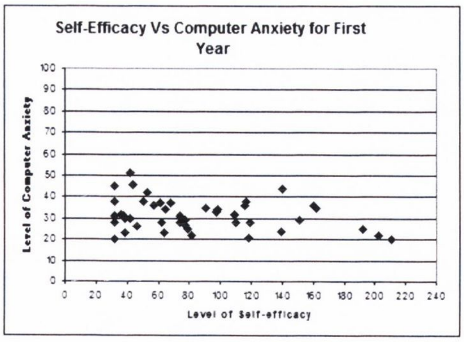

Figure 2.3: Self-Efficacy vs Computer Anxiety for First Year Students

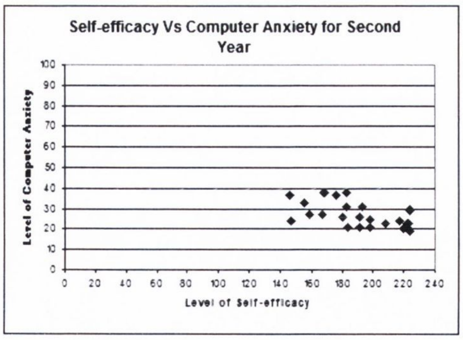

Figure 2.4: Self-Efficacy vs Computer Anxiety for Second Year Students

Finally the question "Does computer experience affect the level of computer anxiety and self-efficacy?" was considered. It was established that there is a

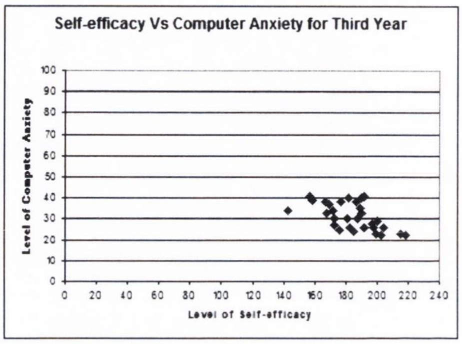

Figure 2.5: Self-Efficacy vs Computer Anxiety for Third Year Students

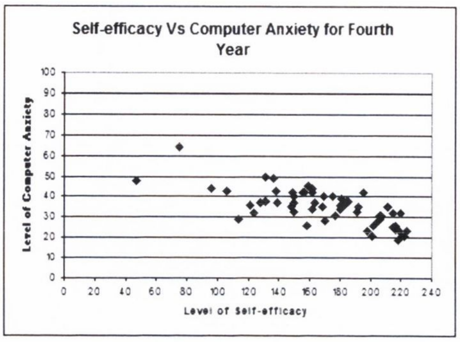

Figure 2.6: Self-Efficacy vs Computer Anxiety for Final Year Students

Pearson Correlation	Computer Anxiety vs Self-efficacy	Computer Anxiety vs Experience	Self-efficacy vs Experience
First Year	-0.262	-0.598**	0.519**
Second Year	-0.546**	-0.513**	0.565**
Third Year	-0.614**	-0.606**	0.534**
Fourth Year	-0.725**	-0.708**	0.612**
** Correlation is significant at the 0.01 level (2-tailed)			

Table 2.5: Pearson's Correlation Coefficients for Computer Anxiety, Self-efficacy and Computer Experience

significant negative relationship between computer anxiety and computer experience. Table 2.5 displays the values of correlation coefficients over the four years, these values indicate that as the level of experience increases the level of computer anxiety decreases. A relationship was expected to exist between self-efficacy and computer experience [34; 102; 121; 227]. Table 2.5 displays the correlation coefficients for these factors. This confirms the existence of a significant positive relationship, which demonstrates that as the level of computer experience increases so does the level of self-efficacy. Thus the students level of belief in their abilities increases as they progress with their course of study.

The findings of this preliminary study revealed that students suffer from higher than expected levels of computer anxiety. Additionally, low self-efficacy levels are very common among first and fourth year students. Based on the results a negative relationship was observed between computer anxiety and experience which means that as computer experience increases, the students level of anxiety decreases. Also a statistically significant positive correlation was identified between experience and self efficacy, which denotes that when the former increases the students become more self confident and their level of self-efficacy increases.

2.6 Ethics

The ethical issues surrounding any research project should be to the forefront from its very inception. Traditional ethical guidelines for research involving human subjects revolve around the principal of informed consent and the protection of subjects from harm [28]. Due to public concerns about the exploitation and harm to subjects many institutions have put in place guidelines and procedures that researchers are required to follow prior to and during the conduct of any studies. Thus every researcher should be fully informed of the ethical implications of their work and take every care to ensure that their research follows their institution's policy on research practice involving human subjects. For example, in the researcher's university current studies are guided by the Trinity College Dublin Policy on Good Research Practice [55] and ethical approval must now be obtained in accordance with the research ethics protocol of the TCD School of Computer Science and Statistics [161]. The data collection phase of this project pre-dated the establishment of these procedures; however the study was carried in accordance with the ethical practices of the School of Psychology in Trinity College Dublin, as its policy was deemed most appropriate for a qualitative study of this type. The consent form used by participants in this project is included in appendix B.

When conducting qualitative, interview-based research the relationship between the researcher and the participants may be viewed as closer to that of a friendship than a contract [28]. This can result in a perceived lack of a written code of ethics, however this is usually far from the truth as, in general, the researcher has received ethical approval from their institution prior to the commencement of their work. As well as the host institution's ethical protocol and policies on good research practice, this study also followed the code of ethics recommended by Bogdan and Biklen [28, p.44]:

- Avoid research sites where the informants may be coerced to participate in your research.
- Honour your informants' privacy.
- There is a difference in informant's time commitment to you when you do participant observation in public places (where people are spending the time they would normally spend there), and when they do an interview with you.
- Unless otherwise agreed to, the subjects' identities should be protected so that the information you collect does not embarrass or in other ways harm them.
- Treat subjects with respect and seek their co-operation in the research.
- In negotiating permission to do a study, you should make it clear to those with whom you negotiate what the terms of the agreement are, and you abide by that contract.
- Tell the truth when you write up and report your findings.

In the following subsections we discuss the key ethical issues and how they were addressed in the course of this study.

2.6.1 Informed Consent

A critical element of the research process is that any participants in a research study are made fully aware of the purposes and implications of the work prior to their engagement with it[28; 129]. This is usually achieved through the practice of informed consent. By contrast, covert research, where the researcher does not disclose the fact that the research is being conducted to the participants or subjects, is usually considered unethical and problematic [142]. More recently, Spicker has argued that covert research is distinct from research involving deception of the subjects and, in informal settings, it should be accepted as a normal part of academic enquiry [191].

The process of informed consent was adopted throughout the course of the work detailed in this thesis. The participants were informed of the aims and objectives of the research and its implications. They were then asked to consider participating in the study and given time to consider the request over the course of one week. Participation was through an opt-in process, where participants were given the email address of the researcher and asked to mail then to find out more about participation in the study. Prior to the interview process the participants provided written consent to the study; moreover they were verbally reminded that they could terminate their participation at any time and request for any data collected to be destroyed. In the course of the study none of the participants terminated the interviews early.

2.6.2 Confidentiality

In any research study it is essential to protect the confidentiality of the participants [28; 129]. In the course of this work the identity of participants was protected through the removal of any identifying personal information including their names. This was done during the transcription and initial coding process, so that all transcribed records were anonymised at the moment of their creation. Care was taken to ensure that the signed consent forms were not linked to the contemporaneous notes taken by the interviewer or to the interview transcripts. The motivation for this is the protection of the participants from any potential embarrassment or harm [28]. It should be noted that providing the participants with anonymity may not be sufficient to protect their confidentiality, if other information provided would allow for their identification through indirect means. Therefore care was taken to remove any such revealing information from the data prior to the analysis process. For example pseudonyms were given to the university, course, name and gender of the participants in the transcribed data.

2.6.3 Emotional Protection

When conducting interview based research it is important to consider the research questions being asked and determine if any of them are likely to be emotional in nature [28]. If this is the case then the researcher must take particular care to be sensitive to the participants emotions when asking these questions. Indeed, the researcher should constantly be concerned about causing emotional distress to the participants and must actively look for the signs of such distress during the course of the interview. If necessary, the researcher should terminate the interview if the participant is distressed.

It was not envisaged that participants would suffer any physical harm during the research process or due to the publication of the research findings. However, it may cause the participants to remember negative experiences associated with the use of computers. During the course of this study the researcher looked out for signs of such distress and would have terminated any interview where the participant suffered such distress. There were no occasions when it was necessary to cut an interview session short due to the emotional distress of the participants.

2.6.4 Data Protection

One of the ethical concerns that must be addressed by any researcher is the collection and storage of data. All personal data collected for research purposes needs to be securely stored and safeguarded. It should be noted that the provisions of the Irish Data Protection Acts [162; 163] apply only to personal data: If the data collected is fully anonymised then the researcher need not concern themselves with the restrictions of the legislation. While the interview data collected as part of the work detailed in this thesis was anonymised during the transcription process it was not done through an irreversible process and so the data was deemed to be covered by the requirements of the then data protection legislation. The primary data sources for this study were the recorded interviews and the contemporaneous notes of the researcher. The interview recordings were stored on an encrypted memory stick which was kept with the contemporaneous notes in a secure location.

2.7 Conclusion

This chapter has provided an overview of computer science education and has explored how the work detailed in this thesis may be situated within this field. The factors that may be used to predict success in computer science were then considered, in particular the concept of computer experience is reviewed and its limitations discussed.

In the following chapter, the theoretical, methodological and practical aspects of grounded theory are presented. The challenges and limitations of using grounded theory are considered and the steps that can be taken to mitigate them are discussed. These are then used to put appropriate safeguards in place during the design and implementation phases of this study to ensure the trustworthiness of the results obtained.

Chapter 3

Grounded Theory

This study uses the grounded theory methodology to seek a definition of objective computer experience that captures the context and effectiveness of key technological skills and knowledge that students have acquired prior to commencing their third level studies. It documents and critiques the tactics and strategies used by undergraduate students in order to adapt to the challenges they encounter as they commence their university education.

Before we address the reasons why a qualitative approach using grounded theory was employed in this work, we must first consider what we mean by a theory and what we consider its key, defining attributes. It is only once these have been clearly articulated that we can hope to identify any theory which emerges from our study. Moreover, not only will it provide a suitable framework for the collection and analysis of data, it will also enable us to recognise any theory that emerges from our work.

3.1 What Constitutes a Theory

The understanding and definition of what is meant by a "theory" is not straightforward: A theory can be considered as a set of interconnected propositions [128], a set of concepts and a set of propositions about the relationships between those concepts [124] and statements of laws or principles which provide an explanation for a group of facts or empirically verifiable observations. A theory seeks to provide insight into a situation or phenomenon by articulating a systematic and rational explanation of it. This usually takes the form of a set of laws or rules which determine or predict what will occur under reproducible conditions [153]. These rules or laws can then provide a basis for future decisions and conduct.

The nature of the theory developed depends on its context and purpose: Descriptive theories simply describe how things took place in a particular setting at a given point in time; explanatory theories seek to capture why things happen in the way that they do; and predictive theories endeavour to predict what will happen in the future given that a certain set of conditions exist within the context of the problem. By the rigorous nature of the Glaser and Strauss methodology, [93], theories developed using grounded theory are all-encompassing in that they are descriptive, explanatory and predictive in nature.

While Glaser and Strauss [93] focus on a sound methodological basis for the development of a theory, they neither define what they mean by a theory nor do they detail how the type of theory developed using their methodology might differ from other theories. Rather, their focus is on setting out a disciplined, structured methodology for the construction of theory from qualitative data. Their emphasis is on the "grounding" of the theory in the data through the use of more methodologically defensible techniques. They argue that theories developed in this way are better by virtue of their grounded nature. However, they do not differentiate between theories developed using their methodology and others that have been developed through alternate means. Thus once a theory has been developed it should not "look different" from other theories just because it has been obtained through the use of grounded theory. Thus the criteria used to characterise a theory and to judge its success are the same regardless of the methods used to derive it.

May [148] sums up the criteria that should be used to judge the breadth, depth and quality of any theory developed as follows: [148]

1. The concepts that make up the theory should form a logical whole i.e. the theory should be dense,

2. The propositions and hypotheses that make up the theory should be clearly supported by the data, interrelated as far as possible and well articulated

i.e. the theory should be clear,

3. The theory should provide novel explanatory, descriptive or predictive perspectives of the phenomenon under consideration, i.e. the theory should be insightful.

3.2 Grounded Theory

The ground-breaking work of Glaser and Strauss, described in their 1967 book "The Discovery of Grounded Theory" [93], had far-reaching consequences for the conduct of qualitative research across a wide range of domains (e.g. social science, health science, information systems). This book was pioneering in that it challenged the dominant focus on quantitative methods that had arisen over the preceding decades. It countered the widely held view that qualitative methods were primarily useful in preliminary studies prior to the deployment of more scientific quantitative methods [45] . Glaser and Strauss originally defined grounded theory as "The discovery of theory for data – systematically obtained and analysed in social research" [94, p. 1].

Glaser and Strauss established systematic guidelines, analytic procedures and research strategies for the conduct of qualitative data analysis that lead to theory development. These made grounded theory attractive to a wider audience; particularly to novice researchers as the methodology provides systematic guidelines and approaches for deriving theories of human behaviour from empirical data. It does not dictate how this data should be collected, a topic which will be considered in greater detail in section 3.3 below.

In the three decades that followed the publication of this work the original authors developed and debated their methodology in numerous books and articles [87; 88; 89; 90; 92; 94; 200; 201; 202; 203]. These debates grew particularly heated with the publication of Strauss and Corbin's book "Basics of Qualitative Research: Grounded Theory Procedures and Techniques" in 1990 [201] and its significantly revised second edition in 1998 [204]. This book provided readers with detailed procedures and methodologies for the conduct of grounded theory based studies. Glaser felt that Strauss and Corbin forced data and analysis through the methodologies they adopted [87] and that this went against his belief in the emergence of theory from gathered data [94]. He expressed his strong opposition to the idea of forcing the theory to address preconceived questions or to the fitting of any emergent theory into existing or predetermined frameworks. He emphasised the need to compare data to data, concept to concept and category to category:

> Categories emerge upon comparison and properties emerge upon more comparison. And that is all there is to it [87, p. 43]

This contrasts with the series of steps and processes that Strauss and Corbin believe will lead to the emergence of a valid theory if executed carefully.

Regardless of which version of grounded theory is followed both assume that there is an external reality that can be revealed and recorded by the researcher, and that this reality does not depend on the researcher or the methods used to reveal it – Glaser's methodology involves the discovery of data, its coding and then the repeated use of comparative procedures while Strauss and Corbin emphasise the use of analytic questions, hypotheses and the careful application of their methodological steps [45].

3.3 Methodology of Grounded Theory

When using grounded theory the researcher does not start out with a preconceived hypothesis and then seek to gather data to enable them to test it, rather they begin gathering data and let the theory emerge or emanate from the data [96; 204]. Glaser and Strauss were not prescriptive in how data should be collected for the development of a grounded theory. While many studies restrict themselves to the collection and analysis of interview data, grounded theory techniques can be applied to data from a wide variety of sources, both qualitative and quantitative, including observations, often in the form of field notes, informal conversations, formal interviews, recorded diaries, journals and autobiographies [45]. As the research evolves subjects are chosen for participation in the study using theoretical sampling techniques that allow the researcher to validate and refine their theory. Theoretical sampling is discussed in more detail in section 3.3.2 below.

The collected data forms the textual dataset that is at the heart of the process of theory development. The dataset is constantly being extended as data is coded immediately after it has been collected. This process involves the careful reading, and re-reading, of the available data in an effort to identify both relevant variables within the data and the relationships between these variables.

Grounded theory requires the researcher to carry out the analysis of the data on a number of levels. We adopt Strauss' three-state approach to analysis that involves open, axial and selective coding [200; 204]. These processes gradually lead the researcher towards the elements that might form the core of any emerging theory and to an articulation of the relationships between these elements.

Open coding requires the researcher to identify, name, categorize and describe any phenomena found within the text [30]. As part of this process the text is interrogated line-by-line and the researcher is constantly noting variables, known as codes, that might be of relevance to the problem being considered. These codes are then reviewed with the aim of grouping those with a common theme together into higher order entities, called concepts [1].

Concepts are then further analysed with the aim of gathering together those with a similar focus into more abstract groups, called categories. The process of relating categories to each other is known as axial coding. Each category is compared with those already encountered using the constant comparative method of analysis. In this way the researcher builds up an appreciation of the similarities and differences that exist between the identified categories. It may also be possible for the researcher to identify subcategories. These are defined as "concepts that pertain to a category giving it further clarification and specification" [204, p.101].

As the researcher identifies central categories and the relationships that exist between them and their subcategories, the focus is usually on the causal relationship between them [30]. When gathering more data will not yield further insights into a category and its subcategories, or into its relationship with other categories, then it is deemed that a point of theoretical saturation has been reached [139].

Selective coding is the process of determining the central phenomenon to which all other categories can be related. Strauss and Corbin [204] view this as the central narrative or storyline which links the categories and their relationships.

Grounded Theory yields an explanatory framework that articulates a central phenomenon (the core category) and the conditions that explain the situation within which it occurs; these conditions may be causal, intervening or contextual [204]. Causal conditions capture events that lead to the development of the phenomenon, contextual conditions capture the specific context of the phenomenon and intervening conditions influence the investigated phenomenon.

The Grounded Theory methodology is an iterative process where the collection of data and its analysis are carried out in repeated cycles. As the researcher proceeds through these cycles they gather an ever richer set of data that allows them to refine, focus and validate their theoretical findings. In the following subsections we explore the techniques that make up the grounded theory methodology in greater detail.

3.3.1 Open Coding

In a grounded theory study data collection and analysis are intertwined, the researcher "jointly collects, codes and analyses his data and decides what data to collect next and where to find them, in order to develop his theory as it emerges" [93, p.45]. This means that the analysis of data begins almost immediately as data is reviewed and coded as it is collected. The coding process involves the use of comparative analysis to classify elements of the data into categories and to identify links between categories [93]. Initially codes are open in nature as the researcher seeks to identify, name and categorise phenomena in the text. This open coding process can be broken into two phases – conceptualising the data and the identification of categories [204].

Glaser suggests three questions that can be used to help conceptualise the data [94, p.57]:

- "What is the data a study of?"
- "What category does this incident indicate?"
- "What is actually happening in the data?"

Such questions help the researcher to separate the data into actions, events, incidents and ideas. The data is interrogated line-by-line and the initial codes noted alongside the relevant data sequence e.g. a paragraph, sentence, phrase or physical action. The initial codes are given suitable labels that help in the later analysis and synthesis of categories. In this process, known as microanalysis [204], the researcher must adopt a rigorous, analytical approach to the data and must review the data from a wide variety of perspectives: from the granularity of short phrases or quick gestures to full sentences, paragraphs and participant demeanour. This requires the researcher to make multiple passes through the data, adopting a different perspective for each pass.

This process of labelling requires the researcher to used their judgement in the identification and labelling of concepts. This requires the researcher to be theoretically sensitive in their awareness of the variety of meanings that can be attributed to the data. Corbin defines it thus: "Theoretical sensitivity refers to the attribute of having insight, the ability to give meaning to data, the capacity to understand, and capacity to separate the pertinent from that which isn't" [201, p.41].

When coding data, each researcher brings their own prior experience, knowledge and attitudes with them; be these from their knowledge of the literature, from being a professional in a related field, from personal experience or from the experience gained by having carried out similar analysis in the past. All these factors will influence the researcher's sensitivity to the meanings that are embedded within the data.

The researcher's reading of the literature heightens their awareness of possible concepts to expect in the data. However, the downside of this is that the researcher must be careful not to approach the data with preconceived ideas of what they expect to discover in their analysis. Similar problems can occur for those who work in a closely related field to that under consideration. While this naturally provides them with a rich insight into the field, they can also become desensitised to things that have become routine and commonplace for them. Equally those who have a similar background to the subjects they are gathering data on have a greater appreciation and awareness of their subjects experiences, however they must be slow to impose their own recall of these experiences onto the subject's personal narrative. The researcher must constantly strive to identify differences and variations within the data that help elucidate possible concepts and connections in the later analysis. They must be constantly critiquing themselves and the data, asking if the concepts they perceive fit the reality of the data.

As Corbin observes "It is theoretical sensitivity that allows one to develop a theory that is grounded, conceptually dense, and well integrated – and it does this more quickly than if this sensitivity were lacking" [201, p.41].

In order to help the researcher suspend their own judgement on the data they can use the process of bracketing [175; 209], as discussed in section 2.3.1. This prevents the data and the analysis becoming a reflection of the researchers preconceptions of what the data should reveal and what the outcomes should be. The researcher must reflect on their own values, and beliefs, and then temporarily suspend these, or set them aside, so that they can become completely immersed in the subject's view of the phenomena [116] .

It is hard to delineate between the process of bracketing and that of theoretical sensitivity. On the one hand the researcher must strive to eliminate their own opinions and bias on the data while simultaneously being theoretically sensitive to all of the nuances and subtleties it contains. This is a potential pit fall that the researcher must constantly confront within their own work, as failure to set aside their own judgements and opinions may lead to a theory that is based primarily on speculation, bias and assumptions. Equally they must avoid becoming over-sensitive where they may attribute a deeper meaning and higher level of sophistication and subtlety to the data than it merits. These issues were addressed in this work through the taping and transcription of interviews, so that that the analysis was not based purely on field notes. The researcher engaged in a period of reflection on their existing research and professional practice in the field with a view to becoming more self-aware of their own preconceptions. Also, the researcher did not engage with the literature that directly related to the topic under consideration until the process of analysis had begun.

Strauss [200] articulates two different methodologies that can be used for open coding: *in vivo* codes are those concepts conveyed through the language and terminology used by the study's participants while *scientific constructs* are those that derive from the investigators knowledge and awareness of the field under consideration. This terminology helps the researcher to distinguish between the constructs that emerge directly from the data and those that the researcher derives based on their scholarly appreciation of the field. The central concepts that emerge from the data are isolated into codes and these are then abstracted into categories [204].

The properties of a category capture the characteristics that are common to all concepts within the category. Strauss and Corbin state that properties are the "characteristics of a category, the delineation of which defines and gives it meaning" [204, p.101]. The dimensions of a category detail how its properties fit into a wider range of possible general properties i.e. the dimensions capture the range of variability of a given property.

When looking at the dimensions of a category, the researcher must also consider the notion of behaviour as "a classification implies, either explicitly or implicitly, action is taken with regard to the classified object" [204, p.104]. The example Strauss and Corbin give is of a flight taking off and landing and that when we consider the concept of a flight these are implicit actions or behaviours associated with the concept. Hence the researcher must also consider the sequence of actions, contextual conditions and action strategies that affect the dimensions of the category. These may not be explicit and it may require the use of the questions "how?", "when?" and "why?" in order to help extract them from the data. For example we might wish to consider what situations provide students with experience of a given phenomenon and the behaviours the students exhibit as a result of this experience.

The researcher identifies, develops and refines the emerging categories through a process of constant comparison of their properties and dimensions. This process is carried out continuously throughout the analysis of the data, as new instances in the data are considered in light of the properties and dimensions of previously identified categories. This process is repeated until saturation is reached.

While engaging with the process of open coding the researcher should also be noting down any reflections and thoughts that they have on the data under consideration. This process of *theoretical memoing* is discussed further in section 3.3.3 below.

3.3.2 Theoretical Sampling

When carrying out research it is often neither possible nor practical to collect data about the population that one is studying; indeed it may not even be possible to identify all members of the population one is considering. To overcome these limitations researchers often work with a subset of the total population and this subset is selected based on the chosen sampling methodology. The type of sampling used will depend on the context of the problem and the analysis performed on the data; it can vary from a sample drawn completely at random to one which is carefully chosen for the potential richness of the data it may yield. The sample is then studied in detail and, based on the results obtained, inferences are drawn about the behaviour of the population as a whole. The way in which a sample is drawn differs greatly depending on the context: quantitative sampling strategies are often random in nature while qualitative sampling tends to be purposeful in the sense that the researcher seeks out information-rich cases to study in detail.

Qualitative sampling techniques have been described in the literature as "selective", "purposeful" or "theoretical" [164]. Selective (or purposeful) sampling is carried out prior to the commencement of a study where subjects are chosen based on a set of preconceived, reason-based initial criteria. The theory that begins to emerge from this initial sample leads to subsequent theoretical sampling where the researcher seeks out specific subjects from which they can learn more about the key characteristics and dimensions of the emerging categories.

When applying the Grounded Theory methodology the researcher is engaged in a highly iterative process, constantly moves between the phases of data collection, coding and theory generation. The collection of data and its analysis form a cyclical process where the researcher must "decide what data to collect next and where to find them, in order to develop his theory as it emerges" [93, p.45]. This is at the heart of the theoretical sampling process [93]. Strauss and Corbin [204] tell us that the purpose of theoretical sampling is to identify people, places or events that will provide the researcher with the best opportunities to capture variation amongst the concepts. Thus the process of theoretical sampling leads the researcher to develop the properties and dimensions of each category, so that that are as dense and rich in meaning as possible.

It is highly unlikely that any researcher will generate a substantive grounded theory through a single data collection phase; rather their analysis of the initial data collected will cause them to identify possible gaps in their data and omissions in their nascent theories. This will require them to go back to the field to collect data that can provide them with a deeper insight into the emerging categories. The aim of this process is to refine concepts, and not simply to increase the size of the original sample. This may require the researcher to return to selected individuals that they have interviewed previously. As Charmaz puts it, theoretical sampling "helps us to tease out less visible properties of our concepts and the conditions and limits of their applicability" [45, p.519].

It is very easy for the researcher to become too caught up in capturing the "perfect" sample for their study; in reality this can be very difficult, if not impossible, to obtain. Strauss and Corbin [204] caution the researcher to be pragmatic in terms of sampling and point out the many physical world constraints that the researcher may encounter, including the availability of suitable resources, time limitations and access to subjects or participants. They recommend that one should not become fixated on gathering data from a particular individual or scenario, rather they advise that the researcher should "make the most of what is available to him or her" [204, p.210].

At some point the researcher will make an intuitive, but informed, judgement to stop collecting data as the category under consideration can not be further refined or embellished. The category is said to be theoretically saturated at this point. This decision may be made during the process of theoretical sampling where the researcher realises that the data collected does not extend or refine the category in question. The researcher must take care that they do not limit the scope of the theory they are generating based on premature theoretical sampling. For example, in a study where all initial participants comment negatively about a particular item then it may be that theoretical sampling leads the researcher to seek out others with negative opinions to gain a greater appreciation of what leads them to make the negative comments. While this is exactly what theoretical sampling is about, the researcher should also be seeking out participants with a favourable opinion of the item. e.g. if all students interviewed comment negatively about computer programming then theoretical sampling may lead the researcher to only question those with negative opinions on programming, thereby missing out on the students who have favourable things to say about computer programming. Charmaz cautions the researcher against "premature commitment" to the emergent categories in the data . She advises the researcher to "fully explore the issues, events and meanings within the research problem, or setting" [44, p.1164]. This relates back to the concept of constant comparison and analysis as the researcher must be fully engaged with the data in order to develop a complete awareness of the dimensions of the emergent categories.

3.3.3 Theoretical Memoing

Glaser states that theoretical memos "are the theorizing write-up of ideas about codes and their relationships as they strike the analyst while coding" [94, p.83]. They capture the researchers thoughts and ideas about all aspects of the data. Memos may:

- articulate the researchers ideas about the data,
- chart a path for future analysis,
- suggest possible new categories to consider and their properties,
- indicate directions to consider for future theoretical sampling,
- capture relationships that might exist between emerging data categories,
- link emerging categories to the existing literature in the field,
- be reflections on the data as a whole, providing a basis for theory generation or refinement.

Memos can vary in form from quick notes jotted in the margin, to written text or diagrams. Memos constitute a resource in, and of, themselves and memoing should take place throughout the open coding phase of the research. Indeed, Glaser encourages the researcher to *always* interrupt coding to memo any thoughts or ideas they may have. Strauss and Corbin [204] also advise that memoing should be done after each round of data collection, as these memos capture the researcher's instantaneous view of the important issues at that point in time.

Memo writing helps the researcher to bridge the gap between the generation of codes and the articulation of their emerging theory. It can be viewed as an extended internal dialogue, where the researcher explores the implicit and explicit features of the data [44]. As Charmaz puts it: "Through memo writing, we elaborate processes, assumptions, and actions that are subsumed under our codes. Memo writing leads us to explore our codes; we expand upon the processes they identify or suggest." [45, p.517].

Gasson [83] cautions the researcher about the use of memos and how it is essential that they be verified through the further analysis of the data and of the relationships between the emerging codes. He reminders the researcher that: "At the end of the day, theoretical insights must be supported by further data analysis, or there is no theory – just speculation." [83, p.83].

3.3.4 Axial Coding

While initial codes are generated during the open coding phase, the researcher must gradually more towards the generation of categories, and theory, from these codes. Glaser [87; 94] felt that this should be achieved through constant comparative methods, while Strauss and Corbin[200] state that this should be achieved through axial coding. The latter requires the researcher to explore possible connections between coded elements of the data, categories and subcategories. These links can be in terms of the context of the category, the circumstances that give rise to it or possible consequences that may follow from the existence of the category.

The term "axial coding" is used because coding pivots about the axis of a given category [204]. Strauss and Corbin [204] view it as a fundamental cornerstone of any study involving grounded theory, however Charmaz[47] cautions researchers not to be too highly structured in their approach to axial coding and indeed suggests that it can be considered optional.

Strauss and Corbin [204] provide the researcher with a paradigm to use in the generation of their theory, whereby the researcher considers a phenomenon (i.e. a higher level category) from a number of perspectives, including its context and its relationship with its subcategories. As axial coding progresses, substantive theories emerge through a combination of inductive and deductive reasoning, with a strong emphasis on the identification of causal relationships[30]. Borgatti [30] provides the researcher with a basic framework, reproduced in table 3.1, which can be used to articulate generic relationships when engaged in axial coding.

Element	**Description**
Phenomenon	This is what in schema theory might be called the name of the schema or frame. It is the concept that holds the bits together. In grounded theory it is sometimes the outcome of interest, or it can be the subject.
Causal conditions	These are the events or variables that lead to the occurrence or development of the phenomenon. It is a set of causes and their properties.
Context	Hard to distinguish from the causal conditions. It is the specific locations (values) of background variables. A set of conditions influencing the action/strategy. Researchers often make a quaint distinction between active variables (causes) and background variables (context). It has more to do with what the researcher finds interesting (causes) and less interesting (context) than with distinctions out in nature.
Intervening conditions	Similar to context. If we like, we can identify context with moderating variables and intervening conditions with mediating variables. But it is not clear that grounded theorists cleanly distinguish between these two.
Action strategies	The purposeful, goal-oriented activities that agents perform in response to the phenomenon and intervening conditions.
Consequences	These are the consequences of the action strategies, intended and unintended.

Table 3.1: Borgatti's Basic Frame of Generic Relationships (source: [30])

3.3.5 Selective Coding and Theory Building

The open and axial coding processes described above lead to the development of categories that capture descriptions of the data. However, they do not provide the researcher with the theory they are seeking. As a step towards the articulation of this theory, the researcher must seek to identify the core categories which lie at the heart of the emergent theory and "explain most of the variation in a pattern of behavior" [87, p.75]. The process of integrating and refining categories to form a theory is known as selective coding [204]. This form of coding is more directed and more concept driven than the coding processes that precede it.

The researcher may find themselves moving between the different stages of coding, memoing and theoretical sampling as they seek to understand individual core categories. The ultimate aim is to have one core category to which all other categories can be related, sometimes called the "storyline" [204]. This core category forms the basis for "a single storyline around which everything else is draped" [30]. This storyline should be sufficiently rich and complex, such that it embraces all of the categories identified as part of the axial coding process.

3.4 Diverging Methodologies

In the early 1990's the co-originators of the Grounded Theory methodology engaged in a very public disagreement on how best to carry out research using Grounded Theory. This difference of views arose from the publication of Anselm Strauss and Juliet Corbin's book. "Basics of Qualitative Research: Grounded Theory Procedures and Techniques" [201]. Strauss and Corbin (a former Ph.D. student of his) had written their book in answer to demands from students for a manual to assist them in carrying out studies using grounded theory. The book contained very detailed guidelines and procedures on how to carry out such a study; including the introduction of axial coding, as discussed in subsection 3.3.4 above.

Glaser was so incensed that he initially requested that Strauss and Corbin withdraw their book. When this did not happen he responded by articulating a detailed counter-argument in his own book "Basics of Grounded Theory Analysis:

Emergence vs. Forcing" [87] where he made the following, withering comment on Strauss and Corbin's work: "It distorts and misconceives grounded theory, while engaging in a gross neglect of 90% of its important ideas" [87, p.2]. Indeed Glaser found fault with many elements of the Strauss and Corbin approach, some of these were substantial in nature and others were of more minor significance. These hotly debated differences between the originators lead to the establishment of two distinct schools of Grounded Theory: the Glaserian School and the Straussian School [197].

The differences between the two Schools manifest themselves from the very beginning of any study: Glaser encourages the researcher to approach the study with an empty mind, while Strauss feels they should have a general insight of the area under consideration. Glaser's concern about Strauss's view is that it will lead the researcher to impose ideas that are known from the literature on the coding of the data. We can put these differences into perspective if we recall that the underlying aim of any Grounded Theory study is that the researcher engage in theory generation. This can involve either the creation of a new theory or the extension and refinement of an existing theory. Both Schools emphasise the fact that Grounded Theory is not a methodology to be used for theory verification or falsification. In their original work Glaser and Strauss indicated that the researcher must have a perspective on the data that will enable them to abstract meaningful categories from it [93]. Dey [70] captures the essence of what Glaser and Strauss are both seeking when he discusses the difference between an "open mind and an empty head".

The notion of what constitutes a research question is another point of difference between the two Schools. Glaser believed that the research question emerges from open coding and theoretical sampling and that "the research question in a grounded theory study is not a statement that identifies the phenomenon to be studied" [87, p.25]. Strauss and Corbin take a very different view: while they feel that the research question will be typically broad and open-ended, its existence helps the researcher to maintain their focus in face of the significant volume of data they amass. Charmaz [44] relates this back to the issue of theory generation vs theory validation by commenting that the researcher uses a general research question as their starting point, rather than a pre-conceived hypothesis.

In Glaser's view Strauss and Corbin's step-by-step methodology is far too narrow and restrictive. He expressed very strong reservations about shoe-horning data into a restrictive, preconceived mould; arguing that it would greatly restrict the articulation and development of any emergent theories. Glaser advocates the use of neutral questions for the elicitation of theory and argues strongly against "forcing the theory" with structured questions. He disagreed with the tools that Strauss and Corbin provided to help researchers conceptualise their data (a coding paradigm and "conditional matrix"). He argued that the idea of forcing data to conform to a given paradigm runs counter to the emergent nature of Grounded Theory [87].

Strauss and Corbin's coding paradigm suggests researchers establish categories and their relationships by looking at "context, conditions, action/interactional strategies, intervening conditions and consequences" [213, p.362]. Glaser considered this to be too limited and narrow as it is only one way of considering a phenomenon and he refers to this as a "conceptual description". Glaser [87; 94] outlined eighteen possible coding families that may be used to establish categories and their relationships e.g. mutual effects and reciprocity, dimensions and elements, and he noted that other such frameworks may exist. Glaser cautioned against the use of the Strauss and Corbin model as "... the more the analyst practices the use of this model, the more he will exclude forever the ability to respond to any theoretical code that may emerge and become relevant. He will always just see a condition or consequence irrespective of relevance and stake his professional identity on it" [87, p.61]. Glaser believed that the Strauss and Corbin methodology would lead the researcher to focus on describing situations and incidents, rather than on the necessary "constant comparative work to generate conceptual properties of the category" [92, p.21]. Glaser felt this would give rise to a lack of analysis, which would in turn hinder the development of a conceptual theory.

Glaser believed in a coding process that involved incremental levels of abstraction (open, selective and theoretical coding), while Strauss and Corbin outlined four, more prescriptive, coding steps (open, axial, selective and "coding for process"). Glaser argued that the participant's worldview should emerge naturally from the data and its analysis, and that the process by which the theory emerges is of little significance. This was summed up well by Babchuk [6] who described it as a "laissez-faire" approach. In short, Glaser is concerned about developing a theory that is revealed by, and grounded in, the data, rather than one that is interpreted from a structuring of the data using a prescribed format.

Strauss and Corbin's methods and tools are intended to assist the researcher in the development of their theoretical sensitivity to the data. Glaser strongly opposes the use of these tools, arguing that theoretical sensitivity should come about through the researcher's immersion in the data and constant engagement in the process of comparative analysis.

From the above discussion it is clear that before commencing any study involving Grounded Theory, researchers must strive to make themselves fully aware of the two conflicting views of the methodology. It is only then that they will be in a position to articulate which variation of grounded theory they are using and be in a position to justify any choices they have made. One of the key initial challenges they face is in trying to determine which of the two Schools they should follow, Glaserian or Straussian. While the Strauss and Corbin paradigm model may initially seem very attractive due to its formulaic nature, many find its rules more restrictive than helpful [150]. Indeed Strauss and Corbin themselves were willing to accept that the methodology would evolve over time [45]. Hence it is not sufficient for the researcher to say that the Grounded Theory methodology was adopted, rather one must clearly articulate the procedures and processes used during the course of the study. As Goulding puts it "preferences regarding the version adopted should be stated to avoid confusion over terminology and procedures" [96, p.57].

At its simplest, Grounded Theory can be viewed as a set of flexible strategies for coding data which in turn lead to the emergence of theory. Grounded theory begins with the collecting and coding of data, regardless of the school of thought that the researcher subscribes to. These codes are then compared and categorised, which may lead to further data gathering in the form of theoretical sampling. This in turn leads to the determination of a core category or categories and the generation of a theory. Both Schools are seeking a theory that is grounded in the data collected, rather than one that arises from the opinions and biases of the researcher involved. As Heath puts it "the aim is not to discover *the* theory, but *a* theory that aids understanding and action in the area under investigation" [104].

Many have found that, in their practical experience of the application of the grounded theory methodology, they do not adhere rigidly to one School or the other [45; 127; 219] . While researchers carefully adhere to the methodological practices of one School, their awareness of the other School's practices and beliefs can prove beneficial. Adherents of the Glaserian School can find it easy to get lost in the data [127] and may find that the Strauss and Corbin methodology helps them gain some initial appreciation of the data upon which they can build using Glaserian process of constant comparison. Whichever method is used, the "depth and usefulness of the theory will depend on the expertise of the researcher" [127, p.756] and on the trustworthiness of the data.

3.5 Philosophical Paradigms

A philosophical paradigm is described by Strauss and Corbin as a "worldview that underlies and informs methodology and methods" [205, p.1]. As noted in Chapter 2, there are many philosophical research paradigms in existence (e.g. positivist, post-positivist, constructivist, post-modern), all contending for "legitimacy and intellectual and paradigmatic hegemony" [138]. Each paradigm has an associated set of methodologies for carrying out research; underpinned by a set of philosophical assumptions and principles [7]. A number of these paradigms have had a key influence on the development of the "Family of Methods" [38] that form the basis of the grounded theory methodology. Hence any researcher wishing to conduct a study using grounded theory needs to have an appreciation and awareness of the philosophical paradigm from which the methods they intend to use have arisen. In this section we will review some of these key paradigms and discuss their relationship to the grounded theory methodology.

There is "considerable disagreement and debate with regard to the underlying philosophical assumptions of grounded theory" [213, p.362]. In its initial formulation it has been classed as positivist/post-positivist or constructivist, while more recently it has been re-cast in social constructivist or post-modern paradigms. These classifications are now explored in more detail.

In the title of original work [93] Glaser and Strauss refer to "the discovery of grounded theory". They argue that, with their methodology, theory emerges from the data and that this theory is substantive in nature as it is grounded in the data. The structured methods of qualitative analysis that they espouse place it firmly in the realm of a post-positivist paradigm. Reflecting over forty years later, Glaser stated that the development of grounded theory was a "response to the extreme violation brought to data by quantitative, preconceived, positivistic research using forcing conjectured theory" [91, p.6]. Glaser [87; 94] bases his School of Grounded Theory on an assumption of an objective, external reality. The researcher is considered as a neutral observer who discovers the data and uses reductionist inquiry to consider manageable research problems though an objectivist rendering of the data [45].

As the Straussian School of grounded theory grew out of the original exposition [201; 204], it starts from a similar positivistic position with the same assumptions of an objective, external reality revealed though the collection of unbiased data. Strauss and Corbin provide a set of technical procedures for the analysis of this data and outline verification procedures that researchers should adopt. Their methodology is more post-positivistic in nature as it relies heavily on these highly structured procedures. However, it can be argued that they adopt a more constructivist stance as they view their methodology as a way of giving a voice to participants and establishing the participants view of reality. Moreover they recognize that there is an art, as well as science, in the analytic processes [45].

Grounded theory may be used to study people in their natural environment and so Charmaz [45; 47] argues that it can be viewed as a social constructivist setting. She places the emphasis on the emergent, constructivist elements of the methodology and considers it as "a set of flexible, heuristic strategies rather than as formulaic procedures" [45, p.510]. This paradigm creates a more natural setting for the use of grounded theory in explorations of the links between research and practice in specific sociological settings. Charmaz views herself as offering "an interpretive portrayal of the studied world, not an exact picture of it" [47, p.10]. Clarke [51; 52] re-casts grounded theory using a post-modernist situational approach. Her focus is intended to shed light on social situations, rather than on basic social processes. She sets out a cartographic situational analysis methodology to "open up the data". Like Charmaz, her variant of grounded theory is designed to accumulate knowledge through the interaction of socially involved researchers and participants.

Grounded theory can be considered through the lens of many philosophical paradigms. The challenge for researchers is to determine which variant of grounded theory to adopt in the course of their own work. It may be that Glaser captures it best when he contends that grounded theory is "paradigmatically neutral" [91]. Charmaz argues that this neutrality makes grounded theory useful for mixed-paradigm research [46]. The variant to use should be guided by the needs of the research, rather than the individuals paradigmatic bias. Glaser tells us: "My bias is clear, but this does not mean I rubber stamp 'ok' or indite any method. The difference in perspectives will just help any one researcher decide what method to use that suits his/her needs within the research context and its goals for research" [91, p.2].

3.6 Trustworthiness

When carrying out a quantitative study researchers describe the experimental methods and quantitative metrics they have used to test the hypotheses under consideration. The focus may be on the choice, or development, of suitable metrics to provide an insight into the effectiveness or quality of the intervention being considered or on establishing a causal relationship between variables. Numerical information is gathered so that it can be easily quantified and summarised. Well-known statistical methods may be used for analysing the data or specialist mathematical techniques may be developed for this purpose. In short the process is about gathering numerical data that back up or repudiate any claims that have been made. The metrics used are chosen in advance of the study and are clearly articulated to enable experiments to be easily replicated, thus allowing others to check the validity of any results presented.

The concept of trustworthiness in qualitative research is equivalent to that of validity in quantitative work. The trustworthiness of the results of any study are usually discussed in terms of their validity, reliability and generalisability, referred to by Kvale [129] as the "scientific holy trinity". However, these derive from a more classical positivist approach to research than that associated with interview-based qualitative research methodologies, such as grounded theory.

There is no fixed, universal definition of validity in the context of a qualitative study, rather it is "a contingent construct, inescapably grounded in the processes and intentions of particular research methodologies and projects" [231]. Qualitative research usually requires the researcher to return to the individual participants to seek validation of individual interview transcripts as a reflection of the participants views or experiences. It may also require participant validation of the interpretations that the researcher has placed on the data [180]. Participant validation is intended to counter any researcher bias or subjective interpretation that may occur during data analysis. However, it is problematic in that it can yield yet more data which in turn needs to be analysed. The cyclical nature of this process limits the extent to which one can ensure the accuracy of the final research outcomes [180].

In the grounded theory research methodology these potential problems are overcome by the fact that this type of validity checking is an integral part of the methodology; embodied in the processes of constant comparative analysis and theoretical sampling. These processes involve the constant checking of concepts, categories and emergent theory to ensure that it accurately reflects the participants' experiences. Triangulation of data involves the use of multiple data sources to maximise the range of data that contributes to the researcher's understanding. The constant comparative methods used in a grounded theory rely heavily on triangulation through the availability of multiple sources of data, both from individuals and groups of participants, as well as from different sources.

Validity may also be considered in terms of communication and pragmatic validity checks [129]. The communicative validity check is centred on the interpretations that have been placed on the data by the researcher. Both the research methods used and the interpretations placed on the data to generate the theory are open to question. Their validity is checked through dissemination amongst the research community. The pragmatic validity check relates to the research outcomes, in particular it focuses on their usefulness and their contribution to the understanding and knowledge of the intended audience.

In qualitative research the concept of reliability is captured by the notion of repeatability or replicability [180]. Replicability is checked through external checks by two or more researchers. If they arrive at findings similar to those of the original researcher when faced with the same data then the original results are considered reliable.

Two forms of reliability that apply to interview-based research methodologies are [129] the coder reliability check and the dialogic reliability check. The coder check is linked to the coding of the data and it involves two or more researchers coding all or a sample of the interview transcripts and then comparing the resultant concepts and categories. This is closely related to Lincoln and Guba's concept of dependability [136]. The dialogic reliability check involves a discussion amongst researchers where they mutually critique the data and each other's interpreted hypotheses.

The concept of generalisation refers to the applicability of the findings of one study to a different setting. This concept is closely assigned with that of transferability, as articulated by Lincoln and Guba [136]. Three forms of generalisability for qualitative research are presented by Kvale [129]: naturalistic, statistical and analytical generalisation. We now look at these in more detail in the context of a study using the grounded theory methodology.

Naturalistic generalisation derives from a person's tacit knowledge of how things are based on their experience in the field (e.g. riding a bicycle is considered a form of tacit knowledge). This tacit knowledge leads the researcher to expectations of how things ought to be; however by its nature tacit knowledge is difficult for one person to communicate to another and this form of generalisation is achieved through the articulation of the emergent theory and the dissemination of results to the wider research community.

Statistical generalisation is more quantitative in nature and is based on the use of inferential statistics. It allows the researcher to establish a confidence level at which they can infer general properties of a population based on those of a sample drawn from that population. This sample is usually drawn at random, or using a known probabilistic sampling frame. In grounded theory the focus is not on the selection of participants at random: theoretical sampling leads the researcher to deliberately seek out participants that can help refine or extend the emergent theory. The aim of a grounded theory study is to develop a theory which is grounded in the participants' experiences and beliefs and so the focus is on capturing the range of experiences and beliefs within the population.

Analytical generalisation refers to a researcher's reasoned judgement about the extent to which the findings of one study can be used to explain what may happen in another setting [129]. This relies on the descriptions and detail provided by the researcher in the original study. The study's participants, methodology, interpreted results and the emergent theory must all be set out clearly and succinctly; so that it may be readily translated and applied in other settings. The level of transfer and applicability of the study's outcomes is determined by the researcher in the new setting and it is up to them to demonstrate the level of generalisation that is appropriate for the new setting under consideration.

The researcher should maintain their dataset in such a way that others can review it and see how the findings were obtained; this is known as the confirmability of the results. The dataset should include not only the data gathered, but also contextual detail and description. This enables others to view how the raw data gathered was linked to the outcomes of the study [164]. The rich contextual information provided in the dataset should also enhance the dependability of the results and allow others to compare the settings of their work with that of the researcher, thus allowing for the data to be used to enhance the trustworthiness of other studies. This concept is referred to as the confirmability of the study by Lincoln and Guba [136].

Trustworthiness is strengthened through the process of bracketing, described in section 3.3.1 above. This requires the researcher to be aware of their own preconceptions, knowledge and beliefs, they should then suspend these so that they can immerse themselves fully in the participant's world [116].

A number of additional concepts that strengthen the credibility and trustworthiness of the data are articulated by Lincoln and Guba [136] . The first of these is prolonged engagement which captures the amount of time spent on the project, both in the field and with the data. The researcher can thus build up a rapport and trust with the participants which increases the likelihood that the information they provide is useful, accurate and rich. It also provides the researcher with time to build up their understanding of the population under consideration, including their language, culture and views.

Another concept that Lincoln and Guba discuss is that of persistent observation: "The purpose of persistent observation is to identify those characteristics and elements in the setting that are most relevant to the object being studied and focusing on them in detail" [136, p.304]. In short, prolonged engagement provides scope while persistent observation provides the depth of a study [116; 136].

The trustworthiness checks detailed above were applied in this course of this work and in Chapter 7 we will look at additional measures to establish the generalisability, reliability and validity of the findings of this study.

3.7 Conclusion

This chapter has provided a comprehensive description of the grounded theory methodology, from its origins in the seminal 1960's work of Glaser and Strauss [93] to the development of the Glaserian and Straussian Schools in the 1990's. The processes involved in the generation of a theory that is "grounded" in the data have been documented: from the initial collection of data, through successive levels of coding and abstraction to the emergence of a grounded theory. The reasons why the researcher must make the reader fully aware of the methodology they have adopted in their analysis have been articulated and strategies for ensuring the trustworthiness of the emergent theory have been considered. In the following two chapters the application of these methods to the collection and analysis of data during this study are described.

Chapter 4

Data Collection

Before data collection and analysis commenced, the researcher had to first gain an appreciation and awareness of the grounded theory methodology and its philosophical foundations, as detailed in chapter 3. This methodology is not prescriptive when it comes to data collection; indeed the researcher is encouraged to gather data from multiple sources using a wide variety of methods as this helps to ensure the trustworthiness of the results obtained.

In adherence with the grounded theory methodology, the researcher actively avoided the use of leading or theoretically motivated questioning. Participants were encouraged to put forward their own ideas, attitudes and experiences; thus data was gathered using the participants' voice, rather than that of the researcher. The underlying objective was to avoid Glaser's criticism of "theory forcing" in the subsequent analysis of the data gathered.

Interviewing is viewed by many as an art, as the conduct of each individual interview often reflects the style and personality of the interviewer. However, this is a rather narrow viewpoint, as many of the techniques and skills needed for the conduct of successful interviews can be learnt. For example, the interviewer can learn how to formulate and structure their questions so that they adhere to the central tenets of the grounded theory methodology. Another key skill required for the conduct of successful interviews is that of listening; the interviewer must train themselves not to interrupt and to actively listen to what the participant is saying. This active listening takes place on at least three different levels [181] and these are explored in more detail below.

While the bulk of the data used in this study was collected through the interviewing of participants, data was also obtained from focus groups and from the use of journals. In the following sections we discuss the types of data collected and detail the data collection practices adopted during the course of this study.

4.1 Collecting Data

The starting point for any grounded theory based study is the collection and analysis of an initial data set. Creswell [63] summarises the Strauss and Corbin methodology [204] for data collection in the table given below (Table 4.1) and this was used to inform the decisions made prior to the collection of data.

Data Collection Activity	**Grounded Theory**
What is traditionally studied? (site/individual(s))	Multiple individuals who have responded to action or participated in a process about a central phenomenon
What are the typical access and rapport issues? (access and rapport)	Locating a homogeneous sample
How does one select sites or individuals to study? (purposeful sampling strategies)	Find a homogeneous sample, a theory based sample, a theoretical sample
What type of information typically is collected? (forms of data)	Primarily interviews with 20-30 people to achieve detail in the theory
How is information recorded? (recording information)	Interview protocol, memoing
What are common data collection issues? (field issues)	Interviewing issues (e.g. logistics, openness)
How is information typically stored? (storing data)	Transcriptions, computer files

Table 4.1: Creswell's Overview of Data Collection (source: [63, p.113])

It was decided that data should be gathered from multiple individuals across the freshman year of an undergraduate computer science degree program. The sample was homogeneous, as all participants had been engaged in pursing the same degree program. In total 19 participants were interviewed, while a further

6 engaged in focus group discussions. All interviews and focus groups meetings were recorded and this data was transcribed prior to analysis. All data was stored securely in accordance with the University's ethical policies. The time frame for data collection was a significant logistical factor, as it was essential that participation in the study should cause minimal disruption to the participants' studies. The researcher was on leave during the period that the data was collected and had not been involved in teaching any of the participants who took part in the study.

4.1.1 Interviews

Creswell [63] places the interview at the heart of any grounded theory study, arguing that all other forms of data collection are secondary to it. Indeed, interviews form the backbone of almost all qualitative research methodologies, although the role of the interviewer in such studies can vary significantly – from that of a detached observer to an active participant in the creation of knowledge. Seidman [181] makes the purpose of interviewing clear, arguing that it is "in understanding the experience of other people and the meaning they make of that experience" [181, p.9].

There are many factors that an interviewer needs to take into account both prior to, during, and subsequent to the interview. These include the way they listen and conduct themselves during the interview process, the setting and the way they interact with participants, the nature of the questions asked, the structure and design of the interview and how any data recorded is stored and used. These factors all affect the type and quality of the data gathered. These are discussed in more detail in the following subsections.

4.1.1.1 Common Issues in the Conduct of Interviews

In their treatise on interviewing, Fontana and Frey [79] consider a number of elements that are common to almost all studies involving semi-structured or unstructured interviews. These must be considered in advance of the conduct of any study in order to ensure the interviewer is well prepared for the work they are undertaking. These factors are discussed below in the context of the study detailed in this thesis.

The first point to be considered is access to the setting, and hence the group, one wishes to study. Fontana and Frey [79] cite examples where the researcher may have to change their mode of dress or conduct in order to be able to meet with possible participants: In one extreme example they refer to a study of those frequenting nudist beaches that required the researcher to undress and walk along such beaches. Fortunately, the setting for this work was not quite so extreme.

In the work detailed in this thesis the researcher worked with a student cohort who were all engaged on the same undergraduate degree program and with whom this author had no prior contact. This ensured that there was no conflict of interest between the researcher and the participants e.g. the researcher would not be involved in the examining and grading of any of the students during their studies and the participants would not already have an academic-student relationship with the researcher.

The second issue that Fontana and Frey raise is that of the language and culture of the respondents. While this can be a significant factor in studies where the interviewer comes from a different ethnic and cultural background to that of the participants, this was not the case in the study described in this thesis. The author grew up in Ireland and completed all of her formal education within the Irish educational system. She has worked as an academic in Ireland for over a decade and has annually lectured to first or second year undergraduate students. This means that she is extremely familiar with the educational background and language of the participants.

Another aspect of the conduct of interviews that Fontana and Frey [79] mention is how the interviewer decides to present herself to the participants. For example in this study the researcher could present herself as a faculty member seeking to study computer science students or she could seek to present herself more humbly as a researcher seeking to gather information that will help future generations of undergraduate students succeed. While the latter was the preferred option in this study, the researcher was concerned that she could be viewed as some form of "spy" seeking to gather information for use "against" students. To counter this concern it was decided to be as upfront about the nature of the study as possible and to emphasise that the research had no input into the grading of students on any module or course they were taking.

Participants were given full information on the nature of the study, its scope and expected outcomes. It was made clear to them that the purpose of the interview was not to grade their knowledge of the field and that there were no right or wrong answers. Rather, the aim was to gain a greater insight into the computing experiences they found benefited them in the completion of their studies.

The interviewer must also gain the trust of the participants, so that they feel comfortable talking about domains where they feel their knowledge or skills to be lacking or deficient. Participants may not be comfortable with their interview being recorded and so it is important that they are provided with information on the collection and storage of the recorded interview data. They must be assured that any comment they make would never be made available in any form that would be directly or indirectly attributable to them. Participants can also feel discommoded by the presence of a recording device, so it was important this be as unobtrusive as possible.

It was essential to establish a rapport with the participants from the very outset of the interview process, so that they are willing to share their real attitudes and opinions. This is particularly difficult in an interview situation that is completely alien to the students. The interviewer must do their best to make the students feel welcomed and a valued part of the process, letting them know that whatever information they shared, no matter how trivial it seemed to them, was important. Participants were also encouraged to bring up factors that they felt might be of relevance if these popped into their head, even if they were not directly related to the topic they were discussing at the time.

The collection of empirical materials is the final feature of almost all interviews noted by Fontana and Frey [79]. The nature of the grounded theory methodology is such that the interviewer is attuned to the collection of such data from multiple sources e.g. from interviews with, and observation of, the participants, from journals kept by participants as well as from other written sources. It was important that the researcher engage in note taking throughout the interview process, however the researcher tried to be as inconspicuous as possible in doing this so as not to interrupt the flow of the participants thoughts and responses.

While the above facts consider the interview process from the macro level, it is also necessary for the interviewer to reflect on the nature of the questioning they engage in and this is discussed further in subsection 4.1.1.3.

4.1.1.2 Awareness of the Interviewer

The interviewer must listen very closely to what the participant is saying, as they need to make sure they understand what they are being told and assess whether the information they are being given is as complete and precise as possible. While it is important that the interviewer does not interrupt the flow of the process, it is often necessary for them to ask follow-on questions that seek to gain a deeper understanding of the participants experience. Hence it may be necessary for the interviewer to keep notes to remind them to return to specific topics at a later point.

The interviewer must also be aware of what Steiner [196] terms the "inner" and "outer" voices of the participant. When speaking with their outer, or public, voice the individual remains conscious of their audience and so may be guarded in what they say and how they say it. This is often reflected in their choice of language, Seidman [181] illustrates this when he contrasts use of the word "challenge" with that of the word "problem": Challenge captures the "positive aspects of a participant's grappling with a difficult experience but not the struggle" [181, p.79]. A participant's "inner voice" is an articulation of their innermost thoughts and beliefs and is allied closely with Vygotsky's definition of inner speech: "Inner speech is to a large extent thinking in pure meanings" [218, p.149]. Interviewers should be sensitive to the language participants' use and, where necessary, find ways to encourage them towards the use of their inner voice to provide a more thoughtful elucidation of their beliefs and feelings.

While listening closely to the participant and keeping note of topics to return to later in the interview, it is also important that the interviewer remain aware of the process and of the passage of time. To this end, the interviewer must constantly reflect on the interview's progress, assess when a topic is close to saturation and look for cues in the participants responses to help move things forward. This is what Wengraf [224] calls "double attention", where the researcher must be both listening to what the participant says and probing for more information when necessary, while at the same time bearing in mind their own needs in terms of the topics that remain to be addressed and the time available

Throughout the whole process the interviewer must remain focused on the task in hand and seek to suppress their natural inclination to engage in a two-way conversation with the participant. The interviewer should try to limit themselves to interactions that help move the process forward e.g. questions that seek further elucidation, or questions that help nudge the participant towards discussion of other aspects of their experience. The interviewer can use the process of note taking and memoing to help them resist their natural urge to engage in conversation and so maintain their focus on the interview itself.

4.1.1.3 The Development of Interview Questions

One of the natural tensions that exists within any grounded theory based study is between the methods that are used to collect data and the "forcing" of data into a preconceived framework. It is important that the interviewer avoid posing questions in a manner that forces the participant towards a particular response. The interviewer must ask their questions in such a way that they establish the topic to be addressed without giving the participant any direction towards the nature of the response required or expected. An example of an open-ended question is: "Tell me about your experience with computers", this encourages the participant to give a full meaningful answer that draws on their own knowledge; it contrasts with the closed-ended question "Do you get on well with computers?" which may encourage the participant to provide answers which express satisfaction or dissatisfaction. Open-ended questions are often statements that encourage the participant to respond, rather than questions in the strictest sense of the word. For example an interviewer may ask a participant to "Tell me about your experience" or to "Describe the events that led up to you doing that" or "How did it happen that you came to study computer science?".

An interviewer must also avoid questions that presume the participant has an opinion on, or knowledge of, a particular subject e.g. "Tell me, what do you understand about wireless networking?". This question presupposes that the participant has learnt about wireless networking and has formed some understanding of this. Overly directive questions of this nature can lead the participant away from the discussion of topics and directions that enrich the quality of the data collected and hence of the theory that emerges. Directive questions should be avoided as they can result in the forcing of theory.

Charmaz [44] puts it well when she notes that "grounded theorists need detailed, vivid data on which to base their analyses" [44, p.1167]. Like Lofland [140], she comments that an interview may be viewed as a directed conversation, however she cautions the interviewer against being overly-directive, as that can "cut off the most interesting leads and rich data". Moreover, she notes that while loaded questions may sometime be appropriate when based on pre-existing knowledge, it is essential that novice interviewers do not unwittingly load assumptions into their questions.

While it is essential that the researcher does not engage in theory "forcing", it is often necessary for them to ask probing questions that deepen the response given. Patton [164] emphasises that probes should be conversational in nature and be asked in a natural manner as follow up to initial responses. He classifies probes under the following headings:

- Detail-oriented probes: "Who", "What", "Where", "When" and "How" questions used to obtain a more complete picture of the participants experience

- Elaboration probes: Used to encourage the participants to keep talking; these vary from silence, a simple nod or a quiet "uh-huh" to a more direct request for the participant to provide more detail.

- Clarification probes: Used if the participants response is ambiguous or maybe contradicts what they said previously. These questions are framed as a failure in understanding on the part of the interviewer, and not as a fault by the participant.

- Contrast probes: These are similar to clarifying probes, but they provide the participant with something to compare against e.g. "How does being a third level student compare with your experience of being a secondary school student?"

Probes are not only useful as a way to elicit more information and as a guide to the participant, they also provided the interviewer with a way to use their awareness to control the flow of the interview.

A suggested ordering for interview questions is given by Charmaz [44]. She classifies questions as (1) short face-sheet, (2) informational, (3) reflective, (4) feeling and (5) ending. The initial short face-sheet questions are intended to be neutral and factual, limited to the information necessary to provide the researcher with some background information on the participant e.g. their age or the subjects they studied in school. As mentioned previously the establishment of a rapport between the interviewer and participant is key to the acquisition of rich and meaningful data. The informational nature of the initial question posed should be non-threatening and help to set the participant at their ease. Once a rapport is established the interviewer can then move on to questions of a more reflective nature e.g. "How did your first experiences with computers affect you?", "What future career did you envision for yourself then?" and "How would you compare your relationship to computers and computing then with how you feel about them today?". These questions are intended to encourage the participant to outline their relationship with computers within as unconstrained a framework as possible. The interview may then continue into questions that ask about the participants' feelings and attitudes towards computers, particularly since they have begun their undergraduate studies in the field. The questions asked at the end of the interview are chosen so as to leave the participant with positive feelings about themselves and their studies. This should also include a closing question that provides the participant with the chance to have a final say on the topics being discussed e.g. "That covers everything I wanted to ask you about, is there anything you feel I've omitted or that you would care to add to the discussion".

4.1.1.4 Interview Design

In the above subsections features common to most interviews and the nature of interview questions have been detailed. The approach to be adopted by the interviewer must also be considered. Patton [164] describes three approaches to the collection of data via open-ended interviews:

- the informal conversational interview,
- the general interview guide approach,
- the standardized open-ended interview.

These approaches are characterised by the extent to which the interview questions are determined in advance of the interview.

The informal conversational approach requires the interviewer to generate questions on-the-fly based on their observation of, and conversation with, the participant. The interview should seem so like a natural conversation that the participant may not even realise they are being interviewed.

In the general interview guide approach, the interviewer draws up a set of issues or topics to be explored with each participant. These topics are decided in advance of the interview and the interviewer keeps track of the responses to make sure all topics have been touched on during the course of the interview.

The final approach Patton [164] described is that of the standardised open-ended interview. Here the interviewer draws up a set of questions that are carefully chosen so that each participant is led through the same sequence of questions. The interviewer is restricted by their inability to deviate from these questions, even if additional probing questions may help to elicit richer data.

From the discussions in chapter 3, it should be apparent that the standardised open-ended approach is unsuitable for use in any study that aims to make use of the grounded theory methodology. Similarly, the use of a fixed set of interview questions that cannot be deviated from would not permit the constant comparison methods that lie at the heart of grounded theory. In grounded theory, the interviewer needs to be able to lead the discussion into areas that have not been previously explored but which may be of direct relevance to the study [96]. Moreover, the generation of a grounded theory is a cyclical process where the researcher constantly moves between the gathering of data, to its analysis and codifying and on to the generation of theory. This constant analysis and re-analysis of data leads the researcher to identify gaps in their data or potential holes in the emerging theory. The researcher may then seek to address this through the inclusion of additional questions in subsequent interviews as part of the theoretical sampling process. It is only through these iterative processes that the researcher can develop the properties and dimensions of each category until they reach saturation.

The notion of theoretical sampling makes Patton's [164] informal conversational interview approach inappropriate for use as the sole methodology within a ground theory study. As analysis progresses, the researcher needs to frame their questions with a view to the saturation of any categories identified. However the informal conversational approach does allow the researcher to pursue topics of interest that arise during the course of the interview and so it is necessary that alternate approaches are adopted for some parts of the interview process.

Similarly Patton's [164] general interview guide approach allows the researcher to plan for the inclusion of topics that may help enrich their emerging categories and theory. However, this approach is clearly too restrictive in the initial phases of data collection where unpredicted responses from participants should drive the process.

Thus for a grounded theory study, such as the one described in this thesis, the approach adopted should be a combination of the general interview guide approach together with that of the informal conversational interview. This is termed a "directed conversation" by Lofland [140], where the interviewer directs the conversation depending on a number of factors including the participant's current physiological state, the rapport between the researcher and the participant, the researcher's awareness of the interview process, the researcher's theoretical perspective and the topic under consideration. As mentioned above, it is important that the researcher should not be overly-directive as this may reduce the richness and quality of the data collected.

4.1.2 Focus Groups

Focus groups are founded on some of the same principles as group therapy, as it is assumed that people with a common problem will feel more secure and willing to talk in a peer group setting where participants share common problems [132]. As a consequence, focus groups allow interviewers to study participants in a more natural setting than that of a one-to-one interview.

Patton defines a focus group as "an interview with a small group of people on a specific topic" [164, p.385]. He goes on to makes it clear that a focus group is not a problem-solving session, a decision making-group or a forum for a discussion amongst the participants. A focus-group enables the participants to hear what other members of the group have to say and gives them an opportunity to agree or disagree with what has been said and volunteer additional information they feel may be relevant. Thus the information provided by focus group participants is given within the context of the views of others in the group.

While focus groups are grounded in the assumption that participants within a peer group will be more willing to share their thoughts, feelings and experiences, there are also a number of disadvantages in the methodology. The presence of others may inhibit an individual from expressing their true feelings and thoughts. Moreover, pre-established group dynamics may cause some to be reluctant to contribute[132]. Kaplowitz [125] illustrates this in his quantitative study where he found that participants in individual interviews were 18 times more likely to raise socially sensitive discussion topics than those in focus groups. However Kaplowitz also established that these two qualitative research methodologies provide the researcher with complementary data [125]. Hence focus groups should be seen as a methodology for enriching any data set obtained from individual interviews alone. As constant comparative methods form a corner-stone of the grounded theory methodology, the use of varied and multiple data sources is an essential part of any study. This triangulation of data strengthens the validity of any emergent theory.

Lederman indicates that focus group participants are selected because they are a "purposive, although not necessarily representative, sampling of a specific population" [132, p.117]. This deliberate selection of participants is consistent with the concept of theoretical sampling, described in section 3.3.2. Theoretical sampling requires the researcher to identify participants that are most likely to enable them to capture the widest variation possible, as that will lead the researcher to a more richly developed set of properties and dimensions for any categories identified.

When considering the conduct of focus groups the term "moderator" is preferred to that of "interviewer" [164]. The moderator's role is to facilitate and focus the group discussion on the topics under consideration in a non-directive manner. The moderator must take care not to impose their own language or ideas on the group [211]. They must also take care to ensure that all group members are given the opportunity to contribute, so that dominant participants do not stifle the contributions of more passive group members. This needs to be done in a subtle manner so as not to alienate or embarrass group members. It can often be achieved by non-verbal means e.g. through a hand gesture, body language or non-directive questioning. The focus group should not turn into a set of simultaneous individual interviews and the moderator needs to encourage participants to comment on, and add to, contributions made by others in the group.

The nature of a focus group is such that the number of questions that can be asked is significantly lower than in an individual interview setting. Hence, if the purpose of the focus group is to seek confirmation of emerging categories or theory, then the researcher must prepare carefully in advance, determining the topics they would most like to touch on. Patton suggests that no more than ten major questions should be considered in a one hour session [164]. However, the moderator needs to be open to the fact that the group interaction may lead the participation into the discussion of other topics that provided rich and meaningful data.

One of the big drawbacks in the conduct of any focus group is that confidentiality cannot be assured. This may make some participants unwilling to volunteer to take part in group sessions. In the context of the study detailed in this thesis a student may not be willing to discuss any difficulties they have encountered as they do not wish to appear inadequate in front of their (possibly) more able peers. This can be mitigated in the way that focus groups are formed e.g. participants may volunteer to take part in individual interviews or in group sessions, also they may be asked to self-assess themselves into a number of broad categories in advance of any group sessions.

The literature offers much advice on the preparation for, and conduct of, group sessions [132; 164; 211] and this was reviewed by the researcher in advance of the collection of focus group data for this study.

4.1.3 Journals

As mentioned in section 3.6, the wider the variety of data sources used in any qualitative study, the more trustworthy its results are likely to be. In this subsection we discuss the use of journals as a means of data collection.

In their simplest form, diaries or journals are a chronological record of events that have occurred, together with a description of the impact of these events and the journal writer's reflections on them. The events discussed may be part of the personal life of the diary keeper or may be drawn from the world around them. More recently, written diaries and journals have often been replaced by the web log, or blog, where writers maintain an online sequence of observations or reflections that may be kept in date order. It should also be noted that journals need not be written, they may also be kept as voice or video recordings.

In qualitative research the nature of the diary one asks participants to keep will depend on the study being undertaken. For example one could ask participants to maintain a learning journal [156], a reflective diary [179] or a process journal [134]. In the case of a learning journals the participant is usually a student who is maintaining a record of their learning as it occurs. This contrasts with the notion of a reflective diary where participants reflect on events after they have occurred with a view to gaining an insight into their own practice and performance. The keeping of reflective diaries is not limited to the educational sphere, indeed many professionals, for example nurses [75] and teachers [80], and researchers are actively encouraged to keep such diaries. Process journals are student logs aimed at tracking their time, activity and thoughts as they work on particular problems or projects [134].

One of the advantages of the use of a diary is that the participant can note an issue or problem they encounter without the need to discuss it with someone else. This means they may be more willing to comment on their experiences, safe in the knowledge they will not be asked to elaborate further.

When asking participants to keep a diary or journal one needs to provide some guidance as to the nature and volume of data to be recorded. In this study participants were asked students to consider their studies and to document any prior computer experiences they had that they felt placed them at an advantage or at a disadvantage relative to their classmates; as well as the computer experiences they drew on that they felt they had in common with others in their student cohort. They were asked to add regular entries to their journal over the course of one week and they were also encouraged to provide any additional thoughts or feedback that they felt might be relevant to their studies. A copy of a suggested template to be used for the keeping of the journal is provided in Appendix C.

4.2 Sampling

As discussed in section 3.3.2, the grounded theory methodology makes use of the principle of theoretical sampling. This means that future data collection is determined by the categories of the emerging theory. Throughout the course of the study the researcher is constantly engaged in cycles of data collection, coding and analysis.

As the researcher concentrates on the development of the emerging theory, their analysis of existing data dictates what data should be collected next [93; 204]. The purpose of theoretical sampling is to assist the researcher in the selection of participants who will provide them with richer data on the categories of a phenomenon. This theoretical sampling must continue until no new categories are found i.e. saturation has been reached.

As they gather data the researcher must identify possible categories within their data and then refine and extend these through the collection of further data. This is achieved by constant comparison, where the researcher is constantly looking to tease out the properties or dimensions of a category [47]. This technique strengthens the validity of the categories identified.

Theoretical sampling is not about searching for repeatedly occurring properties or dimensions; rather it is about testing, extending and refining the categories identified. In this work three types of sampling were employed: open sampling, relational and variational sampling and discriminate sampling [201].

4.2.1 Open Sampling

In this study we are looking to uncover a theory that is grounded in the data collected. At the start of any study the researcher must aim to put aside any preconceptions they may have and look to the data collected for guidance. As there are no pre-existing categories to direct sampling, open sampling is employed. This is the "sampling those persons, places, situations that will provide the greatest opportunity to gather the most relevant data about the phenomenon under investigation" [201, p.181]. In this study this means that we look to students who can share information on how their computer experience has impacted on their undergraduate studies.

It is accepted that at the start of any study it is most common for the researcher to "go to the groups which they believe will maximize the possibilities of obtaining data and leads for more data on their question" [94, p.45]. The initial open sample for this study was based on the literature review, discussions with others familiar with the student cohort and the academic experience of the researcher. Prior programming experience is often cited as a factor that influences success on undergraduate computer science programs [16; 171; 228], while previous studies by the author had shown that mathematical proficiency can be a predictor of success in the chosen undergraduate program [192].

There was no initial presumption that "prior programming experience" or "mathematical proficiency" would be categories as no data had been collected for these to emerge from. Rather, it was accepted that these might be possible categories depending on the results of the data analysis. Many other factors have been proposed and considered from a qualitative perspective (e.g. Bergin and Reilly [15; 16] considered a mathematical model involving 25 attributes, while Cantwell Wilson and Shrock considered 12 factors in their model [228]), while some of these may emerge as possible categories in the later stages of the study, they were not used for initial sampling purposes.

A short student survey was used to gather some statistical data from each class group at the start of the study, and as part of this individuals were asked to indicate if they were willing to be interviewed at a later date. The responses to this survey were then used to help construct the initial data sample.

As mentioned previously the selected students were neither current nor former students of the researcher. This reduced the likelihood of a significant power imbalance between the researcher and the participants [224] and strengthened the trustworthiness of the data collected. It was expected that participants would feel able to speak in a more open and honest way as the researcher had no direct involvement in their academic affairs. Nevertheless it must be acknowledged that the participants were aware that the researcher was a faculty member on research leave, and so efforts were made to reassure them of the confidentiality and anonymity of their participation in the study. Eight students were initially selected according to their self-ranking of prior programming experience and mathematical ability, ensuring that the initial sample exhibited maximum variation across these two areas.

Once the initial data had been collected, data transcription and analysis commenced. Further sampling only occurred once this was complete. This is an essential part of the grounded theory methodology, as the next stage of the process involves the implementation of theoretical sampling.

4.2.2 Relational and Variational Sampling

In the initial phase of the analysis the researcher engages in open coding (see section 3.3.1). Here the researcher tries to make sense of the data by conceptualising it and identifying initial codes within it [204]. The researcher must later seek to use these codes to generate categories (and eventually an emergent theory). To do this the researcher must explore links between the coded elements of the data. This is achieved through variational and relational sampling in the axial coding stage (see 3.3.4). The purpose of this sampling is to not only differentiate between the categories determined but also to explore the relationships and dependencies between these categories. The processes involved in differentiating between the categories can lead to the discovery of these interrelationships [47]. The properties of each category should form a logical whole, so by ensuring that the categories are clearly differentiated the researcher is uncovering the scope of the emerging theory [94].

Relational and variational sampling is not just about the development and refinement of the initial codes and categories, it can also lead to the discovery of new categories. The analysis of the initial data collected suggested a number of codes and categories that may be of relevance to the study. These formed the basis for further variational or relational sampling. For example, the initial data suggested the category of "verbal ability" and the properties and dimension of this category were explored further in subsequent interviews.

4.2.3 Discriminate Sampling

Discriminate Sampling was defined in section 3.3.5 as the process of determining the central phenomenon to which all other categories can be related. At this point in the development of the theory the researcher is constantly re-visiting the data as they seek to identify the core categories. This may lead them to return to participants to gather further information and detail or to engage with new participants to gather new data so that existing categories can be further refined. At this point the researcher is engaging in a form of triangulation, as they are constantly comparing the data from a number of sources to validate and reinforce the emerging theory. In this study the researcher returned to three participants to gather more data and also engaged with a new focus group to help clarify the links between existing categories.

4.2.4 Theoretical Saturation

In the development of each category the researcher must reach a point where the collection of more data will not yield further insights; either into the category and its sub-categories or into its relationship with other categories. This is called the point of theoretical saturation [204]. At this point the researcher can end the data collection phase. This can cause difficulties for the novice researcher as, unlike quantitative methodologies, there is no way of determining in advance what sample size is required for any given grounded theory based study. There is also the inherent risk that the researcher prematurely declares that saturation of categories has been reached, perhaps because their analysis has been overly superficial and limited. Strauss and Corbin [204] are pragmatic in the advice they offer, accepting that continued analysis would almost always lead to the discover of additional properties or dimensions. Moreover, logistical constraints may mean that the researcher's access to participants is curtailed or, as Strauss and Corbin point out, it may be the case that "the researcher runs out of time, money, or both" [204, p.136]. However if the researcher adheres to the grounded theory methodology and does not force the theory, then their findings should yield meaningful results, even if real world constraints have impinged on the collection and analysis of data. This may mean that the researcher needs to accept that their study has not established the theoretical saturation of all categories. In this situation, the researcher is advised to present the saturated categories identified along with others that are not yet saturated and ear-marked for further research, all the while accepting that as the theory evolves the saturated categories may require further revision. As Strauss and Corbin put it "Sometimes the researcher has no choice and must settle for a theoretical scheme that is less developed than desired" [201, p.292].

4.3 Implementation and Logistical constraints

The success of any study depends on careful planning and preparation. As well as preparation for the conduct of interviews, it was also necessary to address a number of more practical issues such as the access to the potential pool of participants, the enlisting of participants and other logistical constraints.

4.3.1 Enlisting Participants

Obviously a study such as this cannot take place without a population of potential participants to sample from. This research was carried out in an Irish university. This university offers many courses with significant technological content ranging across engineering, mathematics and science. Any attempt to investigate computer experience across all such undergraduate courses would be futile as the students' extensive range of possible experiences could not be adequately covered within the time constraints imposed by a study such as this. Thus the sample for this study was drawn from a single degree program within a Department of Computer Science. The sampled students were all enrolled on the same undergraduate degree program in Computer Science. Although the computer experience of students drawn from other programs was not covered in this study, their experience could provide a possible area for further inquiry and for the discovery of more substantive theories in the future.

Once the student cohort had been chosen, potential participants had to be approached at an initial contact session. This gave the researcher an opportunity to introduce herself and build a rapport with potential participants. At this session the researcher briefed each student cohort for about 15 minutes on the nature and the purpose of the study. The students were provided with a printed outline of the project's aims and objectives, as well as any ethical issues related to the study. They were also given a form to complete if they were interested in volunteering to take part in future interviews or focus groups or to keep a journal. A short initial survey was also administered seeking some broad quantitative and qualitative data.

Ethical issues such as confidentiality and data protection were discussed and the students were given the opportunity to ask questions about all aspects of the study. The preliminary session also allowed the researcher to make it clear that the experiences of all students were of interest and value to the study, and that everyone was welcome to volunteer. It was also indicated that the research methodology was such that it might not be necessary to call on all of the volunteers to contribute to this work. For example, they were told that while 100 people may volunteer to take part, it may not be necessary, or possible, to interview all 100 individuals. The researcher also took the opportunity to reassure potential participants that the processes involved were not about exploring the extent of their knowledge but rather about identifying the computer experiences that people drew on during the course of their studies.

Students who were willing to volunteer to take part in the study completed the participation form and returned it to the researcher. They were also given the opportunity of taking this form away with them and returning it at a later date. The information provided to students and forms used are provided in Appendix A. As a result of these preliminary sessions a total of 27 students volunteered to take part in the study on the day and a further six students subsequently indicated their willingness to participate to the researcher by email. From this group 19 students took part in interviews, six students took part in focus groups and four students kept a weekly journal.

4.3.2 Logistical Constraints

As the participants in this study were undergraduate students there were clearly logistical and contextual constraints in relation to access. The study was carried out during the academic year as students are most easily available at that time. However, this meant that it was essential that participation in this study did not disrupt their studies and examination preparation. This had a particular impact on the later stages of the project where the researcher needed to be sure that theoretical saturation had been reached before participants became caught up in preparations for the main examination sessions.

Initial participant recruitment and preliminary data collection had to be carried out and analysed very quickly so as to allow for subsequent theoretical sampling to take place. Within each academic term there is an optimal window for carrying out interviews. There are usually more assignment and coursework deadlines near the end of each term and so data collection took place in the middle of each term. In order to give them some time to settle into university life, it was decided not to recruit first year student participants until after the first four weeks of their studies. It was also essential to ensure that all participants were over the age of 18, as otherwise it would have been necessary to seek parental consent.

4.4 The Interview Process

The interviews and focus group sessions carried out as part of the study detailed in this work were conducted using the methodologies discussed above. The researcher used the first few minutes of every session to describe the purposes of the study and the processes involved. The participants were talked though the consent form, see Appendix B, and were then given time to read and sign it. The researcher explained that the session was not a test or examination, rather its purpose was to gain a greater understanding of how their prior computing experiences impacted on their undergraduate studies.

The interviews were conducted in a quiet, sound proof location. Interviews were set up at a time that suited the participants e.g. when then had a break in their lecture schedule. The interviews varied in length, with most interviews taking between 50 minutes and one hour. All participants were given permission to decline the recording of their interview, however none elected to do so.

The students were told the first few questions they would be asked before the data recording device was switched on. These were extremely straightforward, information gathering questions chosen to help set the participant at their ease. Once the participant had settled into the interview process the questions then asked them to reflect on their computer experiences and the factors they believed were helping them succeed at their undergraduate studies. If the responses given were not very elaborate the researcher followed up with probes that encouraged them to provide more detail on the responses they had given. The initial stages of the interview took the form of a directed conversation. Appendix D below provides a list of questions that were drawn up in advance for use during this phase of the interview process.

The interviewer was always conscious that the pre-determined questions were only to be used as a general guide and that the responses given by the participants were expected to give rise to further probing for clarification on their point of view or more detail on concerns raised [229]. Where necessary, notes were made of issues to return to at a later stage in the interview. This meant that the interviewer had to adopt a flexible approach; consciously bringing the conversation back to the point under discussion or else allowing the participant to shift the focus to issues that are pertinent to them. It is important for the participant to feel that their contribution is valued, and picking up on issues that they raise is one way for the researcher to demonstrate this to them. This can help the researcher to build up the necessary rapport with the participant. The researcher found that it was useful to remind the participants that all of their experiences were relevant to the study, and where necessary they were asked to consider if there were any other factors they were aware of that influenced their experience.

As the data analysis progresses and categories start to emerge, it is important for the researcher to sample in a way that allows them to gain a deeper understanding of the properties of the category, see section 4.2.2. This will usually require the researcher to amend the questions they ask or to probe more deeply.

The researcher must remain sensitive to the feelings of the participant, for example, if the question posed highlighted an area where the participant was not very happy with their progress or slightly embarrassed by their lack of success then rather than asking them to explain where their problems lay, it might be more useful to ask them to couch their response in terms of the advice they would give to someone starting on the same course of study at a future date. Such third party questions allow them to address the same topic from a different perspective.

While each interview was recorded, the researcher also kept handwritten notes. These were used to note topics to be returned to later in the discussion and also to note any aspects of the participants' demeanour or physical behaviour that might be relevant e.g. if the participant sat with their arms folded or if they hit the table to emphasise a point they were making. These notes were used in the subsequent analysis of the data. As with the voice recordings, participants were informed of the data protection policy that was in place for the study and were assured of the confidentiality of both the voice recordings and written data.

The focus group sessions were carried out in a similar way to the individual interviews. There were two focus group sessions, both involving three participants. Each session lasted approximately 50 minutes. Given the nature of group interaction the number of questions that need to be asked at such sessions can be quite limited and in practice it transpired that no more than three distinct questions were needed, along with some careful probing of responses and some measures to ensure that all participants contributed equally. It was found that participants seemed more than happy to contribute to these sessions and the researcher did not get any sense that participants were being wary or reticent about volunteering information on their experience.

A weekly journal was kept by four participants in the study. It was found that the data gained in this way was not as rich as expected. This may be because the participants are not familiar with this form of journal keeping and also due to the pressures of term-time course work deadlines. At the outset the journal-keeping participants indicated they were willing to make regular entries during the week, however only one of the participants actually did this, while the others completed all their entries in one sitting at the end of the week. This meant that the journals did not capture significant day to day variation. The data gathered in this way was sparse and showed the importance of face-to-face contact where the researcher can probe for more detail if necessary.

The four participants provided journals that were mainly descriptive in that they focussed on a narrative history of the week's events e.g. on stating what lectures they had attended, the topics encountered and coursework attempted on a given day. The data provided in terms of the activities undertaken was often quite detailed; in marked contrast to the length and nature of the commentary provided when the participants were asked to reflected on how these activities drew on their prior knowledge and experience. The participants seemed to find it difficult to evaluate and reflect on how their engagement in computer related activities built on their prior knowledge and life skills.

The nature of all STEM disciplines is such that much communication is done through technical and laboratory reports where the emphasis is on factual communication. Great emphasis is placed on the precision of the experimental description and the accurate interpretation and presentation of any data gathered. Reports are used within academia to assess the students' ability to apply their knowledge to a practical task and within the workplace they form the basis for communication between professionals e.g. between a software engineer and those responsible for the implementation of their design. In this context, the difficulties encountered with the gathering of conceptual and reflective data through the keeping of journals is not so surprising as participants have been habituated to writing in a more factual, non-reflective manner.

This is also borne out by the literature on reflective writing where it has been noted that STEM students often experience discomfort with any form of reflective writing [101; 151; 220]. For example, in a study [33] involving over 100 first-year, distance education engineering students it was found that when asked to produce a reflective portfolio, 82% of students produced factual, summary style reports while only 2% of the cohort engaged in deep, meaningful reflection. The study concluded that deep reflection is not a skill that comes easily to Engineering students and that if these students are to successfully engage in reflection then extensive scaffolding and support are needed. There is an inherent risk in the provision of support and guidance to participants who are asked to keep journals as it may result in undesirable theory forcing in grounded theory. If the researcher is over-supportive then they risk influencing the viewpoints expressed by the participants. This was observed in the study reported on by Brodie [33] where qualitative analysis suggested that participants were writing what they thought the facilitators wanted to hear.

In a grounded theory study the fact that data can be collect via multiple methods means that participants can decide to contribute in ways that they feel are best suited to them. It was felt that some participants may not be willing to discuss issues affecting their performance in an open and frank way and that the keeping of a journal may be of more benefit to such students. However, it was found that participants who elected to keep a journal found it hard to fit this into their busy schedule. They also found it difficult to critically analyse and reflect on how their prior knowledge and experience, or lack thereof, influenced their learning. Consequently, the data gathered in this way did not play as large a role in this study as anticipated at the outset.

One of the reasons why grounded theory has become so popular is that the theory emerges from the collection of theoretically rich data. Multiple data collection methods provide the researcher with ways to cross-check the validity of their findings. It is expected that any weakness in one data collection method can be compensated for by the strengths in another [170], so even though the journal data collected was found to be less informative than expected this did not have an undue impact on the emerging theory.

4.5 Field Notes and Memoing

Field notes are written records of observational data recorded during data collection [155]. By contrast, memos can encompass all facets of the research process; from research planning, to the analysis and conceptualising of the data [204]. Memos can range from the researcher's musings, thoughts or feelings on the data, to more formal minute keeping or operational memos, written text or diagrams; however they all serve the same purpose in capturing the interplay between the researcher and the data collected.

Glaser [94] places the keeping of such memos at the heart of the grounded theory methodology, imploring researchers to make the writing of such memos a priority as they ensure the retention of ideas that might otherwise be lost.

Strauss and Corbin [204] classify memos into three distinct categories (i) operational memos, (ii) coding memos and (iii) theoretical memos. Operational memos are the researcher's record of decisions made in relation to research planning and conduct, along with the rationale behind these decisions. Coding memos are specific to the coding and categorisation of data and may be produced during the open, axial and selective phases of coding. Theoretical memos (see section 3.3.3) capture the researcher's thoughts and ideas about all aspects of the research e.g. they may record suggests for future data collection or links between observations and the existing literature.

Memos are subjective in nature as they reflect the researcher's understanding and interpretation of the data, however it is through the analysis of the data that any speculative hypothesis within memos can be validated and become part of the emerging theory. As the researcher draws on data collected from a wide variety of sources, memoing requires them to be theoretically sensitive to the meanings that can be attributed to the data. This forms part of the process of constant comparison that is central to the grounded theory methodology. As Birks et al. put it [24]: "Memos work alongside other sources of data such as transcripts and field notes to provide supportive documentation for a study".

Both field notes and memos can contain descriptive and interpretive components, however they are distinct concepts. Field notes record information occurring within the field, while memos abstract meanings from the field [155]. It is often the case that field notes and collected data are incorporated into written memos, but the reverse is seldom true [24]. Glaser [94] advises the researcher to avoid annotating memos within data sources, as this can blur the distinction between contemporaneous field notes and observations and the researcher's later reflections.

By their nature field notes are usually slightly disorganised and not fully developed, as they are often contemporaneously recorded sentence fragments in the margins of the researcher's notes. The researcher must all set aside time at the end of each data collection session to record additional field notes in the form of observations and impressions. By contrast while memos can also be "partial, preliminary and provisional" [47, p.84], they are kept in a more orderly and systematic fashion so that they are readily accessible at a later date [204]. Both field notes and memos become part of the data available to the researcher during the analysis phase of the study and are essential to the validation of the emergent theory.

4.6 Conclusion

The grounded theory methodology relies on the collection of theoretically rich data. The processes used to collect such data for this study have been detailed in this chapter. The researcher must first decide on the manner in which data will be collected and prepare for the collection of such data. These preparations will also include the selection and ordering of suitable open questions and probes. In this study data was collected using individual interviews, focus groups and journals. Sampling was carried out using the grounded theory methodology. Field notes and memoing complemented the recorded interviews and journal data collected. In the following chapter we review the processes involved in the analysis of this data and in subsequent theory generation.

Chapter 5

Data Coding and Concept Generation in Practice

In contrast to chapter 3 where the theoretical underpinnings of the grounded theory methodology were reviewed, this chapter concentrates on the processes involved in the hands-on analysis of the collected data. It begins with a description of the open coding of the data, where the data is initially analysed for the existence of codes. This is the practical implementation of the processes described in 3.3.1.

When the first analysis of the data is complete, the researcher must engage in the process of axial coding in order to determine the properties and dimensions of these categories. These processes reflect the requirements of axial coding described in section 3.3.4. The researcher also looks to form abstracted, high-order categories during this second stage of the analysis process and the methods used to achieve this are reviewed below. Finally, as the researcher seeks to reveal the underlying theory, they must engage in selective coding. Selective coding, as explained in section 3.3.5, builds upon the initial coding and abstraction processes to provide the researcher with a theory that is grounded in the data.

The purpose of this chapter is to provide the reader with a better understanding of how the grounded theory methodology was applied during this study, enabling them to more fully appreciate the underpinning processes that lead to the articulation of the phenomena and emerging theory in chapter 6.

5.1 Open Coding

Data coding and analysis begun immediately after the first data sample was collected. This followed the Strauss and Corbin recommended coding methodology involving word-by-word, line-by-line, sentence-by-sentence, paragraph-by-paragraph microanalysis [204] of the data to generate the initial codes. These codes were then reviewed to establish their emerging properties and dimensions. The relationships between these codes was also considered.

The data in this research consisted of recorded interviews and focus groups, together with a small number of participant diaries. The interviews were transcribed and analysed line-by-line. As each transcript was reviewed the researcher inserted short notes and comments in the margin of the document. Codes and categories may often be labelled using verbatim terms from the collected data, however as they are refined these labels may evolve to better reflect the properties and dimensions of the code.

The transcribed interviews were analysed using the comment and review features within Microsoft Word. Notes in the margin of each document included potential codes, possible dimensions and indicators of variation. The comment feature was also used to store the researchers thoughts and ideas about the data as theoretical memos (see section 3.3.3). The sections marked and their associated codes and memos were then extracted to a separate document for further sorting and aggregation. Each participant was identified by their associated anonymising identifier e.g. "Participant 8".

The coding process is very slow and laborious as the researcher must keep an open mind and avoid imposing their pre-existing perspectives on the data. While there will naturally be some elements of subjectivity in this analysis it is hoped that the inductive nature of theory generation, combined with researcher's theoretical sensitivity, will reduce its impact.

The grounded theory methodology provides guidelines that help draw the researcher's awareness to their own preconceptions, helping them to bracket this knowledge. The researcher's experience as an academic was both a help and a hindrance during the open coding of the data as it provided the researcher with a theoretical sensitivity to the subject matter, but also with many views and beliefs on the topics under consideration. It was important that the researcher regularly sought to set these beliefs to one side, focusing on what the data was suggesting. These constant reminders helped to reduce the impact of subjectivity during this phase of the analysis, and also helped to reduce the likelihood of theory forcing.

The process of microanalysis is explored in detail by Strauss and Corbin [204, p.65], where twelve of the key facets of microanalysis are reviewed:

1. Microanalysis is a very focussed process, where the researcher must consider the full range of possible interpretations of the data. The process of open coding is very slow as the researcher must look for as many perspectives as possible on the data. It is important that they avoid jumping to hasty decisions on the nature of each category.

2. The researcher must not only look for new codes within the data, but must also be on the look out for variations in existing codes in terms of their properties and dimensions.

3. The researcher must bring their awareness to both the message the participant is communicating and the way in which they are communicating it. This can lead the researcher to a greater understanding of the participants interpretation of events.

4. The researcher must be constantly questioning the data to stimulate the discovery of either the properties and dimensions of existing codes or of new codes. If the answers to these questions do not lie within the data set then it maybe that these questions form the basis for subsequent data collection through theoretical sampling.

5. It is important that the researcher maintain analytic distance from the data – focusing on data that is relevant, rather than the specifics of individual cases.

6. The act of carrying out microanalysis leads the researcher beyond the mere description of events into their theoretical conceptualisation and classification.

7. The researcher must be constantly grouping concepts according to their similarities and differences.

8. There is a natural progression from the differences in the properties and dimensions of each codes that feeds into the establishment of the differences in the properties and dimensions of each abstracted category in the axial coding phase.

9. The researcher must explore the properties and dimension of each code through theoretical comparisons. These occur when the researcher questions what it is they are looking at in the data, e.g. a code, a property or a dimension of a code, and determines their answer based on the properties of what they already know from the data.

10. Comparison of codes observed in the data leads to the identification of variation within and between codes.

11. Provisional hypotheses about the relationships between the concepts are thrown up during the process of microanalysis. The statements of these hypotheses may be overly simplistic at first, but they are naturally refined over time.

12. Microanalysis allows the reader to explore how their own assumptions may influence their view of the data. The should not lead the researcher imposing this viewpoint on the data, rather it should raise their awareness of these assumptions and allow them to rigorously compare them against the data.

Open coding is extremely laborious and time consuming. It is essential that the researcher remains alert and vigilant throughout this process, to ensure that all codes are captured while the researcher's own opinions and beliefs are not imposed on the data. This typically necessitates multiple passes through the data, as the researcher refines the codes and categories previously identified and remains alert to the possible existence of codes that were previously over-looked.

Not all open coding involves microanalysis. If every word or phrase was coded the volume of data generated would be both unnecessary [204] and impossible to deal with. It is essential that the researcher engage in this level of detailed analysis at the start of the project, as it enables the discovery of codes and their properties and dimensions. Once categories have become more clear, the researcher becomes more directed in their analysis, seeking to flesh out the detail of existing categories and explore the relationships between these categories. This means the researcher can parse the data, looking for information that pertains directly to the category of interest.

Open coding gives rise to many categories and it is essential that the researcher be on the look out for duplication or significant overlap between these. This means that the researcher must move beyond the analysis of the data itself to the analysis of the codes generated by that data. Where necessary codes may be merged to form one single code, however the data must be reviewed to ensure that none of the information uncovered in the existing analysis is lost during this process.

While the initial open coding process was very labour intensive, some aspects of it were particularly challenging and time consuming. For example, a number of concepts emerged relating to different aspects of language e.g. verbal reasoning and verbal ability. It was clear that these concepts are closely related, however the researcher was initially uncertain as to whether they should be left as distinct entities (for possible later inclusion in an abstracted category) or merged into a single category.

In order to make this decision it was necessary to engage in what Strauss and Corbin [204] label "theoretical questioning", where the researcher asks questions that help them determine the relationship between concepts. In the case of the language related categories, theoretical questioning allowed the researcher to explore these concepts in detail in order to determine whether they were sufficiently established so as to be able to stand as independent categories if the decision was made to leave them as separate entities. As the categories described distinct elements of language and their associated properties and dimensions were sufficiently clearly articulated, it was decided that it would be possible to leave these as independent concepts. This allowed the researcher to then ask whether the concepts should be merged or not. The next question asked was what would happen if the concepts were combined into a single category i.e. whether the merging of these concepts would reduce the richness of any emerging theory, forcing it away from a more detailed analysis of the elements of language that influence success. An examination of the elements of language uncovered in the open coding phase led to the decision that these codes would not be merged at this point, as it was likely they would form part of a higher-order, abstracted category during the axial coding phase of the analysis.

5.2 Axial Coding

The output of the open coding process are concepts that can then be used as the building blocks of a theory. This labelling and categorising allows the researcher to explore new ways of interpreting the data. The research needs to link these categories so that explanations for the phenomenon and the theory emerge naturally from the data.

Axial coding is recommended by Strauss and Corbin [200] as a mechanism for exploring the connections between codes, categories and sub-categories. Strauss and Corbin required the researcher to ask questions such as what, when, where, why, how and with what consequence or result [204]. Each category is viewed as an axis around which the codes associated with the category pivot. The process of axial coding involves the assembling of the concepts identified into categories and subcategories, leading to the articulation of the phenomena that are revealed by the data.

It should be noted that while the researcher begins analysing the data using open coding and this leads towards axial coding, the two processes are closely intertwined. Indeed it is the exploration of this intertwining that leads to theory building. Theoretical questioning forms a key element of the axial coding process, as the researcher seeks to interrogate the data in order to determine how the concepts are interlinked. While theoretical questioning of this nature is more common during the axial coding phase of the analysis, it may also occur when the researcher seeks to consolidate the categories identified during the open coding phase.

The first stage of the process of axial coding is the identification of the properties and dimension of each category or subcategory. The researcher then proceeds to an exploration of the relationships and interdependencies between categories.

The central categories or phenomena are revealed by these relationships as their conditions, actions and consequences become clear.

While the construction of the categories and the links between them is clearly a creative project and, to a certain extent, depends on the imagination of the researcher; the articulation of these categories should be sufficiently grounded in the data so that the external observer can see their genesis. During the course of the analysis a number of potential abstracted categories were considered. These were then checked against that existing data and it was often found that existing codes did not fit neatly into these categories e.g. it was necessary to distinguish between learning that took place in formal, second-level classrooms and the assistance given by teachers outside of the classroom setting.

During this part of the analysis Borgatti's Basic Frame of Generic Relationships [30] was used as a way of capturing the key features of each abstracted category identified. This framework, discussed earlier in section 3.3.4 earlier and reproduced in table 3.1, will be used in the following chapter for each of the four abstracted categories identified in this study.

While coding and analysis begin once the first data set is collected, the collection of data is not completely separated from the process of analysis. The researcher must be constantly exploring the nature of the data available to them and in what ways it might be deficient. This leads to the collection of more data through the process of theoretical sampling (see section 3.3.2). For example, during the course of this study it was noted that various aspects of language were amongst the codes identified (e.g. verbal reasoning, academic writing). It was deemed necessary to explore this through the collection of further data, so additional questions were included in subsequent interviews and focus group sessions.

5.2.1 Properties and Dimensions

Axial coding requires the research to fully explore the properties and dimensions of each category. In section 3.3.1 the properties of a category were defined as characteristics that are common to all concepts within the category, while the dimensions of a category capture the range of variability of a given property.

When analysing data it was observed that participants experience various phenomena differently e.g. some students had a very supportive home environment where they were actively encouraged to pursue their interest in computer science while others were discouraged as their computer based activities were viewed as a distraction from their studies.

It was expected that it would be difficult to articulate the differences between similar codes that arose in two different transcripts, however it was found that a focus on the identification of key properties and their dimensions brought these differences to the surface. This, in turn, facilitated the process of constant comparison, which enabled the researcher to determine when category saturation had been achieved.

An example of one of the categories found during the study, Academic Influences, along with its associated properties and dimensions is given in table 5.1.

5.2.2 Coding Paradigm

One key element of the Strauss and Corbin [204] axial coding methodology is the coding paradigm (see section 3.4). Strauss' paradigm [200] requires the identification of defining features for each category that include (i) conditions, (ii) interaction among the actors, (iii) strategies and tactics, and (iv) consequences. The aim of this paradigm is to provide a structural framework for the coding of data and the development of categories. This methodology is hotly disputed by Glaser [87] who argues that this framework is far too prescriptive and leads to theory forcing.

Like Urquhart [212], the researcher found that this coding paradigm was particularly difficult to rigidly apply in a practice. It was decided to be slightly more flexible in the application of the paradigm, by using it to draw out differences amongst the elements identified in the data and allowing it to suggest ways in which the data could be abstracted.

Glaser offers an alternate way to assist the researcher with the abstraction of categories through the use of a wide variety of coding families [94]. The eighteen families of codes suggested by Glaser are summarised in table 5.2. These coding families proved to be extremely useful prompts during the axial coding stage

Category Name	**Influential Academics**
Description	Refers to the assistance given to participants by their second level teachers outside of formal classroom settings. This may involve verbal encouragement on a small number of occasions or be more involved encouragement e.g. through the provision of equipment or introductions to others outside of the school setting
Property 1:	Extent of Assistance
Dimensions	Partial – Full
Property 2:	Nature
Dimensions	Negative – Positive
Discussion	The property extent here related to the nature and type of assistance given to the participant by their teacher. At its simplest this may refer to conversations between the participant and their teacher, where the teacher is supportive. At its fullest it extends to support outside of school hours e.g. encouragement to engage with others with a similar interest in the community or with activities run in external institutions (e.g. the Centre for Talented Youth in Dublin City University). The second property of this category is the nature of the influences, these ranged from positive encouragement to active discouragement e.g. where it was suggested that too much time was being devoted to computer based activities.
Related Categories	External Influences, Family Influences, Peer Influences

Table 5.1: Category Overview: Academic Influences

of the analysis and so they were used in preference to rather limiting coding paradigm structure mooted by Strauss and Corbin. However, as the abstract categories came into focus, it was found that Borgatti's Basic Frame of Generic Relationships [30] (see section 3.3.4) provide a very useful mechanism for the researcher to summarise the abstracted categories.

Coding Family	**Examples**
The Six C's	Causes, Context, Contingencies, Consequences, Covariances, Conditions
Process	Progressions, Transitions, Ranks, Ordering, Shaping
Degree	Range, Extent, Grade
Dimension	Elements, Divisions, Aspects
Type	Type, Form, Style, Genre
Strategy	Strategies, Tactics, Mechanisms, Management, Dealing with, Goals
Interactive	Mutual Dependence, Interaction, Covariance
Identity-Self	Self-image, Self-evaluation, Self-worth, Social worth
Cutting Point	Boundary, Turning point, Critical juncture
Means-goal	End, Purpose, Products
Cultural	Social values, Social Beliefs, Social Sentiments
Consensus	Agreements, Opinions, Conflict, Different perceptions, Cooperation
Mainline	Social organization, Social order, Social interaction, Social mobility
Theoretical	Interpretive, Explanatory, Predictive Power, Use of theoretical codes, Relationship to other theory, Relevance
Ordering or Elaboration	Temporal Order, Structural Order, Personal Motivation, Achievement Orientation, Institutional Goal
Unit	Group, Organization, Society, Family
Reading	Concepts, Problems, Hypotheses
Models	Linear model, Property Space

Table 5.2: Glaser's Coding Families (source: [94, pp. 73-82])

5.3 Theoretical Memos and Integrative Diagrams

The two distinct schools of grounded theory (Straussian and Glaserian (see section 3.4)) are both in strong agreement on the use of theoretical memos throughout all phases of the analysis of data. These memos capture the ideas that strike the researcher while in the process of coding the data. These memos can range from a simple phrase or clause (e.g. "Language (again!), everyone seems to be saying they don't understand what they are being asked to do") through to more detailed notes (e.g. "It seems that people seem to feel they need permission to do things on a computer e.g. they need permission from their parents or teachers before they can change settings etc. This seems to suggest they would like to be able to experiment by changing things, but, for example, they don't want to get into trouble with the school authorities if they do so. It's amazing how many of them say they weren't let do things on the school computers as the staff were afraid of what they'd do. How does this fit in? I guess that fits with idea of constructive support from those around them."). As can be seen from these two examples, memos are the unpolished thoughts of the researcher, allowing them to bracket thoughts and ideas about the data as they occur, setting them aside for future consideration and allowing the researcher to re-focus their thoughts on the data.

Memos of this type are very important, as the researcher's exploration of the relationships between the codes and categories is central to the subsequent development of theory. These memos are really the subjective reflections of the researcher, however their validity can be explored through the analysis of the data collected. This is part of the constant comparison process where the researcher uses the data to support and validate any hypothesis that occurs to them during the analysis of the data.

Drawing diagrams formed an important part of the analysis. Diagrams automatically draw the researcher away from the linear lists that arise during open coding. The diagrams ranged from back-of-an-envelope diagrams where the researcher tried to visualise the relationships and communicate them to others to more formal diagrams where the researcher sought a different perspective on the emerging categories and the links between them. Diagrams proved particularly useful at points where the analysis seemed to stall and there was no clear way to resolve the conflicts between the categories proposed by Strauss [200].

Strauss [200] encourages the use of integrative diagrams as a way of making sense of the data by pulling all of different themes that emerge together. The diagrams used may take on different forms depending on the stage of theory development involved. For example, they may be direct graphs or concept maps that summarise the extent of the current analysis. Researchers are encouraged to talk to others about their diagrams as the process of interacting and sharing ideas can help to increase one's understanding and awareness of the information contained in the data.

5.4 Selective Coding

The intended significant outcome of this study is not the generation of codes and categories, but the articulation of a theory. The process by which this is achieved entails the identification of a central category, or phenomenon, and the association of all other categories to it. For each of the associated categories there must be a clear statement of its relationship to the central phenomenon. This process is known as selective coding (see section 3.3.5).

While the researcher is at all times fully aware that one of the goals of the analysis is the identification of this central category, care must be taken not to force its emergence from the data, particularly in the early stages of the analysis when the researcher is grappling with the identification of abstract categories.

One reason why the researcher must be careful not to commit to a few central categories in the early stages of the analysis is that this can have a deleterious effect on the whole study as it may stifle future analysis and impact on the collection of subsequent data. It can be tempting to decide that one should promote an abstract category one has spent many days toiling over to be the core category on which to base the emerging theory. However, if one reviews the data it can quickly become apparent that not all of the codes can be easily associated with this category. For example, it was initially thought that the category entitled "technological experimentation" would be the core category. However, subsequent data analysis revealed that although this was an important category that related to the experience of almost all participants, it was not the core category. In particular, it did not encapsulate the processes captured within the category that was subsequently labelled "Language".

5.5 Conclusion

In this chapter we have reviewed the processes that need to be engaged with during the analysis of the data. In general, the open, axial and selective coding processes from the variant of Grounded Theory espoused by Strauss and Corbin [204] were employed in this study. However, it was found during axial coding that the use of Strauss and Corbin's coding paradigm was particularly challenging. This experience was not unique to this study as similar issues have been recounted in the literature [178; 212]. The difficulties encountered were overcome through the use of the coding families suggested by Glaser [94].

The aim of this chapter was to provide an insight into the processes undertaken during the coding and analysis of data. It should be evident to the reader that the outcomes presented in the following chapter did not spring fully-formed and clearly articulated from the data; rather they were the result of many months of painstaking analysis where the researcher sought to discover the theory embedded in the data.

Chapter 6

Data Analysis and Theory Generation

This chapter documents all stages of the analysis and synthesis of the data collected; from the three stages of coding (open, axial and selective coding) through to the identification of the central category of the theory of successful engagement.

The concepts that emerged during the open coding phase are presented and the mechanisms by which these were reduced into a key set of categories are described. These categories were then abstracted into a set of key higher order categories (or phenomena), from which the central category emerged. The codes, categories and phenomena are evidenced by quotations from the participants. The validity and reliability checks used throughout the study to ensure the trustworthiness of the emerging theory will be considered in chapter 7.

As a qualitative methodology, grounded theory does not have the same straightforward approaches for validity checking that are familiar to users of quantitative methods. If one is carrying out a quantitative survey then the outcomes can be justified by providing a copy of the survey instrument and evidence of the findings is in the statistical analysis of the gathered data. With a qualitative methodology, such as grounded theory, it is simply not possible to provide evidence in a similar way so that the reader can confirm the outcomes for themselves. What is included is a selective subset of the data that creates a rich, meaningful and convincing presentation. Hence, it is important that the researcher chart the evolution of the process from the initial coding to the emergence of the theory, highlighting critical junctures and theoretical insights [95].

6.1 Open Coding

Open coding proved to be a long and involved process requiring multiple passes through the transcribed data. Initial coding involved a microanalysis where each pass through the data looked to view it from a different perspective e.g. one pass through the data focussed on individual words, the next looked at phrases and subsequent passes considered sentences and paragraphs. The researcher needed to return to the data with a fresh, open mind for each pass. It was found that as thoughts and suggestions about the data came to mind that it was best to stop and make memos in the margin of the data. This enabled the researcher to put these ideas to one side for later consideration and then move on with the analysis of the data.

The labels for codes evolved over time; when a new or existing code was encountered it was noted in a shorthand form in the margin. All extant codes were considered as possible and the most appropriate label used. After each pass through the data these codes were reviewed to see if the same concepts had been labelled with different codes. This consolidation of codes helped prevent unnecessary duplication and reduced the number of codes needed. This consolidation of codes was carried out on a number of levels; first for each individual transcript and then as a comparison between transcripts. Some of the codes used were in-vivo codes, arising from the terminology used by the participants; others were chosen as appropriate labels for the concept being articulated.

It was also important to make sure that codes identified really were codes and not just outcomes or action strategies associated with the code. Examples of open coding from the transcript of the interview held with participant 3 are give in table 6.1. It should be noted that in this example of early coding, the codes "Computer Time" and "Computer Use" were used, however it was quickly realised that this was too broad a term that would include other codes such as "Games" and "Social Media" so these items were re-examined and re-labelled to more accurately reflect the associated concept.

I probably thought of computer science as a course choice because **I do spend a lot of time on the computer at home**, but having spent time on the computer it would have been **mostly games and facebook**, there's a lot of **friends on facebook** who spend as much time on it, but they wouldn't be interested in studying it. **It's not just the amount of use**... without **my dad and my cousin** I just wouldn't have been interested.	Home, Games, Social Media, Familial Influences, Peer related access
With my dad and my cousin I saw a lot about computers and that helped... I don't know other than that. It was **helpful to have maths**, I suppose, but **everything else is completely new** to be honest. I was always **into problem solving I think that helps with the algorithms**.	Familial Influence, Mathematics, Alien, Problem Solving
I got a B2 in the Leaving. I would have liked to have done better, but the paper, it was a horrible paper. **I really like maths, I'm not great at it, but I like it**.	Mathematics, Enjoyable
The other girl in the class, she left. **She found it frustrating and she left**. She was **only interested in graphic design**... she was getting really down, she said **she felt like an idiot sitting there when she couldn't understand the stuff**. .. while X, his problem, I think is that **he just doesn't come in. He just says he's too busy.**	Self-affirmation, Challenging, Intrinsic Motivation, Intimidating
Like **there is stuff there I don't want to do**, but I'm going to... **but I'm not going to give up on it because I'm finding it tough**... you know, **I do want the degree** and I want to work with that	Challenging, Self-efficacy, Persistence

Table 6.1: Examples of Open Coding (Participant 3)

As coding progressed, the process became easier and slightly less time-consuming, as the number of new codes needed was reduced. This was to be expected as one should expect to reach theoretical saturation, where collecting more data will not yield any more information.

The outcome of the open coding process were documents with a large number of codes and memos in the margin. These codes and memos could then be extracted into a separate file for further analysis during the axial coding process. The codes identified during the open coding process are give in table 6.2.

Academic writing	Intrinsic motivation
Alien	Linguistic Precision
Challenging	Mathematics
Click moments	Mobile Devices
Competitive peer learning	Office applications
Complex competence	Other access to computers
Collaborative peer learning	Peer related access to computers
Critical listening	Persistence
Enjoyable	Prior Programming Experience
Environmental factors	Problem solving
Extrinsic motivation	Procedural training
Familial influence	School access to computers
Games	School mediated learning
Grammar	Self affirmation
Hardware	Self efficacy
Home access to computers	Sentence structure
Independence	Simple competence
Influential academics	Social media
Informative listening	Verbal ability
Intimidating	Verbal reasoning

Table 6.2: Categories Identified during Open Coding

The properties and dimensions of the categories identified during open coding were explored using axial coding (see sections 3.3.4 and 5.2). For each of the phenomena uncovered, Borgatti's Basic Frame of Generic Relationships [30] was used to summarise and capture its properties. The four distinct abstracted categories identified are detailed below. These were labelled experience accumulation, empowering experimentation, perception of competence and language.

6.2 Experience Accumulation

This phenomenon captures the ways through which participants acquired their pre-undergraduate experience of computers. Computing experience was accumulated in formal and informal learning environments and through hands-on interactions with computers. Participants indicated that those who were most successful had a deeper knowledge of mathematics and had well developed problem solving skills. Others commented on the fact that prior programming experience was extremely beneficial.

The process of axial coding identified four distinct subcategories associated with this phenomenon: technological access, technological interactions, practising and conceptual knowledge. Technological access captures the different opportunities participants had to interact with computers e.g. at school, at home, with their peers or through other sources or venues. Technological interactions looks at the ways in which the participants interacted with computers, for example through programming or social media. The subcategory of practising looked at how experience was gained through practice and persistence. Finally we look at the conceptual knowledge that students felt led to their success at the undergraduate level.

Figure 6.1 provides a hierarchical representation of the category and its associated subcategories. The subcategories of technological access and conceptual knowledge are further broken down in figure 6.1, while the subcategories of technological interactions and practising are portrayed in figures 6.2 and 6.3 respectively. It should be noted that whilst these diagrams follow the traditional hierarchical method of displaying grounded theory phenomena, the interactions between the subcategories are usually more complex in nature.

6.2.1 Technological Access

For many, experience comes through prolonged engagement with a field. The more access students have to computers and technology, the more time they are likely to spend gaining experience with that technology. For many of the participants these experiences came through interactions that took place at school or at home. Others engaged in computer related activity while interacting with

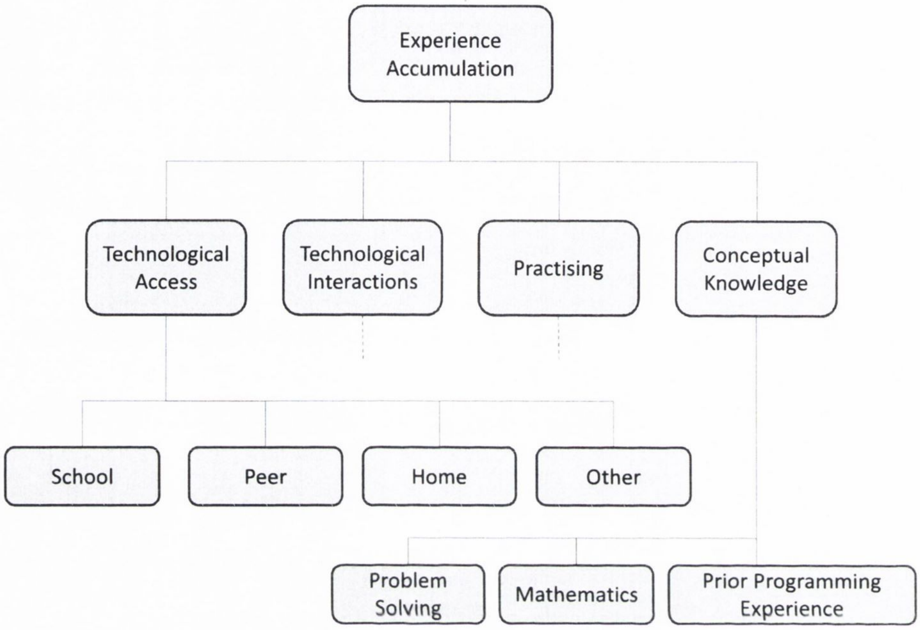

Figure 6.1: Experience Accumulation

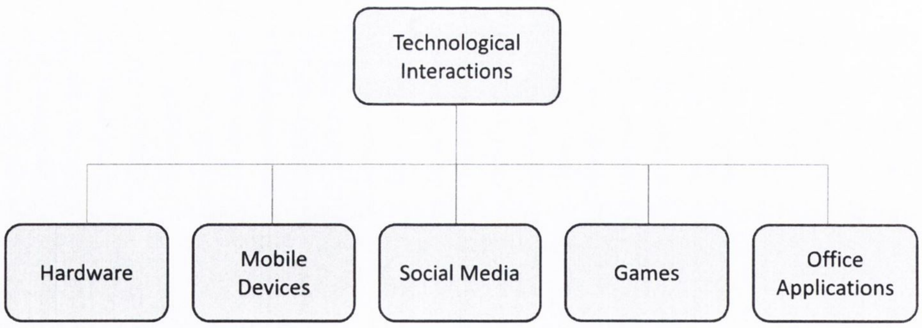

Figure 6.2: Experience Accumulation Subcategory: Technological Interactions

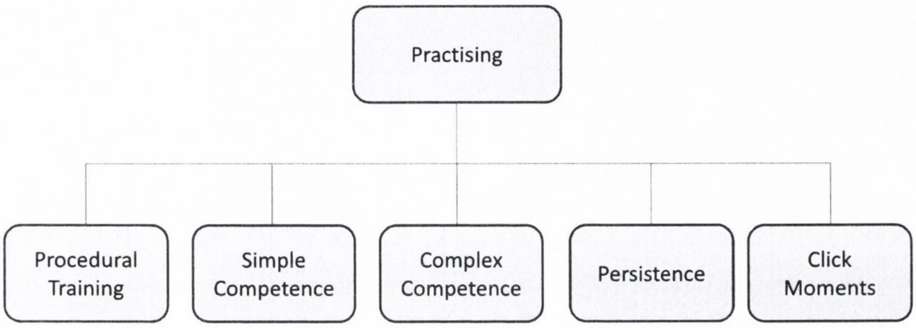

Figure 6.3: Experience Accumulation Subcategory: Practising

their peers in venues outside of school or home. Other opportunities arose when participants took formal courses in computing outside of the traditional school environment.

6.2.1.1 School

Most participants took part in some formal learning related to computing while at school e.g. when taking courses in keyboard skills or on the creation of web pages during their fourth or fifth year of second level education. Others took advantage of the availability of computers in school to gain more experience:

> **Participant 16:** "I'd stay back after school with my friends and we did some programming in the computer room. We played games too, though."

However, not all participants reported on such positive experiences; indeed some reported interactions with teachers outside of the classroom that impeded their progress.

> **Participant 11:** "One of the teachers banned me from the computer room. It was stupid really, he didn't know much about computers and he didn't like me because I did. He kept telling me I was spending too much time there."

Indeed a number of participants report difficulties with access to computers in school as they were viewed as a threat to the system integrity and security. As one participant put it:

> **Participant 12:** "We all knew how to bypass it, so they wouldn't let us do stuff unless they were there to watch, so you'd end up doing stuff at home."

6.2.1.2 Home

For others the main opportunity for gaining access to computers was when they were in their home environment. Spending time doing things with computers formed a hobby for some, as it was something they choose to do during their free time:

> **Participant 16:** "I used to do quite a bit on my laptop at home. When there was nothing else to do, I'd spend a few hours doing stuff."

> **Participant 3:** "I probably thought of computer science as a course choice because I do spend a lot of time on the computer at home."

Another participant stressed the importance of having their own computer at home in order for them to succeed at university:

> **Participant 14:** "It's important to have your own computer at home. I worked all through sixth year so I'd have enough money to buy a good laptop for college. I knew I'd need it if I wanted to do well."

6.2.1.3 Peer and Other

Others spend time engaging with technology when in the company of their peers:

> **Participant 12:** "I spend quite a bit of time at my friends house because his brother was into computers and had a lot of stuff. I learnt a lot by just trying out stuff there, really."

> **Participant 16:** "I used to do stuff with my friends and we'd help one another."

One participant was encouraged to take a course by one of his teachers:

> **Participant 12:** "My physics teacher was really helpful. He'd be interested in what I was doing and was looking for me to enter the Young Scientists. He also suggested I do the CTYI course. Which I did, it was really good. We did stuff on computer games that was really interesting."

The Centre for Talented Youth, Ireland (CTYI) offers courses during the summer months for students with high academic ability. It is based in Dublin City University.

6.2.2 Technological Interactions

The category of technological interactions captured the range of activities the participants were likely to engage in during their time spent interacting with technology. This is different from the previous category of technological access that focussed on the different venues where participants gained experience. There were five key subcategories that captured the broad spectrum of activities participants were likely to engage with: hardware, mobile devices, social media, games and office applications.

6.2.2.1 Hardware

For many participants their encounters with computer hardware before entering university were non existent:

> **Participant 9:** "I'd done some programming but I'd never done hardware and that was all new to me. I don't quite know what to do when it doesn't work."

Those that had gained experience of computer hardware had usually been introduced to it through their interactions with others, rather than through independent learning:

> **Participant 11:** "My friend's Dad let us take an old computer apart, you could really see how everything works. I'd be quite good at fixing things now."

> **Participant 12:** "My friend's brother had a soldering iron, so we used to trying making things."

6.2.2.2 Mobile Devices

For some, mobile devices formed an important part of their interactions with technology. This was typically related to activities involving their mobile phone. For one participant the idea of developing applications to run on such mobile devices was a key factor in their decision to study computer science:

> **Participant 12:** "I can't wait for the new iPhone, I'm definitely going to get one. I'd love to get a job writing games for it."

As noted in the introduction this study took place before the advent of tablet computers and at a point in time when enhanced data rate smart phones were beginning to dominate the market. It is likely that if this study was to be repeated today, then this category would yield richer data. This new data may require the position of this category within the emergent theory to be re-evaluated e.g. it can be speculated that mobile devices might play a much more dominant position in any such study.

6.2.2.3 Social Media and Games

Many participants reported large amounts of time spend interacting with their friends using social media applications such as Facebook or playing computer related games. In the following quotation the participant makes an interesting observation where they note that while these interactions sparked their own interest in computer science, it did not have the same effect on their friends:

> **Participant 3:** "I probably thought of computer science as a course choice because I do spend a lot of time on the computer at home, but having spent time on the computer it would have been mostly games and Facebook, there's a lot of friends on Facebook who spend as much time on it, but they wouldn't be interested in studying it. It's not just the amount of use... without my dad and my cousin I just wouldn't have been interested."

6.2.2.4 Office Applications

When looking at the types of technology that the participants interacted with, the final subcategory that arose from the data was interaction with office related applications for document production, data analysis and the creation of presentations. Others also mentioned learning how to create web pages. Most participants underwent some formal training in the use of these applications whilst at school. It was interesting to note that some of the participants viewed these skills as irrelevant to the study of computer science, as one participant put it:

> **Participant 16:** "We did some computer courses in transition year. I got the Microsoft IC^3 exams. They mainly covered the use of Word, Excel and Powerpoint. It's not really relevant to computer science because it's just how to use Microsoft products."

6.2.3 Conceptual Knowledge

Procedural knowledge relates to a knowledge of the steps required to achieve a given objective. This can be achieved through the rote learning of a given set of rules or procedures. By contrast, conceptual knowledge relates to a deeper understanding of a concept where one moves beyond simple rules to an awareness of the relationships and interdependencies amongst concepts.

The elements of conceptual knowledge that were most frequently cited by participants were problem solving, mathematics and prior programming experience; though some also mentioned that a knowledge of physics would be useful. All participants believed that this conceptual knowledge was key to either their own success or to the success of others.

When participants discussed the importance of prior conceptual knowledge and understanding, they sometimes stressed how the associated skills were attained through practice. It was initially uncertain if "prior conceptual knowledge" and "practising" formed two separate subcategories. Theoretical sampling led to the conclusion that they were distinct, as participants often mentioned the importance of their prior conceptual knowledge without reference to the enhancement of the associated skills through practice.

When asked about what school subjects best prepare them for studying computer science, many students immediately start talking about mathematics and problem solving:

> **Participant 9:** "I know it's about problem solving. I like mathematics and I'm good at problem solving, so that helps."

> **Participant 10:** "It's important to know how to solve maths problems. I actually like maths problems that I can't solve, because you've got to really puzzle it out and think about it. It's much more fun when you can't solve it and then you find a solution. I enjoy doing problems like that."

One student brought up the links between problem solving and programming, although they immediately made reference to mathematics; suggesting that they viewed this as important too:

> **Participant 6:** "Computer science is good if you like problem solving. I like programming, that's my favourite, I like sorting out the algorithm, I like the maths as well."

Others brought up the subject of prior programming experience:

> **Participant 11:** "My friend's Dad works with computers, so my friend did a lot with him and then he showed me. That was really when I learnt about programming and it has made things much easier. I wouldn't like to start computer science without knowing something about programming."

Those who didn't have prior programming experience felt that it gave others a distinct advantage and would recommend that students considering studying computer science learn some form of computer programming:

> **Participant 2:** "I've been doing maths since I went to school. I know its important for computers and you've done it before. If you did programming in school it would really help."

> **Participant 16:** "I'd tell them to do some programming, even computer assembly programming. It would be worth looking up stuff on that."

In summary, the participants tend to believe there is a strong link between problem solving and mathematics and that these are both relevant to the study of computer science. Students with prior programming experience feel they are at an advantage relative to their class mates and those without programming experience would encourage others considering computer science to learn some programming in advance.

6.2.4 Practising

As mentioned above, practising emerged as a distinct subcategory of the phenomenon labelled experience accumulation. The participants found that as well as access to computers and the necessary conceptual knowledge they also needed to develop their experience through practice. As participants put it:

> **Participant 4:** "You have to practice. It's the only way that you can get it. If you are stuck then you need to get someone to help you. If you don't work at it, then you won't understand it."

In response to the question "What do you think makes the good programmers in your class group good?", one participant replied without hesitation:

> **Participant 17:** "Practice. One of the lads is amazing at everything. He says he's been programming since fourth class. Another guy has been programming since he was eight; he'd be a step higher again."

Another participant expressed it in the following way:

> **Participant 1:** "The more you practice, the better you'll be. Some of them have been programming for years and they are much better than I am, or I'll ever be."

One student commented on the similarities between mathematics and computer science in terms of the practice needed to acquire the skill:

> **Participant 10:** "It's like maths, its a subject where you've got to practise. It's not enough to see someone else do the problem. You've got to give it a go yourself. It's only when you've done ten or twenty questions that are the same that you can say you know how to do it."

Practising emerged as a possible category for this study during initial coding and it was subsequently explored further through theoretical sampling. When participants mentioned practising they were asked to elaborate further on the nature of the practice involved. This led to the identification of five distinct subcategories. A set of three subcategories that form a logical progression: procedural training, simple and complex competencies; and two other subcategories labelled persistence and click moments. These subcategories are now explored in more detail.

6.2.4.1 Procedural Training, Simple and Complex Competencies

Three of the sub-categories of practising form a logical progression that moves from an ability to carry out basic procedural tasks through to the application of functional and associational knowledge to carry out more complex tasks. At its most basic level, practising is involved in procedural training where participants acquire a simple skillset through repetition e.g. touch typing. It may also involve some degree of procedural learning where participants learn about the steps needed to carry out a given task e.g. how to set up a wireless access point.

Participants who have engaged in sufficient procedural training, and have developed their basic procedural understanding, may then go on to apply this knowledge to other simple tasks or problems in the domain. This is similar to the study of mathematics where students apply their procedural training in algebra and geometry when tackling calculus problems. The final stage in the logical progression is when students gain an appreciation and awareness of the interdependencies and relationships between the more simple, functional understandings they have developed. These three distinct subcategories of the practice category are illustrated in the quotations from participants given below.

When talking about what they feel makes them successful at programming, two of the participants mention how following a given set of procedural steps helps them achieve a working solution:

> **Participant 5:** "I copy,.... well I don't copy, I make my own version, but I would use the same steps to get something that works."

> **Participant 13:** "I follow the examples given and that helps. If I'm programming then I take a piece of code that works and I change it."

The first year computer science students study linear algebra and one student commented on how the methods used there require a simple procedural knowledge.

> **Participant 11:** "In maths we learnt about Gaussian Elimination and it's like a recipe. You follow the instructions and you get the answer. So you need to be able to follow instructions and I'm good at doing that."

Another student said that they learnt how to touch type in school and that it meant they could work quickly:

> **Participant 8:** "During transition year I did a module in keyboard skills where we learnt how to type and that's been really useful. I'm much quicker than some of the others in the class"

Participants also reflected on how they have used simple competencies and skills to help them succeed:

> **Participant 6:** "I like to start off with some easy problems that I know I can do and then move on to harder questions where I have to think a bit more."

> **Participant 17:** "I'm much better once I see how to apply things. So I would tell people to look at how things are applied and then they will make sense. Some people seem to be able to get it immediately, but I need to see applications."

> **Participant 9:** "There are some things you can learn off, but it doesn't seem to be like other subjects as you need to be able to use what you know. I like to create tests for myself to see if I understand."

In the following quotations we see how participants have begun to move towards an appreciation of the wider context of their studies and appear to have gained a greater understanding of the manner in which topics they were studying linked together. It could be argued that these participants are moving towards a broader, more complex awareness of the subject area.

> **Participant 4:** "At first I didn't see how any of the things we were doing fitted together but I've a better understanding now and I can see why they are relevant. For example, the physics we are doing seems to be out-of-date and twenty years old and I couldn't see the point of it. But then I looked it up and now I can see why we are doing it. I still don't like it, but at least I know why we are doing the course."

> **Participant 16:** "You have to keep going back over things and the more you look at them the more you start to see patterns and similarities. You have to keep going back through the notes and you start to link things together. That way you know you understand."

In this subsection we have looked at three of the subcategories of the category labelled practising. In the following subsection we will consider another of the subcategories, persistence.

6.2.4.2 Persistence

When trying to put what they were learning into practice, many participants commented on how persistence was a key element of their strategy for success. For some students persistence involved focusing on the same task for extended periods of time, while for others it related to the ways they gained more experience with computers. For example, it was noted that one needed to be persistent when seeking out sources of relevant information.

> **Participant 15:** "My advice would be to keep learning things. When I don't know something I google it or I ask someone. That way you learn more. The more you know about computers before you start the better."

> **Participant 16:** "When I was learning how to program, it seemed really difficult at first as there was a lot to get your head around. You have to keep trying things and looking things up. You can spend a lot of time on it. I used to do stuff with my friends and we'd help one another."

> **Participant 18:** "You have to be willing to ask questions. When I don't understand things I ask someone, but you may have to ask a few people. Some of the demonstrators aren't very good, but you have to keep asking. Then you go back and look at the notes again and hopefully they will make sense."

Another element of persistence is the length of time participants spend on computing tasks. For example, some of the participants commented on the length of time they spend studying or the amount of time it took them to find a cohort of peers to engage with.

> **Participant 10:** "Because you don't learn computer science in school, you've got to find out about it for yourself. You have to keep looking on the web or finding books. It takes some time, but then you find others who are doing the same stuff and it gets easier."

> **Participant 8:** "When learning to program you have to be willing to get things wrong. You have to keep looking things up and trying things out. Eventually you will get it to work. It would sometimes take me days to figure something out, so my advice would be don't give up, keep trying different ideas. You'll find the right way to do it."

6.2.4.3 Click Moments

The final subcategory of the "practising" category is that of "click moments". As a result of practice and persistence many students refer to the satisfaction they get when the finally understand a concept they have been studying. For some a series of these "click moments" are needed before they feel they fully grasp a concept, while for others they feel that there is a single "click moment" where everything falls into place.

In the following quotation the participant indicates how their persistence led to one of these "click moments":

> **Participant 18:** "Doing the last assignment I got a much better understanding of it because I was em, ... I read up a lot of stuff, I asked people... I tried my best ... it all clicked an hour before I had to have it in; so I didn't have time to finish it properly which was a pity, but I got it."

> **Participant 1:** "To be honest I'm not finding it easy. There are some bits I've got my head around, but there's other bits that haven't clicked with me yet."

> **Participant 12:** "There have been quite a few times that I didn't get the topic at all. I read through the notes and I looked at the tutorial stuff and it clicked a bit. To really understand it takes quite a bit of time, but once you've got the idea then you just have to work at it"

When asked if "click moments" exist, these participants explain how they come about:

> **Participant 2:** "It does happen, you'd be sitting there wondering what on earth is this about and then something would kind of click with you."

> **Participant 14:** "You'd be looking at a tutorial and suddenly it would all click into place. It's a good feeling."

Another participant is slightly more pragmatic about how they understand things:

> **Participant 9:** "Sometimes you'd be sitting there going... what is going on here, but then you read through it and that helps a lot as you then get it."

In this section the phenomenon of experience accumulation has been explored in detail. Table 6.3 provides a summary of this category and its properties.

Element	Description
Phenomenon	Experience Accumulation
Causal conditions	The student can gain experience through time spent interacting with computers in a wide variety of settings. The nature of this experience may depend on the environment where it is obtained e.g. students may gain basic procedural training in office applications in a formal classroom setting but may engage with social media when at home. Many enjoy the feeling of success they get from "click moments" when they feel they comprehend a given topic and will actively engage in tasks that lead them to experience these click moments.
Context	Experience can be gained in formal and informal settings e.g. at school or at home. For many experience is gained over time in different settings. Initial, procedural training may take place in the school environment and this may lead the students to build on their basic competencies through practice at home or with peers. Those students who have prior programming experience have usually considered it to be a hobby engaged in during their leisure time. Those with the most experience have been persistent in seeking out opportunities to learn more and gain a greater understanding of computer science.
Intervening conditions	For students who actively sought out opportunities to practice their skills and gain experience there were many intervening conditions ranging from text books and learning resources found on the internet, through to engagement with on-line communities and peers who had similar interests.
Action strategies	Students gain experience by persisting at computer related tasks until they succeed. These activities may be carried out in conjunction with others, e.g. peers, teachers and family members, or through solitary time spend working on self-directed tasks. Students may also be persistent when seeking out opportunities to acquire experience. Students who are considered experienced by their peers are likely to have been pro-active in their efforts to gain this experience.
Consequences	Over time, practice leads to the development of complex competencies. Click moments are considered rewarding and those who experience them often look to replicate the experience. Those who don't achieve this success become frustrated and disillusioned with technology and so are less likely to engage in practice at a future date.

Table 6.3: Experience Accumulation

6.3 Empowering Experimentation

The abstracted category of empowering experimentation encapsulated four key subcategories. The first subcategory arose from the learning opportunities presented to students either in formal school room settings or else through contacts with their peers. These peer contacts could be either competitive, where students engaged in learning in order to "keep up" or compete with their peers, or collaborative where students learnt together with their peers. The importance of teachers outside of a formal classroom settings was emphasised by many. Those who were considered most successful were most likely to have received sustained encouragement, direction and extra-curricular advice from these teachers.

All of the subcategories involved participants being presented with opportunities to experiment with, and learn about, computers. The category title includes the word empowering, as these interactions often involved participants initially needing a sense of permission to experiment and engage with computers. This lessens over time so that as the participants confidence in their own ability grew they felt empowered to carry out experiments without recourse to an external, authoritative figure.

6.3.1 Learning Opportunities

This category captures all of the opportunities that participants had to learn about computers prior to the commencement of their studies. These learning opportunities may have arisen in formal school classroom settings, in hands-on practical lessons, through self-teaching and through peer learning. Teachers and peers are the two main facilitators of these experiences.

6.3.1.1 School mediated learning

All participants took part in some form of formal learning involving computers while at second level, ranging from courses on keyboard skills and the use of basic office applications through to courses on computer programming. Almost all participants were disparaging about the courses they took, as illustrated by the following quotation:

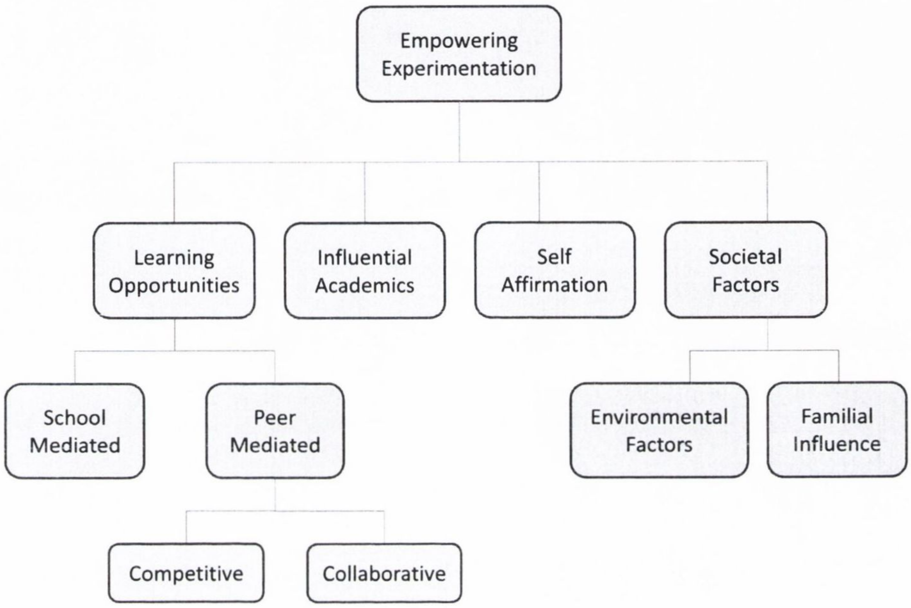

Figure 6.4: Empowering Experimentation

> **Participant 13:** "The computer stuff we did in school wasn't relevant. ... there was choice in fifth year, we had different modules you could pick, or else study periods... one of them was a computer course, it wasn't even ECDL, it was that sort of thing, it was typing up stuff in Word, it was ridiculous."

The European Computer Driving Licence (ECDL) (http://www.ecdl.org/) offers entry level qualifications on the use of computers and covers topics such as computer file systems, word processing, spreadsheets, databases, presentations, web browsing. In Ireland it is offered by many secondary schools to students in the fourth or fifth year of study. Another participant expressed their frustration with the range of computer courses available to them as follows:

> **Participant 9:** "We didn't do anything in school. We learnt how to make a webpage but we didn't do anything useful."

> **Participant 10:** "I did some programming in school, but the teacher didn't really know how to program, so we were all confused. It was only when I tried to learn it myself that I got to like it."

6.3.1.2 Peer mediated learning

For many their opportunities to learn about computer science came about as a result of their interactions with their peers. There were two forms of peer mediated learning identified in this study: competitive and collaborative peer learning.

Some participants said that their initial contact with computer science related topics came about because their friends or relatives were interested in the field. In some cases this peer mediated learning was considered a form of friendly competition:

> **Participant 12:** "My friend's brother was interested in computers and he showed my friend some things. I guess I wanted to be as good as him, so I started to try things out too"

> **Participant 7:** "My cousin was good at programming and I suppose I wanted to do things with him, so I decided I'd better learn some programming"

Participants often worked together with friends on computer science related problems. In the following quotation the participant explains how they learnt to program with the help of their friends:

> **Participant 16:** "When I was learning how to program, it seemed really difficult at first as there was a lot to get your head around. You have to keep trying things and looking things up. You can spend a lot of time on it. I used to do stuff with my friends and we'd help one another."

> **Participant 11:** "I like working with other people, you can share your ideas. If you can work together with other people then you will learn much more quickly."

For some this was a way of spending time with their peers during school hours:

> **Participant 11:** "In school we'd go into the computer lab during breaks so we wouldn't have to go outside and we'd be programming together."

6.3.2 Influential Academics

External influences emerged very quickly as a category during the first round of interviews. Participants described the influences of their families, teachers, peers and others on their engagement with computers. It was these experiences that they felt benefited them during their undergraduate studies.

This category was similar to, but distinct from, that of school mediated learning. Here we look at the influences of school teachers outside of the formal class room setting, where participants were encouraged, helped and praised by their teachers. The following statements capture these influences:

> **Participant 6:** "One of the teachers in my school knew that I was interested and he suggested I have a chat with this man who lives in the town. He was into building computers and he showed me a lot of stuff. It was really useful because he gave me ideas of things to look at."

> **Participant 4:** "I had one teacher and he pushed me because he knew I was interested. He was quite good with computers but in class he was a tough teacher because he wouldn't let you away with anything. But I learnt a lot from him and he kept asking me what I'd done."

> **Participant 16:** "One of my teachers used to give an evening course on Tuesdays. He suggested I go along and he taught me loads of useful things."

All of the above quotes indicate interactions with teachers outside of the classroom that facilitated the participants' engagements with computers. Some participants felt there was no support for their interest in computer science in school.

> **Participant 1:** "Especially in school there wasn't much emphasis on that kinda thing. We were never really told about computer science... they'd mention engineering, there were a few of us who were interested, but when someone came in, well I'd never be one for asking questions in a big group, so I didn't really ask about it"

6.3.3 Self-Affirmation

Self-affirmation [195] refers to the processes by which an individual seeks to make sense of their own actions and of the world around them. These processes are usually considered as a defensive mechanism as they may be activated in response to the typical setbacks and negative events of daily life. For example, a student may seek to rationalise poor academic performance by deciding that they have no aptitude for the subject in question or they may accept their poor performance and use it as a motivation for change. Alternatively, an individual may seek out positive experiences to help them recover from negative events e.g. someone who is feeling unhappy with one aspect of their life may seek out positive experiences to make them feel better.

The self-affirmation processes influenced and shaped participants attitude towards their studies, which in turn influences achievement and success. Self-affirmation varied along a continuum from positive to negative. Participants experienced positive feelings when they were successful in completing a computer related task or when their skill and ability was praised by their teachers. The participants belief in their own ability was affirmed as a result of these interactions.

> **Participant 8:** "Writing computer programs can be really satisfying. When you are doing it and it works, you get a lot of satisfaction. It's really worthwhile and you want to do it again because it makes you feel like you've achieved something."

> **Participant 17:** "I was doing some stuff on the computer and my teacher was really impressed, so that made me realise I was quite good at this, so I guess that made me find out a bit more about the course because I thought I might be good at it."

> **Participant 3:** "When I do well on assignments then I feel really happy because I know I've got it right. So I know that I can do assignments."

At the negative end of the continuum, participants described the influence of negative feedback from their teachers and peers. In relation to their university studies one of the participants rationalised their poor grades as follows:

> **Participant 7:** "It's embarrassing when you don't get things and when you get low marks. I'd done none of this before and I don't want the others to see what mark I got, because then they'll know that I don't understand it."

Another participant described their reaction to negative feedback from a demonstrator:

> **Participant 2:** "The demonstrators aren't the best. One of them is very sarcastic and he always makes me feel like I know nothing. He doesn't seem to know how to get down to your level. He don't explain things and they say you should know it from your notes. I always feel very stupid afterwards."

One participant expressed how they started to doubt their decision to study computer science when they tried to learn how to program on their own:

> **Participant 1:** "I tried to teach myself how to program last summer. I found it really tough and I was really worried that I'd be bad at it."

These negative self-affirmation influences are not only linked to the participants experience on their own; some reported how comments by others caused them to worry:

> **Participant 5:** "My parents didn't really want me to study computer science. They didn't think I'd be good at it, so I was a bit worried at first."

In the following quotation the participant explains how they rationalised their concerns about doing well in mathematics:

> **Participant 9:** "In school I used to feel annoyed when I didn't do well. I don't think he was a very good teacher and I wasn't the only one who didn't like him; though some of the others got A1's. It was quite hard to focus to be honest. I knew I had to get a C in honours maths. That was really my main concern for the whole of sixth year, well for fifth year as well. I didn't think I'd get it. I was really, really concerned about it."

In this study self-affirmation emerged as a category through the participants descriptions of how positive feedback from others influenced their likelihood of further engagement with computer related tasks, while negative feedback was discouraging and led participants to become anxious and worried about their decision to study computer science.

6.3.4 Societal Factors

This subcategory encapsulated the influences in the world around them that lead students to experiment with computers. For some the pervasiveness of computing in the world around them influenced them to learn more:

> **Participant 6:** "If you look at the world, there are computers everywhere. They aren't going to go away. There are always going to be jobs for people who have a computer science degree."

For many participants the ability to do well in computer science was closely linked to mathematical ability. This link appeared to stem from societal beliefs about the relationships between mathematics and computer science. It was interesting to note that while many participants made this association, they were unable to articulate where these links occurred in any meaningful way.

> **Participant 5:** "You need maths to do equations and equations are important for programming"

> **Participant 11:** "Well, it's not so much maths as logic. Computers use logic."

Others were influenced by their family member's interest in computer science or by the pervasiveness of technology in the environment in which they grew up. These formed two distinct subcategories.

6.3.4.1 Familial influence

Some participants gained experience through the encouragement and influence of family members. The quotation below shows how these influences may come from just one family member:

> **Participant 14:** "I'd never programmed before, but my dad is very into all computer stuff. My mother thinks I'm up in the sky."

When asked about how you learn about computer science before you come to university another participant commented:

> **Participant 4:** "I think, well, I suppose people know computers are used for other things, but they kinda... but that's all they have an experience of, they don't really have an experience of that unless they've someone in the family who knows about them."

Another participant credited family members for encouraging them to experiment with computers:

> **Participant 16:** "My parents were very interested and encouraged me. They would buy me things I needed."

6.3.5 Environmental Factors

For some these influences occurred outside the family home for example through their friends and through courses that they took:

Participant 11: "My friend's Dad works with computers, so my friend did a lot with him and then he showed me. That was really when I learnt about programming."

Participant 16: "I did an evening course with some of my friends where we learnt some visual basic. It's very different to Java, but I think it was worth doing."

In this section the phenomenon of empowering experimentation has been explored in detail. This phenomenon captures the way that participants interactions with computers lead them to form a strong belief in their ability to master topics in computer science. Participants felt that those who had been given, or who had given themselves, permission to experiment with computers were most likely to be successful. Table 6.4 provides a summary of this category and its properties.

6.4 Perception of Competence

The abstracted category labelled "perception of competence" is made up of four subcategories: motivation, independence, perception of computing and self-efficacy. For many of the participants their ability to succeed is closely related to their perception of their own competence. These perceptions are sometimes tied to the motivations that led them to study computer science or to confidence in their ability as independent learners. Participants beliefs about their ability to succeed are also influenced by their perception of computer science: some find it challenging but enjoyable, while others find it completely alien and intimidating. Finally, the participants perception of their own competence encapsulates their beliefs of what they can achieve with their existing knowledge and skills.

This is the third category that arose during the axial coding process. During this process the focus is on the development of categories that are distinct and self-contained. The researcher is seeking to abstract the previously identified open codes and concepts into more coherent, abstracted patterns that emanate from the data as discrete constructs. The first two categories that emerged during this process, experience accumulation and empowering experimentation, captured the more objective, quantifiable elements that students' believe are common to those

Phenomenon	Empowering Experimentation
Causal conditions	The student encounters computers in school, through family life and through interactions with others. They may experience computers during formal learning in school, through self-directed learning or through interactions with their peers. They may also recall how opinions given by others e.g. in the form of grades or feedback influenced their likelihood of subsequent engagement.
Context	This can have occurred as influences from others in the world around them, in formal class room settings and through informal self learning and peer learning activities. It may also have arisen through feedback or grades received on computer related assignments.
Intervening conditions	For some the desire to emulate a respected teacher, family member or friend may have motivated their early learning experiences. Equally, competitive individuals may have sought out computer experiences out of a sense of competition or co-operation with their peers e.g. if their friend was interested in computers this may have influenced the experiences they had. Others sought out computer experiences because of the positive feelings they received regarding their abilities or skills.
Action strategies	Participants may have made a conscious decision to engage with computers out of a desire to please or emulate others. The nature of the students' learning was often informal and involved a sense of discovery. Indeed, many commented on the fact that they were self-taught as no formal learning experiences were available to them. Where classroom experience existed, these were sometimes negative and participants indicated that they felt they had to "go against the tide" or "be the odd one out" in their pursuit of knowledge.
Consequences	Successful students felt they developed the ability to self-teach as a consequence of their experiences, but noted that this was encouraged by those around them. The fact that they had overcome obstacles in their quest for knowledge meant that successful students had a sense of self-belief in their ability to tackle harder computing challenges. Participants who had not these experiences felt they did not have the skills or self-belief needed to tackle problems that arose in contexts and settings they were unfamiliar with.

Table 6.4: Empowering Experimentation

they perceive as successful. The category under consideration in this section, perception of competence, is more subjective in nature and forms a distinct, discrete construct. This category does not focus on an individual's skills and how they were gained, but rather on the elements that contribute to their perception of their own competence. This encompasses not only what they believe they can achieve with the skills they have acquired i.e. their self-efficacy, but also the influence of their motivation and their perception of the domain in which they are functioning. Figure 6.5 provides a hierarchical representation of the perception of competence phenomenon and its subcategories are explored in more detail below.

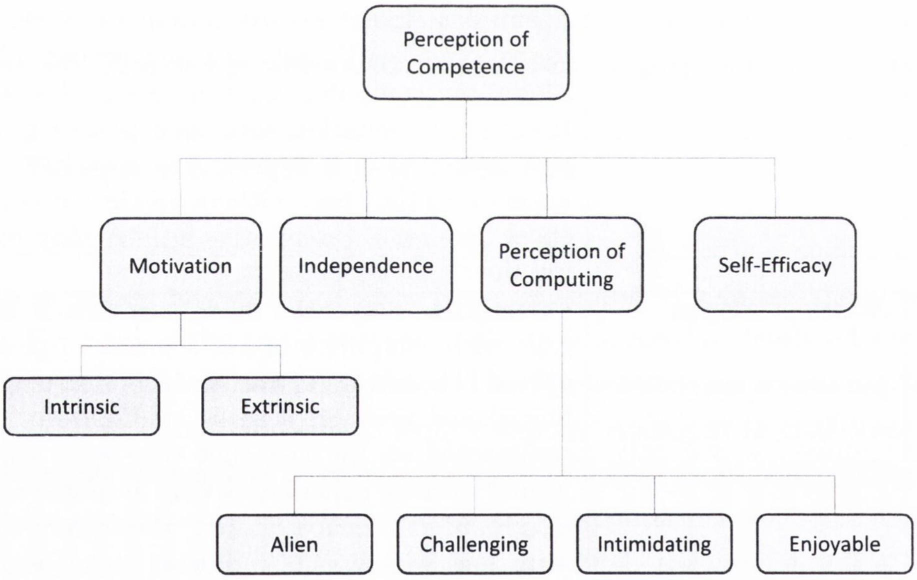

Figure 6.5: Perception of Competence

6.4.1 Motivation

Many of the participants view motivation as a key to success. In the following quotation a student comments on others they view as unlikely to be successful:

> **Participant 3:** "I think it's that they don't put the work in, they are not really that motivated, they don't work."

Another participant expressed their frustration with others that they feel lack motivation and are ill-prepared for the course.

> **Participant 1:** "There are some people in the class who know nothing. I wouldn't have come in to study computer science if I didn't know what it was about. They are finding it difficult and it slows things down."

The four classes of motivation identified by Biggs [22] were discussed in section 2.4.1. Despite efforts to explore these in more detail during theoretical sampling, not all of these classes of motivation emerged as distinct codes within the data collected. The two classes that did emerge were intrinsic and extrinsic motivation, but these were often closely allied with other forms of motivation. It is possible that further theoretical sampling would allow for a more thorough exploration of motivation and of the links between the different classes of motivation.

6.4.1.1 Intrinsic Motivation

Biggs [22] used the term intrinsic motivation to refer to those for whom the process of doing is central. For many of the undergraduate computer science participants in this study this comes across clearly when they are asked about what motivates them to do well:

> **Participant 12:** "I guess I'm interested and that makes it easier."

Another participant expressed their motivation as follows:

> **Participant 16:** "I like getting things working. When you actually get something to work you love it because it is a good feeling. If you are not interested in making things work then you won't like it."

A further participant described their motivation in terms of both achievement motivation,to get good grades, and intrinsic motivation, by the process of studying.

> **Participant 8:** "I'm motivated to get good marks and to study. I want to do well and I know I need good results in my exams."

Of those who indicated that their motivation was due to interest in the field of computer science, it was not clear if this motivation was specifically tied to this subject area. Many indicated that they came to consider the study of computer science though either an enjoyment of mathematics or having previously attained good grades in mathematics.

6.4.2 Extrinsic Motivation

Those who are extrinsically motivated focus on the outcome of their effort e.g. on their future career or job prospects. For some of the participants this is central to their reasons for choosing to study computer science:

> **Participant 6:** "If you look at the world, there are computers everywhere. They aren't going to go away. There are always going to be jobs for people who have a computer science degree."

The following participant appears to have been motivated to study computer science because it has been suggested to them as a suitable future career:

> **Participant 9:** "Because I'm good at maths, the Guidance Counsellor suggested I do computer science. I didn't really know that much about it, to be honest."

In the next quotation the participant indicated that their motivation was both social and extrinsic:

> **Participant 3:** "I'm motivated to do well because my family expect me to well. My Dad and my cousin helped me to get in here, so I want to do well because of that."

6.4.3 Independence

For many students their perception of their ability to succeed is tied to their ability to work independently. Participants often linked the learning strategies they had adopted at second level to success at the undergraduate level e.g. when they ran into trouble they tried to use the methods that had been successful for them in school:

> **Participant 12:** "If I don't get it, I read it myself.... that's what I did in school and when I got stuck"

> **Participant 1:** "I try to make sure I do some homework every day. Like in school you had homework ... but you don't have assignments every day here... so I try to find some homework to do, cause that's what I'm used to doing."

> **Participant 15:** "I usually don't get stuff in lectures, but if I sit down and do it myself then I kind of understand it. You just need the experience to be able to do it."

One student noted their inability to stick to any study plan that they create:

> **Participant 5:** "I do what I decide each day, I'm kind of funny like that. I notice if I spend my time making a plan I never stick to it. So, I've given up making plans."

Another form of independent learning that participants noted was that they taught themselves how to program. Almost all viewed this as beneficial:

> **Participant 18:** "I knew we'd be doing Java, so I got a book and did some programming in Java before I came in. So at the start I found I didn't need to listen, because it was all stuff I knew."

Another student described how they struggled when trying to learn independently:

> **Participant 1:** "I tried to teach myself how to program last summer. I found it really tough and I was really worried that I'd be bad at it."

6.4.4 Perception of Computing

One of the factors that influenced the participants belief in their ability to succeed were their perceptions of computing and computer science. For some these perceptions were extremely negative as they found the concepts to be completely alien to them.

6.4.4.1 Alien

Participants commented on how the concepts they met while studying computer science were completely outside their prior experience. Some participants expressed a fear of the unknown as they met subjects and concepts that were completely alien to them:

> **Participant 13:** "At first it was all different, I didn't get what was going on. I'd sit there in class and wouldn't get what we were supposed to do. I wasn't the only one, none of the group I sit with got it either."

> **Participant 1:** "For some of my friends they are studying things that kind of continue from school, but computer science is nothing like what you do in school. It's all completely different."

> **Participant 7:** "I was struggling because it was something I'd never done before. However, I knew I'd have to learn it so I've been trying to keep studying so it doesn't get away.... so I don't get too far behind."

While some participants noted differences, they didn't seem to be as concerned about it as their class mates:

> **Participant 6:** "Programming uses a lot of syntax, so its very different to the way we speak."

6.4.4.2 Challenging

Participants described how they found their experience as undergraduates to be very challenging in terms of the subject matter they are meeting and the pace at which lectures proceed.

> **Participant 13:** "Things go a lot quicker here. You never go back over it, it's always something new."

One participant commented on how missing one day would cause them concern and they can't understand one of their classmates who does not attend regularly:

> **Participant 7:** "Like I went home for a day, because I had an appointment, I find you miss so much in just one day... now he's just like, I said to him why aren't you doing that assignment, why don't you come in and he's like I don't come in for that class, there's no point, I can't understand it, what's the point.. and I'm like you might understand it if you come and he's like... I don't know."

For some this was a shared experience, while for others they felt they were experiencing problems in isolation:

> **Participant 11:** "We'd go into a tutorial and they'd say "Do this problem" and we'd look at one another cause we wouldn't know how to start. If you don't know where to start you can't even get going."

> **Participant 2:** "I get errors all the time when I try to program. It doesn't seem fair because nobody else seems to get these problems, I'm always missing something easy like a colon or something."

The following participant hadn't learnt to program prior to the commencement of their studies, but had already decided that it was going to be "hard" because this is what a relative had told them:

> **Participant 7:** "I knew I'd have to learn to program and I didn't really know what that was. I asked my cousin to show me and he showed me some assignments he'd written but that didn't really tell me what it was. I knew it was going to be hard though because he told me it was going to be hard."

6.4.4.3 Intimidating

Some students move beyond finding computer science challenging to feeling intimated and fearful. This is often expressed together with a sense of frustration. One participant described how one of their class mates dropped out of the course because they were intimidated by the subject matter

> **Participant 3:** "The other girl in the class, she left. She found it frustrating and she left. She was only interested in graphic design... she was getting really down, she said she felt like an idiot sitting there when she couldn't understand the stuff."

One participant explained how their interactions with a demonstrator during a laboratory class made them feel anxious:

> **Participant 2:** "The demonstrators aren't the best. One of them is very sarcastic and he always makes me feel like I know nothing. He doesn't seem to know how to get down to your level. He don't explain things and they say you should know it from your notes. I always feel very stupid afterwards."

6.4.4.4 Enjoyable

Not all participants were finding undergraduate life difficult as some described the enjoyment they were deriving from their studies.

> **Participant 1:** "Some times you put in a lot of work and you still get stuck, but you just keep at it and then when you get something to work the way you wanted it to it and you've spent ages at it then it feels really good"

The following participant describes how they move from feeling frustrated and stupid to feeling very pleased as they grapple with a challenging programming problem:

> **Participant 14:** "The first time you write a program it never works. It's really frustrating and you feel stupid. Then you learn that programs never work like that, you've got to keep fixing things until they do and then you feel really proud cause you've done it"

For one student the experience contrasted favourably with that of secondary school:

> **Participant 18:** "In school there was a lot of stuff you had to learn like in English you'd to learn those poems and in History you'd to earn all the dates off. It's just stuff you have to learn, while with computers you start of with something that's broken and then you've got to fix it and then when it works you feel really satisfied. You don't get that learning things off."

6.4.5 Self-Efficacy

Participants with high-self efficacy were likely to feel extremely confident about their ability to succeed, even when faced with tasks they might not know how to complete. By contrast, those with low self-efficacy raised concerns about their ability to complete assignments and expressed fear of failure. The following participant exhibits high self-efficacy derived from their pre-university experience as even though they meet a new topic they are confident they can do well:

> **Participant 14:** "We did do matrices in school, but they are very, very easy in the leaving cert. When I realised it was using matrices, I was like 'Yes! This is brilliant', but then you get into the row-echelon stuff and it's weird.... but at least you know you can do them."

Another participant with high self-efficacy rationalises a problem in an unfamiliar domain by comparing it to something they are familiar with. This gives them confidence in their ability, even though they admit the domain is completely new to them:

> **Participant 9:** "I didn't know any programming, but it needs maths, so I try to see if I can make it into a maths problem to make it easier. Like you have a left hand side and a right hand side and you have to make sure they match up so that the program works. It's kind of like those trig formulas we had to prove in maths where the two sides had to be the same."

The two participants with high self-efficacy quoted above are in marked contrast to their classmate quoted below:

> **Participant 2:** "It's not easy. I didn't feel confident when I started the course and I was right. I'd barely got a C in honours maths and I knew that I was going to find it hard, but it's been even more difficult than I thought it would be."

People with low-self efficacy are prone to blame themselves and their lack of ability when they find a subject particularly challenging

> **Participant 5:** "Things go very quickly and I can't seem to keep up. Other people seem to understand things straight away, but I don't. There's times when I'm getting nowhere with it because I'm not able to get it."

In this section the phenomenon labelled perception of competence has been detailed. Table 6.5 provides a summary of this category and its properties. This category differs from its two predecessors as it encapsulates the more subjective codes that emerged during open coding.

6.5 Language

The "language" category that emerged from the analysis captured the various ways in which the participants perceived that English language skills impacted on their success and the success of others. Indeed, the students' attitudes towards individual elements of their undergraduate studies were influenced by their perception of the accessibility of the language involved. The following subsections highlight some of the different elements of language that emerged during the coding of the data, supported by some excerpts from the data that gave rise to these codes. This phenomenon is summarised in Table 6.6.

In the social sciences the broad concept of culture captures aspects of learned and shared standards of thinking, feeling and acting. The abstracted category of language detailed below may be viewed as a specific subset of the cultural differences that participants perceive exist between students and the academic

Element	**Description**
Phenomenon	Perception of Competence
Causal conditions	Students have many motivations for studying computer science and these motivations may be intrinsic or extrinsic. For those entering university for the first time, the requirement to be able to work more independently can be a cause for concern and worry. Perceptions of computer science as an academic discipline may influence students' beliefs about their ability to succeed. Students with low self-efficacy may doubt their capabilities and be uncertain of their own competencies.
Context	For many students their undergraduate modules are the first time they formally meet many of the different aspects of computer science. Topics such a programming, telecommunications and hardware systems are completely alien to them. Students often have expectations and attitudes in relation to these course before they begin their studies and these can negatively impact on their performance. Those who have experience of working independently often find it easier to make the transition to university level study.
Intervening conditions	Participants who have sat through the same lectures and tutorials describe their experiences in very different ways. Some students feel intimidated and alienated. They can't related the material they are meeting to their prior learning experiences. Others find the material they meet challenging and enjoy tackling problems where the solution is not immediately obvious to them. Those who have engaged in independent learning prior to the commencement of their undergraduate studies seem more to have greater confidence in their own abilities. Those with low self efficacy blame themselves when they find course materials challenging.
Action strategies	Many emphasise the importance of keeping up with the course materials and warn against falling behind. Those that find the subject to be alien and intimidating, often emphasise the importance of constantly going over the course materials as a strategy for dealing with their problems. Some just give up as they feel intimidated. Those with high self-efficacy are very self-confident and relish the opportunity to engage in tasks and problems that are academically stimulating.
Consequences	Participants were able to identify that those who were not motivated to succeed or doubted their own ability to succeed were likely to struggle with the course. Students could view their studies from a positive perspective as challenging but enjoyable, while others could view their studies more negatively as challenging and intimidating.

Table 6.5: Perception of Competence

staff who teach them. Such cultural gaps are not unique to computer science classrooms. For example, these differences have been explored in wide range of educational settings and contexts: from kindergarden to university level and from the arts and humanities to science and engineering [26; 35; 76; 154; 166; 169; 216].

Those engaged in scholarship and teaching in specific disciplines naturally form what are known as communities of practice [130; 131] or discourse communities [19]. By their nature these communities have their own lexicon and style of communication. The participants in this study believed that students who are perceived to be successful are those who have the verbal ability and verbal reasoning skills that facilitate access to the computer science community of discourse.

The nature of writing in both university and workplace settings was explored by Dias et al. [71]. In their work they observed that university students were encouraged to adopt a style of writing that closely mirrored that of the recommended written and spoken texts, regardless of the subject area. Their findings are mirrored in the subcategories of "Academic Writing" and "Grammar and Sentence Structure" identified in this thesis.

Like other disciplines, communication within the field of computer science involves culturally learned, implicit assumptions of ontology and epistemology. These presuppositions have evolved and developed since computer science became an recognised research domain. Clearly such cultural differences are not unique to computer science classrooms. Unlike some of the more objective, domain specific elements of the other abstracted categories, this category may be generalisable across a much broader range of disciplinary settings.

6.5.1 Verbal Ability

Verbal ability comprises both verbal understanding and verbal fluency. Many students commented on the range of unfamiliar vocabulary they encountered during their undergraduate studies. Examples they cited included new meanings or context for words they were familiar with e.g. objects and methods in programming, or words and acronyms that they were unfamiliar with. Some commented that they just switched off as they were unable to comprehend what was being said.

Element	Description
Phenomenon	Language
Causal conditions	The student encounters problems in understanding what the lecturer is trying to communicate or in formulating a response to a question asked. This may occur when the student cannot break down a given problem into a sequence of subtasks which they can then complete. Students may be unable to break down sentences into their constituent elements.
Context	This can occur in lectures, tutorials and laboratory situations. It may be that the student finds they cannot fully grasp what the lecturer or demonstrator is saying to them. They find they cannot make sense of what they are being asked to do. The students feel they are unable to verbally reason through the problem and its solution; moreover they are unable to articulate where their problem arises. Not all students experience these problems and those who don't may be unaware of their classmates difficulties. Students experiencing difficulties may or may not be aware of where their problem lies.
Intervening conditions	Lecture notes and course materials given to the student to help them understand a concept. These may then be explained during the course of a lecture. The language used and method of presentation is such that the student feels unable to understand what is being communicated to them, perhaps because they do not engage in informative listening.
Action strategies	Given their lack of understanding, students may decide to disengage from a topic. This may be done consciously, where the student accepts they are no longer paying attention and justifies this to themselves by forming the intention to return to the topic in their own time, or unconsciously where the student lets their mind drift.
Consequences	Students identify this as a factor that causes them to lag behind their more successful class mates. The number of ways in which the student fails to comprehend what is being asked of them impacts on their perception of their success.

Table 6.6: Language

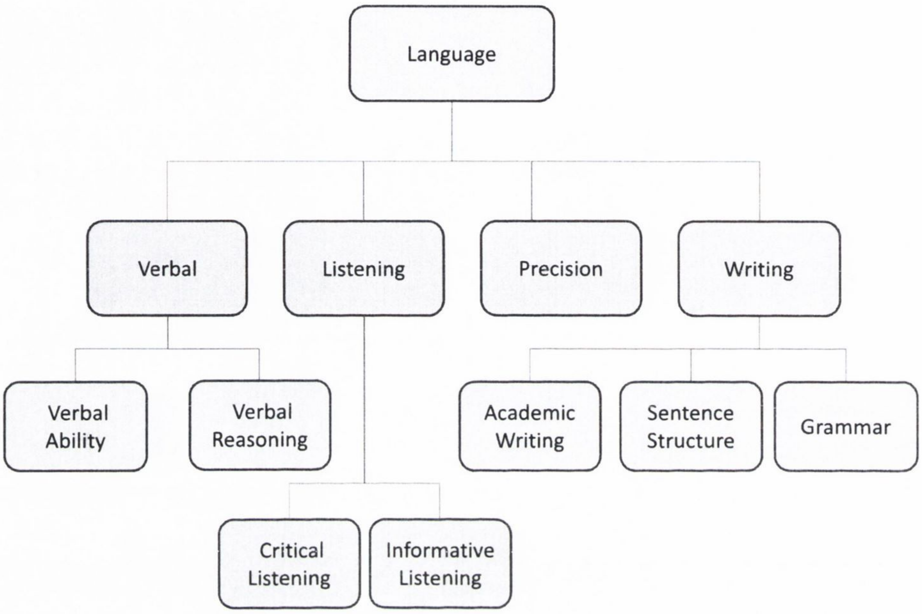

Figure 6.6: Language

> **Participant 2:** "The lecturer was speaking gobbledegook. I mean, each word individually was fine, but the whole lot together didn't make sense."

Students felt those who had met some computer science terminology before starting their studies had a distinct advantage e.g.

> **Participant 10:** "X had programmed before and knew all the lingo."

One student suggested that a dictionary would be useful:

> **Participant 16:** "If you'd a dictionary that explained everything in simple terms, in a way you could understand .... with maybe an example, that would be really useful."

Another participant expressed it as follows

> **Participant 12:** "I've got the lecture notes but it's all just these big words and I look it it for a minute and if it doesn't jump out at me I kinda just move on. Though I know the words must matter, if I don't get it I just move on."

Some of the students had also noticed the precision in language used by the academic staff and were aware that was something they would have to learn e.g.

> **Participant 17:** "The lecturer is very particular about stuff.... like how you define objects"

> **Participant 4:** "You have to be very precise, if you forget a bracket or a comma you're dead."

6.5.2 Verbal Reasoning

The students often expressed frustration about the difficulties they had with problem solving tasks where they were given a series of facts, either verbally or in writing, and were then required to understand and manipulate the information provided to solve a specific problem. This was expressed both in relation to their own work

> **Participant 5:** "You get this problem, you can read it, it all makes sense but you can't work out what to do"

or as an identified ability of those they perceived as successful:

> **Participant 2:** "X just doesn't have to think. He looks at it, sees the solution and just does it. His code is, like, perfect every time. I spend ages trying to work out the steps needed."

In some cases students referred to verbal reasoning as "logical thinking" e.g.

> **Participant 18:** "If you can think logically and break the whole thing down.... then work out what you need...... then it all makes sense. If you can do that then you'll find it easy."

6.5.3 Informative and Critical Listening

When a listener's aim is to understand the message being communicated to them they are engaged in informative listening [232]. A listener is deemed to have been successful when the meaning they give to the information obtained is that which the speaker intended. One of the participants described how they gradually lost interest and even though they attended class they weren't paying as much attention to what was being said:

> **Participant 15:** "At the start I was really interested. It was all new and I wanted to work hard. Then as time went on, I don't know, I kinda stopped trying or wasn't as interested. I'd kinda drift a bit in class, listening but not really getting it as I'd be thinking about something else. You say to yourself you'll look at it later and sort it then, but I never do."

Informative listening contrasts with critical listening where the listener must form an opinion on the information being communicated to them [232]. The critical listener must make decisions on (1) the trustworthiness and expertise of the speaker, (2) the validity of the arguments being advanced and (3) the strength of the logical inferences made or the conclusions drawn. Participants in this study indicated that those with better critical listening skills were more likely to be successful e.g.

> **Participant 4:** "If you can keep your brain focussed then you can just get it, but it goes at a very fast pace, there's a lot to take-in in each class and there's a lot of stuff you just have to learn off and... I just don't know, I can't get it all."

6.5.4 Academic Writing

The style of writing expected from computer science undergraduates in technical reports, dissertations etc. differs to that which they have previously experienced at second level. Students who enter university straight from school were most familiar with the construction of essays for humanities based subjects; though their focus was quite superficial as they referred to the number of facts you needed to include in such essays e.g. when a student was asked how they went about constructing an answer to a question on English literature, they responded:

> **Participant 10:** "It's simple really, you make three points and you give three quotes for each point ."

The students felt that the style of writing they encountered at university was very different in terms of the structure required and the language used. As one student put it:

> **Participant 2:** "You have to structure it right, it's not about just including the right stuff, it's about how you present it."

One of the students commented on the fact that the emphasis is more on the process of getting the result than the result itself:

> **Participant 7:** "It's how you get to the result, how you designed the thing to get the result as opposed to the result. Even in mathematics it's not about the result it's about writing down the steps to get there so people can understand it... then you can go back and reuse it or extend it, cause its got the right format."

6.5.5 Grammar and Sentence Structure

Grammar refers to the set of rules that are used to construct expressions in a given language. While local dialects of a language may involve, sometimes unwritten, rules, there is a set of standardised grammatical rules that are widely accepted as correct. In English these rules vary slightly between different accepted forms of the language, e.g. between British English and American English.

A number of participants observed that the academic staff spoke "differently" to them and felt that this made it harder for them to understand what was being communicated to them. One student commented on the long sentences used by some faculty members while another commented that academics "spoke differently", although they were not able to pin down precisely what these differences were.

The participants were all learning object oriented programming and some students felt that a more through appreciation of the sentence construction, e.g. subjects, predicates, and elements of English grammar, e.g. verbs, nouns, pronouns, prepositions, conjunctions, adjectives, adverbs, interjections would help them be more successful. It was felt that those who could de-construct sentences correctly were most likely to come up with the correct objects, classes and methods to use. Many lamented the fact that they had only touched on sentence construction in school and even then it had often been at primary level. One participant commented that their knowledge of grammar came from learning German as a second language and that they had no recollection of learning English Grammar in a formal school setting.

6.6 Selective Coding

One of the abstracted categories have been identified the researcher must turn their attention to the identification of a central phenomenon. For each abstracted category the researcher must be able to articulate how it relates to this central category. Strauss and Corbin provide the researcher with criteria to use when choosing the central category[204, p.147]:

1. It must be central, that is all other major categories can be related to it.
2. It must appear frequently in the data. So that within all, or almost all cases, there are indicators pointing to that concept.
3. The explanation that evolves by relating the categories is logical and consistent. There is no forcing of data.
4. The name or phrase used to describe the central category should be sufficiently abstract so that it can be used to do research in other substantive areas, leading to the development of a more general theory.
5. As the concept is refined analytically through integration with other concepts, the theory grows in depth and explanatory power.

6. The concept is able to explain variation as well as the main point made by the data, that is when conditions vary, the explanation still holds, although the way in which a phenomenon is expressed might look somewhat different. One should also be able to explain contradictory or alternative cases in terms of that central idea.

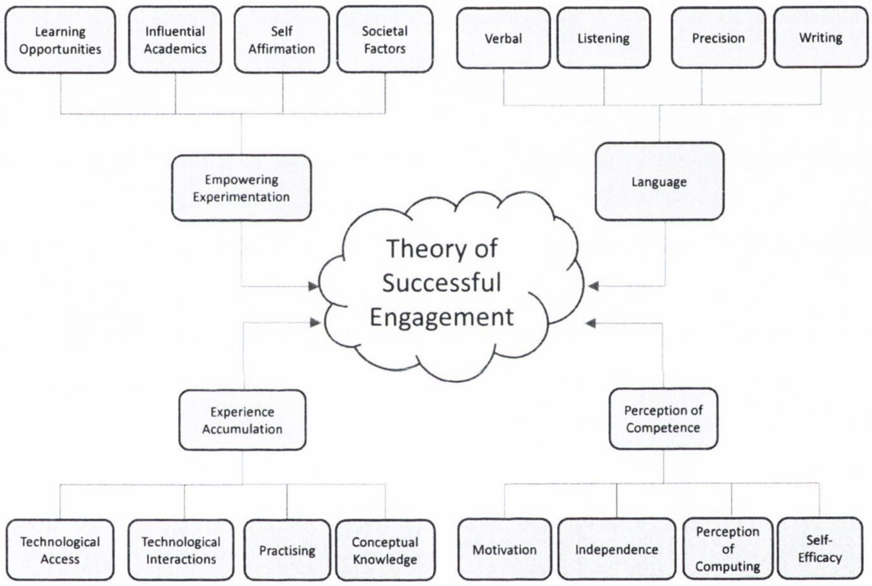

Figure 6.7: Theory of Successful Engagement

6.6.1 The Theory of Successful Engagement

Participants in this study articulated many experiences that they felt were common to those who were successful. One recurring central theme was that of prolonged engagement with computers and technology in a way that positively reinforced the participants self-confidence and belief in their ability to succeed. Moreover, this prolonged engagement meant that those who were likely to be successful have assimilated the language and culture of computer science before commencing their undergraduate studies. For these reasons the theory generated by this study was best encapsulated in a core category labelled "successful engagement". This central category and its associated subcategories are illustrated in figure 6.7. This core category satisfies the criteria set out by Strauss and Corbin [204] as it includes and assimilates all of the key phenomena uncovered during axial coding.

When seeking the central category Strauss and Corbin [204] advise the reader to stand back and get a general sense of that which the data is telling them. They also suggest that the researcher ask "what comes through, although it might not be said directly" [204, p.148]. These two pieces of advice proved very judicious as it is was necessary to step back from the somewhat disparate abstracted categories and return to the data to explore how the concepts revealed by the analysis fit together into one central, recurring theme. This meant not only re-reading some of the transcribed interviews, but also listening afresh to the recorded conversations in light of the analysis conducted. The notion of success is central to all of the abstracted categories; from the success of those with high self-efficacy, to the click moments experienced by those engaged in practising, to the critical listening skills needed and the importance of language. The participants gave voice to the importance of prolonged engagement with technology from the wide range of interactions with computers at school and at home, through to the intrinsic motivation that can lead to the creation of independent learning opportunities. The participants' perspectives provides a rich and compelling articulation of the theory of successful engagement.

6.7 Conclusion

The objective of this chapter was to present the reader with a thorough exposition of the process of data analysis that led to the identification, and emergence, of the central core category of successful engagement. The initial codes and key concepts identified during open coding were presented. These formed the basis for axial coding, where four central phenomena were articulated. For each of these phenomena and its associated subcategories, quotations were used to justify their inclusion in the emerging theory. Finally, it was necessary to return to the data with these abstracted categories in mind to obtain the central core category which integrates these phenomena into the central theory of successful engagement. In the following chapter we review these processes in order to confirm the validity and reliability of the theory of successful engagement .

Chapter 7

Validity and Reliability of the Findings

Qualitative researchers can draw on well-established numerical methods and specialist mathematical techniques for the analysis of data, allowing others to easily confirm the validity of any results presented. As mentioned in section 3.6, when qualitative methodologies are employed there is no single universal accepted definition of validity. This chapter returns to the concepts of trustworthiness previously discuss and reviews how the outcomes of this work were validated.

Many qualitative methodologies require the researcher to validate their theory by presenting it to the participants and asking if they feel it accurately reflects their situation and experience [180]. This is naturally addressed in a grounded theory study as the constant comparative analysis and theoretical sampling require the researcher to return to the participants to check the concepts and categories as they arise during the analysis. A grounded theory is one that emerges from the data, and the participants in the study are usually unaware of data collected on others experiences or observations. Participant validation is usually not considered appropriate as a means of validating the abstracted categories and the emerging theory as they are usually not readily accessible to the participants[92]. Indeed, Glasser [92] indicates that asking participants if the theory reflects their voice is not an appropriate validity check or test. A grounded theory is not expressed in the voice of the participants, rather it is a generated abstraction from the concepts generated by the analysis of the collected data. As a consequence participants may or may not understand the theory; or even if they do understand it, they may not like it or approve of it.

7.1 The Strauss and Corbin Validity Checks

In the context of a grounded theory, validation relates to the process of ensuring that the theory generated during the course of the study is one that "fits" with the collected data i.e. the theory that emerges is an appropriate abstraction of the raw data collected [204]. The theory must not only fit the data, it should also be dense; none of the concepts uncovered should be omitted from the theory. Strauss and Corbin express the notion of fit as follows: "... once an analyst explains in detail how he or she arrived at a conceptualization, other researchers, regardless of their perspective should be able to follow the analyst's path of logic and agree that it is one plausible explanation for what is going on" [204, p.146].

Strauss and Corbin [204] recommend that the researcher return to the inital data and compare the theory generated to it in a form of high-level comparative analysis. This was carried out throughout the course of this study. Numerous passes were made through each data file to make sure that the initial open codes used were a good match for the data. As these codes were combined to form concepts the researcher was constantly checking that vital codes were not overlooked or omitted, as this would result in a reduction in the richness of the theory generated. When an abstracted category was identified the researcher returned to the existing data and labelled all instances of the category. This process helped the researcher to confirm that the category choice was both appropriate and grounded in the data. Strauss and Corbin [201] also provide the researcher with a set of seven specific recommendations for the validation of the research process and a set of seven criteria that can be used to confirm the empirical grounding of the theory. Both of these sets of recommendations are considered in the context of this study in the following subsections.

7.2 Validation of the Research Process

One of the key validity checks for any theory is that those who were not involved in the data collection and analysis are able to form an accurate judgement on the conduct of the study and the analysis of the data. Strauss and Corbin suggest a set of seven criteria that can be used by the reader in the evaluation of any grounded theory study [61]. These criterion are reproduced in table 7.1. To allow the reader to assess the adequacy of this work, the questions posed by Strauss and Corbin are addresed in the context of this study below.

7.2.1 Sample Selection

The initial sample of participants for the study was selected using the open sampling methodology proposed by Strauss and Corbin [204]. This requires the research to identify individuals or objects that may provide data from analysis. The relevance of the initial data collected to the research question is assessed early on, before too much time and effort has been invested. In this study it was also necessary to take account of a number of real world constraints. For example, as the participants were engaged in a course of study it would not have been appropriate to seek to interview them around exam time or around times when a significant number of their course work deadlines fell due. Also, as the study aimed to explore the pre-university computer experiences the students felt were most beneficial, it was necessary to allow them a number of months of experience at third level before interviewing them.

The overall objective of the study was determined before the collection of data begun and so the research questions that guided the study were not completely open. As the study aimed to investigate the experience of freshman students, any first year student of computer science was a potential rich source of data for the study, so the initial sample was drawn from this cohort.

Initial contact was made with the students at a short briefing session given by the researcher at the beginning of one of their lectures. During this introduction it was emphasised that the study sought participants from all those in the class from those doing very well and finding the course very easy to those who were struggling and even to those who had already decided that this course of study

Criterion 1:	How was the original sample selected? What grounds (selective sampling)?
Criterion 2:	What major categories emerged?
Criterion 3:	What were some of the events, incidents, actions, and so on that (as indicators) pointed to some of these major categories?
Criterion 4:	On the basis of what categories did theoretical sampling proceed? That is, how did theoretical formulations guide some of the data collection? After the theoretical sampling was done, how representative did these categories prove to be?
Criterion 5:	What were some of the hypotheses pertaining to conceptual relations (that is, among categories), and on what grounds were they formulated and tested?
Criterion 6:	Were there instances when hypotheses did not hold up against what was actually seen? How were these discrepancies accounted for? How did they affect the hypotheses?
Criterion 7:	How and why was the core category selected? Was this selection sudden or gradual, difficult or easy? On what grounds were the final analytic decisions made?

Table 7.1: Validation Criterion for the Research Process (source: [61])

was not for them. All students in the first year cohort were invited to participate and to specify a preference for the type of data collection activities they might be interested in engaging in.

Before sampling began the researcher was concerned that the voluntary nature of participant selection might give rise to some difficulties as the full continuum of student experience might not be captured e.g. those who were finding the course challenging might not have been willing to take part and to speak openly and frankly about their experiences and their perceptions of how it related to the experiences of others. However, in reality this concern proved to be unfounded as the volunteer participants spanned the whole range of experience.

7.2.2 Major Categories that Emerged

Once interviews commenced a number of major categories began to emerge from the analysis of the data collected. Many students commented on the problem solving nature of computer science and how, when they found something challenging, they needed to persevere in order to understand what was happening. They expressed a sense of achievement and enjoyment in the tackling of such problems. While such problems solving was often related to mathematical ability the students were unable to articulate exactly which aspects of their mathematical knowledge they felt helped them feel successful.

> **Memo:** "Another student who says they don't understand the lecturer. I must explore this more, what aren't they understanding? Is it the subject, the accent, or what?"

> **Memo:** "It's as if they've been told maths and problem solving are important for computer science. Or that they'll like computer science because they like maths and problem solving. But they don't seem to be able to explain the link."

> **Memo:** "Is it the enjoyment they get from persistence at a problem they can't do? With the reward being when they finally get the answer or see how it works? It's a bit like the click moments others have found in work on learning programming. Is mathematics a subject more prone to click moments and so they associate the two?"

The category of language also emerged very quickly as students articulated a number of aspects of language that they felt were holding them back. However, this category did not emerge fully formed, as the initial coding of the data threw up a number of aspects of language in very different settings e.g. the accent and vocabulary used by academics, the breaking down of programming problems into appropriate sub-problems. It was only through subsequent analysis that the associated codes were related through the identification of the category of language.

> **Memo:** "It seems like there is a cultural difference between the student and the academic, as if they come from different worlds and speak in different ways."

> **Memo:** "Language (again!). The student is talking about how they read it but it doesn't make sense. They can't seem to interpret what they are being asked to do or associate the correct meaning with the words."

The major categories found in this study emerged naturally from the data. The examples of theoretical memos given above illustrate how topics for future theoretical sampling were identified during the initial open coding and analysis of data. While the coding of the first data sets gathered was extremely time consuming, it was found that microanalysis of the data and the time taken for memoing provided a wealth of suggestions for further exploration.

7.2.3 Events and Incidents that pointed to Major Categories

Borgatti's Basic Frame of Generic Relationships [30] was found to be of assistance during Axial coding. The tables associated with the phenomena uncovered are given in section 5.2. For each phenomena these tables summarise and capture its properties and dimensions.

The causal conditions, context and intervening conditions for each phenomenon were the pointers that led to their identification, and confirmation, as the major categories of this work. So these frameworks act as a mechanism for the reporting of the key indicators that lead to the creation of these major categories.

7.2.4 Theoretical Sampling Criterion

Theoretical sampling typically arose as a result of theoretical memos recorded during open coding. It was important that the questions asked during theoretical sampling were not "theory forcing" i.e. that the participants were not asked leading questions, rather when the opportunities were presented the researcher asked more specific questions designed to explore the variations in the properties and dimensions of the category. For example, in the language category when students mentioned a difficulty in understanding what they were being asked to do, the researcher then explored this in more detail with the participant, teasing out the sources of the problem and the skills they felt others had that made them better equipped to cope with the problem. For example, in this context some of the questions asked were: "Where do you think the difficulty was for you?" or "What was it about some of the people you consider to be successful that made you think they were better at doing this than you were?".

It should be noted that while participants were selected through open sampling, theoretical sampling only occurred when opportunities presented themselves in later interviews. If a topic of interest for theoretical sampling did not arise naturally in the course of an interview then it was important that the researcher did not deliberately raise the topic as that would be contrary to the inductive method underpinning grounded theory. It was found that it was possible to carry out theoretical sampling in this way and that it proved a very useful tool for exploring possible categories more fully.

7.2.5 Hypotheses Pertaining to Conceptual Relations

The use of the word hypothesis here is slightly problematic, as it has a distinctly different usage to the familiar one used in statistical hypothesis testing. Here we use the term as defined by Strauss and Corbin [204], who call the initial hunches that the researcher has about how the concepts are related to each other "hypotheses". During the early stages of data collection the researcher may begin to formulate ideas about how concepts and codes might be inter-related; and as data collection and analysis proceeds, these ideas may coalesce to help the researcher form the abstracted categories and phenomena. These hypotheses are considered to be abstractions as they explore the links between concepts, not codes.

It was found that the data exhibited many links between the emerging categories. For example, in the following quotation, the student is articulating difficulties associated with "language", while also indicating that they are "persistent" as they will keep going even though they know they are having difficulty understanding the concept:

> **Participant 12:** "I've got the lecture notes but it's all just these big words and I look at it for a minute and if it doesn't jump out at me I kinda just move on. Though I know the words must matter, if I don't get it I just move on."

In the quotation below we see another example of a student being "persistent" and indicating that they feel this contributes to their success. They also mention their reliance on co-operating with "peers" for advice when necessary:

> **Participant 7:** "My advice would be, don't give up on something if you don't see what it means at first. There are things I've learnt that if I'd given up and stopped trying, I never would have got them. Sometimes when I really don't see what it's about, I ask some of the others in the class and they explain it to me"

As data collection advanced it was found that the testing of hypotheses in this manner was more akin to a form of defeasible reasoning [165] than hypothesis testing; as suppositions about the relationships between categories were open to revision if evidence from subsequent data collection did not reinforce the hypothesised relationship. Strauss and Corbin [204] provide some guidance on how to proceed when the evidence appears to be conflicting. They suggest that the researcher may find that some of the inconsistencies found may be attributed to variations in the phenomenon rather than being contradictions within the data.

7.2.6 Where were the Hypotheses Deficient?

The initial data collected was examined using the process of microanalysis. This pain-staking process suggested many relationships between the different codes in the data. In particular it was noted that many students were inclined to view computer science as programming. When participants were asked about factors that influenced success in computer science they often interpreted this as factors that make people good computer programmers. While this link between computer science and programming was very strong, it was felt that it did not form a saturated category. Hence, its exploration fell outside the scope of this work, although it could prove a fruitful area for further research.

Another link that was not explored was that of the perceived relationship between mathematics and problem solving, and how those who are "good" at mathematics should be "good" at computer science. Many of the participants expressed the widely held belief that if one is successful at mathematics and received high grades in this subject in the state examinations then one will automatically be successful when studying computer science. Again, it was felt that the exploration of the origins of this belief and whether it was the experience of the participants, was beyond the scope of this work. This, too, could form an interesting topic for further exploration.

7.2.7 How was the Core Category Selected?

The core category of this work and the methods used to find it were articulated in sections 5.4 and 6.6.1. From these, and from the preceding analysis, it can be seen that this core category emerged through the constant comparative methods of grounded theory where the researcher repeatedly returns to the data to check that the abstractions made are grounded in the data.

7.3 Empirical Grounding of the Findings

The criteria considered in the previous section focused on the validation of the research process, providing the reader with an insight into the different components of the research process from the selection of participants through to the coding of data and the articulation of the core category. In this section, a second aspect of the evaluation of the theory is considered, the empirical grounding of the findings. As before, Strauss and Corbin [61] provide a set of seven criteria that can be used for this validation, and these are summarised in table 7.2. Each of these criteria is examined in the context of this study below.

Criterion 1:	Are concepts generated?
Criterion 2:	Are the concepts systematically related?
Criterion 3:	Are there many conceptual linkages and are the categories well developed? Do they have conceptual density?
Criterion 4:	Is there much variation built into the theory?
Criterion 5:	Are the broader conditions that affect the phenomenon under study built into its explanation?
Criterion 6:	Has "process" been taken into account?
Criterion 7:	Do the theoretical findings seem significant and to what extent?

Table 7.2: Empricial Grounding of the Findings (source: [61])

7.3.1 Are Concepts Generated?

As Strauss and Corbin [61] put it "the basic building blocks of any grounded theory.... is a set of concepts grounded in the data". For a grounded theory study to be convincing it must be clear to the reader that the source of the concepts must be the data collected. Initial coding was carried out using the process of microanalysis and this lead to the creation of a large number of possible codes. During microanalysis, the researcher's objective is to identify all possible codes within the data, regardless of their likelihood of inclusion in abstracted categories at a later stage. It was necessary to constantly review the codes generated, both at the end of the analysis of an individual interview transcript and after the analysis of a number of separate interviews. In some cases it was necessary to relabel codes to more accurately reflect the data, while in others they were joined together to form more meaningful concepts. Many inital codes were in-vivo codes, that drew on the language used by the participants; in some cases these were later re-labelled as they formed part of larger concepts.

An example of open coding was given in table 6.1. Here initial in-vivo codes of "time on the computer" was later re-labelled as "Computer Use", while another in-vivo code was "Frustrating". The initial in-vivo code "Facebook" formed part of the larger concept of "Social Media", while the in-vivo code "completely new" was re-labelled as novel.

As the data analysis progressed, similar concepts emerged and provided evidence for the development of categories. In some cases the labels used for categories were the in-vivo codes from the original microanalysis, though it was more usual to find that a broader label that captured the nature of the category more fully was needed. It was also necessary to review all documented evidence of individual categories to help determine its properties and dimensions e.g. for the categories of computer use it was necessary to explore the impact of duration, the nature of the computer use and the perceived benefit of the use in order to fully capture its dimensions and properties.

The time-consuming process of microanalysis and the subsequent reviewing of the codes and the generation of concepts was necessary to ensure that the analysis was firmly grounded in the data. If the concepts used were merely drawn from common use then they would not be grounded in the data and the theory generated would have no basis in the data. Thus it was important to make sure that the data gathered was accurately reflected in the concepts and categories generated.

7.3.2 Are the Concepts Systematically Related?

For a theory to be firmly grounded in the data, the links between concepts need to be fully explored in the context of the data. While the researcher may hypothesise about the existence of links between concepts if this is not confirmed by the data then the hypothesis must be discarded. The researcher needs to engage in a systematic interrogation of the data to explore the inter-relationships between all of the key concepts identified. This requires the researcher to return to the data over and over again to seek out these relationships and confirm that they are supported by the data.

Strauss and Corbin [61] caution that the systematic relationship of concepts can be hard to present to the reader, as it is usually the case that they are woven into many layers throughout the iterative analysis of the data and it can be hard to untangle them for a systematic presentation to the reader.

One example of the inter-relationship of concepts is contained in the following quotation:

> **Participant 4:** "If you can keep your brain focussed then you can just get it, but it goes at a very fast pace, there's a lot to take-in in each class and there's a lot of stuff you just have to learn off and... I just don't know, I can't get it all."

Here the participant comments on aspects of the language category in terms of their critical listening skills, particularly in relation to the associated dimension of speed. The student also finds themselves over-whelmed by the volume of material they are expected to take on board during one class, again this gives us a dimension for the quantity to associated with the category of informative listening. The student also expresses their frustration with their perceived lack of ability to understand. Frustration was found to be one of the sub-categories of the perception of computer science. From the above quotation we may infer from the data that is possible that the ability to listen critically for a prolonged period of time and quantity of the lecture material one is supposed to take in through informative listening can influence one's perception of computer science as frustrating.

The above example shows that while on the surface there may not be any links between the category of the "perception of computer science" and that of "informative and critical listening" when one explores the data this tentative relationship becomes apparent. Participant 4 was one of those interviewed early in the study and it was deemed necessary to explore further this possible link between categories through the analysis of existing data, and through subsequent theoretical sampling. This is only one example of the types of relationships between concepts that were suggested by the data. This higher level analysis required a different skill set to that of microanalysis and while it was felt that this process threw up many false trails in terms of linkages, it also proved to be a key means of identifying the properties, dimensions and relationships between the categories and the phenomena.

7.3.3 Are there many Conceptual Linkages? Are the Categories Well Developed?

Strauss and Corbin [61] caution that if the number of conceptual relationships identified is small, then this "leaves something to be desired in the overall grounding of the theory". It is expected that the categories identified should be tightly linked to both their subcategories and to the central category. Moreover the categories should be theoretically dense i.e. the properties of each category should be dimensionalised. Strauss and Corbin [61] argue that this is what "gives a theory its explanatory power".

The analysis methods used in grounded theory require the researcher to constantly return to the data to test their hypotheses about possible links between concepts, subcategories and phenomena. This process of constant comparison takes place at all stages of the analysis; from the initial open coding, to axial coding and on to the identification of the core category. It is only when the researcher feels that no more links can be found, and that gathering more data will not yield further information, that it can be determined that saturation has been achieved and that the concept or category is fully developed.

Each of the phenomena developed was articulated in terms of its sub-categories. Indeed, these sub-categories were developed first and it was only through the exploration of their properties that the phenomena emerged. The conceptual linkages between the phenomena and their subcategories were presented along with their properties and dimensions, in section 5.2. From this we can see that the requirement for many conceptual linkages was satisfied through the careful conduct of axial coding.

7.3.4 Is there much Variation Built into the Theory?

Strauss and Corbin [61] explain that, while it is possible for a qualitative study to report on a single phenomenon with limited information on its characteristics, this is the antithesis of a grounded theory study where a number of phenomena should be specified fully in terms of their dimensions and properties. By necessity this means that the phenomena in a grounded theory study should be examined from all variational perspectives to ensure that the resultant theory has captured the full range of conditions and dimensions for each concept.

In this work the grounded theory presented has been fully described in terms of its phenomena and their associated subcategories. Section 5.2 provides a full description of all of the categories uncovered. The categories provided are justified through the provision of supporting quotations from the data collected. Each category is also examined in terms of the variation in its properties and dimensions, as the exploration of these helps to confirm that the category is fully saturated.

If we take the category of "influential academics " as an example, we see that it is part of the "empowering experimentation" phenomena. It contains three properties that differ significantly in terms of their dimensions. The first is the nature of the influence, as the influence of a teacher on the students experience may be positive or negative. It was seen that some teachers are very supportive, introducing students to learning opportunities and others outside of the classroom that will help to foster their interest in the subject. Other teachers were discouraging either because they felt that the time being spent on computer-related activities was detracting from the students performance across the curriculum, or else because they were unable to understand or control the students computer related activities.

Another dimension of "influential academics" is that of duration i.e. how long did this engagement with the student go on, was it just one or two isolated conversations or discussions between the student and their teacher, or was it a sustained involvement through support for activities that spanned the school year e.g. with project work for submission to the Young Scientists competition. This dimension is closely tied to that of extent, where the nature of the influence varies from a number of comments made to the student in conversation to more active engagement with the student in terms of the provision of equipment and support, or the denial of access to computers in the school environment.

7.3.5 Are Broader Conditions that Affect the Phenomenon Built into its Explanation?

While the aim of any work carried out using the grounded theory methodology is to create a substantive theory that is grounded in the data, it is important that the researcher be open to the possibility that factors beyond those directly contained in the data may be at play. For example, not all students who opt to study computer science may be doing so because of their interest in the subject, they may be motivated by the prevailing economic conditions. Many students opt to study computer science due to the perceptions the students, their parents and their guidance counsellors have of the job opportunities open to graduates in the field. Other macroscopic influences may include social trends and cultural values. In order for a theory to be truly grounded it must be open to the possibility that these broader, external factors might need to be considered.

One example of external factors at play is the way that professional bodies, industry and government have sought to influence the perception of computer science; changing it from that of a lone, male programmer who works in isolation to that of a profession open to those who wish to work together to create the knowledge economy of the future. Further examples are the societal perception of mathematics as being a good predictor of one's aptitude for computing and also that those who like the problem solving dimension of mathematics will also enjoy learning how to program. These external factors were mentioned by participants in this study, though most were unable to relate these to their own experience.

Another external factor is that of the second level curriculum, where there are extremely limited opportunities for students to study any form of computer science directly. From the statistics made available by the Irish State examinations commission on their website, www.examinations.ie, only 1.2% of all students study ICT related topics at higher level while 0.6% of all students study them at ordinary level. Thus the Irish Department of Education has a direct influence on the variation in the opportunities students have to gain relevant computing experiences at second level.

Strauss and Corbin [61] advise the researcher to make sure the broader, external factors that influence a phenomenon are directly linked to it and not simply provided as "background" material. The examples above show how such "macroscopic" sources were used to extend the conditions relating to the phenomena beyond those contained directly in the text.

7.3.6 Has "Process" been Taken into Account?

Strauss and Corbin [61] describe process as "stages or phases and also as fluidity or movement of action/interaction over the passage of time in response to prevailing conditions". In order to take this into account, the researcher must be aware of, and sensitive to, the evolving and unfolding nature of events. The notion of process is directly linked to axial coding where the researcher seeks to describe each phenomena in terms of its context, causal conditions, intervening conditions, action strategies and consequences.

Each individual responds to situations in different ways, so it is most likely that the processes involved will vary. For some individuals a given action strategy many give rise to a change in context. In this study some students found that perseverance meant that they grasped a given topic. This led to a change in context for them as it reduced their anxiety when confronted with other problems in the future. It also increased their motivation to attempt to tackle such problems and as they felt rewarded when they managed to find an appropriate solution. It was also noted that students can move between a state of frustration and feelings of achievement and success as they engage with the topic. Participants who viewed themselves as successful were more likely to anticipate, and plan for, feelings of frustration. They were less concerned about this, as they began to feel it was a natural part of their studies. Students who were less successful were more inclined to view these feelings as frustration with a sense of despair, as they felt that they were never likely to find a solution to the problem and be in a position to move forward to the next challenge. This shows that the subcategory of persistence is related to that of frustration and how process is taken into account to explain the links between them.

7.3.7 Are the Theoretical Findings Significant?

Within Ireland, some have used statistical methods to predict programming performance [16], while others have looked at the success of retention initiatives [57; 60] and explored ways to alleviate anxiety amongst those learning to program [58; 59]. Research has also been conducted into the cognitive learning styles of freshman students [158] and provided evidence of trends in mathematics performance of technology students over almost a decade [86]. This researcher has also contributed through her work on supporting novice programmers [109; 112; 192] and on the applicability of the concepts of computer anxiety, self-efficacy and computer experience to computer science and engineering undergraduates [72; 107; 108].

By contrast, there have been a very small number of qualitative studies that involve undergraduate computer science students in Ireland [74; 194], and both of these studies have only focused on the novice programmer. One of these studies [193; 194] was carried out by this researcher and one of her Ph.D. students. It used phenomenography to explore the students' perspective on learning object oriented programming. The second study was a multi-institutional study of the learning of computer programming [74] carried out using the Grounded Theory methodology. The work detailed in this thesis differs from these as it focuses on the computer experiences that undergraduate students feel help them succeed at third level. Thus this study provides an important qualitative assessment of the computer experiences that influence success at the undergraduate level. In the wider, international context this study provides one of the only qualitative explorations of the construct of computer experience as articulated by Smith et al. [187] and Prince [168].

Within this study some of the phenomena, and their subcategories, can be readily associated with factors that have previously been explored quantitatively in the literature. However, the theory produced in this study is firmly grounded in the data and the associated phenomena provide an accurate exposition of the experiences that the participants felt influenced success at the undergraduate level. By its nature, the theory of successful engagement provides a richer, qualitative exploration of the interactions between these factors, and establishes how they can be embedded in more encompassing phenomena.

7.4 Creswell and Miller's Validity Checks

Creswell and Miller [64] provide a different perspective on determining the validity of a qualitative study. They suggest that validity should be viewed through three different lenses: that of the researcher, that of the study participants and that of those external to the study. They observe that these are not the only means for choosing the appropriate validity check to use, and that the researcher's worldview also has a role to play. The three world views that Creswell and Miller draw on are the post-positive, constructivist and critical paradigms, discussed in section 3.5. Using these three different paradigms and the three different lenses, Creswell and Miller set out a two-dimensional framework that encompasses nine different validity procedures. These are shown in table 7.3.

As noted in section 3.5, grounded theory can be considered through the lens of many philosophical paradigms. Indeed, Glaser contends that it is "paradigmatically neutral" [91]. This neutrality becomes apparent when one considers the validity checking procedures suggested by Creswell and Miller. In this section we will review each of these validity checking procedures in the context of the grounded theory methodology and the study presented in this thesis.

7.4.1 Triangulation

The grounded theory approach relies on the constant comparison methodology as a means of ensuring that the emerging theory is grounded in the data. This

Paradigm assumption/Lens	Postpositivist or Systematic Paradigm	Constructivist Paradigm	Critical Paradigm
Lens of the Researcher	Triangulation	Disconfirming evidence	Researcher reflexivity
Lens of Study Participants	Member checking	Prolonged engagement in the field	Collaboration
Lens of People External to the Study (Reviewers, Readers)	The audit trail	Thick, rich description	Peer debriefing

Table 7.3: Validity Procedures Within Qualitative Lens and Paradigm Assumptions (source: [64])

means that the methodology incorporates triangulation across all possible data sources. In this study data was collected through individual interviews, focus group sessions and weekly journals.

Triangulation took place in the constant comparison of data across the interview and focus group sessions. Concepts and categories are only accepted once they are repeated across multiple data sources, if they do not stand up to this continual scrutiny then they are set aside.

7.4.2 Disconfirming Evidence

Triangulation involves a search for a convergence of opinion and corroborating evidence across multiple and different data sources. A closely allied procedure is the search for disconfirming or negative evidence [152] where the researcher returns to the data to search for evidence that contradicts the codes and categories identified. This also forms part of the constant comparative process where the researcher is constantly looking for similarities and differences across the data collected and drawing on these to assist in the articulation of the properties and dimensions of categories. As disconfirming evidence is uncovered the researcher must reconsider, and where necessary rewrite, their hypotheses; resulting in a more accurate articulation of the substantive theory embodied in the data.

Crewell and Miller [64] note that this search for disconfirming evidence is a constructivist approach as the researcher relies on their own lens when examining the data. Moreover, as the constructivist's reality is multiple and complex, it is natural to expect the data to contain both confirming and disconfirming evidence. Provided the later does not outweigh the former, then existence of disconfirming evidence provides further support for the credibility of the emerging theory within the constructivist world.

7.4.3 Researcher Reflexivity

This validity procedure requires the researcher to review their own assumptions, presumptions, beliefs and biases before they commence any study. They must then endeavour to bracket or suspend these biases so that they do not impose these beliefs on the data. In this study, the reflexive bracketing carried out by the researcher has been discussed in sections 2.3.1 and in 3.3.1. Reflexive bracketing falls within the critical paradigm as the individual researcher must reflect on the social, cultural and historical forces that shape their worldview [64].

7.4.4 Member Checking

As part of the grounded theory methodology the researcher moves between phases of data collection, coding and theory generation. In order to explore the properties and dimensions of each category identified, the researcher must engage in theoretical sampling. By its nature, theoretical sampling is a form of member checking as the researcher is seeking to confirm the credibility of the categories already identified and to extend understanding of their properties and dimensions, see section 3.6. During this phase of the study, the researcher must choose the questions they ask careful to ensure they do not engage in theory forcing. Member checking is a form of systematic check as the lens is focused on the study participants.

One of those most common strategies for member checking is the use of focus groups. In this study the focus groups were primarily used as a means of reviewing and validating the findings. This was particularly important near the end of the study as the phenomena because well established and the substantive theory became more refined. However, the researcher must be aware that this form of validity checking may not be appropriate as the abstracted theory is not expressed in the voice of the participants. Hence, it may seem unfamiliar to them. In this study it was found that the focus group participants could see themselves or their classmates in the experiences captured by the abstracted categories.

7.4.5 Prolonged Engagement in the Field

Constructivists recognise that the longer the researcher stays in the field, the wider the range of perspectives they are likely to capture and the greater their appreciation and awareness of the context becomes. Prolonged engagement is an important part of any grounded theory study as, over time, the researcher must return to the participants to check their hunches and extend their perspectives on the data. This extended contact with participants helps to solidify and confirm the codes and categories identified by the researcher. Moreover, the researcher in this study has been engaged in the field of computer science education for many years and so can relate to the culture and context of the participants' world.

7.4.6 Collaboration

Collaboration is part of the critical paradigm because "the intent of the process is to respect and support participants in a study, not further marginalise them" [64]. In this work the collaboration between the researcher and the participants was in relation to how the outcomes of the study might be used. Potential participants were informed of the objective of the study and the aspiration that its outcomes would help future generations of computer science students. By agreeing to participate in the study students were collaborating with the researcher towards the achievement of this goal.

7.4.7 Audit Trail

An audit trail allows individuals external to the project to be formally brought into the study. In grounded theory this is done through the documentation of the research processes and analysis that lead to the development of the theory. One element of the grounded theory audit trail is memos, see section 4.5. These provide the external researcher with a way of understanding and appreciating how the data were interpreted and the theory derived. Grounded theory requires the researcher to provide a systematic presentation of the study, from its inception to the collection, analysis and synthesis of data through to the exposition of the emergent theory. Thus the methodology requires the researcher to present an audit trail that should render the narrative account credible for any external reviewer.

7.4.8 Thick, Rich Description

Denzin [68] defines thin descriptions as ones that are purely factual, and contrasts them to thick, rich descriptions that create the conditions for interpretation and understanding. In grounded theory this form of description lies at the heart of the methodology. In order to provide a credible presentation of their study, the researcher must provide thick, rich descriptions of (1) the participants, (2) the concepts, (3) the categories, (4) the relationships, (5) the phenomena and (6) the substantive theory. This has been one of the guiding principle's underpinning this thesis. For example, when exploring the emerging concepts and categories in Chapter 6, excerpts from the interview data collected were used to reinforce the credibility of the narrative presentation of the theory. Rich description also allows external researchers to determine the applicability of the findings to similar contexts or to make decisions about their generalisation to other settings.

7.4.9 Peer Debriefing

As discussed in chapter 3, there are two forms of reliability checks that apply to interview-based research methodologies [129]: the coder reliability check and the dialogic reliability check. Both of these checks involve a peer review by external researchers. The coder check is where the external reviewer codes a sample of the data and compares the resultant concepts and categories with those obtained by the researcher. The dialogic reliability check involves a discussion amongst researchers where they mutually critique the data and the interpretations placed on it by the researcher. In this thesis this form of peer review was carried out primarily in conjunction with the supervisor of this work and also through discussion with the researcher's colleagues who were not involved in the conduct of the study.

7.5 Conclusion

The validity and reliability of the theory of successful engagement have been assessed from two perspectives in this chapter. Firstly, the Strauss and Corbin [204] recommendations for validity checking within the grounded theory methodology were applied in the context of this study. Secondly, the Creswell and Miller validity checks for qualitative inquiry [64] were considered from the "paradigmatically neutral", grounded theory perspective. Taken together these checks strongly support the validity and reliability of the theory of successful engagement.

Chapter 8

Conclusions

Topics such as computer programming, computer systems and numerical methods are viewed as essential skills for graduates of disciplines where technology plays a key role, such as science, engineering and mathematics. While few argue about the importance of these skills; their acquisition has proven challenging for many students and this has been well documented in the literature over many decades [5; 13; 41; 42; 53; 85; 98; 133; 183; 198; 228]. This thesis was motivated by the observation that existing measures of computer experience [168; 187] were inappropriate and insufficient as they arose from studies conducted in the humanities and social sciences. In short, they did not encompass the more complex computing abilities and skills expected of students on computer science programs.

The aims and objectives of this thesis were to develop a substantive grounded theory based on the computer experiences of first year students of computer science in an Irish university. Dey [69] cautions the researcher about the nature of the questions formulated during the course of a study, commenting that questions that seem vague and ill-posed at the start of study are often redefined and reformulated by the time the final stages of the study are reached. This was certainly the case in this study where it was initially expected that the theory developed would focus purely on an exploration of objective computer experience. Analysis of the initial data collected quickly revealed that the nature of the experiences the participants identified as being key to their success went far beyond simple, objective computer experiences. Thus the research objective evolved to encompass a broader range of experiences that students perceive influence success at the undergraduate level. One of the key aims was to develop and present this theory in a way that will enhance and deepen understanding of the factors that students perceive influence success at the undergraduate level in a computer science program.

Data collection and analysis led to the emergence of a grounded theory with a central category of "successful engagement". This captured the students' perspective of the situations, contexts and settings that they believe influence their success. This theory encompasses four key abstracted categories: experience accumulation, empowering experimentation, perception of competence and language. These categories resonate with much of the existing literature in the domain and this is clearly conveyed in the table provided in appendix F. This table illustrates the nature of each of the studies referenced and the abstracted categories with which they can be associated. Of the 94 studies reviewed, only two captured elements of all four abstracted categories, while 18 reflected three of the four abstracted categories. This work is readily distinguishable from the existing literature as it is a grounded theory study of the students' perceptions of successful engagement in undergraduate computer science education.

The collection and analysis of data was carried out using the qualitative grounded theory methodology. Where possible the variant of grounded theory used was that espoused by Strauss and Corbin [204]. The methods used diverged from those of Strauss and Corbin during axial coding as it was found that the suggested coding paradigm was difficult to apply in practice. This is a common criticism of the Strauss and Corbin methodology [212]. As a consequence a slightly more flexible approach, which was informed by the work of Urquhart [212], was adopted to the application of the coding paradigm. Where possible it was used to draw out differences amongst the elements identified in the data and these were then used to guide the abstraction of categories. The coding families suggested by Glaser [94] proved to be extremely useful during axial coding.

8.1 Significance of this Study

The study presented in this work constitutes a contribution to three different research domains: computer science education research, computer science education and computer science itself. The significant contribution to computer science education research and computer science education arises from the detailed analysis and expertise of the synthesis of the data collected, and the associated theoretical sampling. This provides a novel insight into the undergraduate students' perspective of the computer related experiences that contribute to their perception of success. This represents a significant contribution to the field of computer science education research.

Computer science is a young, dynamic and rapidly evolving collection of research domains that, mostly, pre-dates the allied field of computer science education research. The relative youth of these fields is particularly notable when one contrasts them with the many centuries of research that exist into, for example, mathematics and mathematics education. As presented in the literature review, qualitative studies that investigate student's experiences of learning computer science are few in number and usually limited to explorations of the experience of learning to program. The study presented in this thesis has been conducted using the grounded theory methodology and, as such, it adds to the development and dissemination of this research approach within computer science education. It also provides a qualitative foundation for further quantitative work in the field.

The participants in this study were engaged in the study of computer science and so the theory developed provides educators in this field with a valuable insight into the situations, contexts and settings that students feel influence their success. The grounded theory research methodology employed does not restrict the outcomes of this study to this domain. Therefore the theory generated may be applicable across much wider undergraduate education settings; for example in the fields of engineering and science. Indeed, some of the abstracted categories identified may be applicable across a much broader range of disciplines and in a much wider set of contexts.

In section 6.5 it was noted that the "language" category captures elements of the community of discourse within the discipline of computer science. All academic disciplines develop communities of discourse that reflect their shared culture and hence this element of the theory is likely to have more general implications and relevance across a much wider range of educational settings. Similarly, the more subjective category "perception of competence" contains a number of subcategories that are not tied to the discipline of computer science, but rather are common to learners across all academic settings. Hence these categories may be viewed as more abstracted then the domain specific elements of the categories of "experience accumulation" and "empowering experimentation".

8.2 Further Research

This study achieved its primary objective as the theory of successful engagement emerged from the analysis and synthesis of the collected data. The data was gathered, coded and abstracted using the grounded theory methodology, which relies on the process of constant comparison to ensure the trustworthiness and validity of the emergent theory. This thorough analysis meant that a number of concepts that emerged during data collection were not fully saturated. These unsaturated concepts are the source of a number of additional research questions that extend beyond the scope of this work.

The first of these pertains to the students view of computer science. When asked about factors that influenced success in computer science many participants interpreted this as a question that related solely to computer programming. These answers related only to those factors they perceived as making people good computer programmers. While this link between computer science and programming was very strong, it did not form a saturated category. In particular it appeared that the participants' prior conceptions of computer science as a domain were much broader than the beliefs they held at the end of their first year of undergraduate study. As this did not relate directly to the experiences that influence success, it was deemed to fall outside the scope of this work. Further exploration of the reasons why the students' view of computer science as a domain becomes so narrowly focussed during their first year of study may provide an insight into the primary reasons why some students do not progress beyond the first year of their studies.

Many of the participants noted that those who were likely to succeed were "good" at mathematics and problem solving and had prior experience of computer programming. These formed subcategories of the abstracted phenomenon of experience accumulation. It was noted that participants were unable to explain why they held these beliefs about problem solving and mathematics. Also, when asked for examples of places where good problem solving and mathematical skills would prove useful to them as computer scientists they struggled to provide more than superficial examples e.g. needing to know the mathematical formula for calculating an area in order to write a program to do this calculation. A thorough exploration of the origins of this belief, and whether it was the real-world experience of the participants, was beyond the scope of this work.

One of the limitations of this study was that the progress and performance of the participants was not tracked over a number of subsequent years. This would have added another dimension to the analysis as the researcher could then have explored the insights the data provided in terms of the subsequent level of academic attainment and the professional career paths followed by the participants. Such longitudinal data would also have allowed the researcher to see how the participants attitudes towards their studies and perceptions of success changed throughout their undergraduate career. The category of "language" captured cultural differences between the participants and the faculty members who teach them. A longitudinal study would enable the researcher to explore how this cultural gap evolves over time.

8.2.1 Language Performance as a Predictor

One of the most interesting abstracted categories thrown up by this study is that of language. Consideration of language as a predictor of success in computer science has been mooted over several decades. Gathers [84] conducted a study over twenty five years ago where ten possible factors that influenced success in a freshman computer science module were explored. The factors examined included both English and Mathematics scores from a college entrance examination and scores from a reading test. The latter were obtained using the Nelson-Denny Test

[36] that provides scores for vocabulary, comprehension and reading rate. It was found that while the English and Mathematics scores could be considered reliable predictors of performance, none of the reading scores were shown to influence performance. Gathers concluded that the results indicated that "verbal skills are just as important or maybe more so than mathematical skills" [84].

Rauchas et al. [173] compared students' grades in their university matriculation examinations in Mathematics and English (both for students with English as a first language and students for whom English is their second language) with the grades obtained by the same students in two freshman computer science courses - fundamental algorithmic concepts and basic computer organisation. It was found that, for students with English as a first language, there was much stronger positive correlation between their grades in English and their grades in the computer science subjects than there was between their grades in Mathematics and the same subjects ($r=0.4571$ for English as a first language and fundamental algorithmic concepts vs $r=0.2664$ for Mathematics and the same course). The results obtained for those with English as a second language showed a much weaker positive correlation [173].

Buckland [39] looked at differences in the teaching of the humanities and the teaching of computer science and mathematics. He noted that computer science and mathematics classes are more inclined to focus on factual, quantitative content. By contrast, classes in the humanities spend more time on qualitative issues; leaving students to master the factual material outside of class time. Buckland argued that this means that students of the humanities spend more time gaining a deeper understanding of the course content as more class time is available for discussion, clarification and reflection [39]. From an alternate viewpoint, it could be argued that students who attain good grades in English may have developed the skills needed to independently achieve a deeper understanding of material.

These three studies [39; 84; 173] suggest that elements of language may prove an effective predictor of student success on computer science courses. Beyond these three works, the aspects of language that are relevant to the study of computer science have not received significant attention in the literature. The study reported on in this thesis provides qualitative evidence that evolves and extends the quantitative studies discussed above, suggesting another rich direction for future work and exploration.

8.3 Concluding Remarks

This study provided freshman undergraduate computer science students with the opportunity to describe the experiences they believe contribute to their success in their own voice. The theory presented emerged from, and is grounded in, the rich data collected during the course of this study. It is expected that the "theory of successful engagement" presented will advance understanding of the experiences that help computer science students succeed during the early years of their undergraduate studies. Indeed, it could provide an interesting starting point for future undergraduate computer science curriculum reform.

Bibliography

[1] George Allan. A Critique of using Grounded Theory as a Research Method. *Electronic Journal of Business Research Methods*, **2**[1], July 2003.

[2] Vicki L. Almstrum, Orit Hazzan, Mark Guzdial, and Marian Petre. Challenges to computer science education research. In *Proceedings of the 36th SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '05, pages 191–192, New York, NY, USA, 2005. ACM.

[3] Alastair A. Anderson. Predictors of computer anxiety and performance in information systems. *Computers in Human Behavior*, **12**[1]:61–77, 1996.

[4] L. Anderson, D. Krathwohl, P. Airasian, K. Crusikshank, R. Mayer, P. Pintrich, J. Raths, and M. Wittrock. *A taxonomy for learning, teaching, and assessing.* Addison Wesley Longman, Inc., 2001.

[5] Alex Aravind. Students Recruitment and Retention in Computer Science: Panel Discussion. In *Proceedings of the 16th Western Canadian Conference on Computing Education*, WCCCE '11, pages 43–44, New York, NY, USA, 2011. ACM.

[6] Wayne A Babchuk. Glaser or Strauss? Grounded Theory and Adult Education. In J. M. Dirkx, editor, *Proceedings of the fifteenth annual midwest research-to-practice Conference in Adult, Continuing and Community Education*, pages 1–8. University of Nebraska, 1997.

[7] WAYNE A BABCHUK. Grounded Theory for Practice-Based Application: "Closing the Embarrassing Gap between Theory and Empirical Research". In DANIELA TRUTY, editor, *Proceedings of the twenty eighth Annual Midwest Research-to-Practice Conference in Adult, Continuing, Extension and Community Education*, pages 13–18. Northeastern Illinois University, 2009.

[8] JOANNE M. BADAGLIACCO. Gender and Race Differences in Computing Attitudes and Experience. *Social Science Computer Review*, **8**[1]:42–63, 1990.

[9] J.A. BALLANTINE, P. MCCOURT LARRES, AND P. OYELERE. Computer usage and the validity of self-assessed computer competence among first-year business students. *Computers and Education*, **49**[4]:976–990, 2007.

[10] ALBERT BANDURA. *Social Foundations of Thought and Action.* Prentice Hall, 1986.

[11] ALBERT BANDURA. Self-Efficacy. In *Encyclopaedia of Human Behaviour*, **4**, pages 71–81. Academic Press, 1994.

[12] ALBERT BANDURA. *Self-efficacy: The Exercise of Control.* W.H. Freeman, 1997.

[13] THERESA BEAUBOUEF AND JOHN MASON. Why the high attrition rate for computer science students: some thoughts and observations. *SIGCSE Bull.*, **37**:103–106, June 2005.

[14] J.J BECKERS AND H.G SCHMIDT. The structure of computer anxiety: a six-factor model. *Computers in Human Behavior*, **17**[1]:35–49, 2001.

[15] SUSAN BERGIN AND RONAN REILLY. Programming: factors that influence success. In *Proceedings of the 36th SIGCSE technical symposium on Computer science education*, SIGCSE '05, pages 411–415, New York, NY, USA, 2005. ACM.

[16] SUSAN BERGIN AND RONAN REILLY. Predicting introductory programming performance: A multi-institutional multivariate study. *Computer Science Education*, **16**[4]:303–323, 2006.

[17] Susan Bergin, Ronan Reilly, and Desmond Traynor. Examining the role of self-regulated learning on introductory programming performance. In *Proceedings of the first international workshop on Computing education research*, ICER '05, pages 81–86, New York, NY, USA, 2005. ACM.

[18] Anders Berglund, Mats Daniels, and Arnold Pears. Qualitative Research Projects in Computing Education Research: An Overview. In *Proceedings of the Eighth Australasian Computing Education Conference, Australian Computer Science Communications*, page 33, 2006.

[19] C. Berkenkotter and T.N. Huckin. *Genre knowledge in disciplinary communication: cognition/culture/power.* L. Erlbaum Associates, 1995.

[20] Steven Best and Douglas Kellner. *Postmodern Theory: Critical Interrogations.* Guilford Press, New York, 1991.

[21] Tilmann Betsch, Henning Plessner, Christiane Schwieren, and Robert Gtig. I Like it but I Dont Know Why: A Value-Account Approach to Implicit Attitude Formation. *Personality and Social Psychology Bulletin*, **27**[2]:242–253, 2001.

[22] John B. Biggs. *Teaching for Quality Learning at University.* Society for Research into Higher Education & Open University Press, 1999.

[23] John B. Biggs and Kevin Collis. *Evaluating the Quality of Learning: the SOLO taxonomy.* Academic Press, 1982.

[24] Melanie Birks, Ysanne Chapman, and Karen Francis. Memoing in qualitative research. *Journal of Research in Nursing*, **13**[1]:68–75, 2008.

[25] Arkaitz Bitorika. *Responsive Aggregate Defence for Denial of Service Attacks.* PhD thesis, University of Dublin, Trinity College, 2010.

[26] Patricia Bizzell. College composition: Initiation into the academic discourse community, 1982.

[27] B.S. Bloom, B.B. Mesia, and D.R. Krathwohl. *Taxonomy of Educational Objectives, the classification of educational goals Handbook I: Cognitive Domain and Handbook II:Affective Domain.* Addison-Wesley, 1956.

[28] R. Bogdan and S.K. Biklen. *Qualitative research for education: an introduction to theory and methods.* Pearson/Allyn and Bacon, fourth edition, 2003.

[29] Jeffrey Bonar and Elliot Soloway. Uncovering principles of novice programming. In *Proceedings of the 10th ACM SIGACT-SIGPLAN symposium on Principles of programming languages*, POPL '83, pages 10–13, New York, NY, USA, 1983. ACM.

[30] Steve Borgatti. Introduction to Grounded Theory. Technical report, Analytic Technologies, 2004.

[31] David Boud and Nancy Falchikov. Quantitative Studies of Student Self-Assessment in Higher Education: A Critical Analysis of Findings. *Higher Education*, **18**[5]:529–549, 1989.

[32] Nikos Bozionelos. Computer anxiety: relationship with computer experience and prevalence. *Computers in Human Behavior*, **17**[2]:213–224, 2001.

[33] Lyn Brodie. Reflective writing by distance education students in an engineering problem based learning course. *Australasian Journal of Engineering Education*, **13**[2]:31–40, 2007.

[34] M. J. Brosnan. The impact of computer anxiety and self-efficacy upon performance. *Journal of Computer Assisted Learning*, **14**[3]:223–234, 1998.

[35] Bryan A. Brown. "It isn't no slang that can be said about this stuff": Language, identity, and appropriating science discourse. *Journal of Research in Science Teaching*, **43**[1]:96–126, 2006.

[36] J.A. Brown, V.V. Fishco, and G. Hanna. *Nelson-Denny Reading Test: Manual for Scoring and Interpretation, Forms G & H.* Riverside Publishing, 1993.

[37] CHRISTINE S. BRUCE, LAWRENCE I. BUCKINGHAM, JOHN R. HYND, CAMILLE A. MCMAHON, MICHAEL G. ROGGENKAMP, AND IAN D. STOODLEY. Ways of experiencing the act of learning to program: A phenomenographic study of introductory programming students at university. *Journal of Information Technology Education*, **3**:144–160, 2004.

[38] ANTONY BRYANT AND KATHY CHARMAZ. Grounded theory research: Methods and practices. In *The SAGE Handbook of Grounded Theory*, pages 1–28. Sage, 2007.

[39] RICHARD BUCKLAND. Can we improve teaching in computer science by looking at how English Is Taught? In *Proceedings of the 2nd Australasian conference on Computer science education*, ACSE '97, pages 155–162, New York, NY, USA, 1997. ACM.

[40] PAT BYRNE AND GERRY LYONS. The effect of student attributes on success in programming. *SIGCSE Bull.*, **33**:49–52, June 2001.

[41] PATRICIA F. CAMPBELL AND GEORGE P. MCCABE. Predicting the success of freshmen in a computer science major. *Commun. ACM*, **27**:1108–1113, November 1984.

[42] C. K. CAPSTICK, J. D. GORDON, AND A. SALVADORI. Predicting performance by university students in introductory computing courses. *SIGCSE Bull.*, **7**:21–29, September 1975.

[43] P. CAPUTI, R. JAYASURIYA, AND J. FARES. The development of a measure of attitudes toward computers in nursing. In H. HASAN AND N. NICASTRI, editors, *Proceedings of the Computer Human Interaction Special Interest Group Conference: HCI a light into the future*, pages 138–141, 1995.

[44] KATHY CHARMAZ. Discovering' chronic illness: Using grounded theory. *Social Science & Medicine*, **30**[11]:1161–1172, 1990.

[45] Kathy Charmaz. Grounded Theory: Objectivist and Constructivist Methods. In N.K. Denzin and Y.S. Lincoln, editors, *Handbook of Qualitative Research*, pages 509–535. Sage, second edition, 2000.

[46] Kathy Charmaz. Grounded Theory in the 21st Century: a qualitative method for advancing social justice research. In N.K. Denzin and Y.S. Lincoln, editors, *Handbook of Qualitative Research*, pages 507–535. Sage, Third edition, 2005.

[47] Kathy Charmaz. *Constructing grounded theory: A practical guide through qualitative analysis.* Sage, London, 2006.

[48] Cynthia Chiong and Carly Shuler. Learning: Is there an app for that. *Investigations of young children's usage and learning with mobile devices and apps. New York*, 2010.

[49] Gil Choi, Jan Ligon, and Jim Ward. Computer anxiety and social workers: differences by access, use, and training. *J. Technol. Human Services*, **19**:1–12, March 2002.

[50] Mike Clancy, John Stasko, Mark Guzdial, Sally Fincher, and Nell Dale. Models and areas for CS education research. *Computer Science Education*, **11**:323–341, 2001.

[51] Adele E. Clarke. Situational Analyses: Grounded Theory Mapping After the Postmodern Turn. *Symbolic Interaction*, **26**[4]:553–576, 2003.

[52] Adele E. Clarke. *Situational Analysis: Grounded Theory After the Postmodern Turn.* Sage, Thousand Oakes, CA, 2005.

[53] Valerie A. Clarke and Susan M. Chambers. Gender-Based Factors in Computing Enrolments and Achievement: Evidence from a Study of Tertiary Students. *Journal of Educational Computing Research*, **5**[4]:409–429, 1989.

[54] Louis Cohen, Lawrence Manion, and Keith Morrison. *Research Methods in Education.* Routledge-Falmer, fifth edition, 2000.

[55] TCD Research Committee. Policy on Good Research Practice. May 2009.

[56] Deborah R. Compeau and Christopher A. Higgins. Computer Self-Efficacy: Development of a Measure and Initial Test. *MIS Quarterly*, **19**[2]:189–211, June 1995.

[57] C. Connolly and E. Murphy. Retention initiatives for ICT based courses. In *Frontiers in Education, 2005. FIE '05. Proceedings 35th Annual Conference*, pages S2C –10, October 2005.

[58] C. Connolly, E. Murphy, and S. Moore. Adults Learning Computer Programming, Pedagogical Understanding and the Role of Anxiety. In W Aung, editor, *Innovations 2008 Special Edition, World Innovations in Engineering Education and Research*, pages 199–210. Begell House, 2008.

[59] C. Connolly, E. Murphy, and S. Moore. An Intervention Strategy for Undergraduate Computer Programming Students to Alleviate Programming Anxiety. *Journal of Instruction Delivery Systems (JIDS)*, **21**[3], 2008.

[60] C. Connolly, E. Murphy, and S. Moore. Programming Anxiety Amongst Computing Students - A Key in the Retention Debate? *Education, IEEE Transactions on*, **52**[1]:52–56, February 2009.

[61] Juliet M. Corbin and Anselm Strauss. Grounded theory research: Procedures, canons, and evaluative criteria. *Qualitative Sociology*, **13**:3–21, 1990. 10.1007/BF00988593.

[62] Lyn Corno and Ellen B. Mandinach. The role of cognitive engagement in classroom learning and motivation. *Educational Psychologist*, **18**[2]:88–108, 1983.

[63] J. Creswell. *Qualitative Inquiry and Research Design. Choosing Among Five Traditions.* Sage, 1998.

[64] John W. Creswell and Dana L. Miller. Determining Validity in Qualitative Inquiry. *Theory Into Practice*, **39**[3]:124–130, 2000.

[65] Charlie Daly and John Waldron. Assessing the assessment of programming ability. In *Proceedings of the 35th SIGCSE technical symposium on Computer science education*, SIGCSE '04, pages 210–213, New York, NY, USA, 2004. ACM.

[66] John Davy and Tony Jenkins. Research-led innovation in teaching and learning programming. In *Proceedings of the 4th annual SIGCSE/SIGCUE ITiCSE conference on Innovation and technology in computer science education*, ITiCSE '99, pages 5–8, New York, NY, USA, 1999. ACM.

[67] S. Dehnadi and R. Bornat. Testing Programming Aptitude. In *Proceedings of the 18th Annual Workshop of the Psychology of Programming Interest Group (PPIG)*, pages 22–37, 2006.

[68] N.K. Denzin. *Interpretive Interactionism.* Applied Social Research Methods Series. SAGE Publications, 2001.

[69] Ian Dey. *Qualitative Data Analysis.* Sage, 1993.

[70] Ian Dey. *Grounding Grounded Theory: Guidelines for Qualitative Inquiry.* Academic Press, 1999.

[71] P. Dias. *Worlds Apart: Acting and Writing in Academic and Workplace Contexts.* Rhetoric, Knowledge, and Society Series. Lawrence Erlbaum Associates, Incorporated, 1999.

[72] E. Doyle, I. Stamouli, and M. Huggard. Computer anxiety, self-efficacy, computer experience: an investigation throughout a computer science degree. In *Frontiers in Education, 2005. FIE '05. Proceedings 35th Annual Conference*, pages S2H -3, October 2005.

[73] B. Du Boulay. Some Difficulties of Learning to Program. In E. Soloway and Spohrer J., editors, *Studying the Novice Programmer*, pages 283–299. Hillsdale, NJ: Lawrence Erlbaum, 1989.

[74] Enda Dunican. *Learning Computer Programming in Irish Third-level Institutions: A study of first year students' experiences.* PhD thesis, University of Leicester, 2007.

[75] TALEB DURGAHEE. Reflective Practice: nursing ethics through story telling. *Nursing Ethics*, **4**[2]:135–146, 1997.

[76] FREDERICK ERICKSON. Culture difference and science education. *The Urban Review*, **18**[2]:117–124, 1986.

[77] GERALD E. EVANS AND MARK G. SIMKIN. What best predicts computer proficiency? *Commun. ACM*, **32**:1322–1327, November 1989.

[78] SALLY FINCHER AND MARIAN PETRE. *Computer Science Education Research*. Routledge Falmer, January 2004.

[79] A. FONTANA AND J.H. FREY. Interviewing – the art of science. In *Handbook of Qualitative Research*, pages 361–376. Sage, 1994.

[80] DAWN FRANCIS. The reflective journal: A window to preservice teachers' practical knowledge. *Teaching and Teacher Education*, **11**[3]:229–241, 1995.

[81] M.D. GALL, W.R. BORG, AND J.P. GALL. *Educational research: an introduction*. Longman, 1996.

[82] KATE J. GARLAND AND JAN M. NOYES. Computer experience: a poor predictor of computer attitudes. *Computers in Human Behavior*, **20**[6]:823–840, 2004.

[83] SUSAN GASSON. Rigor In Grounded Theory Research: An Interpretive Perspective On Generating Theory From Qualitative Field Studies. In M. WHITMAN AND A. WOSZCZYNSKI, editors, *Handbook for Information Systems Research*, pages 79–102. Idea Group Publishing, Hershey PA., 2004.

[84] EMERY GATHERS. Screening freshmen computer science majors. *SIGCSE Bull.*, **18**[3]:44–48, September 1986.

[85] EMERY GATHERS. One freshman studies program which improved student retention in the first year computer science sequence for majors. In *Proceedings of the 1988 ACM sixteenth annual conference on Computer science*, CSC '88, page 739, New York, NY, USA, 1988. ACM.

[86] Olivia Gill, John O'Donoghue, Fiona Faulkner, and Ailish Hannigan. Trends in performance of science and technology students (1997–2008) in Ireland. *International Journal of Mathematical Education in Science and Technology*, **41**[3]:323–339, 2010.

[87] B. Glaser. *Basics of Grounded Theory Analysis: Emergence vs. Forcing.* Sociology Press, Mill Valley, Cal. U.S.A., 1992.

[88] B. Glaser, editor. *Grounded Theory: Volumes 1 and 2.* Sociology Press, Mill Valley, Cal. U.S.A., 1995.

[89] B. Glaser. *Doing Grounded Theory: Issues and Discussions.* Sociology Press, Mill Valley, Cal. U.S.A., 1998.

[90] B. Glaser. The future of grounded theory. *Qualitative Health Research*, **9**:836–845, 1999.

[91] B. Glaser. *The grounded theory perspective: Conceptualization contrasted with description.* Sociology Press, 2001.

[92] B. Glaser. Conceptualisation: On Theory and Theorising Using Grounded Theory. *International Journal of Qualitative Methods*, **1**[2]:1–31, 2002.

[93] B. Glaser and A. Strauss. *The Discovery of Grounded Theory: Strategies for Qualitative Research.* Aldine Publishing Co., Chicago, Ill. U.S.A., 1967.

[94] Barney G. Glaser. *Theoretical sensitivity: advances in the methodology of grounded theory.* Sociology Press, Mill Valley, CA, 1978.

[95] C. Goulding. Grounded Theory: The Missing Methodology on the Interpretivist Agenda. *Qualitative Market Research*, **1**[1]:50–57, 1998.

[96] C. Goulding. *Grounded theory: a practical guide for management, business and market researchers.* SAGE, 2002.

[97] P. Green. A Rigorous Journey into Phenomenography: From a Naturalistic Inquirer Point of View. In *Doing Developmental Phenomenograpy*, pages 32–46. RMIT University Press, 2005.

[98] Jiang Guo. Using group-based projects to improve retention of students in computer science major. *J. Comput. Sci. Coll.*, **23**:187–193, June 2008.

[99] Dianne Hagan and Selby Markham. Does it help to have some programming experience before beginning a computing degree program? *SIGCSE Bull.*, **32**:25–28, July 2000.

[100] Dianne Hagan and Judy Sheard. The value of discussion classes for teaching introductory programming. In *Proceedings of the 6th annual conference on the teaching of computing and the 3rd annual conference on Integrating technology into computer science education: Changing the delivery of computer science education*, ITiCSE '98, pages 108–111, New York, NY, USA, 1998. ACM.

[101] J. Hamilton-Jones and T. Svane. Developing research using reflective diaries. In *Frontiers in Education, 2003. FIE 2003 33rd Annual*, **1**, pages T3A–14–19 Vol.1, 2003.

[102] Allison W. Harrison and R. Kelly Rainer, Jr. The influence of individual differences on skill in end-user computing. *J. Manage. Inf. Syst.*, **9**:93–111, July 1992.

[103] Bassam Hasan. The influence of specific computer experiences on computer self-efficacy beliefs. *Computers in Human Behavior*, **19**[4]:443–450, 2003.

[104] Helen Heath and Sarah Cowley. Developing a grounded theory approach: a comparison of Glaser and Strauss. *International journal of nursing studies*, **41**:141–150, 2004.

[105] Terry R. Hostetler. Predicting student success in an introductory programming course. *SIGCSE Bull.*, **15**:40–43, September 1983.

[106] GEOFFRY S. HOWARD AND ROBERT D. SMITH. Computer anxiety in management: myth or reality? *Commun. ACM*, **29**:611–615, July 1986.

[107] M. HUGGARD AND C. MC GOLDRICK. Computer Experience – Enhancing Engineering Education. In *Conference on Engineering Education, 9th International*, pages T4C–21 – T4C–25, 2006.

[108] M. HUGGARD AND C. MC GOLDRICK. Incentivising students to pursue computer science programmes. In *Frontiers in Education Conference, 36th Annual*, pages 3–8, October 2006.

[109] M. HUGGARD AND C. MC GOLDRICK. Lessons Learnt from a Decade of Structured Support for Novice Programmers. In *Conference on Engineering Education, 14th International*, pages FR.A.SB 11.230–7, 2011.

[110] MERIEL HUGGARD. A Control Theoretic Approach to Active Queue Management, 2002. Enterprise Ireland Basic Research Grant.

[111] MERIEL HUGGARD. Structured Programming Support for Undergraduate Students on Technology Based Courses, 2003. Funded by the Higher Education of Ireland as Retention Initiative.

[112] MERIEL HUGGARD. Programming Trauma: Can it be avoided? In *Proceedings Grand Challenges in Computing*. British Computer Society, March 2004.

[113] MERIEL HUGGARD. Wireless Active Queue Management: A Robust, Scalable Approach using Control Theory , 2006. Science Foundation Ireland Research Frontiers Program.

[114] E. HYLAND AND G. CLYNCH. Initial Experiences Gained and Initiatives Employed in the Teaching of Java Programming in the Institute of Technology Tallaght. In *Proceedings of Conference on Principles and Practice of Programming in Java*, pages 101–106, 2002.

[115] MAGID IGBARIA AND ALOK CHAKRABARTI. Computer anxiety and attitudes towards microcomputer use. *Behaviour & Information Technology*, **9**[3]:229–241, 1990.

[116] C.S. Jacelon and O'Dell K.K. Case and Grounded Theory as Qualitative Research Methods. *Urologic Nursing*, **25**[1]:49–52, 2005.

[117] T. Jenkins. Teaching Programming A Journey from Teacher to Motivator. In *Proceedings of the Second Annual LTSN-ICS Conference*. LTSN Centre for Information and Computer Sciences, August 2001.

[118] T. Jenkins. On the Difficulty of learning to Program. In *Proceedings of the Third Annual LTSN-ICS Conference*, pages 53–58. LTSN Centre for Information and Computer Sciences, 2002.

[119] Tony Jenkins. The motivation of students of programming. *SIGCSE Bull.*, **33**:53–56, June 2001.

[120] Colin G. Johnson and Ursula Fuller. Is Bloom's taxonomy appropriate for computer science? In *Proceedings of the 6th Baltic Sea conference on Computing education research: Koli Calling 2006*, Baltic Sea '06, pages 120–123, New York, NY, USA, 2006. ACM.

[121] Nicole Joncour, Kenneth E. Sinclair, and Michael Bailey. Computer anxiety, computer experience and self-efficacy. In *Proceedings of Annual Conference of the Australian Association for Research in Education*, 1994.

[122] T. Jones and V.A. Clarke. Diversity as a determinant of attitudes: A possible explanation of the apparent advantage of single-sex settings. *Journal of Educational Computing Research*, **12**[1]:51–64, 1995.

[123] Robert K. Heinssen Jr., Carol R. Glass, and Luanne A. Knight. Assessing computer anxiety: Development and validation of the Computer Anxiety Rating Scale. *Computers in Human Behavior*, **3**[1]:49–59, 1987.

[124] Abraham Kaplan. *The Conduct of Inquiry*. Chandler, 1964.

[125] Michael Kaplowitz. Statistical Analysis of Sensitive Topics in Group and Individual Interviews. *Quality & Quantity: International Journal of Methodology*, **34**[4]:419–431, 2000.

[126] R.H. Kay. An analysis of methods used to examine gender differences in computer related behaviour. *Journal of Educational Computing Research*, **8**:323–336, 1992.

[127] Judy Kendall. Axial Coding and the Grounded Theory Controversy. *Western Journal of Nursing Research*, **21**[6]:743–757, 1999.

[128] Louise H. Kidder. *Selltiz, Wrightsman and Cook's research methods in social relations*. Holt and Rinehart and Winston, fourth edition, 1981.

[129] Steinar Kvale. *InterViews: An Introduction to Qualitative Research Interviewing*. Sage, 1996.

[130] J. Lave and E. Wenger. *Situated Learning: Legitimate Peripheral Participation*. Learning in Doing: Social, Cognitive and Computational Perspectives. Cambridge University Press, 1991.

[131] Jean Lave. Situating learning in communities of practice. *Perspectives on socially shared cognition*, **63**:82, 1991.

[132] Linda Costigan Lederman. Assessing Educational Effectiveness: The Focus Group Interview as a Technique for Data Collection? *Communication Education*, **39**[2]:117–127, 1990.

[133] R. R. Leeper and J. L. Silver. Predicting success in a first programming course. *SIGCSE Bull.*, **14**:147–150, February 1982.

[134] G. Lewandowski. Using Process Journals to Gain Qualitative Understanding of Beginning Programmers. *Journal of Computing Sciences in Colleges*, **19**[1]:299–310, 2003.

[135] Rensis Likert. A Technique for the Measurement of Attitudes. *Archives of Psychology*, **140**:1–55, 1932.

[136] Yvonna S. Lincoln and Egon G. Guba. *Naturalistic Inquiry*. Sage, 1985.

[137] Yvonna S. Lincoln and Egon G. Guba. Competing Paradigms in Qualitative Research. In N.K. Denzin and Y.S. Lincoln, editors, *Handbook of Qualitative Research*, pages 105–117. Sage, first edition, 1994.

[138] Yvonna S. Lincoln and Egon G. Guba. Paradigmatic Controversies, Contradictions, and Emerging Confluences. In N.K. Denzin and Y.S. Lincoln, editors, *Handbook of Qualitative Research*, pages 163–188. Sage, second edition, 2000.

[139] K. Locke. *Grounded Theory in Management Research.* Sage, 2001.

[140] J. Lofland. *Analyzing Social Settings: A Guide to Qualitative Observation and Analysis.* Sociology-Upper Level Series. Wadsworth/Thomson Learning, 2006.

[141] Brenda H. Loyd and Clarice Gressard. Reliability and Factorial Validity of Computer Attitude Scales. *Educational and Psychological Measurement*, **44**[2]:501–505, 1984.

[142] Peter Lugosi. Between Overt and Covert Research. *Qualitative Inquiry*, **12**[3]:541–561, 2006.

[143] Doug Mahar, Ron Henderson, and Frank Deane. The effects of computer anxiety, state anxiety, and computer experience on users' performance of computer based tasks. *Personality and Individual Differences*, **22**[5]:683–692, 1997.

[144] Lauri Malmi, Judy Sheard, Simon, Roman Bednarik, Juha Helminen, Ari Korhonen, Niko Myller, Juha Sorva, and Ahmad Taherkhani. Characterizing research in computing education: a preliminary analysis of the literature. In *Proceedings of the Sixth international workshop on Computing education research*, ICER '10, pages 3–12, New York, NY, USA, 2010. ACM.

[145] F Marton. Phenomenography - Describing Conceptions of the World Around us. *Instructional science*, **10**:177–200, 1981.

[146] F. Marton and S. Booth. *Learning and Awareness.* Lawrence Erlbaum Associates, 1997.

[147] F. Marton, U. Runesson, and A. Tsui. The Space of Learning. In F. Marton and A. Tsui, editors, *Classroom discourse and the space of learning*, pages 3–40. L. Erlbaum Associates, 2004.

[148] Katharyn Antle May. Writing and evaluating the grounded theory research report. In W. Carole Chenitz and Janice M. Swanson, editors, *From Practice to Grounded Theory: Qualitative Research in Nursing*, pages 146–154. Menlo Park, California: Addison-Wesley, 1986.

[149] Gerry McGhee, Glenn R. Marland, and Jacqueline Atkinson. Grounded theory research: literature reviewing and reflexivity. *Journal of Advanced Nursing*, **60**[3]:334–342, 2007.

[150] K.M. Melia. Rediscovering Glaser. *Qualitative Health Research*, **6**[3], 1996.

[151] U. Melin Emilsson and B. Lilje. Training social competence in engineering education: necessary, possible or not even desirable? An explorative study from a surveying education programme. *European Journal of Engineering Education*, **33**[3]:259–269, 2008.

[152] M. Miles and M. Huberman. *Qualitative Data Analysis : An Expanded Sourcebook.* Sage, second edition, 1994.

[153] D.C. Miller and N.J. Salkind. *Handbook of research design and social measurement.* Sage Publications, 2002.

[154] Elizabeth B. Moje, Tehani Collazo, Rosario Carrillo, and Ronald W. Marx. Maestro, what is quality?: Language, literacy, and discourse in project-based science. *Journal of Research in Science Teaching*, **38**[4]:469–498, 2001.

[155] P. Montgomery and P. H. Bailey. Field Notes and Theoretical Memos in Grounded Theory. *Western Journal of Nursing Research*, **29**[1]:65–79, 2007.

[156] J. Moon. *Learning Journals: A Handbook for Academics, Students and Professional Development.* Taylor & Francis, 2006.

[157] Oliver Mooney, Vivienne Patterson, Muiris O'Connor, and Abigail Chantler. *A Study of Progression In Irish Higher Education: A report by the Higher Education Authority.* The Higher Education Authority, October 2010.

[158] Sarah Moore, Donncha O'Maidin, and Annette McElligott. Cognitive styles among computer systems students: Preliminary findings. *Journal of Computing in Higher Education*, **14**:45–67, 2003.

[159] M. Morgan, R. Flanagan, and T. Kellaghan. A study of non-completion in undergraduate university courses. Technical report, Higher Education Authority, 2001.

[160] Katrin Niglas. Paradigms and Methodology in Educational Research. In *Proceedings of the European Conference on Educational Research*, 2001.

[161] School of Computer Science and Trinity College Dublin Statistics. Research Ethics Protocol. September 2011.

[162] Government of Ireland. Data Protection Act 1988. 1988.

[163] Government of Ireland. Data Protection (Amendment) Act 2003. 2003.

[164] M. Patton. *Qualitative Research & Evaluation Methods.* Sage, third edition, 2002.

[165] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann, 1988.

[166] Karl-Heinz Pogner. Writing and interacting in the discourse community of engineering. *Journal of Pragmatics*, **35**[6]:855 – 867, 2003.

[167] Denise Potosky and Philip Bobko. The Computer Understanding and Experience Scale: a self-report measure of computer experience. *Computers in Human Behavior*, **14**[2]:337–348, 1998.

[168] BRAD PRINCE. Toward a bi-dimensional measurement of computer experience. *Journal of Academy of Business and Economics*, **8**[4], 2008.

[169] CASSIE F. QUIGLEY. With their help: How community members construct a congruent third space in an urban kindergarten classroom. *International Journal of Science Education*, **35**[5]:837–863, 2013.

[170] V. RAFTOPOLOUS. A Grounded Theory for Patients Satisfaction with Quality of Hospital Care. *ICUS Nurse Web Journal*, pages 1–15, 2005.

[171] VENNILA RAMALINGAM, DEBORAH LABELLE, AND SUSAN WIEDENBECK. Self-efficacy and mental models in learning to program. In *Proceedings of the 9th annual SIGCSE conference on Innovation and technology in computer science education*, ITiCSE '04, pages 171–175, New York, NY, USA, 2004. ACM.

[172] VENNILA RAMALINGAM AND SUSAN WIEDENBECK. Development and validation of scores on a computer programming self-efficacy scale and group analyses of novice programmer self-efficacy. *Journal of Educational Computing Research*, **19**[4]:365–379, 1998.

[173] SARAH RAUCHAS, BENJAMIN ROSMAN, GEORGE KONIDARIS, AND IAN SANDERS. Language performance at high school and success in first year computer science. *SIGCSE Bull.*, **38**[1]:398–402, March 2006.

[174] P. RAWSTORNE, P. CAPUTI, AND A. SMITH. Towards the development of a measure of subjective computer experience: Working paper 2. Unpublished Manuscript, University of Wollongong, 1998.

[175] DAVID L. RENNIE. Grounded Theory Methodology. *Theory & Psychology*, **8**[1]:101–119, 1998.

[176] ANTHONY ROBINS. Learning edge momentum: a new account of outcomes in CS1. *Computer Science Education*, **20**[1]:37–71, 2010.

[177] C. ROBSON. *Real world research: a resource for social scientists and practitioner-researchers*. Regional Surveys of the World Series. Blackwell Publishers, 2002.

[178] SUPRATEEK SARKER, FRANCIS LAU, AND SUNDEEP SAHAY. Using an adapted grounded theory approach for inductive theory building about virtual team development. *SIGMIS Database*, **32**[1]:38–56, December 2000.

[179] D.A. SCHÖN. *The reflective practitioner: how professionals think in action.* Basic Books, 1983.

[180] CLIVE SEALE. Quality in Qualitative Research. *Qualitative Inquiry*, pages 465–478, December 1999.

[181] I. SEIDMAN. *Interviewing As Qualitative Research: A Guide for Researchers in Education And the Social Sciences.* Teachers College Press, third edition, 2006.

[182] LILLY SHASHAANI. Gender-differences in computer experience and its influence on computer attitudes. *Journal of Educational Computing Research*, **11**[4]:347–367, 1994.

[183] JUDY SHEARD, ANGELA CARBONE, SELBY MARKHAM, A. J. HURST, DES CASEY, AND CHRIS AVRAM. Performance and progression of first year ICT students. In SIMON AND MARGARET HAMILTON, editors, *Tenth Australasian Computing Education Conference (ACE 2008)*, **78** of *CRPIT*, pages 119–128, Wollongong, NSW, Australia, 2008. ACS.

[184] JUDY SHEARD AND DIANNE HAGAN. Our failing students: a study of a repeat group. *SIGCSE Bull.*, **30**:223–227, August 1998.

[185] B. A. SHEIL. The Psychological Study of Programming. *ACM Comput. Surv.*, **13**:101–120, March 1981.

[186] E. J. SIMPSON. *The Classification of Educational Objectives in the Psychomotor Domain.* Gryphon House, 1972.

[187] B. SMITH, P. CAPUTI, N. CRITTENDEN, R. JAYASURIYA, AND P. RAWSTORNE. A review of the construct of computer experience. *Computers in Human Behavior*, **15**[2]:227–242, 1999.

[188] B. Smith, P. Caputi, and P. Rawstorne. Differentiating computer experience and attitudes toward computers: an empirical investigation. *Computers in Human Behavior*, **16**[1]:59–81, 2000.

[189] Brooke Smith, Peter Caputi, and Patrick Rawstorne. The development of a measure of subjective computer experience. *Computers in Human Behavior*, **23**[1]:127–145, 2007.

[190] K Smith and F. Biley. Understanding Grounded Theory: Principles and Evaluation. *Nurse Researcher*, **4**:17–30, 1997.

[191] Paul Spicker. Ethical Covert Research. *Sociology*, **45**[1]:118–133, 2011.

[192] I. Stamouli, E. Doyle, and M. Huggard. Establishing structured support for programming students. *Frontiers in Education, Annual*, **0**:F2G–5, 2004.

[193] Ioanna Stamouli. *Learning Object-Oriented Programming from the Students' Perspective*. PhD thesis, University of Dublin, Trinity College, 2009.

[194] Ioanna Stamouli and Meriel Huggard. Object oriented programming and program correctness: the students' perspective. In *Proceedings of the second international workshop on Computing education research*, ICER '06, pages 109–118, New York, NY, USA, 2006. ACM.

[195] Claude M. Steele. The psychology of self-affirmation: Sustaining the integrity of the self. In L. Berkowitz, editor, *ADV EXPERIMENTAL SOCIAL PSYCHOLOGY*, **21** of *Advances in Experimental Social Psychology*, pages 261–302. Elsevier Science, 1988.

[196] G. Steiner. *On difficulty, and other essays*. Oxford paperbacks. Oxford University Press, 1978.

[197] P.N. Stern. Eroding Grounded Theory. In J.M. Morse, editor, *Critical issues in qualitative research methods*, pages 212–223. Sage, 1994.

[198] Carolee Stewart-Gardiner. Using peer led team learning to assist in retention in computer science classes. *J. Comput. Small Coll.*, **25**:164–171, January 2010.

[199] Detmar Straub, Marie-Claude Bourdreau, and David Gefen. Validation Guidelines for IS Positivist Research. *Communications of the Association for Information Systems*, **13**:380–427, 2004.

[200] A Strauss, editor. *Qualitative Analysis for Social Scientists.* Cambridge University Press, Cambridge, UK, 1987.

[201] A Strauss and J. Corbin. *Basics of Qualitative Research: Grounded Theory Procedures and Techniques.* Sage Publications, Newbury Park, CA, USA, first edition, 1990.

[202] A Strauss and J. Corbin. Grounded theory methodology – an overview. In N.K. Denzin and Y.S. Lincoln, editors, *Handbook of Qualiltative Research*, pages 273–285. Sage Publications, first edition, 1994.

[203] A Strauss and J. Corbin, editors. *Grounded Theory in Practice.* Sage Publicaitons, London, UK, 1997.

[204] A Strauss and J. Corbin. *Basics of Qualitative Research: Grounded Theory Procedures and Techniques.* Sage Publications, Newbury Park, CA, USA, second edition, 1998.

[205] A Strauss and J. Corbin. *Basics of Qualitative Research: Grounded Theory Procedures and Techniques.* Sage Publications, third edition, 2008.

[206] The Joint Task Force on Computing Curricula. *Computing curricula 2001*, **1**. ACM, New York, NY, USA, September 2001.

[207] The Joint Task Force on Computing Curricula. *Computer Science Curriculum 2008: An Interim Revision of CS2001.* ACM, New York, NY, USA, December 2008.

[208] LYNDA THOMAS, MARK RATCLIFFE, JOHN WOODBURY, AND EMMA JARMAN. Learning styles and performance in the introductory programming sequence. In *Proceedings of the 33rd SIGCSE technical symposium on Computer science education*, SIGCSE '02, pages 33–37, New York, NY, USA, 2002. ACM.

[209] S. THORNE. Data Analysis in Qualitative Research. *Evidence Based Nursing*, **3**:68–70, 2000.

[210] SHEILA TOBIAS. *They're not Dumb, They're Different: Stalking the Second Tier*. Science News Books, 1994.

[211] A. CAROLINE TYNAN AND JENNIFER L. DRAYTON. Conducting Focus Groups - A Guide for First-time Users. *Marketing Intelligence & Planning*, **6**:5–9, 1988.

[212] CATHY URQUHART. An Encounter with Grounded Theory: Tackling the Practical and Philosophical Issues. In E.M. TRAUTH, editor, *Qualitative Research in IS: Issues and Trends*, pages 104–140. Idea Group Pub., 2001.

[213] CATHY URQUHART, HANS LEHMANN, AND MICHAEL D. MYERS. Putting the theory back into grounded theory: guidelines for grounded theory studies in information systems. *Information Systems Journal*, **20**[4]:357–381, 2010.

[214] JOHAN P. VAN BRAAK. Domains and determinants of university students' self-perceived computer competence. *Computers & Education*, **43**[3]:299–312, 2004.

[215] M. VAN MANEN. *Researching lived experience: Human science for an action sensitive pedagogy*. State University of New York Press, 1990.

[216] M. VARELAS, D.B. MARTIN, AND J.M. KANE. Content Learning and Identity Construction: A Framework to Strengthen African American Students Mathematics and Science Learning in Urban Elementary Schools. *Human Development*, **55**[5 – 6]:319 – 339, 2013.

[217] J. D. Vermunt. Metacognitive, cognitive and affective aspects of learning styles and strategies: A phenomenographic analysis. *Higher Education*, **31**:25–50, 1996. 10.1007/BF00129106.

[218] L.S. Vygotsky. *Thought and language.* Studies in communication. M.I.T. Press, Massachusetts Institute of Technology, 1962.

[219] D. Walker and F. Myrick. Grounded Theory: An Exploration of Process and Procedure. *Qualitative Health Research*, **16**[4]:547–559, 2006.

[220] Matthew R. Wallen and Abhay S. Pandit. Encouraging undergraduate engineering students towards civic engagement. *European Journal of Engineering Education*, **34**[2]:141–148, 2009.

[221] Michelle M. Weil and Larry D. Rosen. The psychological impact of technology from a global perspective: A study of technological sophistication and technophobia in university students from twenty-three countries. *Computers in Human Behavior*, **11**[1]:95–133, 1995.

[222] Michelle M. Weil, Larry D. Rosen, and Stuart E. Wugalter. The etiology of computerphobia. *Computers in Human Behavior*, **6**[4]:361–379, 1990.

[223] J.J. Wellington. *Educational research: contemporary issues and practical approaches.* Continuum, 2000.

[224] T. Wengraf. *Qualitative Research Interviewing: Biographic Narrative and Semi-Structured Methods.* SAGE Publications, 2001.

[225] Susan Wiedenbeck. Factors affecting the success of non-majors in learning to program. In *Proceedings of the first international workshop on Computing education research*, ICER '05, pages 13–24, New York, NY, USA, 2005. ACM.

[226] W. Wiersma and S.G. Jurs. *Research methods in education: an introduction.* Pearson/Allyn and Bacon, 2009.

[227] Jeffery D. Wilfong. Computer anxiety and anger: the impact of computer use, computer experience, and self-efficacy beliefs. *Computers in Human Behavior*, **22**[6]:1001–1011, 2006.

[228] Brenda Cantwell Wilson and Sharon Shrock. Contributing to success in an introductory computer science course: a study of twelve factors. In *Proceedings of the thirty-second SIGCSE technical symposium on Computer Science Education*, SIGCSE '01, pages 184–188, New York, NY, USA, 2001. ACM.

[229] Peter Wimpenny and John Gass. Interviewing in phenomenology and grounded theory: is there a difference? *Journal of Advanced Nursing*, **31**[6]:1485–1492, 2000.

[230] Leon E. Winslow. Programming pedagogy – a psychological overview. *SIGCSE Bull.*, **28**:17–22, September 1996.

[231] Glyn Winter. A Comparative Discussion of the Notion of 'Validity' in Qualitative and Quantitative Research. *The Qualitative Report*, **4**[3 and 4], 2000.

[232] A.D. Wolvin and C.G. Coakley. *Listening*. McGraw-Hill, 1995.

[233] Amy B. Woszczynski, Tracy C. Guthrie, and Sherri Shade. Personality and programming. *Journal of Information Systems Education*, **16**:293–300, 2005.

Appendix A

Initial Questionnaire and Form

Department of Computer Science **Michaelmas Term 2006**
Trinity College Dublin
Meriel Huggard **Computer Experience Study**

Gender: ☐ Female ☐ Male Age:______ Course, Year:________________

Educational Background prior to commencing this degree program: (Please tick all that apply)

☐Direct entrant from Second Level

☐Mature Student

☐Exchange Student

☐Transfer student, if yes, please specify initial course choice:__________________

☐Other:______________

Prior to commencing this degree program had you completed any courses, certificates, etc., related to computers or computer programming?

☐ Yes ☐ No

If yes, please specify:

Prior to commencing this degree program had you any previous work experience related to computer science?

☐ Yes ☐ No

If yes, please specify:

Would you be interested in being interviewed once or twice during this academic year for our study on Computer Experience? (A small token will be offered in thanks for your participation).

☐ Yes ☐ No

If yes, please provide contact details (Name, e-mail address, mobile number)

Please Turn Over Page 1 of 2

Figure A.1: Initial Questionnaire/Contact Form used for this study, page 1

State three things that motivated you to choose a Computer Science based degree program:

What are the main computer experiences you have gained during this degree program?

What are the main computer experiences you wish you had prior to commencing this degree program?

What are the main skills and abilities you have acquired and developed as part of your degree program?

Upon completion of your degree, do you expect your future career to be in the computer science/ICT sector?

☐ Yes ☐ No

Please say why:

Page 2 of 2

Figure A.2: Initial Questionnaire/Contact Form used for this study, page 2

Computer Experiences that help Undergraduates to be Successful in their Studies
Researcher: Meriel Huggard

The objective is this study is to gain an insight into the computer experiences that first year undergraduate computer science students believe help them to be successful during the course of their studies. This data collected will be analysed and used to develop a theory of computer experience for a Ph.D. thesis being prepared by the researcher. Data will be collected via individual interview, focus groups and reflective diaries. To ensure confidentiality all data will be anonymised and you will be referred to by a number in any reports or documents arising from this study e.g. you could be referred to as, say, "Participant 7''.

Interviews
Participants (undergraduate students) will be asked open-ended questions on the experiences they think help them and others to be successful. Participants will be asked to talk about their own personal experiences and how these have helped them to succeed. There are no right or wrong answers, all that is required is of each participant is that they be willing to talk openly and in confidence about their experiences. All data collected will be of relevance for this study. Note: This is not an exam and you won't be asked any exam style questions. All information provided will be strictly confidential.

Focus Group
These are group interview session where open questions are posed and the group members discuss the computer experiences they feel are relevant to the study of computer science. It is expected that there will be no more than five participants in each group.

Reflective Diaries
Participants will be asked to keep an electronic diary of their computer based activities and the experiences they think are relevant to them. Participants will be asked to provide data that covers seven, not necessarily consecutive, days. The diary should document the experiences they think are relevant to their studies. There is also a section in the diary for general comments, where any information you think might be relevant to this study can be noted and recorded

Participation Form

I am interested in participating and would like to participate in the following (you can participate in any one or more of the following):

- Interview □
- Focus Groups □
- Journals / Diaries □

E-Mail Address ____________________________

You will be contacted by e-mail to arrange a suitable time and day to meet up, or else you will be emailed about the completion of a reflective diary.
Please note, there may be more volunteers than are needed for this study and if your contribution is not required you will receive an email to let you know.

Figure A.3: Study Overview Form

Appendix B

Consent Form

Consent Form

I give my informed consent to participate in this grounded theory based study on defining the concept of computer experience.

I consent to the publication of the study results so long as the data is kept confidential, so that no identification can be made. I understand that my responses are anonymous and will be identified by number only.

1) I have been informed that my participation in this study will require my involvement in a semi-structured interview where I will be answering questions.

2) I have been informed that my interview will be tape-recorded and transcribed (it will be typed up and anonymised).

3) I have been informed that this grounded theory study is involved in identifying the relationship between objective computer experience and undergraduate success. Thus the task of the research study is not an IQ or performance test.

4) I have been informed that there are no known expected risks or discomforts involved in my participation in this study. This judgement is passed upon a relative large body of research with people undertaking tasks of a similar nature.

5) I have been informed that participation is voluntary. If at any point during the course of the project, I wish to withdraw from the study, my decision will be respected immediately.

6) I have been informed that the researcher will gladly answer any questions about the research when the interviewing and experimental session is completed.

Concerns about any aspects of this study may be referred to Meriel Huggard, a lecturer in the Computer Science Department at Trinity College Dublin.

Signed: ____________________ ____________________

(Researcher) (Participant)

Figure B.1: Consent Form used for this study

Appendix C

Reflective Diary Template

Computer Experience Study – Reflective Diary

Instructions to Participants

1. In this diary there is a section for each day where you record in your own words what computer science related activities you did, how long you spent on them (approximately) and your comments on computer experiences relevant to the activity.
2. The comments section is where you document in your own words what it might have been useful to know in advance of carrying out the activity.
3. If you carry out more than one activity in a day then you can discuss and comment on all of the activities together. If you need to go onto a new page, or pages, then that's fine.
4. If you don't carry out any relevant activities in a day, then there is no need to record anything in the diary. I would ask you to record seven days where you engage in computer science related activities.
5. There is a section at the end where you can make general comments on your experiences that you have found relevant.
6. Please document activities and experiences in your own words. Please document anything that comes to mind, even if you don't think it is relevant.
7. Please fill in the form electronically and e-mail it back to me once you have it completed.

Figure C.1: Information provided on how to complete the Reflective Diary

Day ____

What Computer Science related activities I did: e.g. programming/activities/study

Duration: How long did you spend at it?

Comments: What prior knowledge and experiences did you draw on to complete the tasks/activities/study. What would it have been helpful to know?

General Comments

Use this box for any general comments and issues you want to raise, even if you think it might not be relevant.

Figure C.2: Reflective Diary, Sample Page

Appendix D

Initial Interview Questions

The initial section of each interview took the form of a directed conversation. The following list of questions were drawn up in advance to help guide this process.

Tell me a bit about your self and how you came to study computer science?

Where did you use computers before you came to College?

What sort of things did you use computers for/do with computers before coming to College?

Did you take any computer courses before you started your studies here?

What did you know about computer programming before undertaking this course?

Why did you choose Computer Science?

What were your expectations of the course before it started?

Were these expectations realised?

What interests you most about computer science?

What interests you least about computer science?

Has anything proven to be particularly easy for you?

Has anything proven to be particularly challenging for you?

Were there things that your lecturers presumed that you know that you hadn't met before?

How have you found your studies so far?

How have you found lectures?

How do you go about learning/understanding things? What approach to study have you adopted?

How does this compare to the approach to study you adopted in school?

If you would describe the characteristics of students in computer science what/how would they be? Is this you? Do you fit in/not fit in?

What would you say are the most important things for somebody to know in advance of starting this course?

Suppose you were talking to a leaving certificate student, how would you describe computer science to them?

If you were talking to someone who was going to accept a place to study computer science next October, what advice would you give them?

More broadly, how are you finding life as a university student?

What do you regards as your academic strengths and weaknesses?

Appendix E

Computer Experience Questionnaire

Brad Prince's computer experience questionnaire [168] is included here for completeness

Objective Computer Experience Scale

DIRECT COMPUTER INTERACTION

Amount of Computer Use

Appropriate scales are provided on the actual survey.

1. How long have you used a computer at home (in years)?
2. How long have you used a computer at work/school (in years)?
3. NOT INCLUDING INTERNET OR E-MAIL. On average, how many hours during a week have you used a computer for school/work purposes?

4. NOT INCLUDING INTERNET OR E-MAIL. On average, how many hours during a week have you used a computer for personal purposes?

5. On average, how many times during the week have you checked your work/school e-mail?

6. On average, how many times during the week have you checked your personal e-mail (if it is different from the above e-mail account)?

7. On average, how many hours during the week have you spent using the Internet for school/work purposes?

8. On average, how many hours during the week have you spent using the Internet for personal purposes?

9. What percentage of your time at school/work was spent using the computer for personal purposes?

10. What percentage of your time at school/work was spent utilizing the computer for school/work purposes?

11. What percentage of your time at home was spent utilizing the computer for personal purposes?

12. What percentage of your time at home was spent utilizing the computer for school/work purposes?

Opportunity to use computers

Appropriate scales are provided on the actual survey.

1. In years, how long has a computer been available to you at school/work?

2. In years, how long has a computer been available to you at home?

3. On average, what percentage of your week at school/work has a computer been available to you?

4. How much of that time have you utilized that access (at school/work)?

5. On average, what percentage of your week at home has a computer been available to you?

6. How much of that time have you utilized that access (at home)?

Diversity of Experience

Consider the following list of computer uses. Based on the following criteria, indicate how often have you engaged with these types of computer uses for PERSONAL REASONS?

(1) Never, (2) Rarely, (3) Often, (4) Very often, (5) Very Frequently.

1. Word Processing (Microsoft Word)
2. Spreadsheet (Microsoft Excel)
3. Database (Microsoft Access, Oracle)
4. Presentation (Microsoft PowerPoint)
5. Statistics (SPSS, AMOS, Minitab)
6. Personal Assistance (TurboTax, Quicken, etc.)
7. E-mail
8. Games
9. Instant messaging/chat

10 Internet for downloading music

11. Internet for downloading videos
12. Internet for online gaming
13. Internet for shopping
14. Internet for research

15. Internet for news/weather

Consider the following list of computer uses. Based on the following criteria, indicate how often have you engaged with these types of computer uses for SCHOOL/WORK REASONS?

(1) Never, (2) Rarely, (3) Often, (4) Very often, (5) Very Frequently.

1. Word Processing (Microsoft Word)
2. Spreadsheet (Microsoft Excel)
3. Database (Microsoft Access, Oracle)
4. Presentation (Microsoft PowerPoint)
5. Statistics (SPSS, AMOS, Minitab)
6. Personal Assistance (TurboTax, Quicken, etc.)
7. E-mail
8. Games
9. Instant messaging/chat
10. Internet for downloading music
11. Internet for downloading videos
12. Internet for online gaming
13. Internet for shopping
14. Internet for research
15. Internet for news/weather

INDIRECT COMPUTER INTERACTION

Sources of Information

Please answer the following questions.

1. How many computer courses/training sessions have you completed?
2. How many computer courses did you take because you wanted to?
3. How many computer courses did you take because you had to for school/work?
4. On average, how many computer-related magazines do you read each month?
5. On average, how many computer-related TV programs do you watch each month?
6. On average, how often have you engaged in computer-related conversations with family?
7. On average, how often have you engaged in computer-related conversations with coworkers/ schoolmates?

Subjective Computer Experience Scale

COGNITIVE COMPONENT

Previous Successes or Failures

Based on the following criteria, indicate the amount that you recall experiencing the following emotions when interacting with the computer: (1) never, (2) rarely, (3) often, (4) very often, (5) all the time.

1. When using a computer because I chose to, I successfully completed the tasks that I set out to do.
2. When using a computer because I had to for school/work, I successfully completed the tasks that I had to do.

3. More times than not, I have been able to complete an assignment on the computer without help.
4. When I have approached a computer task, I have generally been able to find a way to complete it.
5. When problems have come up with a computer task, I have typically found a way to solve the problem.
6. When I failed at a computer task, I was extremely disappointed.
7. When things have gone wrong I have typically felt that it was my fault.
8. I have felt like something has always gone wrong when I touch a computer.
9. Successfully completing a task has brought me great pride.

Others' Successes or Failures

Based on the following criteria, indicate the amount that you recall experiencing the following when observing others' interaction with the computer: (1) never, (2) rarely, (3) often, (4) very often, (5) all the time.

1. I have typically surrounded myself with people who are good with computers.
2. I have been unaffected by the successes or failures my friends have experienced while using a computer.
3. I have felt jealousy or anger when I observed my peers succeed at a computer task that I did not succeed at.
4. I have felt happy when I observed my peers fail at a computer task that I succeeded at.
5. Generally, I have tried to learn from others' experiences (good or bad) with the computer.
6. I have tried not to make the same mistakes others have made with a computer.

Verbal Persuasions

Based on the following criteria, indicate the amount that you recall receiving Verbal Persuasion of your computer skills: (1) never, (2) rarely, (3) often, (4) very often, (5) all the time.

1. I have often been told that I am "good with computers".
2. My friends have often brought their technical questions to me.
3. When my friends considered making a computer related purchase, they have often asked my opinion first.
4. My professors/boss often compliments me on my computer skills.

AFFECTIVE COMPONENT

Based on the following criteria, indicate the amount that you recall experiencing the following emotions when dealing directly with a computer: (1) never, (2) rarely, (3) often, (4) very often, (5) all the time.

1. Nervous
2. Amazed
3. Excited
4. Awkward
5. Pleased
6. Eager
7. Frustrated
8. Self-conscious
9. Successful
10. Uncomfortable

11. Dumb
12. Happy
13. Gratified
14. Intimidated
15. Uncertain
16. Triumphant

Based on the following criteria, indicate the amount that you recall experiencing the following emotions when dealing indirectly (through conversation, reading a magazine, etc) with a computer: (1) never, (2) rarely, (3) often, (4) very often, (5) all the time.

1. Nervous
2. Amazed
3. Excited
4. Awkward
5. Pleased
6. Eager
7. Frustrated
8. Self-conscious
9. Successful
10. Uncomfortable
11. Dumb
12. Happy
13. Gratified

14. Intimidated
15. Uncertain
16. Triumphant

Revised Subjective Computer Experience Scale

COGNITIVE COMPONENT

Previous Successes or Failures

Based on the following criteria, indicate the amount that you recall experiencing the following when interacting with the computer: (1) Strongly disagree, (2) Disagree, (3) Neutral, (4) Agree, (5) Strongly Agree

1. When I have searched for something on the Internet, I have successfully completed my search.
2. I have successfully completed basic tasks with the computer (E-mail, Microsoft Word).
3. I have successfully completed advanced tasks with the computer (Statistics, Database, Spreadsheet).
4. I have successfully used complex programming languages (JAVA, Cobol, C++).
5. When a problem came up with a computer task, I have successfully found a way to overcome it.

Others' Successes or Failures

Based on the following criteria, indicate the amount that you recall experiencing the following when observing others' interaction with the computer: (1) Strongly disagree, (2) Disagree, (3) Neutral, (4) Agree, (5) Strongly Agree

1. My peers (friends) have been successful with searching the Internet.
2. My peers have successfully completed basic tasks with the computer (E-mail, Microsoft Word).
3. My peers have successfully completed advanced tasks with the computer (Spreadsheet, Database, Statistics).
4. My peers have successfully used complex programming languages (JAVA, Cobol, C++).
5. I have been affected by the successes or failures my friends have experienced while using a computer.
6. Generally, I have tried to learn from others' experiences (good or bad) with the computer.
7. I have tried not to make the same mistakes others have made with a computer.

Verbal Persuasions

Based on the following criteria, indicate the amount that you recall receiving Verbal Persuasion of your computer skills: (1) never, (2) rarely, (3) often, (4) very often, (5) all the time.

1. I have often been told that I am "good with computers".
2. My friends have often brought their technical questions to me.
3. When my friends considered making a computer related purchase, they have often asked my opinion first.
4. My professors/boss often compliments me on my computer skills.

AFFECTIVE COMPONENT

Based on the following criteria, indicate the amount that you recall experiencing the following emotions when dealing directly with a computer: (1) never, (2) rarely, (3) often, (4) very often, (5) all the time.

1. Nervous
2. Amazed
3. Excited
4. Awkward
5. Pleased
6. Eager
7. Frustrated
8. Self-conscious
9. Successful
10. Uncomfortable
11. Dumb
12. Happy
13. Gratified
14. Intimidated
15. Uncertain
16. Triumphant

Appendix F

Relating the Generated Theory to the Existing Literature

Links between the existing literature and the theory of successful engagement presented in this thesis are briefly considered in this appendix. The table below takes the literature cited within this thesis and identifies the key categories with which it each study is most closely allied.

Study	Focus	Associated Categories
Anderson [3]	Predictors of Computer Anxiety & Performance	Perception of Competence, Experience Accumulation
Ballantine07 [9]	Computer Usage & Self-assessed competence	Empowering Experimentation, Perception of Competence, Experience Accumulation
Badagliacco [8]	Computing Attitudes & Experiences	Experience Accumulation
Bandura [10; 11; 12]	Self-Efficacy	Perception of Competence

Study	Focus	Associated Categories
Beckers [14]	Computer Anxiety: A six-factor model	Perception of Competence, Experience Accumulation
Bergin [15; 16]	Factors Influencing Introductory Programming Performance	Empowering Experimentation, Perception of Competence, Experience Accumulation
Bergin [17]	Influence of Self-regulated learning on programming performance	Empowering Experimentation, Perception of Competence
Berkenkotter [19]	Disciplinary Communication	Language
Bizzell [26]	Initiation into Academic Discourse Community	Language
Bonar [29]	Principles of Novice Programming	Language, Experience Accumulation
Bozionelos [32]	Computer Anxiety & Computer Experience	Perception of Competence, Experience Accumulation
Brodie [33]	Reflective writing by Engineering Students	Language
Brosnan [34]	Computer Anxiety & Self-efficacy	Perception of Competence
Brown [35]	Language, Identity & Science Discourse	Language
Bruce [37]	Experience of learning to Program	Perception of Competence, Experience Accumulation
Buckland [39]	Comparison of Teaching Methods for English & Computer Science	Language
Byrne [40]	Student Attributes & Success in Programming	Perception of Competence, Experience Accumulation
Campbell [41]	Predicting Freshman Success in CS	Language, Experience Accumulation
Capstick [42]	Predicting Undergraduate Performance in Introductory Computing	Language, Experience Accumulation
Clarke [53]	Gender Based Factors in Computing Enrolment & Achievement	Empowering Experimentation, Perception of Competence
Compeau [56]	Computer Self-Efficacy	Perception of Competence

Study	Focus	Associated Categories
Connolly [57]	Retention Initiatives	Empowering Experimentation, Perception of Competence, Experience Accumulation
Connolly [58]	Adults learning Programming, Pedagogical Understanding & Anxiety	Perception of Competence, Experience Accumulation
Connolly [59]	Strategy for Alleviating Programming Anxiety	Perception of Competence, Experience Accumulation
Connolly [60]	Programming Anxiety & Retention	Perception of Competence
Corno [62]	Cognitive Engagement & Motivation	Perception of Competence, Experience Accumulation
Davy [66]	Research-led innovation in teaching & learning programming	Empowering Experimentation, Perception of Competence, Experience Accumulation
Doyle [72]	Computer Anxiety, Self-Efficacy, Computer Experience	Empowering Experimentation, Perception of Competence, Experience Accumulation
Dunican [74]	Learning to Program, First year students' experience	Empowering Experimentation, Perception of Competence, Experience Accumulation
Erickson [76]	Culture Differences in Science Education	Language
Evans [77]	Predicting Computer Proficiency	Empowering Experimentation, Language, Experience Accumulation
Garland [82]	Computer Experience & Computer Attitudes	Perception of Competence
Gathers [84]	Screening Computer Science Majors	Language, Experience Accumulation
Gathers [85]	Retention of Computer Science Majors	Empowering Experimentation, Language, Experience Accumulation
Guo [98]	Group Project & Retention	Empowering Experimentation

Study	Focus	Associated Categories
Hagan [99]	Prior programming experience	Experience Accumulation
Hagan [100]	Discussion class for teaching introductory programming	Empowering Experimentation, Language
Harrison [102]	Influence of Individual Difference on computing skill	Empowering Experimentation, Perception of Competence, Experience Accumulation
Hasan [103]	Influence of specific Computer Experiences on computer self-efficacy beliefs	Empowering Experimentation, Perception of Competence, Experience Accumulation
Hostetler [105]	Predicting Success in Introductory Programming	Empowering Experimentation, Perception of Competence
Howard [106]	Computer Anxiety	Perception of Competence
Huggard [107]	Computer Experience	Perception of Competence, Experience Accumulation
Huggard [108]	Recruitment of Computer Science students	Empowering Experimentation
Huggard [109]	Supporting Computer Science students	Empowering Experimentation, Perception of Competence, Experience Accumulation
Huggard [112]	Programming Trauma	Empowering Experimentation, Experience Accumulation
Hyland [114]	Teaching Programming	Empowering Experimentation, Experience Accumulation
Igbaria [115]	Computer Anxiety & Attitudes to Computers	Perception of Competence, Experience Accumulation
Jenkins [117]	Teaching Programming	Empowering Experimentation, Perception of Competence, Experience Accumulation
Jenkins [118]	Difficulty of Learning to Program	Empowering Experimentation, Language, Perception of Competence, Experience Accumulation

Study	Focus	Associated Categories
Jenkins [119]	Student Motivations	Perception of Competence
Joncour [121]	Computer Anxiety, Computer Experience & Self-efficacy	Perception of Competence, Experience Accumulation
Jones [122]	Diversity as a determinant of attitudes	Perception of Competence
Heinssen [123]	Assessing Computer Anxiety	Perception of Competence
Lave [130; 131]	Situated Learning & Peer Learning	Empowering Experimentation, Experience Accumulation
Leeper [133]	Predicting Success in Programming	Language, Experience Accumulation
Loyd [141]	Computer Attitude	Perception of Competence
Mahar [143]	Computer Anxiety, Computer Experience	Perception of Competence, Experience Accumulation
Marton [147]	Classroom Discourse	Language
Melin Emilsson [151]	Social Competence	Empowering Experimentation, Experience Accumulation
Moje [154]	Language, Literacy & Discourse in Science	Language
Moore [158]	Cognitive Styles among Computer Systems Students	Language, Perception of Competence
Pogner [166]	Writing and interacting in the Engineering Discourse Community	Language
Potosky [167]	A measure of Computer Experience	Empowering Experimentation, Perception of Competence, Experience Accumulation
Prince [168]	A bi-dimensional measure of Computer Experience	Empowering Experimentation, Perception of Competence, Experience Accumulation
Quigley [169]	Discourse Community building	Language
Ramalingam [171]	Self-efficacy & Mental Models in learning to program	Perception of Competence, Experience Accumulation
Ramalingam [172]	Computer Programming Self-efficacy	Perception of Competence

Study	Focus	Associated Categories
Rauchas [173]	Language performance & success in first year CS	Lanuage, Experience Accumulation
Rawstorne [174]	Subjective Computer Experience	Empowering Experimentation, Perception of Competence
Robins [176]	An account of outcomes in CS1	Experience Accumulation
Shashaani [182]	Computer experience & Computer Attitudes	Perception of Competence, Experience Accumulation
Sheard [183]	Performance and progression of first year ICT students	Empowering Experimentation, Language, Perception of Competence, Experience Accumulation
Sheard [184]	A study of a repeat group of CS students	Perception of Competence, Experience Accumulation
Sheil [185]	A psychological study of programming	Experience Accumulation
Smith [187]	The construct of Computer Experience	Empowering Experimentation, Experience Accumulation
Smith [188]	Computer Experience & Attitudes towards computers	Empowering Experimentation, Perception of Competence, Experience Accumulation
Smith [189]	Subjective Computer Experience	Empowering Experimentation, Perception of Competence
Stamouli [192]	Supporting Programming Students	Perception of Competence, Experience Accumulation
Stamouli [193; 194]	Learning Programming from the Students' Perspective	Language, Perception of Competence, Experience Accumulation
Steele [195]	Self-affirmation	Empowering Experimentation
Stewart-Gardiner [198]	Peer led team learning & retention	Empowering Experimentation, Experience Accumulation
Thomas [208]	Learning Styles & Computer Programming	Empowering Experimentation
van Braak [214]	Self-perceived computer competence	Perception of Competence

Study	Focus	Associated Categories
Varelas [216]	Content Learning and Identity Construction, Mathematics and Science & Elementary School	Empowering Experimentation, Language
Weil [221]	Technological Sophistication & Technophobia in University students	Empowering Experimentation, Perception of Competence, Experience Accumulation
Weil [222]	Computerphobia	Perception of Competence, Experience Accumulation
Wiedenbeck [225]	Factors affecting programming success	Perception of Competence, Experience Accumulation
Wilfong [227]	Computer Anxiety, Computer Use, Computer Experience & Self-efficacy belief	Perception of Competence, Experience Accumulation
Cantwell Wilson [228]	12 factors contributing to success at the undergraduate level	Empowering Experimentation, Perception of Competence, Experience Accumulation