Fall 12-2016

The Phenomenological Exploration of User-Design Among Junior Undergraduate Students Majoring in Computer Science in a Medium Sized Liberal Arts University When Developing a Game Prototype

David Tucker

THE PHENOMENOLOGICAL EXPLORATION OF USER-DESIGN AMONG JUNIOR UNDERGRADUATE STUDENTS MAJORING IN COMPUTER SCIENCE IN A MEDIUM SIZE LIBERAL ARTS UNIVERSITY WHEN DEVELOPING A GAME PROTOTYPE

A Dissertation

Submitted to the School of Graduate Studies and Research

in Partial Fulfillment of the

Requirements for the Degree

Doctor of Philosophy

David Tucker

Indiana University of Pennsylvania

December 2016

Indiana University of Pennsylvania
School of Graduate Studies and Research
Department of Communications Media

We hereby approve the dissertation of

David Tucker

Candidate for the degree of Doctor of Philosophy

_____ _____
Mark J. Piwinsky, Ph.D.
Professor of Communications Media
Advisor

_____ _____
James Lenze, Ph.D.
Professor of Communications Media

_____ _____
Anna V. Ortiz, Ph.D.
Assistant Professor of Communications Media

ACCEPTED

_____ _____
Randy L. Martin, Ph.D.
Dean
School of Graduate Studies and Research

Title: The Phenomenological Exploration of User-Design Among Junior Undergraduate Students Majoring in Computer Science in a Medium Size Liberal Arts University When Developing a Game Prototype

Author: David Tucker

Dissertation Chair: Dr. Mark J. Piwinsky

Dissertation Committee Members: Dr. James Lenze
Dr. Anna V. Ortiz

The purpose of this dissertation was to discover the themes of computer science students as they designed their own instructional systems. This was done by exploring the use of User-Design as a method of instruction for college computer science students. Data shows that there is a high attrition rate in college computer science programs. Perhaps, to solve this problem, this new form of instruction could be applied.

I investigated to find patterns while the students designed and developed computer games. In addition, I looked at how they design and cultivate their learning. I also observed and noted their experiences during the process. I used a phenomenological approach to observe computer science students as they developed games. I specifically chose upper level students with a history of success to study because they have already developed good learning strategies. I observed them as they developed their games and conducted interviews to capture their experiences used to develop themes.

This study suggests that User-Design has positives and negatives. Students are genuinely motivated and show a heightened sense of ownership when developing their games. But their learning lacked some needed structure that traditional class room instruction provides.

ACKNOWLEDGEMENTS

I would like express appreciation to my committee chair Dr. Piwinsky for all his excellent suggestions and motivation to complete this dissertation. I'd also like to thank my committee members Dr. Lenze and Dr. Ortiz for all their help to polish the writing. In addition, I'd like to thank Dr. Almeida for his suggestion of the study and help getting it started.

There are also some fellow students I'd like to thank, Chrissy Schaeffer for pushing me to continue; Dave Porter for his support during the coursework and Jennifer Forrest for her assistance in bringing the dissertation to completion.

Likewise I would like to thank Dr. Bennett of Edinboro University for his assistance getting participants for the study. I want to acknowledge the Edinboro Math and Computer Science department for their flexibility and Edinboro University's financial support.

To my children Max and Ivan and step daughter Katie, thank you for your patience while Dad was away.

TABLE OF CONTENTS

Chapter		Page
1	PROBLEM STATEMENT	1
	Introduction	1
	Statement of the Problem	3
	Need for the Study	3
	Purpose of the Study	4
	Significance of the Study	5
	Research Questions	6
	Terms and Concepts	7
	Limitations	9
	Conclusion	10
2	REVIEW OF THE LITERATURE	11
	Introduction	11
	Retention in Science Fields	12
	The Need for New Teaching Methods	13
	User Design/System Theory	15
	Systems Theory and How it Fits to Computer Science Curriculum	15
	Empirical Research Relating to User-Design	18
	Design-Based Research	21
	User Centered Design	22
	Participatory Design	23
	Rapid Prototyping	25
	Computer Games	26
	Why Games?	28
	Games as an Alternate Learning Method	30
	Why Browser-Based Games?	32
	Implications for Instructional Design	32
	Phenomenology as a Means to Study Computer Science Education	33
	Conclusion	34
3	RESEARCH DESIGN	36
	Introduction	36
	The Qualitative Research Paradigm	38
	Why a Qualitative Design is Appropriate	38
	Rationale for Conducting a Phenomenological Study	39
	Protocol/Methodology	40
	Unit of Analysis	48
	Background of the Participants	48
	Researcher Relationships and Power	59
	The Site	61

Chapter Page

Rational for Choosing the Site 62
Data Collection 62
Observations 63
Interviewing 64
Data Analysis Process and Procedures 64

4 RESULTS 66

Introduction 66
Content Selection 66
Process of design 67
Themes 70
Theme #1: User-Design Provided Genuine Enthusiasm for Learning. 70
Theme #2: User-Design Encouraged Focused Problem Solving. 77
Theme #3: User-Design Leads the Learner to Emphasize on Visual Outcomes. 85
Theme #4: User-Design has Two Phases, Participatory then Individual 90
Theme #5: User-Design Can Lead to a Lack of Structure Some Students Require. 94
Conclusion 97

5 CONCLUSIONS AND RECOMMENDATIONS 98

Summary of Findings 99
Answering the Research Questions 99
RQ-a: What Are the Patterns Found in College Students from a Medium-Sized University When Designing and Programming Their Own Computer Game? 99
RQ-b: How did the Students Design Their Instruction So That They Cultivated Advanced Programming Concepts Required for a Computer Science Class? 101
RQ-c: What Are the Experiences of College Students As They Design Their Own Instruction for Developing a Computer Game? 101
Implications for Practice 102
Study Reflections and Contributions 103
Limitation of the Study 106
Recommendations for Research 107

REFERENCES 109

APPENDICES 122

Appendix A - Letter of Invitation 122

Chapter Page

Appendix B - Informed Consent Form 124
Appendix C - Log Sample 128

LIST OF TABLES

Table Page

1 Participant Distribution of Characteristics....58

LIST OF FIGURES

Figure		Page
1	Susan's Game, Capture	68
2	Tim's Pong Game	69
3	Chris's Quiz Game	69

CHAPTER 1

PROBLEM STATEMENT

Introduction

With the desire to improve education to create innovators and problem solvers from students, educators are exploring new methods of instruction and curriculum delivery. Fresh approaches can help universities retain students in their chosen discipline and inspire them to achieve academic success (Monge, Fadjo, Quinn, & Barker, 2015). According to a U.S. Department of Education study which looked at beginning bachelor's degree students in the 2003-2004 academic year, 59% of students who initially entered into a computer or information science degree program left the major within the 1st year (X. Chen & Soldner, 2013a). Of the 59%, 31% switched to a different major and 28% left school. The same study also looked at students from 2003 to 2009 and found that 44.3% of computer/information science students changed majors after their 1st year (X. Chen & Soldner, 2013b). In addition, the Bureau of Labor statistics reports that computer and mathematical occupations will have an 18% increase in number of jobs from 2012 to 2022 ("Employment by major occupational group, 2012 and projected 2022," 2013). This is significantly higher than the average percent change in job growth of 10.8% ("Employment by major occupational group, 2012 and projected 2022," 2013). These statistics demonstrate an occupational need for more computer science graduates and therefore a reason to retain the students who had an initial interest as a computer science major.

Edinboro University is no exception to these statistics. Edinboro University is a medium-sized, four-year, primarily residential, public university with balanced undergraduate arts and sciences and professional programs, with thirty-four graduate programs ("The Carnegie Classification of Institutions of Higher Education, 2010 edition," 2011). In 2012, Edinboro

University had 139 computer science majors, yet only 11 graduated with a degree in computer science that year, demonstrating a decline in the number of majors each year. In comparison, 70% of all students at Edinboro returned after the 1st year and of those, 73% graduated or were still enrolled after 4 years ("Edinboro University Fact Book," 2013a). While there are many reasons for the high attrition rate in computer science, studies such as Beaubouef & Mason's (2005), suggest that the instructional design systems in use did not motivate the students.

This dissertation explores students' perceptions as user-designers using a phenomenological study. It will expand on the concept of User-Design, an approach where the end-users, or students, are empowered with the opportunity to design the process of their education (Alison Carr-Chellman, 2007). Typically, this role is left to the instructor. This dissertation will evaluate the results of what happens when students design their own system of learning. It will examine the methods students choose for learning and their level of success in comprehending the required curriculum. Armed with this information, computer science instructors can adjust their approach to teaching the curriculum by incorporating some of the students' own practices or letting the students design a personal approach to learning. Optimistically this will increase the retention rate in computer science.

Carr-Chellman and Savoy (2004) state that empirical research using User-Design is almost nonexistent. Studies of similar approaches -- such as Inquiry Based Learning (Aditomo, Goodyear, Bliuc, & Ellis, 2013), Instruction by Design (Jackson, Madison, Vangelisti, Daly, & Friedrich's, 1999), and Flipped Classroom (Tune, Sturek, & Basile, 2013) -- indicate that User-Design has potential as an effective instructional design that can facilitate learning.

Because this dissertation aims to further investigate the User-Design instructional model (Alison Carr-Chellman, 2007) and understand the perceptions that influence students' behavior,

I have chosen a qualitative approach. A group of students with a record of academic success will receive a topic to learn that will require high order thinking, defined as using critical, logical, reflective, metacognitive, and creative thinking (F. King, Goodson, & Rohani, 1998). They are asked to create an on-line game using previously unknown skills. During their self-directed learning, I will observe and interview the students to capture their methods of instruction and further understand their perceptions as they design the process of their education. The goal of this dissertation is to generate improvements to more traditional methods of instruction.

Statement of the Problem

Most instructional design models exclude the learner from the development of instruction (A. Carr-Chellman & Savoy, 2004). The User-Design model, however, empowers the learners to make decisions allowing them to design their own instruction. There have been fewer studies in the area of User-Design as an instructional model as compared to the number of studies in similar approaches such as Blended Learning (Staker & Horn, 2012) and Flipped Classroom (Strayer, 2007).

This dissertation attempts to contribute to the research in the field of User-Design as an instructional system. It will specifically observe college level computer science students' approaches to User-Design when creating a computer game. Computer games require proficiency in advanced computing concepts. This dissertation will observe the techniques the students used to acquire the knowledge required of advanced programming and record their perceptions of User-Design as a means of learning.

Need for the Study

User-Design was first advanced as an instructional method by Allison Carr in 1997 (Carr, 1997). User-Design is a model that empowers students to design their own learning environment

(B. Banathy, 1991). Almeida (2008) conducted a phenomenological study with high school students creating 2-D prototype games in a classroom environment using the User-Design instructional model as the form of learning. This dissertation will advance the User-Design model further by investigating the lived experiences (Van Manen, 1990) of college computer science students in a more informal environment. This dissertation is designed to serve as a catalyst for systemic change by advancing how computer concepts are taught and learned (Alison Carr-Chellman & Almeida, 2006). There have been few empirical studies regarding User-Design as a model (Alison Carr-Chellman, 2007) such as the ones investigated by Carr (2007) and Almeida (2008) . This dissertation is designed to add to the field of User-Design and contribute to the overall advancement of the field of Instructional Technology.

Purpose of the Study

The purpose of this dissertation is to discover the themes of junior, undergraduate, computer science students in a medium-sized liberal arts university as user-designers of their own instructional system. The themes discovered can be used to make improvements to the instructional design of the IT undergraduate program to attract and retain students to the major. Junior, undergraduate students were targeted because they are far enough in the Computer Science program to have developed their own study skills. This dissertation specifically looks at the students' own design process and decisions as they discover ways to construct and to solve complex problems in the context of on-line game prototypes. This dissertation attempts to advance User-Design as a learning model (Alison Carr-Chellman, 2007) in the context of game development (Almeida, 2008) in higher education.

This was accomplished by recruiting students from a computer science course where the instructor offered extra credit for participation. Seven students responded. They created a game

using JavaScript, a language, with which none of the participants were familiar. They were observed and interviewed to discover themes on how they designed their approach to learning.

This dissertation used a phenomenological approach (Corbin & Strauss, 2014) where the researcher attempted to understand the process by which upper-level computer science students teach themselves. This was accomplished by an analysis of observations, participant logs and participant interviews. I gave each participant a goal of programming a game using a programming language that is unfamiliar to them. The participants were free to use any method to learn the programming language that was new to them. By observing and interviewing the participants, I discovered common themes of learning.

Significance of the Study

The majority of studies in instructional design attempt to test models which are already established (C.M. Reigeluth, 1999). Common instructional design models include the Dick & Carey, ADDIE, and Gagne's Nine Events of Instruction (Gagne, Wager, Golas, Keller, & Russell, 2005). All of these instructional models advance the idea that an expert will guide the design process. With the high dropout rate in science, technology, engineering, and math (STEM) fields, including computer science, it is important to consider new approaches (Rask, 2010). This dissertation will look at alternate models of instruction in the context of computer programming. Computer programming requires higher order thinking skills (Williams, Wiebe, Yang, Ferzli, & Miller, 2002). The strengths of User-Design are consistent with the abilities of advanced computer science students because this method will pair these students' natural ability of self-direction with their innate ability to think critically (P. M. King, Wood, & Mines, 1990). This study is designed to contribute to the preliminary steps of helping instructors find an alternate method of instruction to teach computer science by gaining a better understanding of

the ways in which students naturally acquire knowledge through User-Design (B. Banathy, 1991).

The study in this dissertation is relevant to the topic because it leads to further studies that demonstrate User-Design (B. Banathy, 1991) being employed in the context of higher education. This study could be useful for other college instructors or administrators who are responsible for the design and implementation of computer science courses. Furthermore, this dissertation may lead to a systemic change in the way computer science skills are taught to students in the 21st century.

Research Questions

This dissertation's research question will be: What are the patterns within computer science undergraduate students' thought processes while using the User-Design model of instructional design when developing an on-line game prototype? The three questions that guide this study will be: (a) What are the patterns found in college students from a medium-sized university when designing and programming their own computer game? (b) How did the students design their instruction so that they cultivated advanced programming concepts required for a computer science class? (c) What are the experiences of college students as they design their own instruction for developing a computer game?

In order to investigate the research questions, specific underlying questions and issues needed to be included in the study such as exploring how computer science students are practicing User-Design and how their surroundings played a part in their own instructional design. How did they use the available resources? What engagement with fellow participants did they have? How did students of a medium-sized, rural university respond to User-Design? Can the results of this dissertation help comparable institutions? Does local culture and context,

such as a rural setting and medium-sized school with a mostly rural student body, have an effect on the participants (Von Bertalanffy, 1969)?

Terms and Concepts

The terms and concepts used in this study are:

Theme – A theme is a repeated pattern of responses or meaning (Braun & Clarke, 2006). In the case of this study found in observations, interviews and participant logs.

Object oriented programming – A computer software programming concept where the process of design revolves around objects and data used in an application such as a game. The advantage of this over procedural, or step-by-step programming, is that it allows for better design and reuse of proven procedures. For example, in object-oriented programming of a game of war, the programmer thinks in terms of tanks or soldiers and the actions only they can do. When a programmer creates an instance of a soldier, he is creating all of its properties and actions per soldier such as level of damage or fight ability (Dale & Lewis, 2012).

HTML 5 – Hypertext Markup Language, the 5th major revision of the core scripting language. HTML is a set of symbols and tags that are used in a text file to describe to a web browser how the document should look. The HTML 5 version is significant to this study because of its ability to animate and run the advanced JavaScript features that earlier versions could not do. Since the release of HTML 5, it along with JavaScript, have become common tools used to create advanced on-line games (W3C, 2014).

JavaScript – A dynamic programming language that is primarily used to program web browsers. Programs that are developed using the JavaScript language are used within the HTML 5 environment (ECMA, 2014). An HTML document is used to form webpages and provide a place to run the JavaScript programs. For example, an on-line currency converter

program would be written in JavaScript but displayed using an HTML document. JavaScript is significant for this study because of the community of programmers that have implemented libraries of source code and resources to learn from for game programming.

Browser-Based Game – A game played using a web browser such as Internet Explorer or Google Chrome as its platform. Typically, these games are hosted on a web server. To play a browser-based game, the users connect to a website where the game is located and start playing from there.

Inquiry based learning – Independent based learning by investigation of questions (Bruner, 1961). This study looks at this method of learning as a comparison to User-Design. Both are similar in that the student explores answers to questions. Inquiry based learning is used for more focused topics whereas User-Design is used for broader learning objectives (Alison Carr-Chellman, 2007).

Instructional design - A systematic process by which instructional materials are designed and course instruction methods are built. The result is an entire instructional experience for the acquisition of knowledge (Merrill, Drake, Lacy, Pratt, & Group, 1996).

Lived experience – First-hand accounts and impressions of living as a member of a group (Bernard, 2011)

User-Design – Engaging the users in the creation of their own systems of learning. The idea of a User-Design approach is that the student will see meaning and value in their learning since they are invested with creating their instructional design. This is beyond learning systems that consider the needs of the end users during the development stage. The hope is that User-Design will result in the student being highly engaged in his or her learning (Alison Carr-Chellman, 2007) .

Limitations

This dissertation does not attempt to, however, implement User-Design across a discipline or even a particular course. This dissertation's purpose is to discover the themes within computer science undergraduate students' thought processes while designing their own instructional system. This is done using the User-Design model of instructional design when developing an on-line game prototype. Some of the limitations encountered are discussed below.

One limitation was the participants' level of commitment. The IRB approval that was granted in this study limited the amount that could be given as extra credit from the instructor of the course in which the students were recruited. Extra credit was needed in order for the participants to have motivation for a high level of participation. The limit was set at a maximum of 2% depending on how I evaluated their engagement in the study. The extra credit percentage was suggested to be set low so that the students who completed the alternate assignment or chose to do nothing were not adversely affected or treated unfairly compared to the student who decided to participate. Two percent is not much extra credit from the perspective of the students. When considering their exams were worth 25% of their final grade, the extra credit from this participation can be compared to one question on one exam. The participants were asked to come in for observations on five Saturdays for four hours, work outside of observations on their game and see me for two interviews. During the study, I questioned if the students had lost some interest because of the amount of work and time required in relation to the extra credit received.

Because of the narrowness of the subjects of the study and the lack of sufficient reward, it is possible that the participation level of the participants and the quality and completeness of the data was impacted. The researcher believes that this did not significantly hamper the study's data collection or invalidate the results; but the possibility is worth mentioning here.

Conclusion

This study looked at applying the principle of User-Design using a phenomenology. The participants were all computer science students in their junior year. The participants were observed as they went through the process of learning a new computer programming language to create a web-based game.

In chapter two, the study examines relevant literature in the area of user-design and also compares it to related instructional design theories such as Rapid Prototyping, User Centered Design and Participatory Design. In addition, chapter two discusses the use of games and the implications for instructional design with User-Design. Chapter three reviews the method used in this study followed up by themes discovered in chapter four. Chapter five will then discuss the findings.

CHAPTER 2

REVIEW OF THE LITERATURE

Introduction

This chapter examines the literature regarding the field of User-Design and similar approaches of instructional design. This dissertation focuses on computer science college students learning higher-level concepts, specifically computer programming and game design. The focus of this literature review is on research that has implemented innovative instructional design for computer science and related curricula. More specifically the chapter looks at innovative ways that involve the student to a greater extent.

An overview of the relevant research will show how the research community understands User-Design and learning models related to User-Design. It will offer the theoretical foundations for this study. This chapter will provide a better understanding of what literature currently exists relevant to current research and practices in User-Design (Alison Carr-Chellman, 2007) and related teaching methods. This literature review will lay the framework for the necessity of this study in light of the absence of empirical data in this area. Innovation and advancement in this realm of user centric learning is needed to expand the tools and methods that are currently available for instruction and learning.

This chapter begins by looking at User-Design and how it relates to systems theory. This is followed up with a discussion of empirical research related to having students more involved in the design of instruction. Methods such as Design-Based Research, User-Centered Design, Participatory Design and Rapid Prototyping are examined. The chapter then discusses browser-based computer games and why having the students develop one is a good approach for this study. The chapter concludes with possible implications of implementing User-Design in a

college level computer science curriculum and a discussion of why a phenomenological study is appropriate.

Retention in Science Fields

There are several studies that look at retention in the sciences. These studies conclude that the rate of students leaving science majors, such as computer science, is quite high. According to a study by the U.S. Department of Education, 48% of bachelor's degree students who entered a STEM (Science, Technology, Engineering, Math) field between 2003 and 2009 had left by 2009 (X. Chen & Soldner, 2013a). There are many reasons for their departure. One study at Georgia Tech (Biggers, Brauer, & Yilmaz, 2008) attempted to quantify why students leave the computer science major. Some of the top reasons given were "I did not feel as if I belonged," "I am unhappy with my CS grades" and an "excessive workload" (p. 5). Important to this study were six reasons that relate to curriculum. These included: "CS curriculum did not provide enough flexibility," "CS classes were unfriendly," "Poor teaching by CS faculty or teaching assistants" and "CS curriculum is too narrow and could not bring together my interests outside of CS" (p. 6.) The authors of the study had recommendations but none that involved a change in the way curriculum was developed. A similar study done at Penn State came to parallel conclusions. Most notably, Penn State identified curriculum difficulty, poor teaching and advising. This study could contribute to improving the teaching aspect by suggesting a new approach to instruction (Marra, Rodgers, Shen, & Bogue, 2012).

Beaubouef and Mason (2005) conducted a study at Southeastern Louisiana University focusing on the causes of high attrition rates for computer science majors. They note as an example one year when their department had four hundred declared majors but only about 15 to 20 graduates. Their freshman computer science classes were full with multiple sections; but the

upper level courses were low enrolled, indicating a high attrition rate between the first and fourth year. The authors made several conclusions, a few of which are relevant to this study, specifically, poorly designed CS1 lab courses. In this context, a CS1 lab course is a first-semester, freshman course where students get hands-on practice writing programs. Because of the advantage computer science students get from a customized approach to learning in a lab environment (Konak, Clark, & Nasereddin, 2014), this type of course is a good candidate for User-Design. Graduate student teachers were also cited as a problem. While not directly related to this study, the paper points out that graduate student teachers lack focus on curriculum and instruction design. Very specific to this study is their conclusion suggesting the introduction of object oriented programming early on in the entire computer science curriculum. This is an especially interesting detail for this study because the nature of what the participants are learning is how to program using the object oriented methodology with the JavaScript language in the development of their game.

These studies point out that student attrition is a problem in the STEM fields including computer science. This, coupled with the expected high demand of tech jobs, will leave a gap in technology field employment (Guba & Lincoln, 1981).

The Need for New Teaching Methods

Educators need to continue to research new and innovative ways of teaching. The United States is falling behind countries such as Finland, the Netherlands, Singapore, and China in education achievement because these countries have invested in innovated teaching methods leading to better outcomes (Darling-Hammond, 2010). Darling-Hammond (2010) also stated that if educational conditions in America remain substantially stagnant, education achievement would eventually fail. Further, Darling-Hammond asserted it is the educator's responsibility to

place the nation on the correct path of education proficiency by experimenting with novel techniques of instruction.

Loewenberg et al. (2009), state that most quality teaching efforts should be focused on recruiting and providing new pathways to becoming a better teacher rather than developing new and innovative methods of teaching. They also offered explanations for why educators may be hesitant to innovate. For instance, using experimental methods can be perceived, by students, as the teachers are losing their teaching skills. Some may even insist that formal detailed teacher instruction is crucial to building quality educators (Loewenberg Ball & Forzani, 2009) who have an impact on creating innovative methods of teaching.

In a like manner, Goode, Margolis and Chapman (2014) expand on a previous research study relevant to the goals of this study. In order to make higher-level computer science courses more attractive, an innovative, educational technique that focuses on inquiry-based instruction was implemented. Students were told the topics for the assignment in advance. This strategy served as a motivator, giving students an appreciation of the overarching goal. Each student group could select any of the topics, which gave them a buy-in and a feeling of being empowered in their own learning. The teachers were coached to lead the students to query any resource, including the teacher, and the teacher continually queried the students to guide them. The results were positive with findings including an increase in students' perceptions of the usefulness of computer science, more satisfaction, finding it stimulating, and increasing motivation to stick with difficult problems.

Given these points, new and innovative instructional techniques are necessary to keep students engaged. Also shown is that the field of computer science in particular has

opportunities in its method of instruction that lend itself to an innovation approach such as User-Design.

User Design / System Theory

This study looks at an application of User-Design with respect to systems theory. System theory is the study of how interconnected parts work together to form a complete structure (D. Chen & Stroup, 1993).

Systems Theory and How it Fits to Computer Science Curriculum

This dissertation focuses on college level computer science instructional design, where Edinboro University is part of a larger system. The following studies suggest a revisit to science education and look at it as a system beginning from there and considering new approaches at every level. One study suggests starting by implementing change in the way lectures are done (Handelsman et al., 2004). Instead of the traditional approach, use discovery-based laboratories to engage the students. The researchers cite an example at Dickinson University where they offer a course called *Calculus-Based Physics without Lectures* and a North Carolina State University technique that uses a pure problem-based format for one of its courses. Here students work together to make observations and analyze lab results. The lab and lecture environment is physically shaped to promote this type of collaborative working by arranging the workstations to promote group problem solving. The study also recommends inquiry-based labs where the students are thought of as scientists. The lab assignment included questions promoting the students to research for solutions themselves. The authors conclude by suggesting this type of systemic change can start by better informing research universities of the resources and studies that have been done in science education. The authors suggest these new instructional techniques can filter down to other schools through their graduates (Handelsman et al., 2004).

In another study closely related to application of systems theory, a group of university professors attempted to reform how Los Angeles high schools teach computer science. Though this dissertation's focus is on college-aged students, it is relevant because of the approach to curriculum change and the use of student-centered instruction. The researchers wanted to change and add to the existing computing curriculum (Goode & Margolis, 2011). They took an approach to do a complete reform. This included summer workshops for the teachers on topics of computer science and how to teach using inquiry-based learning and programming coaches. The first attempt was not as successful as they wanted so they modified the curriculum with guidance from students and teachers to adjust the topics. With the addition of student input, they found that the students required more engagement and needed to see the importance of the material in order to increase learning (Goode & Margolis, 2011). This experiment supports a case for this dissertation because of the complete instructional reform of the system and the use of student-centered instruction and instructional design that closely relates to User-Design.

In a landmark study by Ausubel et al. (1968), the concept of the advance organizer was introduced. The advance organizer is a tool that students can use to facilitate User-Design. The instructor gives information to the students, who then categorize it into smaller sub categories of learning concepts that the instructor wants them to obtain. It provides a preliminary structure upon which the students build their method of learning (Ausubel et al., 1968). The subject matter is expressed in the form of higher levels of abstraction rather than direct instruction. This tool helps students organize and categorize information as it is introduced and gives them a means to chart their own learning process. This has been shown to help students obtain the learning objective (Korur, Toker, & Eryılmaz, 2016).

Science students with a record of academic success will take a systemic approach to the design of their own education (Leonard, 1997). Typically, the student will apply active learning strategies by engaging physically, emotionally, and mentally with the topic. They may use the concept of the advance organizer (B. Chen & Hirumi, 2009) and categorize the required learning in order to create a system of learning. Using it as an instructional tool to show students the relationship between what they are currently learning and putting it together with what they have already learned gives the student a bigger picture (Korur et al., 2016). This process involves developing categorizations of the subject matter then applying priorities on what to focus on. Each category could then be broken down further, building a framework from which the student will have designed his or her own learning system. The student becomes more intellectually vested in this method of learning because they have ownership of it and feel empowered with their own learning (Chan, Graham-Day, Ressa, Peters, & Konrad, 2014). This research directly relates to User-Design (B. Banathy, 1991) in that the user has designed his or her own learning environment.

Given these points, User-Design is used as an approach where stakeholders, in this dissertation, students, are involved in the design of their instruction more than what is customary (B. Banathy, 1991). They are responsible for the design of their own instruction (Charles M. Reigeluth, 1993). This approach requires a complete method to the design of the instructional system. Bringing this aspect of system theory (Von Bertalanffy, 1969) together with User-Design (Alison Carr-Chellman, 2007) as an instructional design model is shown to be successful.

Empirical Research Relating to User-Design

A recent study looked at developing curriculum by means of User-Design (Jordan & Carr-Chellman, 2014). At the Bureau of Labor Statistics when an agency wide team was tasked with developing a curriculum for its economists in order to teach a wide variety of tasks, User-Design was chosen as the development method. A team of experienced economists was assembled and one was placed in the position of team leader and acted as a facilitator rather than an expert in the subject matter. As User-Design requires, the team members were also to be the curriculum's end users (A. Carr-Chellman & Savoy, 2004). Using Microsoft SharePoint, the team collectively designed the curriculum as an iterative process. As the team progressed, the team leader noted that the resulting curriculum was better than expected and better than what he could have done himself. In addition, the curriculum was easily distributed and understood by its end users because many of the staff economists participated in its development (Jordan & Carr-Chellman, 2014).

Lim (2008) suggested that students playing games such as *Never Winter Nights*, a role playing game that immerses players in medieval history, can lead them to steer their own learning of history as they make decisions in game play. Lim referred to this as the "designed play experience" (Lim, 2008). Two other games mentioned were *Rollercoaster Tycoon*, a game that helps teach business strategies and *Rock Band*, a game that players can use to learn to play an instrument. Both of these games are structured to allow the player to design their own learning by deciding on difficulty level, scenario selection, and when they engage. Lim (2008) also mentioned that when students are empowered to take charge as co-designers of their own learning with the instructor, they become more interested and determined. Thus, one way in which to empower students is to allow them to design a game based on school curriculum.

Levy and Petrulis (2012) used a similar education approach to User-Design known as inquiry based learning. The method encourages learning by the use of investigating questions. The researchers wanted to know how students experience and understand curriculum-based inquiry and research. In addition, they wanted to know how these experiences relate to desirable educational outcomes. In this qualitative study, first year college students in the arts, humanities and social sciences were asked to come up with answers to certain questions and left to design their own methods to acquire the knowledge to answer the questions (Levy & Petrulis, 2012). The researchers found that the majority of the students were highly motivated to learn the material and found relevance in what they were doing. The study contained many quotes from the students embracing the idea of researching for themselves the solutions to projects given to them by their instructors. For example, one student mentioned that he liked that the university is not all about learning passively (Levy & Petrulis, 2012). Another student observed that he was freer to examine a topic(Levy & Petrulis, 2012). More specifically, he discovered a theory that he disagreed with and was able to discover several examples supporting his opinion (Levy & Petrulis, 2012). This is indicative of achieving a rich and diverse learning experience.

In Almeida's (2008) phenomenological study, rural high school students created their own game using the principles of User-Design where the students developed their own method of learning. He showed that User-Design could be successful and provided empowerment and ownership for the students. Almeida discovered five themes from his observations; (a) The use of User-Design can be implemented though empowerment and ownership of the student's work, (b) The students found the User-Design to be a fun experience, (c) by nature User-Design is a participatory activity, (d) User-Design can be a tool for problem solving, and (e) implementing

User-Design is demanding. In Almeida's study the overall goal of students learning how to program in Game Maker was achieved. The students were engaged more than what was typical.

An extensive literature review of data from workshops, student focus groups, and case studies found that students should be participants in curriculum design (Bovill, Morss, & Bulley, 2009). These researchers found that when students were involved in curriculum design decisions, they found learning to be more relevant and authentic. A common form of student participation in curricular design comes from student feedback on courses via staff-student liaison committees, questionnaires, focus groups, and electronic voting systems. The input from these tools can drive changes to teaching design.

Following a review of the relevant literature on User-Design, and its primary contributors Carr (2007) and Almeida (2008), four instructional design techniques emerged as being closely related to User-Design and accepted by the instructional design community. The first is Design-Based Research. This technique is used to continually improve instructional techniques by continual review among researchers and educators in a real-life setting (Barab & Squire, 2004). This technique is similar to User-Design in that there is a continual revisit of the design process but differs from this technique because of the lack of student input. The second is User-Centered Design. This method closely focuses on the requirements and limitations of the students when designing instruction (Abras, Maloney-Krichmar, & Preece, 2004). It is related to User-Design in that the student is the focus; but, the instructors are still the main instructional designers. The third instructional design method considered is Participatory Design. This design process has its roots in community projects where every stakeholder, not just the end users, can contribute to the design. Rapid Prototyping, another instructional design method has spun off of this (B. H. Banathy, 1992). Similar to the others, this method differs with User-Design because of its lack

of focus solely on the end-user. Rapid Prototyping has its roots in design engineering and has off-shoots in various areas including instructional design (Chua & Leong, 2003). Similar to the others, this technique uses continual end-user input during the development process. The difference here is the end products are always being rebuilt at each stage and only moves on when the designer and end user agree (Tripp & Bichelmeyer, 1990). Rapid Prototyping and how it relates to User-Design will be explored further in this chapter.

User-Design was chosen for this study because of the lack of research done in this area and more specifically in the context of college-aged students. Because of the need to explore new techniques as outlined earlier, this study investigates the possibility of User-Design as an instructional design method for college level, computer science curriculum.

Design-Based Research

This project will incorporate design-based research. As defined by Wang and Hannafin (2005), design-based research aims to improve educational techniques by design evaluation and implementation followed by analysis of proposed educational techniques using real-world settings. Design based research requires facilitation in an authentic setting and the involvement of the participants in the study. As such, it requires the collaboration of the researcher and participants for successful application. Additionally, there is an inherent requirement for documentation of the outcomes as they relate to the development process (Wang & Hannafin, 2005). Because of the continuous iteration in design-based research, the researcher is a participant in addition to the person who generalizes the results. One downside is that special care is needed to recognize that the findings are part of this interaction. The benefit is that the knowledge gained can be applied right away. Because of these characteristics, the results are easily and quickly translatable to actual practice (Hoadley, 2004). Barab (2004) elaborates on

this further stating the importance of research in real-life settings, where most learning actually occurs, is vital to the implementation of design based research results. Design based research focuses on characterizing the situation and all relating complexity with an emphasis considering different variables that can affect learning as it is happening. Often there are complex social interactions in the learning process among students and their peers. What Barab (2004) says is directly what this research is attempting to do, to gather the data and consider any aspect of learning and possibly discover some previously unknown characteristics of learning.

In one example involving design-based research, science teachers in a secondary school designed and developed games for use in their classrooms as an instructional aid (Annetta et al., 2013). The study found that during the game creation process, teachers thought of new forms of instruction they may have not realized without the game development. In addition, they discovered that past practice with computers, in general, greatly affected the outcome of the game quality. Interestingly the researchers mentioned that a next step in research was to consider asking the students for guidance on the instructional design (Annetta et al., 2013), similar to what I intended to do with this study.

In the same way as the previous studies, this study aims to investigate real-world problems by designing interactions and expanding on the theory of User-Design. The nature of this study makes it interactive and flexible, which aligns well with design-based research (van den Akker, Gravemeijer, McKenney, & Nieveen, 2006).

User Centered Design

User-Design differs from User-Centered Design in some fundamental areas. User centered design has an instructor creating the instructional system with an emphasis on the users at the center of the creation process in all the stages of design. User-Design (Alison Carr-

Chellman, 2007), on the other hand, leaves the strategy of instruction completely up to the end users with minimal guidance by a subject expert (B. Banathy, 1991). User-Design is process-oriented in that the end-user is continually considered when developing course delivery and is actively involved in the entire instructional design process (Baek, Cagiltay, Boling, & Frick, 2008; Alison Carr-Chellman, 2007).

An example implementation of User-Centered Design involved the use of a game called EverQuest II as a language learning game (Rankin, McNeal, Shute, & Gooch, 2008). The game is a massive on-line roleplaying game. The students were required to create accounts where the opposing players' native language was set as the language for the game. The study was designed as a pre/post-test experimental design; some students acted as a test group where traditional instruction was replaced with game play. Playing the game allowed the students to design their own learning, deciding what quests to undertake while being required to communicate with others in a different language in both oral and written communication (Rankin et al., 2008). This study showed that students, using a User-Centered game based approach, could learn as an alternative to traditional lecture. In addition, the research found that these students enjoyed learning and became more engaged than they would have with traditional lecture (Rankin et al., 2008). As has been noted, User Centered Design which has a similar approach to User-Design, both using student designed learning, is an effective method of instruction.

Participatory Design

Participatory design integrates the end-user with the instructor as co-designers in an instructional system. The objective is to have the resulting technology more suitable to the needs of its users. This differs from User-Design in that the end-users are co-designers at the preliminary stages of instructional design. Users meet with the originators of the technology --

for this paper that would be the teacher -- to design the learning system (Simonsen & Robertson, 2012).

In one example implementation, a group of researchers experimented with several secondary school classes with the purpose of finding out if participatory design could work (Könings, Brand-Gruwel, & van Merriënboer, 2010). They randomly selected a small group of students from each of the courses in the study. These students then met with the teacher to brainstorm ideas for the instructional design. The researchers then surveyed the students and teachers to see how satisfied they were with the design and if the course met the needs of the curriculum. The results showed that, although the students felt uncomfortable at first, they eventually became comfortable with the meetings and felt empowered regarding their education. (Könings et al., 2010).

In another example of participatory design, a group of faculty at Virginia Tech wanted to determine if high school teachers could be effective by jointly designing a computer network learning community where students and teachers could share educational technologies between different schools (Carroll, Chin, Rosson, & Neale, 2000). Typically, the participatory design process does not last much past the initial design phase meeting, but surprisingly in this case, the teachers were involved for the entire process. The teachers and students were pleased with the outcome and inter-school student engagement proved to be an effective means for learning (Carroll et al., 2000). The results of this study show a possible outcome related to User-Design in that students were observed designing their learning throughout the time they were building their game.

Rapid Prototyping

Rapid prototyping has its roots in software engineering where the developer creates a user interface prototype in a short amount of time for a client to consider. The developer then gets feedback on whether the prototype meets the client's specifications. This process is repeated until the client is satisfied with the skeletal structure and interface of the software. The software developer then proceeds to write the computer code, which is the bulk of the work (Dale & Lewis, 2012). This type of development cycle is also used as an instructional design tool. In this case, the instructional designer consults the users, or the students, to see if the proposed design of instruction is clear and will be effective for learning. For example, this can include designing and organizing the interface of a learning management system. Typically, only the interface and references to objects are created at this point (Tripp & Bichelmeyer, 1990). The instructional designer takes feedback, makes rapid changes according to the reaction, and repeats the cycle until the end-users are satisfied with the interface and design. The instructional designer will then use that interface and create the objects and resources for which the prototype calls (Tripp & Bichelmeyer, 1990).

One example of rapid prototyping is a student who created a computer-based grammar tutorial for foreign students (Tripp & Bichelmeyer, 1990). The designer began to notice that his system was becoming too complex for students to understand. The prototype took the designer hours to produce and was tested soon after with end-users. The feedback answered many of the designer's questions and he was able to move rapidly toward an effective finished product. This form of instructional design only works if the designer is using a system that can be easily modified, such as a computerized learning management system (Tripp & Bichelmeyer, 1990), or rapid prototyping.

Another study that implemented rapid prototyping looked at creating instructional design at the commercial level, where the company's business was instructor-led training. In this case for the automotive industry (T. Jones & Richey, 2000). They create instructional designs for Internet certification testing, hypermedia software and any other custom instructional needs. The instructional designers are also researchers with a PhD or working on a PhD. They looked at two particular projects and applied rapid prototyping design as close as possible. The study describes the success they had with the continual interaction with the customer (student) in each phase of the project. The researcher did not want to be in a situation where their design was done before the customer reviewed it. This proved to be very successful in designing a quick, high quality instructional design. The paper's main focus was on one project. In it they had meetings with the customer, referred to as a "Designer/Customer Task" a total of 11 times and had only five exclusively designer tasks. The research charts the interactions and demonstrates the effectiveness of rapid prototyping. Some specific points they note were that every time they met with the customer/student, they kept in mind the final product, user needs, objectives, and a continual revisit of the previous design (T. Jones & Richey, 2000). This study is an interesting comparison to User-Design because of all the student input and the demonstrated success of the end result. Their entire design was student driven and when the design was first finished, both the instructor and student were satisfied with the outcome.

Computer Games

Instructional designers are borrowing from various new sources, such as games, for their user environments to provide new and innovative ways to deliver instruction. This provides an alternate new media for the design of education and interactive learning. Dickey (2005) observed that the importance of the design of these games has been ignored in research and that

educational game design is particularly important. Dickey (2005) also mentioned that the design, and not just the playing, of educational games helps learning. In a study involving students immersed in a 3-D game environment it was shown that motivation to complete a task was much greater using a game than the desire to earn a better grade (Barab, Pettyjohn, Gresalfi, Volk, & Solomou, 2012).

The creation of game design can be integrated into various types of learning environments and activities such as problem-based learning, project-based learning, case studies, and education games and simulations (Rollings & Adams, 2003a). Electronic games require active engagement in environments that support discovery, observation, trial and error, and problem solving. Prensky (2003) mentioned that games are graphical environments that require players to read the visual environment and interpret symbols. Elements of challenge, fantasy and curiosity found in successful games could be used in the development of instructional technology (Dickey, 2005).

According to researchers (B. Jones, Valdez, Norakowski, & Rasmussen, 1994; Schlechty, 2001) learners require the following for success: focused goals, challenging tasks, clear and compelling standards, protection from adverse consequences for initial failures, affirmation of performance, affiliation with others, novelty and variety, choice, and authenticity. User-Design works well with this idea in that the instructor will provide goals, tasks, standards, and affirmation while the learner provides the rest by their own design. The use of designing and programming a game as the assignment for the participants will support this study because of the various dimensions of game development. Game development requires creativity, passion, intelligence and programming skill (Murphy-Hill, Zimmermann, & Nagappan, 2014). Because the games will be web-based, they will be programmed using the JavaScript language which uses

an object-oriented paradigm. In addition, college aged, computer science students tend to be interested in games and their development (Sweedyk, deLaet, Slattery, & Kuffner, 2005).

In relation to those components and goals stated above, game design provides focused goals through creation of narrative, character roles, and perspective. The following is a list of how these learning needs should be incorporated into gaming and game design (Howland, 2002; Rollings & Adams, 2003b). First is the inclusion of challenging tasks. Here designers set action, resource, tactical, strategic and time hooks. Hooks are something that make the game unique and draws in the players (Fullerton, 2014). The player must have initial protection from adverse consequences, so they do not lose interest. The player must be rewarded for good performance from the game and other players if possible. More advanced games also incorporate a story of some sort for the player to follow and usually provide a goal.

The process of design and programming a game includes a variety of aspects which are valuable as an assignment for this study (Jonas, 2013). The participants must research topics ranging from good game play to learning the syntax of JavaScript.

Why Games?

Students in recent years have grown up with the concept of computer games. It is common to begin playing games as soon as children develop the motor skills to do so. Today's students are accustomed to computer games and already have some ideas of their own on what makes a good game (Koster, 2005). Students interested in games are familiar with learning how to accomplish tasks in this environment and understand complex ideas (Schell, 2014). They take in information from many sources and make decisions quickly to deduce a game's rules. Instructional designers can take advantage of this enormous potential for learning (Prensky, 2003) by empowering users to design their own system of learning (Banthay, 1991).

There have been various studies done in the area of computer game education. Shabanah (2009) has created several games to teach introductory computer science topics, by implementing games that teach the binary search concepts via a pong-like prototype. In addition, Shabanah (2009) created a game to teach the selection sort algorithm, where the player must sort dominos in a finite amount of time. Shabanah (2009) concluded that with the computer game, creation and play is a natural motivation for students to become engaged in learning complex concepts.

The creation of a serious game provides a means to focus on the psychology of learning as it relates behavioral, cognitive, and motivation theories. Not only will the game creators need to learn game design but must also grasp effective principles of learning. In addition, serious game designers need to consider how information is stored, retrieved, and represented in order to build a quality educational game. Carefully designed and developed interactive game experiences can provide the discovery aspect of how to design while at the same time learn the material involved in the game (Greitzer, Kuchar, & Huston, 2007).

Game creators must also consider cognitive principles that lead to active learning. According to Guillén-Nieto & Aleson-Carbonell (2012), games should follow active learning instructional guidelines such as concepts including: starting small and building up, giving immediate feedback, setting realistic goals, and making problem-centered activities challenging but not so much that the player cannot succeed. The participants of this dissertation will need to consider these principles as they design and create a quality game. This directly ties back to me wanting to see how they learn to design this type of game. More than just the programming aspect, they must also research the theory of good game play.

Prensky (2008) demonstrated that students can design and build games for education. He mentions how they prefer to build games for entertainment and even create games for education when in the right environment, e.g. *MeCHeM, Waste of Space,* and *Elemental* for middle school science. Prensky (2008) also noted that students, as game designers and builders, could be very effective learners because they are being empowered to become instructional designers facilitated by their recent experience as students.

Games as an Alternate Learning Method

For this dissertation, the project of a game was selected because of my opinion that students prefer programming assignments that develop games and that programming games would attract more participants. One study that closely examines this assumption suggests this to be true. Cliburn (2006) conducted a study in an introductory programming course where he developed ten different projects for five different assignments. For each assignment, the students could choose between a game-based or non-game based project. Of the total assignments turned in, 78.9% were the game options. The author did note that the performance on the non-games was higher than the game options. It was discovered that some of the better students selected the non-game option because of the perceived increase in challenge. A survey was given at the end of the semester. The results showed that the students preferred the game assignments because they had a better understanding of the problem and they thought it would be more fun than the alternative.

Further research (Sung et al., 2011) suggest the same results as the previous study. In a study involving the use of games as projects for programming courses, the researchers looked at first and second year programming courses and developed potential game and non-game assignments (Sung et al., 2011). This study went into more depth making sure the assignments

were teaching the appropriate topics of each class and involved an independent reviewer. Similar results were shown as the previous study (Cliburn, 2006). In particular, the overall pass rates were higher, between 7% and 10% and participants had consistently higher scores on the individual assignments (Sung et al., 2011). This study also demonstrated that, on average, students spent less time on the game projects then the non-game projects. Interestingly, this study's survey showed no significant difference in students' interest and satisfaction in game programming at the end of the course (Sung et al., 2011).

In another paper (Sweedyk et al., 2005), four experienced professors who have taught computer science courses using games gave their personal experiences and reasons for choosing games as programming assignments. The common theme among the instructors is that the games motivate the students to work on the project more than assignments they have previously given. In addition they discussed that the game programming assignment can cover the programming topics of the course as well, if not better than, the traditional assignments. Often mentioned was that students enjoyed the game-related assignments because they are domain experts since they already had experience playing and understanding games (Sweedyk et al., 2005).

This inquiry into using game development as a project confirms my opinion that this will motivate the participants to complete their game and maintain interest in the project. The literature suggests that game development would be as good, if not better, than traditional programming assignments because of students' interest in games and their experience as users. As can be seen from the research, a game development project will also serve well for a User-Design study.

Why Browser-Based Games?

As games have progressed through historical development, browser-based games have emerged as one of the most common game types to play. This is because of the availably of the Internet, Internet browsers and minimal computer hardware needed to play (Klimmt, Schmid, & Orthmann, 2009). In addition, browser-based games do not require any special software to develop. The programmer can use any standard text editor but there are text editors designed for JavaScript, such as the language HTML5 browser-based games are developed in for this study (Mikkonen & Taivalsaari, 2007).

The participants in this dissertation study were asked to develop browser-based games in JavaScript because none of them had much, if any, experience with JavaScript and the participants would not require any special software or hardware to create their game. This freed them up to work anywhere, at any time, to design their instruction and eventually their game. In addition, all of the participants had an interest in developing a browser-based game and learning the JavaScript programming language, helping with motivation.

In summary, it has been shown that the choice of having participants in this dissertation study creating a game has many advantages. Games will keep the participants self-motivated to work on the project and learn a new language.

Implications for Instructional Design

This project focuses on giving students the means to learn computer programming and game design techniques on their own. Observing students designing their own learning could provide important feedback to instructors on how to design learning for future classroom initiatives. It is fair to say that in a college setting, pure User-Design is impractical because of curriculum requirements and learning objectives. For example, at Edinboro University every

course has a pre-defined outline of learning objectives and curriculum to follow. If an instructor would employ User-Design with no other instructional methodology, chances would be high that students would not be introduced to all of the required material (Alison Carr-Chellman, 2007). Nevertheless, the principles of User-Design can be used to enhance the traditional guided instruction often found in a typical college classroom, thus enriching the student experience and scholastic environment.

Phenomenology as a Means to Study Computer Science Education

The following looks at studies that use phenomenology in computer science education research. In one example the researcher wanted to explore the use of Internet cafés as an educational tool (Cilesiz, 2009). The study took place in Turkey where it is common for students to use Internet cafés because of a lack of their own computer and Internet connection. The researcher wanted to understand and describe the phenomenon when students are using computers in the Internet café environment as an educational tool. This study was conducted using six participants with three interviews per participant. The author came to several previously unknown conclusions and uncovered themes. Namely that computer games were used as a stepping stone into the formal study of computer science, Internet cafés became communities for people with an interest in computer science and Internet cafés became a context for identity building (Cilesiz, 2009). This study is included as a comparison because of its similarity in goals and methods to this dissertation study showing a successful use of phenomenology in computer science education. As with this dissertation, the author used interviews and analyzed the data to develop common themes.

In a related study using phenomenology, Rose (2011) attempts to get insights into students' experiences of reading electronic texts. The researcher gathered data from ten students

who responded to a volunteer request by conducting one interview per participant. The researcher then analyzed the data thematically until what she calls "units of meaning" started to emerge. Her data revealed six main themes. She lists these themes along with excerpts from the interviews to back up her assertion. In this study, the author wanted to understand the lived experiences of students as they read text on a computer screen. This study demonstrates that phenomenology can be an appropriate method for this type of study. Rose's (2011) study is similar to this dissertation study because of the method of collection of data and the focus on discovering themes from lived experiences.

Though they are not directly related to this study, the two previously mentioned studies are similar in that I am looking to analyze the lived experiences of students as they attempt to learn. The previous studies use phenomenology as a means to gather and analyze data about the lived experiences of students. Both approaches use interviews and effectively analyze the data to recognize previously unknown and relevant information. This dissertation study will also analyze interviews and, in addition, observations I made.

Conclusion

The literature reviewed illustrated that implementing User-Design in higher education, could be incorporated into the traditional classroom to assist in successful student outcomes. If implemented with care, User Facilitated Learning supported the overall enrichment and benefit to the learner's experience. User-Design supports the higher thought functions and processes that are necessary for learning complex concepts (Alison Carr-Chellman & Almeida, 2006). User centric learning projects from both the student and teacher perspectives have proven valuable supplemental components for learning (Alison Carr-Chellman, 2007).

Based on the supporting literature, observing the way in which the participants develop browser-based games will provide insight into how students design their own learning to enable the discovery themes in their learning while using user-design, which is ultimately the purpose of this research.

CHAPTER 3

RESEARCH DESIGN

Introduction

This is a phenomenological study (Van Manen, 1990) involving the observation of computer science students at Edinboro University creating browser-based computer games. The project is building upon Almeida's (2008) work where the application of User-Design (Alison Carr-Chellman, 2007) with rural high school students creating computer games was studied (Almeida, 2008). This study differs in several ways. First, the participants are all college students majoring in computer science at a medium-sized, liberal arts university. All the participants have taken the upper-level programming requirements of the degree. All participants were juniors. This is significant because they all have experience in programming and have a proven record of success in doing their own complex programming projects. In addition, this study's participants are not meeting in a classroom environment continually on a daily basis as part of their normal routine that is typical of a high school setting. Other differences include the tools used to create the game. Almeida (2008) had his participants use the Game Maker software development package to create the games. Game Maker is only used in game development. This study's participants will be using JavaScript with HTML5. These are much more generic programming environments and are used for a variety of applications. One similarity is that both studies investigate the effectiveness of User-Design as an alternative to traditional teaching of programming concepts.

This study is exploratory in nature; the goal is to describe what the learning experience of undergraduate students majoring in computer science is when developing a computer game. The participants developed their computer games on their own computers and in a university

computer lab where they were observed. The decision to collect the participant's information *in vivo* via interviews, observations, and artifacts, was used because this project is qualitative in nature and explorative in design. I was more concerned about observing the nature of the experience of creating games to advance the understanding of User-Design and discovering what this learning experience was like for the student.

Because the intent of this study is to describe the lived experiences of the participants as they create games, the phenomenological research method (Van Manen, 1990) was used in this study. There are many forms of phenomenology. The hermeneutical form of phenomenology is used here versus others such as transcendental or existential because this study is looking for themes derived from interpretation and textual meaning (Kafle, 2013). In addition, this form of phenomenology is a good choice for looking at multimedia systems such as game and software development interfaces (Kafle, 2013).

This chapter provides a detailed description of the components of this study's research design. It begins with an explanation of the qualitative research paradigm and why I chose this approach. This is followed by a description of the study, starting with the unit of analysis. It will profile the relationships of the participants and the context in which they did the study. Then, this chapter will explain the rationale for choosing the site and provide information about where the participants worked. Next, it will include a detailed description of how the data was collected and analyzed. Lastly, this section will discuss known issues of quality and any pertinent ethical considerations.

The Qualitative Research Paradigm

According to Patton (1990) qualitative research uses natural settings as the source of data. The researcher wants to observe, describe, and interpret settings as they are. The researcher is a human instrument of data collection. Qualitative research uses inductive data analysis to discover important patterns, themes and interrelationships rather than deductive conclusions. The participants will describe their lived experiences when creating a game; the researcher will garner themes from these experiences.

Quantitative studies start out with *a priori* assumptions or hypotheses that are tested to determine their validity. Qualitative studies, on the other hand, want to discover the meaning of experiences certain events have for the participants and then perform the analysis after the data has been collected. This data will hopefully lead to new discoveries by me. Creswell (2009) described in a general way how to perform qualitative studies: (a) gather information, (b) ask open-ended questions, (c) analyze the data to form themes or categories, (d) and look for patterns or generalizations with the intent to relate the results to past experiences and existing literature.

There is no prediction of what will be discovered in this study. The intent is to realize previously unknown characteristics about how college students go about learning in a User-Designed environment.

Why a Qualitative Design is Appropriate

A qualitative approach to this study is appropriate because this study is looking at lived experiences in a natural setting. There are no assumptions in advance of the study; the aim here is to discover the nature of using User-Design learning within the framework of creating a computer game. This study is not testing or quantifying a hypothesis; there are too many combinations of variables to consider. The purpose is to gain meaning and perspectives from the

participants in their own setting to discover any processes or patterns that may arise using the User-Design approach of instructional design.

Qualitative research uses various methods such as predicting cause, determining cause, explaining factors, discovering if a design will work, observing, and describing (Marczyk, DeMatteo, & Festinger, 2005). Since this study is designed to observe and describe the lived experiences of university students using a naturalistic observation as they develop games, a qualitative method is appropriate.

Rationale for Conducting a Phenomenological Study

Van Manen (1990) stated that phenomenology looks for the very nature of a phenomenon that makes sense of something. Phenomenology is a philosophy with a focus on the lived experience and those engaged in this type of research focus on an in-depth meaning of a particular aspect of experience (Rossman & Rallis, 2012). Phenomenological studies focus on descriptions of what people experience and how they experience a phenomenon (M. Patton, 1990). Creswell (1998) stated "Researchers search for essentials, invariant structure (or essence) or the central underlying meaning of the experience and emphasize the intentionality of consciousness where experiences contain both the outward appearance and inward consciousness based on memory, image and meaning" (p. 52). Phenomenology allowed me, the researcher, to see the lived experiences through interviews, observations and student logs. Phenomenology also allowed me to journal during the observations as students experienced the actual process of learning and creating their games. For these reasons, phenomenology was the most appropriate method to describe the human experiences of the participants as they worked to create their games.

There are several general assumptions that must accompany the phenomenological approach: that the study will be able to accomplish its tasks by observations and interviews; that meaning can be gained by focusing on the participant's individual perceptions; and the researcher can construct interpretations based on what the participants are thinking (Creswell, 1998). In order to accomplish this, as much data as possible was collected and classified to assure that an accurate understanding of the participant experience is being documented.

This dissertation explored the experiences of computer science students as they developed computer games. The phenomenological method lends itself well to this task. The method allows participants to describe their experiences as they go through the process of computer game development. The questions that were developed for use in this project have been created in a manner that reflects phenomenological research: parsing and evaluating data that will ultimately support the goal of this study.

In general, phenomenological studies carefully look at the characteristics behind change and bring to light any common thread or themes that are observed by the researcher. In this case, one of the goals was to observe and evaluate the change in knowledge of each participant and how they came about that change as they employed User-Design as an educational technique. The participants of this study were asked to create something using tools and techniques they did not possess. The participants set their own path for knowledge. This path is the focus of this study and research.

Protocol/Methodology

This study aimed to capture patterns or themes about what the students are doing to acquire advanced computing knowledge. When selecting participants, the researcher employed purposeful and convenience sampling techniques (Van Manen, 1990) to better capture their lived

experiences. To achieve thick descriptions (Guba & Lincoln, 1981), a small representative number of participants was chosen for the study.

The first step was to solicit a faculty member teaching an upper level computer science class for a source to recruit students. This faculty member also had to agree to offer extra credit for participation and an alternate assignment for those who did not wish to participate yet still wanted extra credit. A faculty member teaching the systems programming course agreed to this and developed an alternate work assignment of writing a paper on a topic relating to systems programming logic.

For the next step, I went into the class and explained that I was looking for volunteers for a study. I explained that they would be creating a game from scratch and I was looking to see how they learned on their own and what resources they used to teach themselves how to program in a new language. I intentionally did not give any more details about the programming language because I did not want any participant to look into it before the study started. I also explained that they would be required to come in to the projects lab once a week for observations. In addition, they would need to log their activities when they worked on the game outside of the lab. I also explained that there would be an interview near the middle of the study and a concluding interview at the end. A sign-up sheet was passed around to put their name and e-mail address on. There were approximately thirty students in the class and about half put their name on the list. Afterwards the course instructor explained that they could earn two extra credit percentage points on their final grade if they participated in the study or completed the alternate assignment. A few more students from the class contacted me later about the study following the instructor's explanation of extra credit. After I received the list of student emails, I sent out an email to all the potential volunteers requesting times to meet as a group. After many varying

responses, I realized it would be better to meet individually with each potential participant to explain in more detail items such as what they will be asked to develop, work location and times. At the end of recruitment there were about fifteen responses of which seven committed to the required work.

The study began by meeting with the participants individually in my faculty office. During these meetings I repeated the same basic information to each participant. They were introduced to the possible tools they were to use for game development. I made sure that the participants knew they were free to use any resource available. I also made sure they did not have prior experience programming in JavaScript. Aside from their usual resources, I mentioned that they were free to consult faculty members and other students as a resource. This allowed freedom for the participants to discover without ties to a traditional classroom setting yet still have all the resources of one. Additionally, during the initial meeting, the participants were informed of what was expected of a final produced game, review of the incentive to complete it, and the details about when and where the interviews would take place. The students were provided with a thorough explanation of the process as a whole and given the chance to ask questions at this point.

I provided a consent document (Appendix A) that explained the study to each participant. This document described the purpose, procedures, and benefits of the study. After providing adequate time for the participants to read the document, they were asked if they had any questions and I left time for a discussion about the study to answer any concerns they had. The participants signed and dated the consent form. I signed and dated the form and gave a copy to the participant.

During the initial meeting with each participant there was discussion about the game to be developed. The participants were encouraged to create an educational game but were also allowed the freedom to choose their own topic. This will help motivation to work attentively on the game (Wright, 2011). I wanted some similarity in game development difficulty among the participants, as the discovery of common themes is one of the desired outcomes. If the projects varied too much in difficulty then it would be expected that their User-Design would also be significantly different (Land, 2000). I approved each student's game idea in order to ensure that it was possible to develop the game in the amount of time given and that their game was extensive enough to ensure the participants will need to learn new material and develop their own method of instruction. Approval was either done during the first meeting or they were given two days to think about it, email or visit me and I would answer with approval or required changes. I wanted to get each participant's game idea established quickly so they could start working on it immediately. None of the game ideas were too small and several were too big. I recommended cuts and the participants agreed to the changes. This was usually done in person.

I gave the expectation that the games should run without any errors and that the game should be entertaining to play or educational. In addition, the game should include some fundamental game principles such as, challenging but not impossible, easy to learn, and conducive to player decision making. Each participant was asked to create a game that requires only a web browser to play. This game development environment was chosen for its ease in making games within a short amount of time, availability of resources to learn from and the object-oriented nature of the programming required. Additionally, I asked the participants to pay special attention and make note of any resources they used and the thought process while developing their game.

After talking with the students and learning their schedules, I realized that we would have to meet on the weekends for the observations. I didn't want to take up any student's Saturday since it is common for them to go home on the weekend. For this reason, I suggested Sunday afternoons. We decided on 1:00 in the afternoon. By this time, it was two weeks after spring break with five weeks left in the semester. I had the participants start their games the first week then we meet for the next three weekends over approximately a four-hour period each meeting. The length of meetings depended on the students' demands at that time. It always ranged from two to four hours that each participant could stay. We did not meet during finals week but they had that week to complete their games. These meetings were set up only to allow me to observe them as they worked on their games to gain another perspective other than logs and interviews. They were encouraged to continue working between meetings. The participants had a total of five weeks to complete their game. During the observations, I sat at the instructor station and took notes using a text editor of any significant activities the students were doing and any resources they were using. I would do this by walking around noting the resources they were using then logging it. Occasionally, I would query them about the status of their game and what means they used to figure out a programming problem. Then, I would log their responses. I collected approximately 2,500 words of data in the observational logs.

During each initial, individual meeting, I stressed the importance of logging their work as they programmed outside of the observation meetings. I had set- up a website location using D2L[1] where they would all have access. I created an essay quiz assessment with settings of unlimited time and unlimited number of submissions. This was not used as a quiz but only a place to log their activities. The intent was they would keep answering an essay quiz question as a means to log their data. I choose this because I knew they were all familiar with using D2L. In

[1] D2L – Desire to Learn, a course management website used by Edinboro University

addition, D2L provides a means to archive logged data that can be accessed any time online. To clarify what I wanted in the logs, I wrote an extensive example of what was expected and emailed this, along with instructions, to the participants (see Appendix C for a screen shot of my sample log). Throughout the 5 weeks, I sent out periodic email reminders to log their activity in the D2L quiz.

To obtain another form of phenomenological data, two sets of interviews were conducted with each participant. The first was conducted approximately half way through the study and the second at the end. Each interview was conducted in my office. Times and dates were scheduled through email. All interviews took place during business hours at the convenience of the participants. All interviews were recorded. Using audio editing software, I edited out any part of the recordings that were clearly unrelated to the study. For example, conversations ranging from other courses or advising question to what they did that weekend. The recording times ranged from three to eight minutes with an average of six minutes per participant of relevant data for the first interview. I applied the same approach for the final interviews. The final interviews averaged ten minutes per participant. Total recording time after filtering out non-related data was 110 minutes for all recordings. I used an on-line transcribing service to transcribe the recordings to MS Word format. I randomly selected three samples of text from each transcription and compared to the audio recording to double check that the transcription was accurate.

The interviews used a set of pre-defined questions aimed to gain the participants' perspectives as they developed their games. The questions were designed using established protocols (M. Q. Patton, 2001). The questions for the mid-study interview were:

1. How would you describe your experiences as user-designers?
2. Do you feel more invested when designing your own game prototype?
3. Can you talk about an experience you had to design something?
4. What is it like to be the creator of your own system?

For the final interview, the researcher asked more open-ended questions that captured additional detail and their thoughts on being user-designers. They fall into three categories: game design, programming, and User-Design:

Game Design:

1. Did you research good game design principles?
    a. If so what resources did you use?
    b. How did you go about finding these resources?
2. Did you implement good game design principles?

Programming:

1. What were your first steps in learning how to get started with programming the game?
2. What resources did you most commonly use to learn how to program for browser-based games?
3. Is there anything you wanted to implement but could not because you could not find a resource to show you how to implement it?

User-Design:

1. If you were to design a course about browser-based game programming what major topics would you include?
2. If you were to design a course about game design (not programming games), what major topics would you include?

I occasionally asked the participants to elaborate during the interviews to collect more detail. In addition, I collected demographic data from the participants including gender, age, and school status. This was done to help describe the participants' context.

At the end of the study, all the participants sent me their final games. These are used to compare themes, methods and resources as well as the quality of the game produced. In addition, the final game was used as a reference when analyzing the interviews regarding details of the game.

Because each participant was a student in the Systems Programming course and participating in the study for extra credit in that course, the instructor requested a rating for level of participation of zero to ten. At the end of finals week, I gave a list to the instructor, containing each student's name and their rating.

All of the concluding interviews from the seven participants were conducted during the spring 2015 final exam week. All were done individually at various times within a three day span. I had the participant deliver or email me their final game by the end of finals week.

Following the collection of data, I reviewed the interviews, and began coding the data, using open and axial coding (Miles & Huberman, 1985). I was looking for distinct concepts and categories in the data. I also looked at the games to see if they had met their goal and their quality.

Unit of Analysis

The unit of analysis for this study is undergraduate students majoring in computer science at Edinboro University of Pennsylvania, a medium-sized, liberal arts university. Seven students volunteered for the study. All seven had junior status. All of the participants had the department's object-oriented-programming course. This course was significant for two reasons. One, the computer programming language in which the game was developed used this object-oriented paradigm. This study looked at how these students learned a new programming language without having to learn an entire new method of programming because of the time constraint and desire for an end product. Second, the object-oriented-programming course is one of the more challenging courses in Edinboro's computer science program. Students who have progressed far enough to take and pass this course are likely to graduate with a computer science degree (Hillman, 2010).

Background of the Participants

There were seven participants in the research study. The seven were all computer science students in their junior year with the exception of one. All but one were recruited from a network programming and scripting class. The descriptions that follow are depictions of them based on my interactions with them as their instructor and in some cases as their advisor.

At the time of the study, Edinboro University's Computer Science department had five tracks; Theory, Game, Web, Networking and Applied. The Web, Networking and Applied tracks were introduced two years before the study. ABET is an accrediting body that accredits programs in applied science, computing and engineering (Parrish, 2016). The Theory track is ABET accredited and is a traditional B.S. in computer science in that it requires considerable upper level math courses such as Calculus II and Discrete Math along with a strong upper level

CS (Computer Science) elective requirement. The Game track is based off of the Theory track and has the same level of rigor. The other three tracks are not as math or CS elective intensive. The department used to have a track titled "Applications" that was very similar to the Applied track and was replaced by it. Students who were enrolled in the Applications track can graduate under its requirements if they wish, instead of switching to the Applied track.

Each CS track at Edinboro University has a common core of courses that all students must take. These include fundamental programming theory courses and some other areas such as database and general networking. The Applied track at Edinboro University is designed to give the students the most flexibility choosing their courses after taking the core. This track has the largest block of computer science electives, allowing the students to choose their own depth or breadth of courses. This track graduates more students than the other tracks. It is common for students who are struggling with the Game or Theory track to switch to the Applied track.

The Applications track was discontinued with the creation of the Applied, Web and Networking tracks. It is most similar to the Applied track and gave the students significant flexibility in computer science elective courses. The difference between the Applied and Applications tracks is the required computer science courses. The Applications track included courses that are not offered anymore and did not require some of the others in the current common core such as *Programming in COBOL* and *Systems Analysis*. In order to make uses of all of their coursework and to avoid taking additional courses in the common core, most CS students who enrolled in the Applications track tend to stay in that track.

The Game track is designed for students who wish to pursue a career in game programming. It is considered more rigorous than the Applied or Applications tracks because of the additional upper level math and upper level computer science courses required.

The participants in this study came from three of the tracks the computer science department offered. Three students were in the Applied track, three in the Applications track and one in the Game track. The Applied and Applications tracks are considered less challenging then the Game track, mostly because of the additional math courses required in the Game track.

All but one of the participants in this study were enrolled in CSCI 310 Script and Systems Programming. Students in CSCI 310 were offered extra credit to participate in this study. Interestingly, the CSCI 310 course is not required of any track the participants of this study were taking. Only the Networking track requires this course. In order to protect the participants, I use pseudonyms in this dissertation to preserve confidentiality.

A short biography of each participant.

Tim. Tim is a junior, Computer Science major in the Applications track. He is a twenty-two year old, male, Caucasian from a small town outside of Pittsburgh, PA. Tim has taken four classes that I have taught and is also my advisee. His QPA is 2.91 which is within the average range for Edinboro Computer Science students. From my perspective as his instructor and advisor, Tim is a hard-working student who turns in quality work. He often stops by my office to ask questions about anything, even if it is not academic related.

I have not heard much feedback from other faculty members about Tim. Therefore it is hard to judge faculty perception. From brief conversations, no faculty members have reported having any negative interactions with this participant.

Tim's motivation for participating in the study is twofold. He wanted to earn extra credit in the network scripting course. He was also interested in web development tools and was therefore curious about the project. Tim seemed very committed to the study. He truly enjoys learning new lessons in Computer Science. In addition, I believed Tim's commitment came

from being his advisee and having been a student in four different courses I taught. Tim participates not only for the extra credit but feels he is doing me a personal favor.

In conversations with Tim, it appears that he would prefer a career in web development but would consider other programming or networking types of jobs. Tim is not academically active in the department other than being a member in the Computer Science Club. Tim has a small group of Computer Science students he tends to spend his spare time with socially. In addition, he is always involved in any Computer Science club social activity such as bowling competitions with the Math club.

Tammy. Tammy is a twenty-three year old, female, Caucasian from a small town with no large city close by it. Her hometown was a prosperous town for its size in the seventies and was the headquarters for three major corporations which have since relocated. Since that time, the town has suffered economically. Tammy is a junior in the Applied track Computer Science major with a QPA of 3.23. Her academic performance is above average in the Computer Science department. I have not heard much about her from other faculty members so gauging faculty perception is difficult. I have had her as a student for three different classes. Tammy is quiet compared to most students but tends to sit up front and rarely misses class. I consider her a very good student, completing her assignments and turning in good, complete projects.

She is participating in the study to earn the extra credit from the scripting class. In addition, she is interested in learning about more applications that run on the Internet such as the JavaScript required in this project. I had Tammy twice as a student and believe she enjoyed my classes. As the study progressed, I could see this participant was losing focus. In talking with this participant, it was apparent that this participant was becoming overwhelmed with other coursework.

Tammy's career objective is to find a job in web programming. She is not very active in the department with the exception of attending the Computer Science club meetings. She is quiet in class and does not interact often with other students. After class, she rarely talks with the other students before leaving.

Allen. Allen is a twenty-eight year old male, Caucasian in his junior year studying in the Applied track. He is older than most computer science majors at Edinboro. Allen is from an urban area of more than 100,000 with a faltering industrial economy. I have has known Allen most of the time he has been at Edinboro having had him as a student in two classes.

His academic performance is about average compared to the other Computer Science majors at this point in their degree. At the time of the study, he had a 2.7 QPA. The faculty perception of Allen is generally not favorable. The most common comments are that he tends to do what he feels should be done, not necessarily what the faculty member wants him to do. This has come up in discussions on several occasions.

His motivation for participation is to gain the extra credit for the scripting class in exchange for participation in the study. In addition, I have a good working relationship with Allen compared to most faculty. Allen is also very interested in game programming and this study provided a structured framework to accomplish that goal. Allen is different than most in that he seemed reluctant at first to participate in the study but, towards the end, he was very motivated to create a quality game.

His career goals are fairly simple. He would like a career as a computer programmer. Opposed to the other participants, he has mentioned that the only thing he does not like is web programming. He mentioned to me that his ideal job would be in game programming. He is

realistic, knowing that the field is competitive. He is also interested in most other areas of programming.

He is only involved in the department academically through the computer science club. He attends meetings occasionally but is less active than most members.

Charles. Of all the participants, I know Charles the least since I never had him as a student in class. Charles is a twenty year old, male, Caucasian from an urban area. His hometown is a medium sized city with a metropolitan population over 100,000. The economy there has been hit with high unemployment in the last several years.

He is a junior Computer Science major in the Virtual World and Game Development track with a QPA of 3.26. He is the only participant in this track. His academic performance is very good, with a high QPA in one of the more challenging computer science tracks.

I have not heard other faculty members mention him so I am unfamiliar with the faculty perception of him.

I believe Charles's only motivation to participate was out of curiosity and because one of his closest friends, Allen, had agreed to participate. When Charles heard about the study, he approached me to ask if he could participate. This participant always seemed eager to build a quality game and it appeared he enjoyed the time spent working on it.

His career goals are to become a game programmer. He is one of the few students who enjoy playing games and also has the academic ability to become a game programmer. He is aware that jobs in game programming are competitive; and he is also interested in pursuing other programming careers.

Charles has worked with a faculty member through undergraduate research and presented a poster on his work at a regional conference. He has a small group of close friends that are computer science majors that he seems to spend most of his time with socially.

Chris. Chris is a twenty-one year old, male, Caucasian from a rural area. The closest town has a population of approximately 1800 residents. The closest town above 50,000 is a three hour drive away. Chris is a junior Computer Science Major in the Applied Computing track.

He has a QPA of 2.84 and a very strong student though his QPA is below a 3.0. I have had Chris as a student in two different classes and he is one of my advisees. All of my experiences with him have been very good. Chris is a hard working student who gets very interested in his projects. He often goes beyond the expectation of the assignments once he gets started. I have not heard anything regarding Chris from the other faculty. This could be considered good since there have not been any complaints.

His motivation for participation in the study was mostly driven from the desire for bonus points in the scripting class. He is also interested in learning all he can about web development and since the project for this study was to write a web based scripting game using the Java Script, this participant was motivated to participate. This participant seemed to maintain the same level of interest throughout the study. He may have not been overly motivated, but he always showed up and gave it his full attention while working on it.

He would like to get a job in the area of computer science in the Harrisburg area. He is not picky about the type of job and is looking into working with a relative at the Pennsylvania Department of Transportation. He is drawn to web programming; but he would work in another area of computer science if a good opportunity opened up.

He is not very active in the department academically. I have not heard any comments good or bad from other faculty members about his academic work. In addition, I do not know of any research projects or conferences Chris has attended with faculty or students. Socially, he appears to have a few close friends but does not participate in the Computer Science Club activities. He does get along with the other computer science students and interacted well with the other students while working on the study's project.

Todd. Todd is a twenty-three year old, male, Caucasian from a small town about forty miles outside of Pittsburgh PA. His hometown is a rural area that has seen much of its manufacturing plants close and has seen sharp declines in population and employment.

He is a junior Computer Science major in the Applications track. I had Todd as a student in class on two different occasions. My impression is that Todd is very intelligent and a hard worker if he's interested in the assignment. More than most students, he can lose motivation if the project doesn't interest him. His academic reputation among the faculty is good, although his QPA of 2.88 does not reflect that opinion.

His motivation to participate in the study was twofold. He wanted the extra credit in the scripting class and since I knew him better than most students, I personally asked him to participate. So part of his motivation may have been to do a favor for me. Todd maintained a steady level of interest in creating the game and ended up with one of the better games of the group. This participant stood out more than others in that he performed more background research than any of the others. This participant enjoyed sharing his findings with others in the group.

His career goals are similar to most Computer Science majors, to find a programming job right after graduation. He is not particular on what type of programming he would like to do as some students are who want to hold out for positions such as game or web programming.

Typically students get involved in the department by participating in research projects and traveling to conferences with faculty. Todd did not do any research projects but did attend some of the regional conferences with faculty and other students. Socially, Todd had a group of computer science students that he regularly hung out with who are at about the same academic level. He frequently attended the computer science club meetings and any social events they had such as bowling nights.

Susan. Susan~~,~~ is a twenty-eight year old, Caucasian female. She is from an urban area just outside of Pittsburgh PA, an old steel mill town where the economy suffered for many years. She is a junior studying Computer Science in the Applications track. She has a QPA of 3.66, the highest of the participants.

She is different than most Computer Science students in that she came to Edinboro already having an undergraduate degree. Her first degree is in Art History from a private university in Philadelphia. She looks at Art History as more of a passion than a career. She spent some time looking for a job in that field but quickly realized it was overly competitive. After talking with her, it seems she had more interest in Computer Science than other marketable skills and that is why she choose this field. Though she is from the Pittsburgh area, she chose Edinboro because she has family that lives nearby and provided a low cost place to live while she was going back to school.

She has an excellent reputation among the Computer Science faculty. She is known as a hard worker, producing good programs and turning in assignments on time. In addition Susan is

very respectful to all the faculty. I had her for one class and agree with the other faculty on her reputation.

Her motivation to participate in the study was mostly to earn some bonus points in the Scripting class. She commented that she did not like the material covered in that class and was beginning to struggle. Also, Susan is very interested in all aspects of web development and looked at this as an opportunity to learn a new part of web programming.

Although Susan took only one web programming course, it appears that web development is the area of computer science in which she would like to focus her career. Susan's internship required mostly web development. She also organized several speakers for the computer science club; and they were all focused on web development.

She is quieter than most students but still social with the rest of the Computer Science students. She held offices in the computer science club, and worked hard at organizing events for it. She focused on bringing in speakers to the Friday meetings. In addition to the CS Club, Susan also was involved in computing events in Erie PA through her contacts as an intern. She actively recruited students and faculty to attend presentation by the local programming groups. More than most, it seems that Susan is very involved and enjoys the field of computer science.

The study drew students from *CSCI 310 Script and System Programming* course, commonly called the scripting class.

Table 1

Participant Distribution of Characteristics

Characteristic	Placement		
Academic Performance	Lower – 2.0-2.7 Allen	Moderate 2.8-3.1 Tim Chris Todd	Higher 3.1 + Tammy Charles Susan
Hometown	Rural Chris	Small Town Tim Tammy Todd	Urban Allen Charles Susan
Motivation for Participation	Extra Points All participants	Web Skills Tim Tammy Chris Susan	Game Interest Allen Charles
Engagement toward Study	Low Tammy	Medium Chris Todd Susan	High Tim Allen Charles
Department Involvement	Minimal Tammy Allen	Club Tim Chris Todd	Club + Other Charles Susan
Computer Science Track	Applied Tammy Allen Chris	Applications Tim Todd Susan	Game Charles
Career Goal & Area of Interest	Web Programming Tammy Chris Susan	General/Not Specific Tim Todd	Game Allen Charles

I accepted any student who volunteered. I was fortunate that there turned out to be a good distribution of traits in the group of students who decided to participate. Table 1outlines the different traits among the participants. These include; academic performance, type of area

they are from, motivation for participation, how engaged in the study they were, how involved they are in the computer science department, what computer science track they are studying, and their area of interest in computer science.

The table and descriptions above show the distribution of student types among the participants regarding the categories listed. Academically, most fit into either highly or moderately successful. There is a good split between urban and small town students. Also participants had varied reasons for involvement in the study. The table also shows a split of how engaged the students were during the study. There is an almost even distribution of level of involvement in the department. One weak distribution is that most of the students are either from the Applications or Applied track, with one Game track student. Note that there are five tracks available in the department. There is also an even three-way splitting of the students' area of interest. As expressed, the distribution of participations is wide with one exception. This will help show that there is a saturation of data relative to student types. Consequently, college students at or near their junior year are good candidates for this dissertation.

Researcher Relationships and Power

The relationship between the researcher and the study's participants needs to be taken into account. The participant may look at the researcher as a person of power and this perspective may skew the responses during an interview (Kvale, 1996). Bourdeau (2000) mentioned in her article four principles that should be taken into account as it relates to the researcher and participant relationships. The first principle is that the researcher must have respect for, and make allowances to create an atmosphere of independence for the participant apart from the researcher. The second point addressed by Bourdeau (2000) is that there can be no threat of harm to the participant. The third and fourth issues indicate that there should be

benefit to both the researcher and its participants. These must be addressed in order to facilitate a successful study.

In this study, I was aware that these students look to me as an authority figure because of my position as a faculty member in the department. Five of the seven participants were previous students in an introductory web development course taught by me. Three of the seven had been students more than once in other courses taught by me, such as *Advanced Web Development* and *Visual Basic Programming*. Two of the participants are my advisees. I have known them since their freshman year. Only one had not been my student or advisee. This will be helpful in that these students are already comfortable with me and the fact that they volunteered for this study indicates there is no animosity between the participants and me. As mentioned in the limitations section, these students are not earning a lot of extra credit to participate in this study. I felt fortunate that these seven volunteered for the study. Perhaps most of them volunteered as a favor to me in addition to the extra credit. My courses have never had issues getting students to sign up and from what I have perceived my reputation among the students is one of fairness and I provide a relaxed atmosphere in the classroom.

I learned how to program from traditional methods of instruction. The instructor lectured on the aspects of a programming language then gave an assignment to work on. Often I would struggle, then ask the instructor for some guidance. The lectures were straight to the point and covered the necessary material. I wonder if this type of instruction is not optimal and if there is a better way. I tend to teach in this same manner with the exception I give more examples and I prefer hands-on activities to do with the students in class. As an instructor, my general observation of my students' weaknesses is that they are not good at fixing glitches and problem solving. I want to keep these biases in mind as I observe and interview the participants of this

study. Overall the intent is to create a relationship of openness during the study where the participants will feel free to explore issues as they wish during their game development and honestly answer the questions regarding their experiences.

The Site

Edinboro University is a public liberal arts university located in northwestern Pennsylvania approximately twenty miles south of Erie, PA. It is one of fourteen state schools in the Pennsylvania State System of Higher Education. Edinboro has a total enrollment of approximately 7000 students ("Edinboro University Fact Book," 2013b). Edinboro University is described by U.S. News and world reports as a Regional University (North). This category has a full range of undergraduate programs, some master's, and a few doctoral programs ("Edinboro University," 2012). Edinboro's student body is generally from rural areas with a significant amount of students being first generation college students.

The Computer Science department had approximately 200 majors during the time of the study, which is typical ("Edinboro University Fact Book," 2013b). The department has about a 50% attrition rate after the second year ("Edinboro University Fact Book," 2013b). The department is housed entirely on the top floor of Ross Hall on the Edinboro main campus. There are two computer labs designated for Computer Science majors only on this floor. One is a general purpose lab where students spend much of their time working on homework assignments or socializing between classes. The second is a projects lab with computers stations, instructor station and space to store any equipment needed for projects. The purpose of this lab is for students to have a place to work on big projects such as an independent study for a faculty member. This second lab is usually quieter and not used as much as the first lab. This second lab was chosen to conduct the observations because of its quiet atmosphere and low usage. The

layout is more open with computer stations along two walls, white boards and closets on the other walls. There is a table in the middle with no computers on it.

Rational for Choosing the Site

The main reason for choosing Edinboro University is that it houses the type of participants that are ideal for this study. This study's objective is to explore how college students from a medium-sized, liberal arts university apply User-Design when being asked to learn advanced computer skills required to create a computer game. Edinboro is a good resource for those types of participants. This study is not looking at uniformity in participants in terms of age, gender, or race; they only need to be currently enrolled computer science majors in an institution of this type. Some generalizations can be made of participants from this site. For example, the participants have higher than average motivation for learning, they are career oriented, and consider higher education important and a means to further advance or start their career (Clark & Schroth, 2010).

In addition, Edinboro University provides resources such as Internet access, libraries, and other students and faculty that the participants may turn to as a resource to acquire knowledge. This study wants to discover and further analyze available resource utilization.

The second reason why I chose this site was out of convenience. In a phenomenological study, the researcher needs to observe and interview the participants on a regular basis. Selecting students from Edinboro University provided me with easy and frequent access to them as I am a professor employed in the Computer Science department at the university.

Data Collection

The data was collected through interviews, observations and participant logs. The interviews were recorded for analysis and review. Recommendations given by Creswell (2009)

and Bernard (2011) were followed. Specifically, Creswell suggested that interviews are of heightened importance when capturing anecdotal evidence of the student discoveries. The use of semi-structured interviews (Bernard, 2011) allowed for the documentation of rich dialogue to reflect best the experiences of the students. There was one mid-study interview followed by a final, more comprehensive, interview which was led by pre-determined questions and topics which were covered during conversation. While a predetermined set of questions was used during the inquiry, due to the nature of qualitative studies, further questions arose. In addition, reactive direct observation with continuous monitoring (Bernard, 2011) was used as the observational approach, where I was fully engaged with the participants as they created their games, but acted as a neutral third party. I was in the room walking around and interacting with the participants but not participating in game development. This was done on three different occasions to gain another qualitative perspective.

After a period of five weeks, the game prototype artifacts were collected in order to triangulate the study. Member checks, where the participants are asked to restate or clarify their position (Guba & Lincoln, 1981), and thick description (Denzin & Lincoln, 2011) where a detailed account was kept during observations and putting them in context, were the methods employed to reach trustworthiness. The data was analyzed by constant comparison and by moving from meaning units to themes as recommended by Graneheim and Lundman (2004).

Observations

Gathering observational data can contribute to a more in-depth understanding about the participant's decisions (Bernard, 2011). The researcher used Bogdan & Biklen's (1997) recommendation for conducting observations. I kept observer comments, recording my thoughts about what the participants were doing. In addition, I planned data-collection sessions in-light of

what I found in previous sessions. I recorded verbal as well as non-verbal behaviors. Though I was conducting official observations, all but one participant knew me as an instructor and advisor. All of them became comfortable with my presence as they continued to work on their games. The objective with the observations is to complement data by collecting additional data that I did not discover through the interview process (M. Patton, 1990).

Interviewing

I conducted interviews to supplement the observations and collecting of the finished games to research. Creswell (1998) mentions that conducting interviews is necessary in phenomenological studies. I captured the participants' experiences by recording our conversations based around semi-structured questions (Bernard, 2011). The questions are listed in the following section.

I interviewed each participant twice, a mid-study interview and a final interview. Each interview was conducted individually in my faculty office. The interviews were recorded for later analysis. The first round of interview lasted from ten to fifteen minutes in length each. The final interviews typically lasted about twenty minutes per participant. In contrast to Almeida's (2008) study, of which this study is a follow up, it was easy to get the participants to talk about their experiences. The participants were willing and eager to discuss their methods for creating games.

Data Analysis Process and Procedures

The researcher chose to use Colazzi's (1978) approach to data analysis. This model is appropriate for discovering patterns and themes in phenomenological studies that collect data via interviews and observations. The steps are: Read though the transcripts at least twice to gain an understanding of each participant's background and experiences. Second, identify significant

statements which relate to the questions for this study. Third, develop interpretive meanings for each of the significant statements. Forth, group the meanings into clusters which allow themes to emerge. During this step the researcher seeks validation in addition to avoiding repetitive themes. Fifth, the researcher integrates the themes into a comprehensive description while referring back to the protocols that validated them. Next, produce concise statements about the themes.

Additionally, the researcher used Hycner's (1985) approach of considering the data chronologically to determine any commonalities. This started the process of bracketing the data and reducing the data to its meaningful parts.

As described in this chapter, the games were developed over a five week period from April 12th to May 8th of 2015 with two weeks prior used to recruit students and a week following for the final interviews. The resulting data was analyzed over the following summer. The results of this study will be described in detail in Chapter 4.

CHAPTER 4

RESULTS

Introduction

In this chapter, I will discuss how the participants in this dissertation study came to choose the games that they developed, the progression of their game design, and a discussion of this dissertation's outcomes. The outcomes will be the themes that developed from the observations and interviews the researcher conducted with the participating college computer science majors in this study as they created their games.

Content Selection

From my observations and interviews, it appears that the participants' method of choosing what type of game to develop was based on their experiences with their own game play and the level of difficulty they perceived in each of the game options. At the start of the study, I instructed the students to be careful with game selection because of the limited time-frame to develop the game. I noticed that the students searched the Internet for web based games. They would look at many samples and then tended to choose to create a game similar to one that already existed but with a new twist added to make the game their own. For example, Susan used the web sites HTML5 Games[2] and HTML5 Rocks[3] to get ideas for games. Todd used a web page that offered free online books to learn about JavaScript syntax; and, Tammy made extensive use of the W3C School's web site[4] for its tutorials. After discussions with the participants, I believe they chose to create similar games to ones that already existed because they knew it would be possible to create that type of game. They did not overtly state this; but they alluded that seeing a similar game existed meant that it was possible to create. In addition,

[2] http://html5games.com/
[3] http://www.html5rocks.com/en/
[4] http://www.w3schools.com/

it appeared that the participants were not motivated to be creative. Choosing to revise an existing game idea relieved them of the added step of creating something new.

Process of Design

My first observation was a week after the participants had a chance to start their games. Most of the students mentioned that they went through several ideas, even starting a few, before eventually settling on the one they ended up creating. Tammy was the exception. She changed her mind several times and did not settle on a game idea until the first observation meeting. Susan started with a plan to create a game involving a treasure hunt. She described several features to me that she was excited to implement. In the end, she had narrowed the game down to a much simpler version. In her game, the player would catch only one item and use a timer to see how fast the player could do it. When talking to Tim, he had an idea of implementing a Stratego-like game[5]. When I was talking to him, I realized that his game idea would take far more time than he had available. I did not mention the time constraint to see what he would do. In the end, he decided to build a modified version of a pong game, something he could complete during the time available for the study.

I noticed that four of the seven participants chose to develop a game by enhancing an existing idea, starting with skeleton code they found on-line, then repeatedly tweaking it to fit their design idea. Opposite of what Almeida (2008) found with high school students, my participants seemed apprehensive and questioned their ability to finish the game at first. At the end, all had lost that apprehension and felt comfortable using Java Script. When the study started, there was a lot of interaction between the participants. But as time when on and their games started to take shape, there was less conversation between the students and they asked one another fewer questions. Once started, all of the participants had a similar routine. First,

[5] Stratego is a strategy board game where 2 players battle to capture the other team's flag.

determine a task, look something up on-line, implement what they found, and test. If it did not work, repeat the process. Or if it worked, they would move on to the next task. Their method of design was, what I would describe as, a continual process of a mini rapid prototyping.

Some sample games:

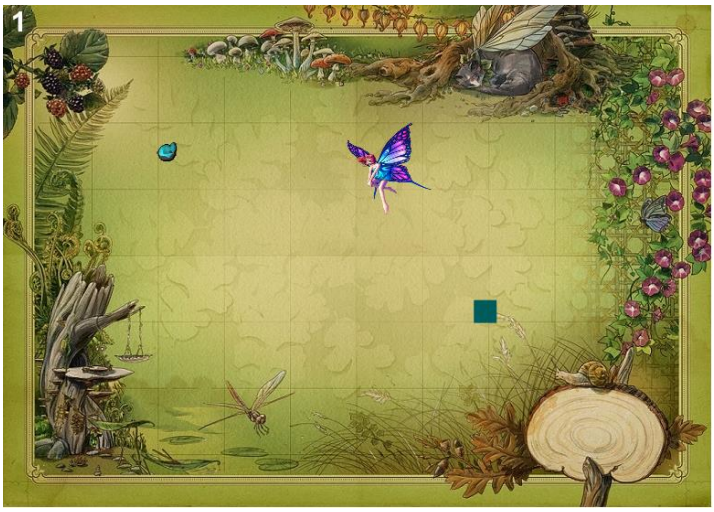

Figure 1. Susan's Game, Capture. This figure illustrates the graphical complexity of Susan's game.

In Susan's game (see Figure 1) the user manuvers the fairy to capture the gem that is on the screen. Timing is imporant and you must avoid the obstacle, in this case the dark green square.

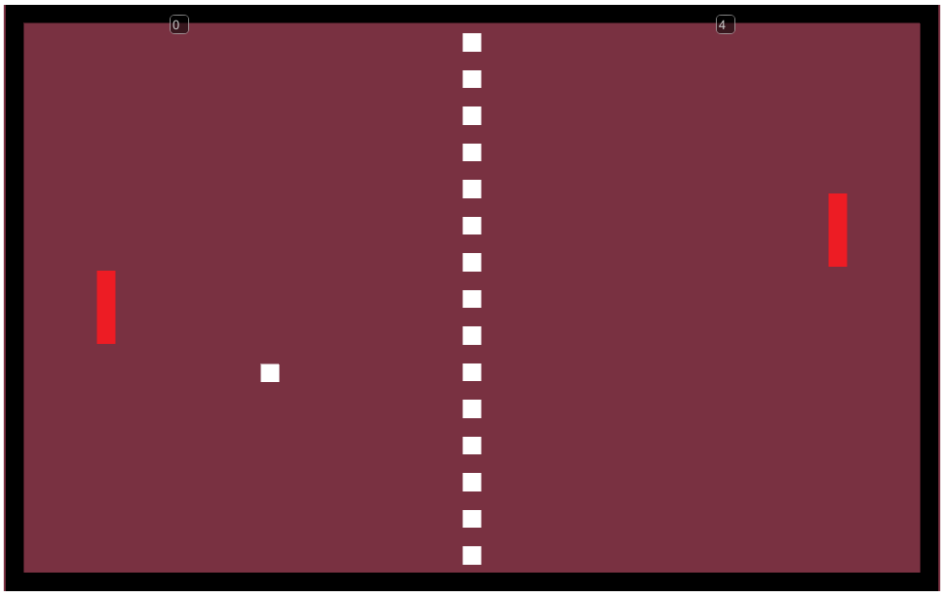

Figure 2. Tim's Pong Game. Tim made an advanced pong game. He started from a simple well know idea then modified it.

Above is Tim's game (see Figure 2). He duplicated a pong game then added some artificial intelligence to play the computer. At first, Tim had a bigger project in mind but settled on this when he realized the complexity of the original idea.

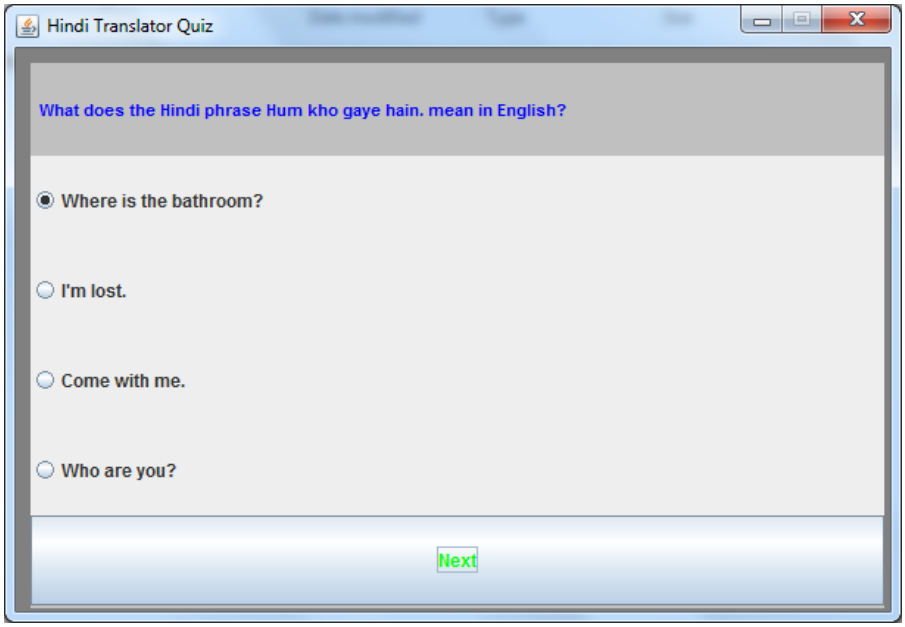

Figure 3. Chris's Quiz Game. Chris resorted to making a simple trivia game.

Chris wanted to make an educational game so he decided on a trivia game that tests the users on the Hindi language. Above is a screen shot of that game (see Figure 3).

Themes

This study had five themes. The themes were: 1) User-Design provided genuine enthusiasm for learning. 2) User-Design encouraged focused problem solving. 3) User-Design encouraged the learner to emphasize on visual outcomes. 4) User-Design had two phases, participatory then individual 5) User-Design led to a lack of structure some students require.

Theme #1: User-Design Provided Genuine Enthusiasm for Learning.

This theme stood out more than any others starting with the initial meeting with each participant. During the individual meetings to introduce the participants to the study, I explained that they would be choosing their own game to develop and this would be the process in which they learned the programming language JavaScript. All of the participants seemed very interested and eager to start the project. I think meeting with them individually provided more of a realistic impression on their enthusiasm, avoiding group think (Turner & Pratkanis, 1998). Each student immediately started suggesting game ideas; and, I could tell they were eager to start. I felt as if they had been waiting for an excuse to create a game of their own. The idea of creating their own project, truly their own creation, was appealing when compared to a traditional class assignment that was prescribed to them and over which they had no control. Their only restriction was they had to learn JavaScript.

During the interviews, several of the participants reflected on their experiences that related to this theme. All participants were in agreement that they felt more vested in the project and the learning of a new language than they usually were with a traditional class project. This

was not surprising since Almeida (2008) came to a similar conclusion. Chris provided a revealing description of how he looked at his project.

> *Yes. It becomes your own ... it becomes my own project. It's something to call my own. I can attach my name to it and say, "That's mine, I did that," and take full credit and I don't have to say [it was someone else's assignment] (Chris, interview #1).*

Chris continued to elaborate with an additional comment that is very relevant, the idea that a student truly enjoys and gets gratification from taking credit for his creations. Such as, "I like to take credit for things that I do. What's the gratification in that if you don't?" (Chris, interview #1). In addition, Chris went on to reflect more on how he is in charge and the power he has over the creation of his game. Interesting here is how he sees it as a living and growing entity, something you nurture and help grow.

> *Basically it's like having a child. It becomes my baby. You have to find the time to care for it and love it and work on it and you invest yourself in it and you basically just develop it into what you want it to be and overtime [it develops] ... (Chris, interview #1).*

Allen gave a similar account to Chris's and emphasized that since he was the originator, he wants his project to be excellent. He feels it will reflect on his personality since the game was his idea and his creation. In an interview he says, "Yes. Overall, I'm a very, one of those, [types of people] if you say you're going to do something, do it and take pride in what you do, so do it right." (Allen, interview #1) Below it is an example Allen gave that points to his enthusiasm for creating his own game. For me personally, this is very telling. It is the idea that a student wants to work on an assignment beyond what is required.

I have a personal enjoyment in it, to continue working on it even outside of this project, because I thought of putting it out on my web page, just to say, "Look what I did." [Yes] I really enjoyed it. I surprised myself. Never done it before (Allen, interview #1).

In our interview, we were talking (above) about how he was excited to complete a game that he designed from scratch. He states that he is proud enough to post it on his personal web page. Note that Allen uses his personal web page as kind of an on-line resume. As a computer student, it is a place to show off his work for potential employers and colleagues.

Another participant, Susan, agrees with the previous students in that she feels more vested in the outcome of the project because she was the originator of it. In the excerpt below, Susan is relating her work on the project to how she feels about creating web pages. I had Susan in class for web design and she enjoyed creating web pages. For example, her homework page is a page I had the students create for me to be able to access their exercises. For most students this is a very simple, easy to create page. Susan went far beyond the requirement applying advanced web programming techniques simply because she enjoyed learning that type of material.

In general I'd say I usually feel more invested in working on things that I'm personally motivated to create. I guess for website design, it's what I'm most interested in. When it's like an open ended thing, like just make a website about whatever you want, I'm definitely more interested and more invested in that than a specific, like make java jam (Susan, interview #1).

Above, Susan is referring to a semester-long project in the web design class called *Java Jam*. This is a prescribed assignment from the book that is well defined with clear end goals. She is mentioning how she much prefers working on something that is more open-ended that she can make her own.

In the following excerpt, Tammy brings up an interesting point that not only does she create her own game but that she is also free to choose the tools needed to create the game. Her comment about the curriculum at Edinboro is telling. She is directly mentioning how she can choose her own curriculum in full support of this theme that students are enthused with making their own choices.

> *I do like it. I like being able to choose what I'm going to do and the system that I'm going to use, the tools that I'm going to use, mostly because I think I can go for some more of the modern stuff that isn't really involved in the curriculum here, but I do always like having a deadline, so I like that that lends some pose on me from someone else (Susan, interview #1).*

She further explains her preferred method of learning. She likes to guide her own project development and have the freedom to try different things. In this example, she's talking about the actual process of programming the game. How she can work on one aspect, write some code for it, see what happens, then it is up to her whether or not she approves of the result. It is strong evidence that user-design is empowering and that she likes that aspect of it.

> *Whereas, when it's my own, when it's like, "Okay, I want to kind of do this, but maybe I want to do a little bit of this; let's see what I can do with this." ..." It's more interesting. (Tammy, interview #1).*

In the following quote, Tammy also briefly explains that she may reject an instructor's assignment. Knowing Tammy, I doubt she would actually refuse, as stated below, but clearly she is saying she doesn't like to be forced to do something that is of little interest to her. She would much rather come up with the assignment herself. She says, "If someone told me to do

something, I completely didn't want to, then I would be like no. I don't want to do this" (Tammy, interview #1).

Charles, the only game track participant of the group, has the most experience with developing games. He also remarks that he feels more vested in his project. His use of the word "chore" is very powerful. It speaks to his lack of enthusiasm for assignments that he did not design especially if he lacks interest in them. He was much more satisfied developing this idea than if he had had a prescribed assignment.

> *Yeah, I feel way more invested. It doesn't feel like as much of a chore. If something's required, and you have deadlines, it feels more like stress that's added onto things you have to do. In the back of your mind, you're thinking, "I've got to finish this. I've got to do this". Me personally, I don't like the feeling of something in the back of my mind, sitting there, just saying, "I've got to do this eventually". I like being able to work on something that's my own (Charles, interview #1).*

Tim continued with a similar remark. In his first comments, when asked about designing his own game as a project, mention "fun." It is a clear indication of his enthusiasm for the project. In our conversation, he related that he would get excited to create and work on his project.

> *Yeah. I had a lot of fun writing the game. When I decided, "Okay, I'm going to dedicate some time editing my game." I'm like, "All right, let's do this." I always felt optimistic about what I was going to do. It was definitely good timing. Yeah, I would definitely say I felt invested in what I was doing (Tim, interview #1).*

I have had Todd as a student on two occasions; and he was much more of a procrastinator than most students. I especially valued this comment from him. He addressed his self-

motivation problem. Possibly User-Design helped him with motivation to keep working on the game to see what he could do.

> *I like being able to work on my own stuff, so that way I can design whatever I want to do. It goes back to the whole self-motivation problem. If it's something they gave me, I have a hard time willing myself to do it in a timely manner. It's just low on the priority list. It's like, "Well, something I want to do." I'll put it below the stuff I have to do for other things (Todd, interview #1).*

Later Todd comments that he is more invested in his project since it is his own, similar to the other students' comments. In his second comment below, he describes that it became fun to work on, but hints that not explicitly having an assignment may have caused him to lose focus. Then again, in the third quote, Todd mentions his motivation problem and how he has designed his learning to squeeze into his schedule because he is enthusiastic about it.

> *I am more invested in it, in the product itself, what I'm working on, but I'm not exactly invested a whole lot in getting it done any time soon if it's just for myself (Todd, interview #1).*

Similar to Tim, Todd also mentions about how fun it was to work on a project that he had designed as stated in his second interview.

> *Overall, it was fairly fun. Not having any structured goals is both a blessing and a curse, because I have nothing I'm really aiming to get finished, but I'm learning the system as it is. Right now, I'm fairly confident with understanding JavaScript now. My product, the game I made, I don't know if I'd actually be proud to show that, but I learned a lot using it (Todd, interview #2).*

And again here, Todd talks about how much he enjoys working on his own design and how it helps with time management of the project.

> *Overall, I liked it. I liked having my own initiative to do my work on my own time when I felt, "I'm ready to do it now, so I'll just go ahead and do this in the twenty minutes that I have free" (Todd, interview #1).*

Allen is the one participant that most faculty did not favor. He often wanted to do things in class the way he thought they should be done and not necessarily the way the instructor assigned. Some faculty members commented that it was a struggle to have him in class. I was especially curious to see how User-Design would work for him. I suspected it would work very well since he was designing his assignment via creating a game. This comment supports the idea that he enjoyed this project more than his typical class assignment. "Learning at my own pace. It's a freedom you rarely get. I really enjoyed it." (Allen, interview #1)

Tim's work log notes how much he enjoyed the project. He says that he is glad he had an excuse to motivate himself to learn something he would have otherwise not spent time doing. Note that Unity is a software package used to help write JavaScript and is specifically designed to help with game development, commonly referred to as a game engine.

> *This has been quite a project and I really hope that this helps you with our work. I'm glad I had an excuse to start learning Unity, although a bit tricky to learn, I think it's something that is very useful and there is a lot of potential in this engine. If I hadn't already been so invested with the application track, I'd honestly consider switching to game track (Tim work logs).*

In my observations, I noted "[It] helps to let students choose game type for motivation." I could see that the participants were more invested and were having more fun compared to what

I've experienced in the past when giving students programming assignments. In addition, I noted from my observation logs "[the] group seems to be enjoying making the game." It was a clear indication that User-Design promoted enthusiasm. During the observations, I also noted "[they] would talk with me and show me some interesting things they found". When I would walk around or as I was sitting at the instructor station, it was common for a student to get excited about a resource they discovered. They would often share that source with me and the rest of the participants. I could tell that User-Design was creating an atmosphere of enjoyment and a desire to want to share information. It was apparent that User-Design created enthusiasm in the students to start their projects and see what they could do. It was satisfying to see them eager to learn. In the same way, User-Design also provided other behaviors as mentioned in Theme #2.

Theme #2: User-Design Encouraged Focused Problem Solving.

During my observations, one of the key themes I noticed was that all of the participants were eager to have something they could see on their screens. This may be partially due to the assignment being a game which by its nature is very graphical. Given it was a game, it still struck me as unusual that the main focus was on the look on the screen. The alternative would be to read or watch tutorials on how to program using JavaScript and building some typical 1st programming applications. Instead, all of the participants went immediately for the end result. It appeared that once they had something on the screen, even if it didn't work, they would go back and tweak it to make something happen. None of the participants started by reading about how to program and implementing the basic constructs as they would typically do in a programming course (Robins, Rountree, & Rountree, 2003).

The participants searched the Internet for ideas for games. Once the participants got an idea of what they wanted to create, they then immediately searched on-line for code they could make minor changes to in order to create a running mockup of what they envisioned. The concept of starting with a mockup and working from it to develop a product in Software Engineering is call rapid prototyping (Pressman, 2005). My observations noted that they were doing a smaller version of that. I referred to it as "mini rapid prototyping." Mini because the project is on a small scale and it is very easy to generate another revision of the game compared to a full software engineering project. But interestingly, they were not just modifying the project incrementally then testing the small change. They wanted to run the entire game again, even if the game was not playable. In general, I observed that the participants made small changes to their projects, something they could see on the screen, and then continued to use this approach to move it in the direction of what they envisioned.

I also believe their vision was continually changing. For example, they would add features that made the game more fun to play. This is where most of the "harder" learning took place. I would consider harder learning being advanced programming that they had not been exposed to in their coursework. They used new methods of interacting with the graphics of the computer (Duckett, 2014). I observed that they spent more time with this than any other part of the development. Development parts typically include Specification (the assignment), Design, Coding, Refine, and Test & Debug (Pressman, 2005). My observations noted that it looks like their approach was to get something going quickly then continually modify it.

In the following quote, Susan is referring to the design portion of the software engineering process. In this step, software developers do not program, they gather requirements, talk to end users and run through use cases with the end goal of developing a design for the

software. It is generally thought of as the most time consuming part of software development. Having taught computer students for years, I know firsthand that students tend to minimalize or totally avoid this step when completing programming assignments. Susan's quote supports my observation of their focused approach for the end product which removes much of the design process. This is not necessarily considered poor software development, just a different approach. I observed that she and the other participants did the design as they developed.

> *In terms of what I actually do, I don't do as much design as ... I don't spend a lot of time working on design but once I'm working and actually doing the implementation I wish I would've spent more time on the design, so I think I do value it (Susan, interview #1).*

Susan's comments below seem to confirm this theme. Her comment in the following quote points to the ability to build in small increments and in any area of the project. "It's more engaging I think overall, just because you can do whatever, wherever your interests take you, you can kind of follow that natural pathway" (Susan, interview #1).

The following quote by Susan describes a typical rapid prototyping experience. The programmer tries something, quickly checks results, makes a quick decision, and depending on the result, keeps going or changes path. Here Susan talks about being very focused but with small tasks, one at a time. She mentions that they can lead to dead ends but she is creating small prototypes quickly.

> *It can also be hard because there can be dead ends down that road. You can spend some time working on something, and realize "oh, this is not as feasible as I thought it was going to be", so you lose that time. You have to kind of back track to some kind of base point (Susan, interview #1).*

During the first interview with Tammy, she revealed that she does not like the standard approach to education with lecture followed up by a sample exercise. Based on our interviews, Tammy prefers an approach where she is free to experiment and, when she wants to, focus on parts of a project Couple this with an assignment of her choosing and it appears this is an ideal situation for learning for her. I include this quote with this theme because, again, it points to the fact that students prefer User-Design.

> *I guess I did it at different times. I did this one on Sundays at one o'clock until whenever. Normally, in a class, we have a certain set period where we have to just sit there and listen to lecture, listen to lecture, listen to lecture. Okay, now we can do it. Where when I'm doing it myself, I can do like, "I can do like, okay, here's a little piece of script, let's see what it does." "Here's a little piece, let's try this. I wonder can I put these two together, also." I know how to combine these two now; instead of here's a little piece, learn it, here's a little piece, learn it, here's a little piece, learn it. Couple of days later, take everything you learned, put it all together (Tammy, interview #1).*

Perhaps this quote below best demonstrates this theme. It was clear to me that the participants were learning at their best when given only what the final product should be and allowing them to focus and explore as they wanted. This allowed them to organize the development of the product. Below Tammy is explaining how she prefers project development that allows her to discover and work on various parts at various times. She even mentions that it may take more time; but she is willing to work the extra time in exchange for the ability to explore on her own, make her own mistakes, and learn from them. In addition, she states that having her own decision on what the project will be makes it more interesting.

I'm more interested, but I probably don't like jam it all in as much. When a teacher says, "Okay, you have to do this project." I'm like, "Okay, sit down and I'm going to do this project all right now." Whereas, when it's my own, when it's like, "Okay, I want to kind of do this, but maybe I want to do a little bit of this; let's see what I can do with this." Maybe later I'm like, "Oh well, I kind of want to change this." It's more interesting. It takes longer. It takes more time. I have to put in it, but at the same time, when I have the teacher, it's like, "Okay, learn this part. Done."

[Interviewer] Do you feel you learn better?

[Tammy]Yes.

[Interviewer] with the teacher, or looking it up yourself?

[Tammy] At myself. Definitely, definitely. I kind of zone out when teachers are up there just talking, "Okay, okay, yes use all these parts." I'm like, "Okay. What?"

(Tammy, interview #1).

I included some dialog with myself that points out that she prefers to look-up something. In this context we were talking about learning how to program. For example, instead of the instructor going over in detail how to use a feature in JavaScript, such as the canvas, she feels she learns better by being introduced to the idea then allowed to explore for herself. This allows her to focus on the things she wants to and that are important at that part of the game development.

Charles gives another example below of how the participants focus on something small, then immediately see if they like it before a full implementation is complete. During this interview we had deviated from one of the questions to talk about his game programming class and how he prefers to develop for the projects in that class. It is clear Charles's views are similar

to how the rest of the participants of this study approached design and development, by focusing on one small part, seeing how to implement it and if they liked the output. Charles reiterates this thought in the second quote below about going back and forth between implementation and design in short very focused steps.

> *Generally, I like to block out a main idea for a game, and see how it works as an actual game, by then implementing that little portion of the design. So design something small, and not fully fleshed out. See if it's fun, and if it's not fun, scrap it. If it is fun, then go back to the design, and continue from there, making more in depth descriptions of how I'd want it to be done (Charles, interview #2).*

Later, Charles comments how he likes very specific, small focus on tasks as mentioned in this quote "Design, implement, go back to design, implement, go back and forth" (Charles, interview #2).

In the following quote from Charles, we were discussing how he started working on his game. He discusses how he looked for something that existed to see how it works. The interesting part of the discussion that relates to this theme is the last part where Charles talks about tinkering. In our discussion, he is referring to the idea of trying out small, focused, programming ideas and seeing if they do what he wants.

> *Instead of just starting from scratch, I wanted to see what was already made, and learn by interacting with something that was already coded. It's easier to see how things work when you've got a mindset of how programming works in general, over the years of learning C++ here. Then applying that by looking at something else that's similar, but different, and seeing how it works, and tinkering with it, to learn how it's put together.*

Instead of just, "Here's the syntax. Here's the rules and the syntax, now try and code it, and make something work" (Charles, interview #2).

In the following excerpt, Tim explains a similar concept as the previous participants. He is trying out different strategies, testing and quickly seeing if that works. What is most interesting to me in this passage is his mention of spending five hours trying something. In computer science, I have observed on many occasions this is where much of students' learning takes place. They try and fail until success and with every failure they have learned something. I believe that the five hours Tim put in were enjoyable and did not seem like an annoyance.

There are some routes where I'd go something like, "Well, maybe this wasn't such a good idea. Let's scrap that. Let's try this. I spent five hours trying to do this and it turns out it's way harder to do than I thought it was. For time saving purposes, let's not do that" (Tim, interview #2).

Another participant, Allen, discussed with me how he likes having control over what he is learning. I include this quote for this theme because he is referring to the ability to focus on what he needs to do and learn for a particular task. When Allen is talking about pace in this quote he is referring to programming pace which is affected by how long it takes to focus on a task and learn it. He is in control of the speed, and therefore the focus of his learning.

I like it because I like having control over the pace that I'm learning and what exactly I'm learning. I'm not saying that all knowledge isn't useful, but for the context of what I was doing, I didn't need to know everything (Allen, interview #1).

From my observations, I also noticed the trend in the participants' game development I refer to as "mini-rapid prototyping" which is many short, focused problem solving sessions followed immediately with checking the results. I want to point out here that this is different

than simply writing some programming code and doing a quick test. Here the participants want to see the end product every time they make a change. I found this to be an atypical way to develop a project. This comment will also apply to the next theme, where User-Design leads the learner to focus on visual outcomes. As I noted in my observation logs, "Each has beginning focus on the graphics and getting something working quickly. Afterwards tweaking it to meet their needs." (Researcher observation notes)

I made the following observations about Tim. He used a mini-rapid prototyping method of development when he first started his game. The second note is slightly different in that he is learning the IDE, the integrated development environment. An IDE is a software package that helps programmers program by providing code hints, various tools and the means to immediately run the program in one place. It was interesting that he is using the quick focused approach to learning not just JavaScript but the IDE as well. "[Tim] Starting with a well-known game and then tweaking to make it an original game." (Researcher observation notes) And then learning the IDE he mentions the following, "[Tim] Still has mostly used the unity game engine site only and the IDE itself is open. He's using trial and error with the IDE to see if it complains." (Researcher observation notes)

Similar to the other students, Todd and Susan spared no time in getting something up on the screen so they had something immediately to see. Then, they made small focused incremental changes. From my observation "[Todd] mostly has game and code open, tweak, see result, then switch over to read about the code, maybe google another source." (Researcher observation notes)

Here Susan is having a similar experience. She is quickly focusing on one task and explaining that in her interview.

[Susan] switched background image, learned the collision detection from a short video tutorial, she just followed that and wrote the detection as she was going through the tutorial. Says it's not perfect but wants to make it better (Researcher observation notes).

As stated above in the participants' interviews, logs, and my observations, it is evident that the participants chose the development model of implementing small focused changes then immediately seeing the results. In programming, it is common to write code then immediately perform a cursory test; but this was different. They all wanted to see an end product and watch it grow as they worked on it. As mentioned earlier, I like to call this mini-rapid prototyping.

Theme #3: User-Design Leads the Learner to Emphasize on Visual Outcomes.

Another theme that stood out was the participants' drive for good visuals. Even though this project was to develop a game and the visual look is important, it still struck me as unusual. In most cases, it was as if the visual drove the entire project. Good visual did not always work out for each participant; but all were focused on it at the beginning and spent considerable time and energy in this part of the game. For example, a participant could have spent more time on strategy, physics or rule development; but all participants wanted a good look immediately. It was common for them to envision a look then work toward coding that look. Not a game idea but an image of what the game would be.

In Chris's interview when discussing design strategies, he talks about needing to see the end goal. What he is talking about is graphically, he wants to work toward a picture of what the end product should look like, not necessarily the game play aspect of his project.

I didn't say that before but it's just I found that the design lets me see the whole big picture. I always work better when I'm ... I know the end goal. It's always better to work

towards, I work better when I'm working towards a bigger end goal and I know the big picture I guess (Chris, interview #2).

Todd's experience is very similar to Chris' and coincides with this theme. Here during an interview, we were discussing how he came up with his design. He says that he immediately started looking at the visuals of other games. From there, he decided what to pursue. Other options could have been to choose something similar to what he already knew, maybe query different strategies, maybe query what JavaScript can do and go from there. Instead Todd and the others tended to go right for good visual appeal.

Oh. For a while, I didn't really have a very strong idea of what I wanted to do. I just knew I wanted to work with JavaScript in some way. I looked around and found that Quintus HTML and was like, "Okay." My thought process from that design point was mostly to understand the engine, because I am not a terribly creative fellow. I started looking at what was already made, and was like, "What if I modified and made my own version of that?" I guess, as far as the design part of it, I really liked designing it and coming up with what I wanted to do after I looked at other projects, but I enjoyed the implementation far more (Todd, interview #2).

Later in the interview, Todd and I were talking about how he would design a course on JavaScript. I found it interesting that he would have his course focus on the graphic appeal first instead of the theory of fun. He goes on to talk in detail about character development and more specifically about the look and how important it is for the player's ability to understand graphically how the characters should look. Traditionally in game programming courses, the instructor would start out with programming the environment not necessarily the visual charm.

(When asked about creating a course) Making sure that everything's graphically coherent, that the difficulty of the game that you're designing it in mind for people with the exception if you're doing a multiplayer game (Todd, interview #2).

And

The other thing is character design, making sure if you just look at the silhouette of the character, you should be able to tell what that character is (Todd, interview #2).

When interviewing Tammy about what she would include in a course about game development using JavaScript, she often mentions the visual aspects instead of programming or playability of the game. I would expect a student to recommend learning basic JavaScript elements and how to apply them to use on the screen before moving on to the details of graphics.

Theory of fun. I guess learning in general about ... What was the term he used? I think it was ambiance, like knowing what colors mean. What it means to the player when they're playing certain things that calm, like things outside of what you would think of the game. When you think of the game you think of the player moving…. The textures used, the colors used, the sounds used. I think that's a lot more important than the base design of the game in general because you can have the most awesome, fun to play, kill everything game, and then do it all in white and it won't be very fun. I think if you would be able to find your way that might actually be kind of interesting. No because then you just push W and walk into infinity. You'd never know. You'd be walking into a wall (Tammy, interview #2).

I made the note about Todd at the end of the project. Todd was eager to show me another resource he had found. Interesting, this particular game engine focused on the graphical aspect of game development. "At the end, Todd showed me another game engine that he is working

with, very graphical." (Researcher Observation Logs). In addition, I noted the following about Tammy "leaning towards brain games. Looking at graphics of games." (Researcher Observation Logs). Susan worked a similar approach. She immediately went for an image that she liked and downloaded it from the Internet. This was the basis of how she started her game development. Below are comments in the observer logs about it.

> *[Susan] she found a graphic of a map and quickly figured out the JavaScript for that. We discussed her game. And she is leaning towards making a game of the type that she likes. One of her early things was to get the graphics done.*
>
> *[Susan] got the graphic going in short amount of time (Researcher Observation Logs).*

Similarly, Allen and Charles, they started their design process by envisioning what they wanted the game to look like. This is a log entry I made when they were starting the design process. "[Allen & Charles] definitely looking at it more graphically. They want to code what they envision the look should be" (Researcher Observation Logs).

The following are entries from Chris's work log. He did not have many entries but the ones he noted describe researching how to make the graphical interface.

> *Practicing with some GUI examples from the page listed above. Needed more insight, Found some Oracle documentation.*
>
> *https://docs.oracle.com/javase/7/docs/api/java/awt/Frame.html.*
>
> *Created my own simple GUI with help from a tutorial on this page:*
>
> *https://www3.ntu.edu.sg/home/ehchua/programming/java/J4a_GUI.html*
>
> *When I got compiler errors that I couldn't figure out on my own, I just copied and pasted them into the almighty Google and usually obtained my results from stackoverflow.com.*

Some examples of errors I encountered were solved by the pages listed below (Chris work log).

Still messing around with my GUI, it's proving to be a slight pain. Did some more reading on flow layouts again.
http://www.tutorialspoint.com/swing/swing_flowlayout.htm (Chris work log).

All of the participants commented that the visual part of development was important. In some cases, it was the most important part of the project development. I do agree that when creating a game, the visual aspect is very important. Nonetheless, I include this as a theme because of my experience as an instructor. Typically, when giving a project as an assignment in which the students have no prior knowledge of the syntax of the language, I have them start with the fundamentals and we build from there. It would be well into the course before I would give an assignment that would require graphics.

This theme stood out for me as an observer on the first day. Almost all of the participants looked for JavaScript code that produced a graphic as soon as they started working on their project. I immediately noticed this and was a little surprised. I wonder if project development in User-Design creates an atmosphere of wanting to see something right away. I suspect this is true for subject material that requires large projects and has a visual part to it. The participants designed their project and designed their method of learning. Perhaps while thinking of how to do those parts they envisioned the end product and therefore started with that. In the same way, the participants showed examples in the next theme where they worked together at the beginning then transitioned to individual work.

Theme #4: User-Design has Two Phases, Participatory then Individual

As the games developed, the group moved from looking stuff up online and having many open discussions about what they are finding to using their own knowledge and working individually to add functionality to the game. The participants got more focused and talked less as time went on.

I offered a suggestion to borrow books from faculty during the initial interview when discussing possible sources for learning JavaScript and game development. I noticed that no participants used text books on JavaScript or game theory, just web sources. Below is an ordered list of usage of what I perceived during my observations and reading their work logs:

1. Forums
2. Tutorials
3. On-line books
4. Videos

From my observations as a computer science instructor I have noticed that computer science students tend to procrastinate then work diligently to get something done quickly. I cannot help but wonder how much their previous experience as a Computer Science student is shaping the way the participants are developing their game. In this case, even though there was time, it looks like they fell into that model.

In Susan's interview, she describes how she is finding sources to learn how to program in JavaScript. I include this here because as an observer I remember this. What she didn't mention is how interactive she was with the other participants in helping and asking questions of each other on sources they were using to find information.

I started googling programming a web game, programming a game in Java script, programming gaming in HTML 5. I was just kind of reading about things like various tutorial type things. A lot of things that would come up would be libraries that you could integrate into your game or game engines, that kind of stuff like that. I was on a website, I think it was called HTML 5 games maybe? Or HTML 5 rocks? I was on there and I was looking around and then what I ended up doing to actually when I decided what I wanted, because that's when I was opened ended and wasn't sure what I wanted to do. When I actually decided, I used a YouTube tutorial to get the real basic structure of the game (Susan, interview #2).

This did not only apply to Susan, I noticed this behavior in one of my logs about the whole group, "Lively getting organized. Soon all started working on their games. Working quietly most of the time." (Researcher Observation Log) Similarly, I noted the following during the first part of project development: "Lots of collaboration between the students, tend to talk to neighbor some discussion with me." I included here four log entries from me relating to the idea that the student work during early game development cycle is participatory in nature. So much interaction prompted me to make note of it.

[Todd] Introduced a web page of free books to the group we thought was funny, He is helping other find sources (Researcher Observation Log).

[Allen] seems like the concept of paired programming that is used in industry. Bouncing ideas off of each other. Everyone couldn't do that in this room would be too disturbing. But works for now and produces some interactions between the other students that I think is good (Researcher Observation Log).

[Allen]More bouncing ideas between students, Susan getting some good ideas from Charles (Researcher Observation Log).

[General]More discussion among the group with the worm game, discussion on different types of games (Researcher Observation Log).

Note my reference to paired programming. Paired programming is a common technique in industry where two programmers sit together with one computer to program code. The purpose is to produce better quality software because the programmers are continually consulting each other and checking each other's work (Williams, 2001). In the quote above, the worm game mentioned is one of the games being developed where the user steers the worm to eat food but also has to avoid obstacles.

In my later logs, I made entries that the participants shifted from being talkative and asking questions between them to using their own experiences and knowledge to solve the programming problems they were facing. The room started to get noticeably quieter and the participants appeared to be more focused with their own games. "[General] notice this time student looking up stuff less and relying on their existing programming knowledge and what they have learned to write the code." (Researcher Observation Log). Observing Tim, I noted the following: "[Tim] working on implementing the scoring still. Not using much web resources at this point. Just prior programming experience" (Researcher Observation Log).

In the following log entry, I make a similar comment about how the work has shifted from participatory to more individual. The second sentence is relevant for this age group compared to Almeida's (2008) study of high school students. My participants are more independent and tend to live on their own. I note that it is more likely, at this point of the project development, that they would prefer to work in their own environment and at various times.

[General:] seems like most of work has shifted to coding using their previous knowledge about programming. Quieter, mostly debugging at this point. Probably would do this at home using familiar environment. For the most part games have been designed and most of the way done, the semester is winding down there is only 1 more week to work on this (Researcher Observation Log).

I reiterate what I had observed previously about the move to a more independent development method. "[General:] For the most part the group is quieter than usual, fixing bugs and final implementations" (Researcher Observation Log). And another observation from when I was walking around the room about how two of the participants have shifted to using their previous experience. This could be the difference in choosing older Computer Science majors for this study. I wonder if Computer Science freshman would have relied on their previous programming experience. For example, I noted the following in my log, "[General:] 2 of the 4 here are working on getting it going on the web. Not using much resources for that, just past experience and previous course work and resources" (Researcher Observation Log). From my past experience as an instructor, freshman computer science majors would have worked independently at this point.

This theme that shows a pattern of User-Design shifting from participatory to individual was very clear to me during the final day of observation. It was interesting to see the dynamic of the room change so dramatically. Almeida (2008) and Carr (2007) mention how User-Design can be participatory. However, in this study with college aged students, there was a clear shift to independent work on learning the JavaScript language and integrating their knowledge into the game.

Theme #5: User-Design Can Lead to a Lack of Structure Some Students Require.

Although the participants enjoyed the freedom of choosing their own project and methods of learning the programming language not all were completely successful with their games. Many struggled with time management. Others struggled with not having enough guidance for their learning. For example, Chris remarks below how it was hard for him to teach himself things. When asked of his experiences designing his own learning, he eludes to the idea that he likes the organization that a traditional classroom provides. More specifically he feels that material covered in traditional instruction has been thought of and organized ahead of time to cover important subjects.

> *It's hard for me, it was always hard for me to learn, to teach myself things and ... It's always been an issue for me throughout my academic career. Picking up the pieces that a professor doesn't give me, it's kinda hard but this helped me develop my own, better process for teaching myself and helped me. I feel better prepared for if I need to teach myself things in the future. Because it's always been a problem (Chris, interview #1).*

When asked to describe her experiences as a User-Designer Susan admits it was overwhelming and she felt some direction would help. She also comments that having deadlines and milestones would be helpful. This is particularly significant from Susan because she is a non-traditional student in that she already has an undergraduate degree and is a little older than the rest of the participants. Of all the participants I would think she is the most likely to be self-directed.

> *I'd say it can be a little overwhelming, because you don't necessarily know what direction you want to take your plan of what you're going to do. Otherwise I tend to procrastinate and I don't like the feeling of knowing that I have something that I have to get done but*

it's not done yet. If I were, if it was more like a self-directed project, that feeling is stretched out over a much longer period because I tend to take much longer at things. I like having a deadline but also having the freedom to kind of know within that deadline I'm doing to try to set my own milestones that go leading up to that (Susan, interview #1).

In an interview with Tammy, I asked the following open-ended question, "... I guess I'm looking at making this game as a course. A course that you chose the topic. You chose the assignment, and you chose the resources. If you think of it like that, what is it like to be in that situation?" It is evident from the answer and dialog below that though she enjoys the freedom user-design provides she also struggles with a lack of structure in her assignment. She does not want a fully detailed programming assignment, but a little more than total freedom.

It's definitely interesting, but it's a little lost at first when you're just told, "Hey, this is your language. Find something. Make something." It's like I haven't even given it a slight bit of thought before, let me think about this. It would be better if it was like, "Okay, you're going to use JavaScript, and you're going to make a game that's going to be ... I don't know, say a quiz game." It doesn't have to say, "Oh, you're going to do a trivia game, or you're going to do a math game." Sort of this is the idea, this is what I want, this is where I'm going; instead of "Here, learn, do" (Tammy, interview #1).

[Interviewer:]So you would prefer to have a little bit of an assignment?

[Tammy:]Yes.

[Interviewer:]Okay, all right, cool.

[Tammy:]I spend more time trying to figure out what to do than I spend doing it. Well, yes. The other week I spent four or five hours just trying to figure out what I want to do, what game to do.

Yes, I already know what I'm doing, basic set down, or even just an option of, "Okay, you can do option one, option two, or option three." That way, you still got options. You can choose what you want to do, but at the same time, it's not completely, "I've never even thought about what I want to try and do in this language that I barely knew existed; and now I'm not even sure what I want to do" (Tammy, interview #1).

When asked of Charles what he thought about User-Design he said "I want to say that, it being really open ended makes it a bit more difficult, because then there's so many more options" (Charles, interview #1). It was interesting to hear Charles indicate that it was difficult. Charles is likely the brightest student in the group and the only participant studying in the Game track. I noticed a similar experience with Tammy.

During the last observation, I noticed that Tammy had switched the type of game she was developing. She was now working on a much more simple trivia game. I had mentioned this in the first interview as one of many possible options. A trivia game is relatively easy to implement compared to the other games and it is likely that there are a lot of sample JavaScript games available on the Internet. I noted the following in my log.

[Tammy]I think she designed her game so it wouldn't need much looking into as far as knowledge resources. She got off to a late start because she scrapped her initial idea because it was too big to accomplish in such a short amount of time (Researcher observation log).

Reading one of Tammy's comments in her work log, I could sense some confusion about what she should be doing and how complex the project should be. She writes "Also, I started looking into everything I would need to do the game I want to do; however, I don't think that there will be enough to type about." Susan also makes a similar comment, she is clearer with

what she is worried about. I could sense that she would like some advice on what can be done in the given amount of time. Appropriate project size for a given amount of time is something that an instructor would know from past experience with students.

> *I definitely would still like to do this. I was doing more research on my game last night though, and I'm worried that I picked something a little out of my league. I was actually going to come and talk to you about this today, but would it be ok if I change the game I'm doing at this point? (Susan's work log).*

Conclusion

It is clear that User-Design has some benefits that students enjoy and can thrive in this type of instruction. Unfortunately there is also some potential drawbacks that can make some students uncomfortable. I believe that it is evident that the students were genuinely motivated to learn something new as a result of User-Design, also that students become much focused. When students were given a large programming project, User-Design led them to aim for something visual. It was also evident that when applying User-Design in a group, the group worked cooperatively at the beginning then each student was inclined to work on their own. Some students made it clear that the openness of User-Design made them unconformable at times especially when they did not know a direction to go with their projects.

In Chapter five I will summarize my findings and answer the research questions. I will then discuss how the results of this dissertation can be used in practice and conclude with recommendation for further research.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

This study explored the experiences of seven, junior, computer science majors as they were given the freedom to design their own games while at the same time designing their own method to learn a new programming language. This study gave the students the autonomy to explore and for me, the researcher, to observe and collect data on the way the students designed their learning.

The study was performed to discover patterns computer science college students used when given the freedom to design their own learning. The results can be used to make improvements to instructional design in a computer science college curriculum. The results of this study would likely be applied at the course level possibly changing the way in which instructors design large projects typically found in computer science courses (Monge et al., 2015).

I collected data through observations, reading the participants' work logs and conducting interviews. I listened to the recordings of the interviews along with reading and rereading the interview transcripts. While reading, I compared the participants' data to each other's to see if there were matching concepts or any repeated behavior in general. Along with my observation logs and data from the participants' logs, I isolated meaning units, condensed these down to sub units, then eventually focused on five themes (Graneheim & Lundman, 2004). Including all of these, I was able to develop themes that were consistent across most, if not all, of the participants. Triangulation and member checks were used to validate the data. When completed, five themes emerged. I feel I had reached data saturation by the end of the study, noting that, at the end, I was observing the same behavior repeatedly.

The rest of this chapter will include the summary of my findings where I reiterate the five themes and give a brief synopsis of what I discovered about each of them. This will be followed up with a discussion where I answer the research questions presented in chapter one. I will then discuss how the findings of this study can be used in practice. Next, I will reflect on the study and recognize the limitations I found. Finally, I will offer recommendations for further research.

Summary of Findings

With the purpose of discovering themes of college aged computer science majors when using User-Design as an instructional method, five themes emerged. This section gives a summary of each research question, its related theme and comparisons to related studies. Some of these themes confirm the findings of previous studies while others are unique.

Answering the Research Questions

This study focused on three research questions, patterns when designing and programming a game, how students designed their instruction, and what their experiences were. Five themes emerged. The following explores how these themes helped to answer this study's research questions.

RQ-a: What Are the Patterns Found in College Students from a Medium-Sized University When Designing and Programming Their Own Computer Game?

There were clear patterns discovered during this study. These came out as themes in the observations and interviews. For this discussion, I'll focus on the design and programming topics discovered. The first and most immediate pattern was that the students enjoyed the experience and they felt empowered when given the option of choosing their own game and how to learn to program it. Secondly, I noticed that there was a very high level of interaction and sharing of ideas and resources which appeared to help everyone advance quickly with their initial

design. Thirdly I noticed that the students focused on the visual appeal of the final product more than if they were given a more traditional in-class assignment.

The first theme was, User-Design inspired genuine enthusiasm for learning. This theme was evident immediately from my observations. The participants, without exception, commented on how they felt empowered and enjoyed working on their game. This coincides with Almeida's (2008) findings from his study with gifted high school students. Another study with adults at a professional setting also found that giving them the independence to design their curriculum for a company training created an atmosphere of interest in the project (Jordan & Carr-Chellman, 2014).

When the group of participants initially got together in the lab to work on the games, the atmosphere was of one of collaboration and open involvement. Students were sharing ideas and resources they had found online. There was a lot of talk and comparison of each other's games. This supports the idea of a community of designers (B. Banathy, 1991). As the observations evolved, I began to notice less interaction and more focus on their own projects. The participants began using their previous knowledge with little research on how to program their games, supporting the fourth theme that User-Design starts out participatory then moves to individual work. This was very consistent with all the participants. Their logs also indicated they were working alone when not in the computer science department.

I noticed right away that the participants were interested in the visual outcome as stated in the third theme. When looking on-line for game ideas, their focus was on the visual appeal of the game instead of other game aspects such as game strategy, number of players, or passive play (Barab et al., 2012; Gee, 2007). This theme continued throughout the study.

RQ-b: How did the Students Design Their Instruction So That They Cultivated Advanced Programming Concepts Required for a Computer Science Class?

This research question had two overarching answers. This type of college student becomes very focused on small incremental steps when creating their game. I believe more so than if they had been given a traditional class project. When left to their own time frame and project, the college student focuses and drills down in more depth to find solutions. Secondly, they tended to search on-line for existing code that they could modify. This points to the idea that they believe they learn better from examples.

The second theme, User-Design encouraged focused problem solving, was observed following the start of the study. I noticed this during my observations. When listening to and reading the transcripts of the interviews, I also noticed their pattern of short, focused, problem solving. I found this behavior to be different than the typical project design where, at first, a more general system wide approach is used for problem solving (Pressman, 2005). I referred to this as mini-rapid prototyping where the programmers are making quick focused adjustments to the project then seeing the final result. This agrees with what Banathy (1992) observed in a similar study relating to creating a culture of design involving the end user.

RQ-c: What Are the Experiences of College Students As They Design Their Own Instruction for Developing a Computer Game?

The third question, led me to observe that User-Design is not good for all students. Many of the students had cited in their final interview that they had wished for more structure, most notably due dates and recommendations on game projects that could be completed within the given amount of time. The participants tended to have grand designs that could not be completed during the study.

The fifth theme, User-Design can lead to a lack of structure some students require, became obvious as the study was coming to a conclusion. During the final interviews a majority of the participants had commented that they did not progress as far as they would have liked. One, realizing she was not going to complete her original design, started over with a new, simpler game so she could have something completed. The idea that User-Design is difficult is reinforced by previous research (Almeida, 2008; Alison Carr-Chellman, 2007; Carr, 1996; McLaren, 1999). Observing this phenomenon, I realized that User-Design is not good for every student and that a hybrid of User-Design and instructor led instruction is something that could be researched further.

Through my observations and interviews it appeared that the participants did learn the programming language and truly enjoyed the freedom to explore their own designs. In the end, User-Design provided a capable means of instruction.

Implications for Practice

This dissertation is a direct continuation of Almeida's (2008) work on User-Design and explores how it can be used as an instructional technology. His study focused on gifted high school students. My approach was on computer science students in their junior year of college. I choose this group because they have all made it past the most challenging courses in the curriculum.

My study offers suggestive evidence for letting college students choose their own coursework projects. This gives them a feeling of empowerment, a commitment to the project, and a feeling that it is their own. This leads the student to feel proud of the final result. There is also evidence that too much autonomy can lead to frustration if the student chose a project that is too complex or a student tends to put things off, even things they take ownership in.

This study appears to support previous research that User-Design is a viable way to modify curriculum at the college level, allowing the students to have input on what and how the material is learned. The evidence also suggests that if User-Design was to be used to modify or completely design curriculum that some input is needed from a seasoned college instructor who has experience with setting milestones and can limit the scope of projects.

This study proposes that students be encouraged to collaborate at the design phase of project development. Student collaboration benefited all of the participants in that it provided a breadth of ideas to which students may have otherwise not been introduced. .

This study is a small piece of a much larger body of research where end-users are empowered to design their own systems of learning (B. Banathy, 1991). However, the study does show that there is potential for college-aged students, particularly ones that are advanced in their curriculum, to effectively use User-Design. The participants had a common goal of creating a game; and through that, they created a community that was mutually beneficial. The overarching conclusion of this investigation is that empowering advanced college students with their own instructional design is a viable alternative to traditional classroom instruction.

Study Reflections and Contributions

When I first started, I envisioned I would be looking at the participants' actual methods of instruction. For example did they use videos, sample code, tutorials, books, me, other professors, etc. as resources to learn JavaScript. Instead, the themes were quite different. They all used similar tools; but, that is not an accurate description of User-Design. The study of User-Design is about seeing if students will create, for themselves, a meaningful program of study. This is a much more open question.

I do not think User-Design would work for every discipline. But I truly believe it fits well into any curriculum where large projects are used to teach special skills. For example, courses in Computer Science, Media Production and Instructional Design may contain good candidate projects. My approach to User-Design may be different than researchers before me in that I am looking at it in a smaller scale, as a subset to a traditional college course (Almeida, 2008; B. H. Banathy, 1992; Alison Carr-Chellman, 2007).

As I was observing the participants working on their games, I could not help but wonder if User-Design worked well here because they were developing games; and college aged students tend to be interested in games more than most groups. Certainly some of the initial enthusiasm was because they were asked to program a game. But perhaps it was more than I imagined. In the recommended research section, I propose an idea as a further study to use a different final project instead of a game.

Another question I reflected on is, if I gave the students more autonomy as User-Designers and monitored their progress much less frequently, as I did as a result of observations, would they procrastinate? Would they wait until the end of the semester to start, when it would be too late to finish. This led me to consider a hybrid where the instructor could require milestone checks throughout the semester but with the student's instructional design. Tammy, for example had high hopes in the beginning and a big project idea but had to create something simple because of lack of time. I truly believe she learned a lot through her mistakes but the final project of this study was not what she had hoped to accomplish.

I wonder if the third theme, that User-Design promotes visual thinking, exists only because they were developing a game and games are naturally graphical. Of all the themes, this is the one I found most surprising. When I was observing the participants, this theme was very

obvious. During the observations, I continually noted that all of the participants were preoccupied with the visual look instead of other aspects of their game; and I did not understand why that was the case. This question could be cleared up with the proposed research discussed below about changing the topic.

There were several contributions of this dissertation. The first is that this study extended the work of testing User-Design as an instructional technology. This is different than other User-Design studies in that the subjects were junior college students majoring in Computer Science. For example, previous studies used gifted high school students (Almeida, 2008), and studies that looked at User-Design in the work place (Jordan & Carr-Chellman, 2014). In addition this study's finding support the idea that User-Design is useful to empower and motivate students in courses that use large projects for student learning. The participants especially enjoyed working on projects they designed themselves. Likewise, the findings support the idea that User-Design encourages students to collaborate in the early stages of project development. The study also suggests that the use of a hybrid of User-Design and traditional instruction for a course with large projects could be an ideal instructional method. Additionally, this study introduced the idea of mini-rapid prototyping with project development. This is a method of development the students used where they were focused on small aspects of the development. The subjects would develop to completion small parts to test if an approach was appropriate. Moreover this study considered the input from junior Computer Science students. I have taught freshman through senior Computer Science students. From my observations as a Computer Science instructor, junior computer science majors will have more useful feedback when it comes to study habits and an ability to design their own instruction then younger students such as freshman or sophomores. Typically junior Computer Science students are either in or past their most

challenging course work. This paragraph summarized that several conclusions were developed with his study than can help or give hints for the overarching goal of better instruction.

Limitation of the Study

The IRB limited the amount of reward the participants could get to a two percent bonus in a class they were taking. In this case I drew students from a computer networking scripting course. This could have a potential impact on the study because they may not put their full effort into the development of the game. The students quickly realized that compared to the other ways to earn points in the course, such as exams and homework, participation in this study was far more time and effort than earning the same credit by other means. I was pleased that seven decided to stay and complete the study. One simple solution would be to offer more credit for participation; but I also realize the reason the IRB imposed this limitation. It was to make it fair for the other members of the network scripting class if they did not want to participate or were unable to do so. On a side note, the instructor did offer an alternate assignment for the same two percent bonus. Another option would be for me to conduct the study with my own class. This was considered but the IRB would not allow it. I cannot help but wonder if I conducted this study during one of my classes how having more observation time, more participants and more importance on the end product would have affected the outcome.

Another limitation I found conducting this study was the lack of response from the work logs. The students were asked to reflect in a log immediately following working on their game. The purpose was to get any insights from the participants when they were not being observed, likely at home, working on their game. One reason for the lack of response could be the nature of the student. All of the participants were computer science majors and as such may be less likely to want to write out what they were feeling. I may have received a good response here if

the participants were from a writing oriented field. In addition, I could have expanded on the instructions to ask them to include occurrences of feelings of frustration, accomplishments or failures at various steps. I did get some good responses but overall the participation in this activity was minimal. I do not think this had much of an impact on the study because the logs I did collect generally contained references to on-line resources they were using to learn JavaScript or to get sample code. This type of information was not useful for this study. In the future, I could incentivize the log portion of the study or have them write a reflection paper to get the extra credit reward.

Recommendations for Research

An intriguing future study would be to test User-Design as a hybrid instructional technique that would include minimal guidance from an instructor. The student is given the freedom to design their own instruction but with limitations on the scope of the projects and a requirement of milestone completion.

Games are very appealing to college aged students. What happens when User-Design is applied to a course where a project in games is not appropriate? Is there still the same enthusiasm and sense of empowerment? A follow up study that applies User-Design to a course where the project was not creating a game is important. For example, instead of game development, the assignment could be database processing where the students develop an idea and implement it using a database system (Kroenke & Auer, 2016).

During the study, I felt that some of the participants were overwhelmed by being suddenly thrust into being designers of their own learning system. It is important to investigate having User-Design gradually brought into the curriculum where the students are asked to design small aspects of their learning while there is an instructor with a traditional class. This may

provide some interesting and useful results. This method would likely prevent the downside of User-Design, the lack of instructor experience as it relates to appropriate size & complexity of projects (McLaren, 1999).

How might User-Design work in other skill based disciplines? For example courses in media production that have large video or audio production projects. It would be interesting to see how these compare to applying User-Design to more theoretical or observational disciplines found in the social sciences.

How well would User-Design instruction translate to the workplace? After having learned a skill with a User-Design form of instruction would the student have the skills required to follow workplace design and requirements and produce a professional product? A longitudinal study following students with this form of instruction could answer that question.

Based on the findings of this dissertation, computer science college students may benefit from User-Design learning techniques. The results of future research will continue to explore and expand on this research adding to the benefits and bringing to light limitations of User-Design.

References

Abras, C., Maloney-Krichmar, D., & Preece, J. (2004). User-centered design. *Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications, 37*(4), 445-456.

Aditomo, A., Goodyear, P., Bliuc, A.-M., & Ellis, R. A. (2013). Inquiry-based learning in higher education: Principal forms, educational objectives, and disciplinary variations. *Studies in Higher Education, 38*(9), 1239-1258.

Almeida, L. C. (2008). *The Phenomenological Exploration of User-Design In Gifted Rural High School Students When Designing Their Own Game.* (Doctor of Philosophy), The Pennsylvania State University, State College PA.

Annetta, L. A., Frazier, W., Folta, E., Holmes, S., Lamb, R., & Cheng, M.-T. (2013). Science teacher efficacy and extrinsic factors toward professional development using video games in a Design-Based research model: The next generation of STEM learning. *Journal of Science Education and Technology, 22*(1), 47-61. doi: 10.1007/s10956-012-9375-y

Ausubel, D. P., Novak, J. D., & Hanesian, H. (1968). Educational psychology: A cognitive view.

Baek, E.-O., Cagiltay, K., Boling, E., & Frick, T. (2008). User-centered design and development *Handbook of research on educational communications and technology* (pp. 659-670). academia.edu

Banathy, B. (1991). *Systems design of education: a journey to create the future*. Englewood Cliffs, NJ: Educational Technology Pubns.

Banathy, B. H. (1992). Comprehensive systems design in education: Building a design culture: From conception to realization. *Educational Technology, 32*(8), 32-34.

Barab, S., Pettyjohn, P., Gresalfi, M., Volk, C., & Solomou, M. (2012). Game-based curriculum and transformational play: Designing to meaningfully positioning person, content, and context. *Computers & Education, 58*(1), 518-533.

Barab, S., & Squire, K. (2004). Introduction: Design-Based research: Putting a stake in the ground. *The Journal of the Learning Sciences, 13*(1), 1-14. doi: 10.2307/1466930

Beaubouef, T., & Mason, J. (2005). Why the high attrition rate for computer science students: some thoughts and observations. *ACM SIGCSE Bulletin, 37*(2), 103-106.

Bernard, H. R. (2011). *Research methods in anthropology: Qualitative and quantitative approaches*: *Fifth Edition* ;Rowman Altamira Publishers, Inc.

Biggers, M., Brauer, A., & Yilmaz, T. (2008). Student perceptions of computer science: a retention study comparing graduating seniors with cs leavers. *SIGCSE Bull., 40*(1), 402-406. doi: 10.1145/1352322.1352274

Bogdan, R., & Biklen, S. K. (1997). *Qualitative research for education*: Boston: Allyn & Bacon.

Bourdeau, B. (2000). Dual Relationships in Qualitative Research. *The Qualitative Report, 4*(3),1-6 .

Bovill, C., Morss, K., & Bulley, C. J. (2009). Should students participate in curriculum design? Discussion arising from a first year curriculum design project and a literature review. *Pedagogic Research in Maximising Education, 2*, 17-26.

Braun, V., & Clarke, V. (2006). Using Thematic Analysis in Psychology. *Qualitative research in psychology, 3*(2), 77-101.

Bruner, J. S. (1961). The act of discovery. *Harvard educational review,* 31, 21-32.

The Carnegie Classification of Institutions of Higher Education (n.d.). About Carnegie Classification. Retrieved (2010) from http://carnegieclassifications.iu.edu/.

Carr-Chellman, A. (2007). *User Design*. Mahwah, New Jersey: Lawrence Erlbaum Associates.

Carr-Chellman, A., & Almeida, L. C. (2006). User-Design for systemic change. *TechTrends, 50*(2), 44-45.

Carr-Chellman, A., & Savoy, M. (2004). User-design research. In D. H. Jonassen (Ed.), *Handbook of research on educational communication and technology: A project of the association for educational communications and technology* (2nd ed., pp. 701-716). Mahway, New Jersey: Lawrence Erlbaum Associates.

Carr, A. (1996). Distinguishing systemic from systematic. *TechTrends, 41*(1), 16-20. doi: 10.1007/bf02812077

Carr, A. (1997). User-design in the creation of human learning systems. *Educational Technology Research and Development, 45*(3), 5-22. doi: 10.1007/bf02299726

Carroll, J. M., Chin, G., Rosson, M. B., & Neale, D. C. (2000, August). The development of cooperation: Five years of participatory design in the virtual school. In *Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques* (pp. 239-251). ACM.

Chan, P. E., Graham-Day, K. J., Ressa, V. A., Peters, M. T., & Konrad, M. (2014). Beyond involvement: Promoting student ownership of learning in classrooms. *Intervention in School and Clinic, 50*(2), 105-113. doi: 10.1177/1053451214536039

Chen, B., & Hirumi, A. (2009). Effects of advance organizers on learning for differentiated learners in a fully web-based course. *International Journal of Instructional Technology and Distance Learning, 6*(6), 3-12.

Chen, D., & Stroup, W. (1993). General system theory: Toward a conceptual framework for science and technology education for all. *Journal of Science Education and Technology, 2*(3), 447-459. doi: 10.1007/bf00694427

Chen, X., & Soldner, M. (2013a). STEM Attrition: College students' paths into and out of STEM fields. In C. f. E. statistics (Ed.). Washington, DC: Institute of Education Sciences.

Chen, X., & Soldner, M. (2013b). STEM Attrition: College students' paths into and out of STEM fields (U. S. D. o. Education, Trans.). Washington D.C.: Institute of Educational Sciences.

Chua, C. K., & Leong, K. F. (2003). *Rapid prototyping: principles and applications* (Vol. 1): World Scientific.

Cilesiz, S. (2009). Educational computer use in leisure contexts: A phenomenological study of adolescents' experiences at Internet cafes. *American Educational Research Journal, 46*(1), 232-274.

Clark, M. H., & Schroth, C. A. (2010). Examining relationships between academic motivation and personality among college students. *Learning and Individual Differences, 20*(1), 19-24. doi: 10.1016/j.lindif.2009.10.002

Cliburn, D. C. (2006). *The effectiveness of games as assignments in an introductory programming course.* Paper presented at the Frontiers in Education Conference, 36th Annual.

Colaizzi, P. F. (1978). Psychological research as the phenomenologist views it.

Corbin, J., & Strauss, A. (2014). *Basics of qualitative research: Techniques and procedures for developing grounded theory*: Sage publications.

Creswell, J. W. (1998). *Qualitative inquiery and research design: Choosing among five traditions*. Thousand Oaks, CA: Sage.

Creswell, J. W. (2009). *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*: Sage Publications.

Dale, N. B., & Lewis, J. (2012). *Computer Science Illuminated* (5th ed.). Burlington, MA: Jones and Bartlett.

Darling-Hammond, L. (2010). Teacher education and the American future. *Journal of Teacher Education, 61*(1-2), 35-47.

Denzin, N. K., & Lincoln, Y. S. (2011). *The SAGE Handbook of Qualitative Research*: SAGE Publications.

Dickey, M. (2005). Engaging by design: How engagement strategies in popular computer and video games can inform instructional design. *Educational Technology Research and Development, 53*(2), 67-83. doi: 10.1007/bf02504866

Duckett, J. (2014). *JavaScript and JQuery: Interactive Front-End Web Development*: Wiley Publishing.

ECMA. (2014). ECMA-262, Edition 5. ECMAScript Documentation: ECMA.

Edinboro University. (2012, 2010). Retrieved 7/31/2012, 2012, from http://colleges.usnews.rankingsandreviews.com/best-colleges/edinboro-university-3321

Edinboro University Fact Book. (2013a). Retrieved 9/15/2013, 2013, from http://www.edinboro.edu/about_eup/

Edinboro University Fact Book. (2013b) (pp. 156): Edinboro University.

Employment by major occupational group, 2012 and projected 2022. (2013). In N. E. M. t. a. code (Ed.). Washington D.C.: U.S. Bureau of Labor Statistics.

Fullerton, T. (2014). *Game design workshop: a playcentric approach to creating innovative games*: CRC press.

Gagne, R. M., Wager, W. W., Golas, K. C., Keller, J. M., & Russell, J. D. (2005). Principles of instructional design, 5th edition. *Performance Improvement, 44*(2), 44-46. doi: 10.1002/pfi.4140440211

Gee, J. P. (2007). *Good Video Games + Good Learning*. New York: Peter Lang Publishing.

Goode, J., & Margolis, J. (2011). Exploring Computer Science: A case study of school reform. *Trans. Comput. Educ., 11*(2), 1-16. doi: 10.1145/1993069.1993076

Goode, J., Margolis, J., & Chapman, G. (2014). *Curriculum is not enough: the educational theory and research foundation of the exploring computer science professional development model*. Paper presented at the proceedings of the 45th ACM technical symposium on Computer Science education, Atlanta, Georgia, USA.

Graneheim, U. H., & Lundman, B. (2004). Qualitative content analysis in nursing research: concepts, procedures and measures to achieve trustworthiness. *Nurse Education Today, 24*(2), 105-112. doi: 10.1016/j.nedt.2003.10.001

Greitzer, F. L., Kuchar, O. A., & Huston, K. (2007). Cognitive science implications for enhancing training effectiveness in a serious gaming context. *J. Educ. Resour. Comput., 7*(3), 2. doi: 10.1145/1281320.1281322

Guba, E. G., & Lincoln, Y. S. (1981). *Effective evaluation: Improving the usefulness of evaluation results through responsive and naturalistic approaches*: Jossey-Bass.

Guillén-Nieto, V., & Aleson-Carbonell, M. (2012). Serious games and learning effectiveness: The case of It's a Deal! *Comput. Educ., 58*(1), 435-448. doi: 10.1016/j.compedu.2011.07.015

Handelsman, J., Ebert-May, D., Beichner, R., Bruns, P., Chang, A., DeHaan, R., Tilghman, S. M. (2004). Scientific teaching. *Science, 304*(5670), 521-522.

Hillman, P. (2010). *Edinboro University Computer Science Program Review*. Edinboro University, School of Natural Sciences and Health Professions: Edinboro University of Pennsylvania.

Hoadley, C. M. (2004). Methodological alignment in design-based research. *Educational Psychologist, 39*(4), 203-212.

Howland, G. (2002). Balancing gameplay hooks. In C. R. Media (Ed.), *Game design perspectives* (pp. 78–84).

Hycner, R. H. (1985). Some guidelines for the phenomenological analysis of interview data. *Human studies, 8*(3), 279-303.

Jackson, S. A., Madison, C., Vangelisti, A., Daly, J., & Friedrich's, G. (1999). Instruction by design: Technology in the discourse of teaching and learning. *Teaching communication: Theory, research, and methods*, 393-408.

Jonas, M. (2013). Teaching introductory progamming using multiplayer board game strategies in Greenfoot. *J. Comput. Sci. Coll., 28*(6), 19-25.

Jones, B., Valdez, G., Norakowski, J., & Rasmussen, C. (1994). Designing Learning and Technology for Educational Reform. Publications Department, North Central Regional Educational laboratory.

Jones, T., & Richey, R. (2000). Rapid prototyping methodology in action: A developmental study. *Educational Technology Research and Development, 48*(2), 63-80. doi: 10.1007/bf02313401

Jordan, R., & Carr-Chellman, A. A. (2014, 2014 Spring). Do-It-Yourself design: A user-designed training curriculum can lead to innovation programs that are more readily adopted by users. An interconnected approach to design uses selected team members and a facilitator. *The Public Manager, 43,* 61+.

Kafle, N. P. (2013). Hermeneutic phenomenological research method simplified. *Bodhi: An Interdisciplinary Journal, 5*(1), 181-200.

King, F., Goodson, L., & Rohani, F. (1998). Higher order thinking skills. *Retrieved January, 31*, 2011.

King, P. M., Wood, P. K., & Mines, R. A. (1990). Critical thinking among college and graduate students. *Review of Higher Education, 13*(3), 167-186.

Klimmt, C., Schmid, H., & Orthmann, J. (2009). Exploring the enjoyment of playing browser games. *CyberPsychology & Behavior, 12*(2), 231-234. doi: 10.1089/cpb.2008.0128

Konak, A., Clark, T. K., & Nasereddin, M. (2014). Using Kolb's experiential learning cycle to improve student learning in virtual computer laboratories. *Computers & Education, 72*, 11-22.

Könings, K. D., Brand-Gruwel, S., & van Merriënboer, J. J. G. (2010). An approach to participatory instructional design in secondary education: an exploratory study. *Educational Research, 52*(1), 45-59.

Korur, F., Toker, S., & Eryılmaz, A. (2016). Effects of the integrated online Advance Organizer teaching materials on students' science achievement and attitude. *Journal of Science Education and Technology, 25*(4), 628-640. doi: 10.1007/s10956-016-9618-4

Koster, R. (2005). *A theory of Fun for Game Design*. Scotsdale, AZ.

Kroenke, D., & Auer, D. J. (2016). *Database processing* (14 ed.): Pearson.

Kvale, S. (1996). *Interviews: an introduction to qualitative research interviewing*: Sage Publications.

Land, S. M. (2000). Cognitive requirements for learning with open-ended learning environments. *Educational Technology Research and Development, 48*(3), 61-78. doi: 10.1007/bf02319858

Leonard, W. H. (1997). How do college students learn science. *Methods of Effective Teaching and Course Management for University and College Science Teachers*, 5-20.

Levy, P., & Petrulis, R. (2012). How do first-year university students experience inquiry and research, and what are the implications for the practice of inquiry-based learning? *Studies in Higher Education, 37*(1), 85-101. doi: 10.1080/03075079.2010.499166

Lim, C. P. (2008). Spirit of the game: Empowering students as designers in schools? *British Journal of Educational Technology, 39*(6), 996-1003. doi: 10.1111/j.1467-8535.2008.00823_1.x

Loewenberg Ball, D., & Forzani, F. M. (2009). The work of teaching and the challenge for teacher education. *Journal of Teacher Education, 60*(5), 497-511. doi: 10.1177/0022487109348479

Marczyk, G. R., DeMatteo, D., & Festinger, D. (2005). *Essentials of research design and methodology*. Hoboken, New Jersey: John Wiley & Sons.

Marra, R. M., Rodgers, K. A., Shen, D., & Bogue, B. (2012). Leaving engineering: A multi-year single institution study. *Journal of Engineering Education, 101*(1), 6-27. doi: 10.1002/j.2168-9830.2012.tb00039.x

McLaren, P. (1999). *Schooling as a ritual performance: Toward a political economy of educational symbols and gestures*: Rowman & Littlefield.

Merrill, M. D., Drake, L., Lacy, M. J., Pratt, J., & Group, I. R. (1996). Reclaiming instructional design. *Educational Technology, 36*(5), 5-7.

Mikkonen, T., & Taivalsaari, A. (2007). Using JavaScript as a real programming language (pp. 17): Sun Microsystems, Inc.

Miles, M. B., & Huberman, A. M. (1985). *Qualitative data analysis*: Sage Newbury Park,, CA.

Monge, A. E., Fadjo, C. L., Quinn, B. A., & Barker, L. J. (2015). EngageCSEdu: engaging and retaining CS1 and CS2 students. *ACM Inroads, 6*(1), 6-11.

Murphy-Hill, E., Zimmermann, T., & Nagappan, N. (2014). *Cowboys, ankle sprains, and keepers of quality: how is video game development different from software development?* Paper presented at the Proceedings of the 36th International Conference on Software Engineering, Hyderabad, India.

Patton, M. (1990). *Qualitative Evaluation and Research Methods*: SAGE Publications.

Patton, M. Q. (2001). *Qualitative Research & Evaluation Methods* (3rd ed.): SAGE Publications.

Prensky, M. (2003). Digital game-based learning. *Comput. Entertain., 1*(1), 21-21. doi: 10.1145/950566.950596

Prensky, M. (2008). Students as designers and creators of educational computer games: Who else? *British Journal of Educational Technology, 39*(6), 1004-1019. doi: 10.1111/j.1467-8535.2008.00823_2.x

Pressman, R. S. (2005). *Software engineering: a practitioner's approach*: Palgrave Macmillan.

Rankin, Y. A., McNeal, M., Shute, M. W., & Gooch, B. (2008). *User centered game design: evaluating massive multiplayer online role playing games for second language acquisition*. Paper presented at the Proceedings of the 2008 ACM SIGGRAPH symposium on Video games, Los Angeles, California.

Rask, K. (2010). Attrition in STEM fields at a liberal arts college: The importance of grades and pre-collegiate preferences. *Economics of Education Review, 29*(6), 892-900. doi: 10.1016/j.econedurev.2010.06.013

Reigeluth, C. M. (1993). Principles of educational systems design. *International Journal of Educational Research, 19*(2), 117-131. doi: 10.1016/0883-0355(93)90022-c

Reigeluth, C. M. (1999). *Instructional-design Theories and Models: A New Paradigm of Instructional Theory*: Taylor & Francis.

Robins, A., Rountree, J., & Rountree, N. (2003). Learning and teaching programming: A review and discussion. *Computer science education, 13*(2), 137-172.

Rollings, A., & Adams, E. (2003a). *Andrew Rollings and Ernest Adams on Game Design*: New Riders Games.

Rollings, A., & Adams, E. (2003b). *Game Design*: New Riders.

Rose, E. (2011). The phenomenology of on-screen reading: University students' lived experience of digitised text. *British Journal of Educational Technology, 42*(3), 515-526.

Rossman, G. B., & Rallis, S. F. (2012). *Learning in the field : an introduction to qualitative research*. Los Angeles : SAGE.

Schell, J. (2014). *The Art of Game Design: A book of lenses*: CRC Press.

Schlechty, P. C. (2001). *Inventing Better Schools: An Action Plan for Educational Reform*. San Francisco: Jossey-Bass Publishers.

Shabanah, S., & Chen, J. X. (2009). Simplifying algorithm learning using serious games. Paper presented at the *Proceedings of the 14th Western Canadian Conference on Computing Education*, (pp. 34-41). ACM, Burnaby, British Columbia, Canada.

Simonsen, J., & Robertson, T. (2012). *Routledge handbook of participatory design*: Routledge.

Staker, H., & Horn, M. B. (2012). Classifying K–12 blended learning. *Innosight Institute*.

Strayer, J. F. (2007). *The effects of the classroom flip on the learning environment: a comparison of learning activity in a traditional classroom and a flip classroom that used an intelligent tutoring system.* (Doctoral dissertation, The Ohio State University).

Sung, K., Hillyard, C., Angotti, R. L., Panitz, M. W., Goldstein, D. S., & Nordlinger, J. (2011). Game-themed programming assignment modules: a pathway for gradual integration of gaming context into existing introductory programming courses. *Education, IEEE Transactions on, 54*(3), 416-427.

Sweedyk, E., deLaet, M., Slattery, M. C., & Kuffner, J. (2005). Computer games and cs education: why and how. In *ACM SIGCSE* Bulletin (Vol. 37, No. 1, pp. 256-257). ACM.

Tripp, S. D., & Bichelmeyer, B. (1990). Rapid prototyping: An alternative instructional design strategy. *Educational Technology Research and Development, 38*(1), 31-44.

Tune, J. D., Sturek, M., & Basile, D. P. (2013). Flipped classroom model improves graduate student performance in cardiovascular, respiratory, and renal physiology. *Advances in physiology education, 37*(4), 316-320.

Turner, M. E., & Pratkanis, A. R. (1998). Twenty-five years of groupthink theory and research: Lessons from the evaluation of a theory. *Organizational behavior and human decision processes, 73*(2), 105-115.

van den Akker, J., Gravemeijer, K., McKenney, S., & Nieveen, N. (2006). Introducing educational design research. *Educational design research*, 1-14.

Van Manen, M. (1990). Researching Lived Experience : Human Science for and Action Sensitive Pedagogy*: State University of New York Press, Albany, NY*

Von Bertalanffy, L. (1969). General Systems Theory:Foundations, Development, Applications *Archives of General Psychiatry, 21*(2), 251.

W3C. (2014). HTML5 A vocabulary and associated APIs for HTML and XHTML *W3C* Recommendation. *World Wide Web Consortium*.

Wang, F., & Hannafin, M. J. (2005). Design-based research and technology-enhanced learning environments. *Educational Technology Research and Development, 53*(4), 5-23. doi: 10.1007/bf02504682

Williams, L. (2001). *Integrating pair programming into a software development process.* Paper presented at the Software Engineering Education and Training, 2001. *Proceedings. 14th Conference on* (pp. 27-36). IEEE.

Williams, L., Wiebe, E., Yang, K., Ferzli, M., & Miller, C. (2002). In support of pair programming in the introductory Computer Science course. *Computer Science Education, 12*(3), 197-212. doi: 10.1076/csed.12.3.197.8618

Wright, G. B. (2011). Student-centered learning in higher education. *International Journal of Teaching and Learning in Higher Education, 23*(1), 92-97.

Appendix A

Letter of Invitation

Title of Study: The Phenomenological Exploration of User-Design Instruction among Undergraduate Students Majoring in Computer Science when Developing a Web Game Prototype.

Principal investigator: David Tucker.

You have been invited to participate in this dissertation study whose purpose is to explore the development and experiences of undergraduate computer science students when they conceptualize and design a computer game because you are a student of computer science at Edinboro University of Pennsylvania. Your participation in this study is voluntary. You can withdraw or discontinue your participation in the project at any time by or emailing the principal investigator at dtucker@edinboro.edu or by calling his number at 814-732-2552. There will be no penalty or questions asked. No one will stop you from withdrawing from the study.

Once you arrive at the Hamilton Lower Level laboratory, the principal investigator will meet you and ask you to choose a seat in the computer lab that you will be using for the next six weeks. You are going to have to design and develop a Web game using JavaScript, be interviewed three times throughout this study, and be observed once every two weeks. The principal investigator will also collect your game artifacts in this study. Nobody other than the principal investigator and the dissertation adviser will have access to the data. All data will be retained for a period of three years in compliance with federal regulations and later destroyed. The data will be kept in the principal investigator's office, in a locked cabinet.

This is a User-Design study. You will be designing and developing your game on your own with minimal guidance. The entire game design experience will last six weeks. In conducting this research, the principal investigator is very careful to safeguard your information. If you have any questions or concerns, you can contact the principal investigator, or his committee chair. You can also contact the IRB office via the website http://www.iup.edu/irb/default.aspx or via email at *irb research@iup.edu* if you have any questions in regards to research studies.

Principal Investigator:

David Tucker
Instructor, Math & Computer Science
Doctoral Candidate, Communications Media & Instructional Technology (IUP)
Edinboro University of Pennsylvania
215 Meadville Street
Edinboro, PA 16444
814-732-2552
dtucker@edinboro.edu

Faculty Sponsor/Committee Chair:

Dr. Luis Almeida
Assistant Professor, Department of Communications Media
Indiana University of Pennsylvania
121-C Stouffer Hall
Indiana, PA 15705-1087
724-357-3295
Luis.Almeida@iup.edu

Appendix B

Informed Consent Form

CONSENT TO PARTICIPATE IN A RESEARCH STUDY

Title of Study: The Phenomenological Exploration of User-Design Instruction among Undergraduate Students Majoring in Computer Science when developing a web Game Prototype.

Principal Investigator: David Tucker

Introduction

You have been asked to participate in a doctoral dissertation research project. This consent form describes the protocols and procedures of this endeavor.

Purpose of The Study

The purpose of this study is to explore the development and experiences of undergraduate computer science students when they conceptualize and design a computer game. This will take place at a medium sized liberal arts university and the students will act as user-designers of their own instructional system (B. Banathy, 1991). This study specifically looks at the student's own design process and decisions as they discover ways to solve complex computing tasks. Additionally, this study will also advance research on student empowerment in the context of a college class and analyze the methods utilized by the students as they accomplish their assigned task.

What Will Happen During the Study?

Multiple dates (Monday to Friday spots) will be provided in order to make participation convenient for most students. Selected students will have to design and develop a web game using JavaScript within six weeks. The subjects will then be interviewed three times throughout this study. The researcher will also observe the students throughout the six week period once every two weeks. The researcher will also collect the subject's artifacts to triangulate the data. Subjects will have access to the Hamilton Lower Level Computer Lab throughout the six week period.

What Are the Possible Risks or Discomforts?

There are minimal psychological, personal, or academic risks associated with this study.

What Are the Possible Benefits of Being in This Study?

The potential benefits will be the design and development of a web game using JavaScript, an innovative game engine. In short, they will have the chance to conceptualize and design their own computer game. Students will get one extra point for participating in this study. Subjects will have the chance to win a $100 amazon gift certificate after completing the JavaScript game.

How Will the Data Collected Be Kept Confidential?

Nobody other than the principal investigator and the dissertation adviser will have access to the data. All data will be retained for a period of three years in compliance with federal regulations and later destroyed. The data will be kept in the principal investigator's office, in a locked cabinet. The electronic information will be housed on Edinboro University of Pennsylvania servers, which are secure as per federal regulation. Data collected will be analyzed and prepared for publication.

What Happens If I Have More Questions?

You can ask questions about this doctoral dissertation research project. Contact David Tucker at (814) 732-2550 with questions. You can also call this number if you have complaints or concerns about this research. If you have questions about your rights as a research participant, or you have concerns or general questions about the research, contact IUP's Office for Research Protections at (724) 357-7730 or by sending an email to *irb-research@iup.edu.* You may also call the number above if you cannot reach the research team or wish to talk to someone else.

What Will Happen If You Decide Not To Be in the Study?

Your decision to be in this research is voluntary. You can stop at any time simply by sending an email or calling the principal investigator's phone number at (814) 732-2550. You do not have to answer any questions you do not want to answer. Refusal to take part in or withdrawing from this study will involve no penalty or loss of benefits you would receive otherwise.

SUBJECT'S STATEMENT

I have read and understand the information on the Informed Consent form and I consent to volunteer to be a subject in this study. I understand that my personal information is completely confidential and that I have the right to withdraw at any time. I have received a copy of the Informed Consent form to keep in my possession. I understand and agree to the conditions of this study as described.

_____ _____

SUBJECT'S SIGNATURE **DATE**

Appendix C

Log Sample