A QUANTITATIVE EXAMINATION OF CRITICAL SUCCESS FACTORS COMPARING AGILE AND WATERFALL PROJECT MANAGEMENT METHODOLOGIES

by

Mitra Pedersen

RICHARD LIVINGOOD, PhD, Faculty Mentor and Chair

CASEY BROWN, PhD, Committee Member

GEOFFREY LAENDNER, PhD, Committee Member

Barbara Butts Williams, PhD, Dean, School of Business and Technology

A Dissertation Presented in Partial Fulfillment

Of the Requirements for the Degree

Doctor of Philosophy

Capella University

September 2013

Abstract

This study investigated the rate of success for IT projects using agile and standard project management methodologies. Any successful project requires use of project methodology. Specifically, large projects require formal project management methodologies or models, which establish a blueprint of processes and project planning activities. This study asked what methodologies should be utilized to improve the chances of project success, as project managers are engaged in decision making. A deficiency in reliability and consistency correlate to a lower success rate of projects. The objective of this quantitative study was to relate the use of the agile or waterfall methodologies and specific critical success factors (CSFs) to IT project success for a sample of IT project managers who have used these methodologies. The results of the analysis of variance showed that there were significant differences between the extent of use of effective communications, user involvement, and use of a quality plan between the models. The results of the Pearson's correlation test showed that project success is significantly and positively associated with effective communication, user involvement, and use of a quality plan in the agile methodology. In addition, the results of the Pearson's correlation test showed that project success is significantly and positively associated with only effective communication and use of a quality plan in the waterfall methodology. Lastly, the results of multiple linear regression analysis showed that none of the CSFs of effective communication, use of quality plan, and user involvement significantly influences the success of the project in both the agile and waterfall project management models.

Dedication

This work is primarily dedicated to the most influential people in my life: To my late mother, Molood Abadian, who served 30 years in the field of education as an instructor and school principal, thank you for all you have done to teach me how to write my first letter, prepare me in life, and for instilling a level of courage and perseverance that have guided me through the years. To my father, Yagoub Amirsoleimani, has encouraged me to be well educated, strong, and brave. It is also dedicated to my children, Roza Kalantari my daughter, for her love, her tremendous encouragement, and always believing in me that gave me the strength to continue embracing this journey and Karl Kalantari, my son, for his inspiration and support to achieve high and influence future generations in my family. This work is also dedicated to my husband Ole Martens Pedersen for his support and encouragement.

Acknowledgments

My thanks and deep appreciation go out to Dr. Richard Livingood (Mentor/Chair of Committee), for constantly encouraging me and providing invaluable support throughout this process, along with committee members Dr. Geoffrey Laendner and Dr. Casey Brown, for their guidance and assistance on this journey.

Table of Contents

List of Tables

List of Figures

CHAPTER 1. INTRODUCTION

Introduction to the Problem

Project Management Solutions has conducted research that indicates more than one third (37%) of information technology (IT) projects were troubled and at risk of failure (Standish Group, 2010). Many IT projects are not completed on schedule, on budget, and within scope, resulting in cost overruns and missed business opportunities (Standish Group, 2011). The CHAOS studies undertaken by the Standish Group recognize the three primary constraints of time, scope, and cost that determine project success. Proposals to properly manage these constraints include the application of appropriate project methodologies (Standish Group, 2008). This study investigated critical success factors and the extent to which they are associated with perceptions of project success for information technology projects that use agile, waterfall, or mixed agile and waterfall project management methodologies.

Chapter 2 reviews the literature on agile and waterfall software development methodologies, including Prince2, Six Sigma, spiral model, extreme programming, crystal, unified process (RUP), and rapid applications development (RAD). Most project methodology has moved from waterfall to agile. Agile and waterfall are the most commonly used methodologies in software development and information technology. Agile and waterfall are the selected methodologies for this research project.

The Project Management Institute (PMI) is a professional organization that offers a range of services for project management professionals, including project management standards and best practices recognized by the American National Standards Institute (ANSI). PMI (2013) defined project success as the "collective assessment by project stakeholders of the degree to which the project has achieved each of its objectives" (p. 37). Oisen (1971), more than 40 years ago, suggested cost, time, and quality as the success criteria bundled into the description. In later years, the modified definition of project success also included the completion of the project and customer perceptions of the quality of the solution in production (Cohen, Dori, & Haan, 2010; Praeg & Spath, 2009). Project success can be achieved by the customer/users performing an acceptance test without business interruption during project execution and by managing organizational changes through implementation of organization change management processes (Camilleri, 2011).

Any successful project requires use of project methodology (Nelson, 2005). Specifically, large projects require formal project management methodologies or models, which establish a blueprint of processes and project planning process. Among standard project management methodologies, the waterfall project management model was developed for software development and is guided by the metaphor of downward development in sequential order (*Waterfall Model,* 2012). Developed by Royce (1970), the waterfall model has been popular for its relative ease of use. Software programmers have found this model to be efficient for short-term project periods and development of programs that are already stable (Mumford, 2010). The waterfall model documents varying stages of development and facilitates the transfer of a project midway from one team of developers to another (*Waterfall Model,* 2012). Departing from this traditional model, the agile model identifies various aspects of a problem and its potential solutions (*When is a Model Agile?,* 2013). The agile model develops solutions that are considered minimally adequate, with the details relatively minimal to ensure ease of utilization by a broad scope of audiences (Griffiths, 2007).

The waterfall model represents the original software development model, and is less commonly used in contemporary applications than it used to be (Benediktsson, Dalcher, & Thorbergsson, 2006). Benediktsson et al. (2006) noted the particular preference for use of agile in small to medium sized projects in a context of rapid change, uncertain needs, and high demand for business value. The agile model is preferred due to its smaller team sizes and adaptability for competition and rapid change (Leffingwell, 2007). Each model offers project managers something unique and should be applied based on distinct organizational and project contexts (Fewell, 2009).

In an effort to help increase the success rates of IT projects, this study seeks to determine, based on the experience and perceptions of IT managers who have used the agile and waterfall methodologies, which among the identified critical success factors (CSFs) are related to successful projects. In line with this overall purpose, this study also seeks to investigate the extent to which the identified CSFs are associated with project success for IT projects that use agile or waterfall project management techniques. Somers and Nelson (2001) defined CSFs as "those factors that are necessary to meet the desired deliverables of the customer on a project" (p. 3). Primary critical success factors include cost, scope and timeline. In addition, critical success factors involve quality and the appropriateness and timing of user acceptance signoffs (Hirshfield, 2010).

Background of the Study

Among standard project management methodologies, the waterfall project management model was developed for software development and is guided by the metaphor of downward development in sequential order (*Waterfall Model,* 2012). Developed by Royce (1970), the waterfall model has been popular for its relative ease of use. Software programmers have found this model to be efficient for short-term project periods (Mumford, 2010). The waterfall model documents varying stages of development, and this facilitates the transfer of a project midway from one team of developers to another. In contrast to this traditional model, the agile model identifies various aspects of a problem and its potential solutions (*When is a Model Agile?,* 2013). The agile model develops solutions that are considered just good enough, with the details relatively minimal to ensure ease of utilization by a broad scope of audiences (Griffiths, 2007).

This study elaborated upon the earlier findings of Cao (2006). Cao addressed only the project success and critical success factors of agile methodologies. Thus, there is a need for an evaluation of critical success factors that contribute to project success in other methodologies and in integrated waterfall–agile methodologies. This study attempted to collect and analyze data that could answer the question of what methodologies prove most beneficial in contemporary project environments.

Statement of the Problem

The rate of failed IT projects is too high to tolerate given cost, schedule, and quality objectives. Projects continue to suffer from a myriad issues and problems, resulting in failure to meet schedule, budget, and quality (Pinto & Slevin, 1989). This study should also help organizational leaders to understand the importance of change management, and in agreeing on project plans that are focused on user involvement. This should help lead to overall improvements in the rate of project success. The results will also help stakeholders agree on project goals, help project managers develop clearly defined plans with assigned responsibilities and accountability, and help project managers manage changes to scope and user needs. In turn, project success could improve the organization's return on investment (Verner & Evanco, 2005). This research study should provide researchers and IT managers with quantitative research using project management methodologies and add to the existing literature with regard to quantitative research studies on the uses of agile, waterfall, and mixed methodologies.

Purpose of the Study

The purpose of this quantitative, nonexperimental, descriptive study was designed to relate the use of the agile or waterfall methodologies and specific CSFs to IT project success for a sample of IT project managers who have used the agile and waterfall methodologies. The independent variables are the use of a specific project management methodology, which is generally defined as the use of either the agile method or the waterfall method, and the specific CSFs, which are defined as critical factors that are required to meet customer's expectation for a product or services (Somers & Nelson, 2001). The dependent variable is project success, which is generally defined as "the collective assessment by project stakeholders . . . of the degree to which the project has achieved each of its objectives" (PMI, 2013, p. 37).

Additionally, the study sought to understand differences in the role that CSFs play in the delivery of successful IT projects that use waterfall or agile methodology, in other words, which CSFs are used by each methodology that potentially result in successful projects. The researcher used an online survey instrument to collect data on IT project success rates and methodology used for project delivery. The findings of this research could contribute to a better understanding of IT project success for agile and waterfall methodologies and contribute to the existing body of project management research.

Project success was the result of two independent variables used in this study, namely agile and waterfall. This study examined project success of agile and waterfall development methods in light of user involvement, quality, and communication. Since the waterfall model has become less popular among contemporary users, it is necessary to test whether this development is valid (Benediktsson et al., 2006). In addition, this study has the purpose of testing the findings of various researchers regarding project success. The purpose of this study is also to evaluate the findings of the PMI (2013), which emphasizes the importance of a collective assessment by stakeholders of whether the project meets objectives. Kerzner (2009) defined project success as project completion within performance, cost, and time determinants. Figure 1 identifies the focus of this study in light of project success (DV) using either agile or waterfall methodologies.

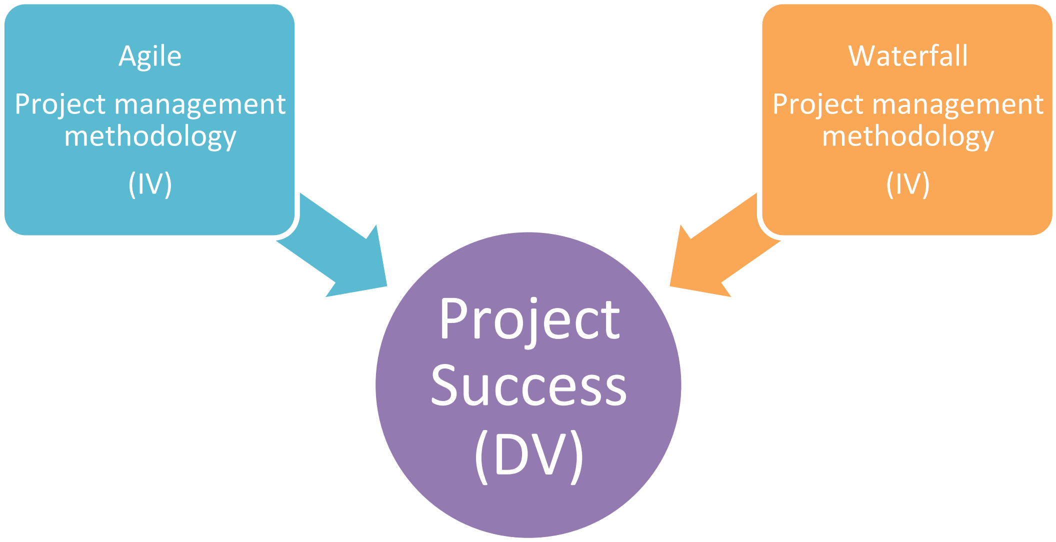


Figure 1. Project success and agile and waterfall methodologies.

Rationale

This study is based on the elaboration of a study conducted by Cao (2006) and was selected given changes in the organizational and market environments since the original effort. The findings of the Cao study were instrumental in recommending future project design.

In addition, development of a theoretical explanation for project success became the focus of this study. The outcomes are expected to contribute to the literature by documenting nontechnical problems that continue to plague software projects (Johnston, 1995). Information technology projects are the focus of significant management effort, resulting in high costs to organizations. Technology is constantly changing, and this rate of change becomes a factor in IT project success (Johnstone, Huff, & Hope, 2006). Managing change is essential for IT managers and projects (Kuruppuarachchi, Mandal, & Smith, 2002).

Technology projects are also increasing in number, scope, and size, as organizations must implement new innovations. As a part of good project management, managers should be utilizing communication, quality strategy, user involvement, business requirement identification, implementation analyses, and dedication of resources (Legris & Collerette, 2006). Without these additional factors implemented in the project management process, IT project failure is commonplace. A 2004 review published by the Standish Group identified a failure rate of 71% for IT projects that relied on traditional critical success factors (Nelson, 2005). Time, budget, performance, and functional requirements are identified as essential critical success factors (Bennington & Baccarini, 2004; Nelson, 2005; PMI, 2013; Rockart & Bullen, 1981).

Some theorists have claimed that IT project failure is the result of low levels of user involvement, resulting in the lack of business value for an implemented system (Camilleri, 2011; Nelson, 2005). Contemporary IT systems should be evaluated based on their ability to foster learning, value, and use. The IT system should improve effectiveness, efficiency, and stakeholder knowledge (Rodriguez-Repiso, Setchi, & Salmeron, 2007). This study allowed IT and project managers to share knowledge that may contribute to more successful projects, based on use of critical factors that support proper use of methodologies, best practices, evaluation of life cycles, and a higher likelihood of project success, achieving better returns on investment.

The scholarly literature included findings that recommended additional research into project outcomes (Kendrick, 2009). This researcher designed this study to continue research still needed on the relationships among communication, user involvement, quality, and organizational change management by surveying IT project managers in the

United States and identifying critical success factors used in their past project methodologies. It is expected that this research will contribute to the literature on the use of these methodologies in IT project partnering (Kendrick, 2009).

Research Questions

This study was designed to determine whether there is a relationship between IT project success and the implementation of an agile or waterfall project management methodologies, based on critical success factors. The research questions that drove this study are as follows:

RQ 1: What is the difference between the extent of use of the identified CSFs in the agile model and in the waterfall model?

RQ 1.1: To what extent do project managers in IT for each methodology, agile and waterfall, report using effective communication?

RQ 1.2: To what extent do project managers in IT for each methodology, agile and waterfall, report using user involvement?

RQ 1.3: To what extent do project managers in IT for each methodology, agile and waterfall, report using a quality plan?

RQ 2: Which among the identified CSFs are correlated with successful projects in the agile and waterfall model?

RQ 2.1: In the waterfall project management model, which among the identified CSFs are correlated with successful projects?

RQ 2.2: In the agile project management model, which among the identified CSFs are correlated with successful projects?

RQ 2.3: In the waterfall project management model, how influential are the identified CSFs on the success of the project?

RQ 2.4: In the agile project management model, how influential are the identified CSFs on the success of the project?

This quantitative study's null hypotheses are as follows:

H01: There is no significant difference between the extent of use of effective communication between the agile model and the waterfall model.

Ha1: There is significant difference between the extent of use of the effective communication between the agile model and the waterfall model.

H02: There is no significant difference between the extent of user involvement between the agile model and the waterfall model.

Ha2: There is significant differences between the extent of user involvement between the agile model and the waterfall model.

H03: There is no significant difference between the extent of the use of a quality plan between the agile model and the waterfall model.

Ha3: There is significant difference between the extent of the use of a quality plan between the agile model and the waterfall model.

H04: Project success is not significantly associated with effective communication in the agile model.

Ha4: Project success is significantly associated with effective communication in the agile model.

H05: Project success is not significantly associated with user involvement in the agile model.

Ha5: Project success is significantly associated with user involvement in the agile model.

H06: Project success is not significantly associated with the use of a quality plan in the agile model.

Ha6: Project success is significantly associated with use of a quality plan in the agile model.

H07: Project success is not significantly associated with effective communication in the waterfall model.

Ha7: Project success is significantly associated with effective communication in the waterfall model.

H08: Project success is not significantly associated with user involvement in the waterfall model.

Ha8: Project success is significantly associated with user involvement in the waterfall model.

H09: Project success is not significantly associated with the use of a quality plan in the waterfall model.

Ha9: Project success is significantly associated with use of a quality plan in the waterfall model.

Figure 2 illustrates the interactions among two project methodologies, three CSF and project success for this research study.

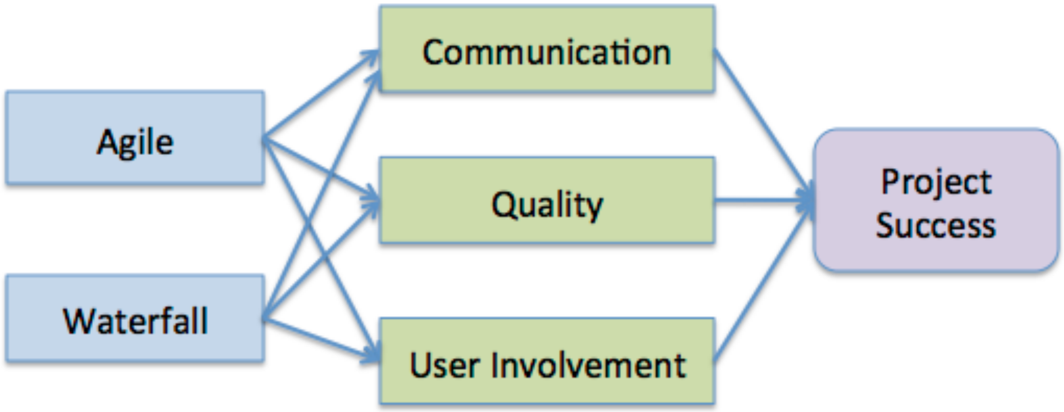


Figure 2. Potential interactions of type of project methodology leading to project success.

Significance of the Study

The rate of failed IT projects is considered too high to justify based on cost, schedule, and quality objectives. Projects continue to suffer from a myriad of issues and problems, resulting in failure to meet schedule, budget, and quality (Cook-Davies, 2002). This study should help IT functional and project managers engage end users, manage change during the project life cycle (PLC), and employ user acceptance testing prior to production release. The results may also help stakeholders agree on project goals, help project managers develop clearly defined plans with assigned responsibilities and accountability, and help project managers manage changes to scope and user needs. In turn, project success will help improve the organization's return on investment. This study has the potential to provide practitioners and scholars with additional support to the current body of knowledge through an investigation of project management methodologies, based on the use of critical success factors, by studying CSF impact on agile and waterfall.

Definition of Terms

The following definitions are important to this study:

Communication. The provision of an appropriate network and necessary data to all key actors in the project implementation.

Critical success factors (CSFs). Factors necessary for project success, including schedule, budget, quality, and change control processes, along with appropriateness and timing of signoffs (Kerzner, 2006).

End users. Ultimate intended users for which the system is developed (Laudon & Laudon, 2002).

Information technology (IT). The comprehensive management, design, development, study, implementation, and support of computer systems, including software and hardware.

Information technology project. A project that implements information technologies for a clear and stated benefit (Bennington & Baccarini, 2004).

Information technology project success. A project completed within deadline, budget, and specifications, as stated by the users, resulting in organizational improvement (Nelson, 2005).

Organizational change management. The various processes and policies intended to manage an organization through a period of change toward a specific objective and without interrupting efficiency and effectiveness.

Program. "A group of related projects managed in a coordinated way to obtain benefits and control not available from managing them individually. Programs may include elements of related work outside of the scope of the discrete projects in the program" (PMI, 2013, p. 579).

Program management. "The centralized management of a program to achieve the program's strategic objectives and benefits" (PMI, 2013, p. 579).

Project. "A temporary endeavor undertaken to create a unique product, service, or result" (PMI, 2013, p. 579).

Project life cycle (PLC). "The accumulated collection of project phases identified in sequence" (PMI, 2013, p. 580).

Project management. "The application of knowledge, skills, tools, and techniques to project activities to meet the project requirements, objectives, and completion" (PMI, 2013, p. 580).

Project Management Body of Knowledge *(*PMBOK®*).* "The total knowledge in the project management profession, including proven practices identified in published and unpublished materials yet which is constantly evolving" (PMI, 2013, p. 580).

Project management information system (PMIS). "An information system that includes techniques and tools to organize and distribute project processes and outputs, including automated and manual systems" (PMI, 2013, p. 580).

Project Management Institute (PMI). International association for project management, provides standards, publication and credential (PMI, 2013).

Project management lifecycle. The accumulated collection of project phases identified in sequence (PMI, 2013).

Project Management Office (PMO).

> An organizational body or entity assigned various responsibilities related to centralized and coordinated management of those projects under its domain. The responsibilities of a PMO can range from providing project management support functions to actually being responsible for the direct management of a project. (PMI, 2013, p. 580)

Project management professional (PMP). A person certified as a PMP by the Project Management Institute (Schwalbe, 2010).

Project manager (PM). "The person assigned by the performing organization to achieve the project objectives" (PMI, 2013, p. 581).

Project schedule. "A detailed specification of the individual actions and steps for project implementation, including dates required for activities that achieve milestones of the project" (PMI, 2013, p. 581).

Project success. This is determined by the reports by project managers of their perceptions of success.

Quality. "The degree to which a set of inherent characteristics of a product or process meets the needs of a customer or user" (PMI, 2013, p. 581).

Stakeholders. Those involved or affected by the project (Schwalbe, 2010).

Systems development life cycle (SDLC). The traditional methodology used to develop, maintain, and replace information systems (Hoffer, George, & Valacich, 2011).

Triple constraint. Project scope, time, and cost used as criteria for managing and evaluating projects (Schwalbe, 2010).

User involvement. Involvement of users during the project to gain input, clarification, and test acceptance.

Assumptions and Limitations

Assumptions

The positivist research tradition is the basis of the methodological study approach for this study as the relationship between CSF, project methodology, and project success were measured. The researcher utilized the positivist research assumptions of cross-sectional, nonexperimental, quantitative survey studies using a descriptive and correlational research design (Nardi, 2005). These assumptions are based on reality-based ontology, including the view that observed elements to be studied are measurable, fixed, and representative of one defined reality (Neuman, 2011). This design also included epistemological assumptions that define knowledge as objective, quantifiable, and achievable through testing "causal relationships" (Creswell, 2009, p. 18) between the variables and theories (Creswell, 2009; Creswell & Plano Clark, 2011).

The assumptions for this research study were in alignment with the study design. The study research design was to investigate differences and qualify the relationships of involved variables. (Creswell & Plano Clark, 2011). Two assumptions were considered that guided this correlational, nonexperimental, quantitative research study. First, the theoretical framework for this study asserted that conditions for communication, user involvement, and quality were tested for project success. There are two assumptions that guided this quantitative, nonexperimental, correlational study. First, the theoretical framework for this study asserted that conditions for communication, user involvement, and quality were tested for project success. The second assumption was that more research study was required to measure other critical success factors, such as schedule, cost, and scope (Atkinson, 1999).

Researchers have noted the potential for respondent bias simply as a result of negative attitudes about taking surveys (Swanson & Holton, 2005). This might be particularly true of surveys administered to professionals whose time and energy are limited. Thus, an acceptable time for taking the survey was reported. In addition, the study is based on the use of survey methodology with a relatively large sample, allowing the results to be generalizable to the target population. The study also assumed the field test would provide evidence of the measures' validity and reliability, along with the relevance of constructs, as assessed by a small group of subject matter experts.

Another assumption was that the study's online survey instrument would provide obscurity to participants, the ability to conduct many surveys over the study timeframe, and could provide the short period of time needed for data collection (Cooper & Schindler, 2011). The study also assumed research using a critical factors approach can help managers of software development and IT organizations assess which project management methodology is applicable for their organization.

Limitations

Among the limitations of this study, a possible bias in the results may exist due to the limited outreach in the survey to project team members in the United States. The English language was used in preparing and using the survey; therefore, only individuals in English-speaking, Western, or developed countries where the English language is common were solicited. However, it should be noted that English has broad enough usage globally to provide some indications of the use of CSFs in IT projects.

Another limitation of this study is that the data came from preselected organizations, in this case the Audience database available through SurveyMonkey and a local west coast project management association. There are potential problems with this research plan due to threats of internal validity as noted in the discussion of convenience sampling (Gray, Williamson, & Karp, 2008). In addition, this survey was based upon self-reported use of different project management methodologies on an individual IT project. The project managers in the study selected the project to report on and it can be anticipated that some participants may have tended to report on projects that were successful, whereas unsuccessful projects may purposely have been ignored. Other functions of an organization may have also influenced the participant's use of different methodologies, therefore impacting success of the project, including the organizational culture or the maturity and size of the organization. Since this study was limited to a single industry, that being information technology and all study data were cross-sectional; the external validity of the findings might be limited.

Conceptual Framework of the Study

This study constructs a conceptual framework that involves the relationship among project management methodologies, their various features, and project success. This research concentrated on and narrowed itself to selected variables and quantitative methodology, ideal for this objective.

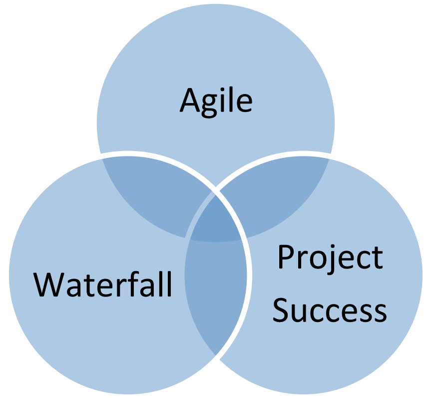


Figure 3. Conceptual framework.

Research Study Model

Figure 2 illustrates the conceptual framework for this research study and is based on the Project Management Institute's definition of project framework for project success in relation to critical success factors. The research survey instrument incorporated two independent variables, four dependent variables, and three independent covariates using analysis of variance (ANOVA) and multiple regression. The variables, including dependent, covariate, and independent, were collected from self-reports. Variables were mapped to the survey data, as outlined in Table 1.

Potential casual or correlational relationships can present and clarify the nature of relationships between variables. This study identified multiple correlations involving nine independent variables (Cooper & Schindler, 2011). This correlational study attempted to answer the central question: Based on reports of project managers who have used the agile or waterfall methodology, is there an association between CSFs (communication, user involvement, and quality) and project success, and how do these associations differ for each methodology?

Table 1. *Types of Variables*

Measure	Data type	Variable type
Agile method	Item-nominal	Independent
Waterfall method	Item-nominal	Independent
User involvement	Scale-interval	Independent/dependent
Quality	Scale-interval	Independent/dependent
Communication	Scale-interval	Independent/dependent
Project success	Scale-interval	Independent/dependent
Cost	Item-nominal	Independent covariate
Scope	Item-nominal	Independent covariate
Timeline	Item-nominal	Independent covariate

Table 2. *Use of Variables as Independent or Dependent Variables*

Hypothesis	Dependent variable & level of measurement	Independent variable & level of measurement
1	Communication	Agile vs. waterfall
2	User involvement	Agile vs. waterfall
3	Quality	Agile vs. waterfall
4	Project success	Communication (agile)
5	Project success	User involvement (agile)
6	Project success	Quality (agile)
7	Project success	Communication (waterfall)
8	Project success	User involvement (waterfall)
9	Project success	Quality (waterfall)

This research used a quantitative, nonexperimental, descriptive research design and sought to understand the role that the identified variables play in the delivery of successful waterfall or agile projects in information technology. A multiple, correlational relationship between two variables relates one variable with the others in a synchronized way and does not interpret whether one variable causes the other two.

A multiple regression relationship refers to identifying the relationship between two variables (Cooper & Schindler, 2011). Correlational method was the most appropriate for this study's data analysis (Swanson & Holton, 2005). In addition, multiple regression analysis was used to identify the relationship between a single dependent variable and an independent variable, while holding other independent variables constant (Cooper & Schindler, 2011).

Organization of the Remainder of the Study

The reminder of this manuscript provides a review of the literature regarding agile and waterfall methodologies and project success. In addition, Chapter 2 includes an exploration of gaps in the literature related to the interaction between waterfall and agile, with an emphasis on the relationship of project management processes implemented during the project life cycle and their contribution to project success.

Chapter 3 outlines the model variables, survey instrument content and construction, and other details related to the research methodology used in this study. Chapter 4 presents the results of the study and an analysis of the results, including issues related to validity and reliability testing of the instrument. Chapter 4 includes a presentation of interpretation of the results, with the conclusions based on the findings in the data collected. The chapter also will suggest recommendations for future research related to the topic.

CHAPTER 2. LITERATURE REVIEW

This chapter reviews the literature examining software development project methods, specifically agile and related CSFs. The first section of this review examines the various software development process models, from past to present, including prototyping, spiral, and waterfall. Next, the literature review includes an examination of the agile model, particularly as used by software development projects. Lastly, the chapter examines the literature related to critical success factors and their relationship to success of agile projects.

Introduction

This chapter examines only a portion of the literature related to project management methodologies, particularly agile and waterfall, in software development. A search of scholarly databases using terms related to project management methodologies and critical success factors yielded thousands of research articles and studies. This literature review is only intended to examine the history of project methodologies, particularly as they relate to software and information technology projects, the emergence of agile and waterfall as the preferred methodologies used on projects, and the identification of critical success factors.

Literature Review Research Approach

The articles and studies for this literature review came from three sources: peer-reviewed journal articles based on theory and research findings, books, and professional standards. Association standards were retrieved primarily from association websites. Journal articles were retrieved from scholarly databases, such as EBSCOhost.

Project Life Cycle

One of the seminal works in project management describes project schedules, scopes, plans, and facilities, but is notable for its lack of any reference to a project lifecycle (Baumgartner, 1963). The first reference to the project lifecycle emerged in Stuckenbruck (1981), which established four phases of a project. These phases provide the foundation to develop and evaluate the project.

Other seminal works related to the project lifecycle expanded on these phases, such as Kerzner's (1982) five phases. The author notes the absence of industry practices that directly correlate to specific phases. Rather, the differences over the phases of the lifecycle relate to scholarly discussions. Cleland and King (1983) noted the range of three to six phases for the lifecycle in the scholarly literature, with agreement only on the general purposes of the lifecycle, which is to organize stakeholders in a discussion about management responsibilities, deliverables, and actions.

Systems engineering has also contributed to the scholarly exploration of the project lifecycle (United States Air Force, Department of Defense, 1969). This seminal work noted the necessity of identifying a lifecycle for defense projects, which were large-scale and complex activities. The primary emphasis of this focus on lifecycle was cost concerns. More recently, a web-based emphasis on lifecycle has become evident throughout the industry, profession, and in scholarly literature. The International Council on Systems Engineering (2010) noted the use of five common lifecycles, continuing the emphasis on large-scale and complex government and military projects, with three to nine phases identified.

The PMI's first reference to lifecycle occurred in the 1983 publication by Adams and Lovedahl, though significant discussion did not occur until Wideman (1986) discussed the lifecycle in the prelude to the 1987 *Project Management Body of Knowledge* (*PMBOK*). PMI compared and contrasted lifecycles at the level of organizations and products. A four phase sequence was identified as consisting of concept, development, implementation, and termination (Wideman, 1986).

Today, the project lifecycle is broadly discussed; however, there is still debate in the literature about its relevance to project management. Harpum (2004) warned of limitations that might occur as the lifecycle fails to account for unpredicted issues outside the normal phases that are a part of process. The project manager must pay attention to the design phase, since design constraints are a threat. Harpum emphasized a more detailed concept of the lifecycle that imagines cycles within the primary cycle, particularly in the design phase.

The *PMBOK Guide* (PMI, 2013) emphasized a four-phase lifecycle. This begins with project start, organization and preparation, project work, and the project close. These four generic phases are presented only as a baseline, although there may be other phases associated with each of them. The number and nature of the cycles depends on the environment, industry, and organization. Even though this framework is relatively simple, it is beneficial to facilitate project comparisons and communications among various stakeholders, particularly those at the senior level.

The project start includes initiation, which involves documenting a business, case, conducting a feasibility study, establishing terms of reference, gathering the team, and establishing facilities (PMI, 2013). Planning involves organization and preparation, which includes creating a roadmap through a set of project, resource, financial, acceptance, quality, and communications plans. Execution includes the project work, which builds the deliverables and controls delivery, scope, quality, costs, risks, and other issues. Lastly, the project closure requires the release of staff, transfer of deliverables, and evaluating the implementation in a formal review.

Project Management Methodologies and Processes

The PMI (2013) noted the necessity of using a specific project management methodology. Indeed, a common cause of project failure is the fact that a coherent methodology was never employed or the wrong methodology was applied. “One fact that stands out clearly is that reasons for failure can be totally attributed to project management methodologies” (Al-Ahmad et al., 2009, p. 95). Johnson (2006) and Milosevic (2003) supported the implementation of a singular management methodology that is integrated into project and business needs.

Kendrick (2009) emphasized the importance of accounting for and integrating human elements into the methodology. For example, the methodology must address the social environment of the project and the resulting conflict, negotiation, and teamwork that is inherent to this environment. Shenhar and Dvir (2007) and Kendrick and conceived of the ideal project management methodology as one that is suited to the unique needs and qualities of the organization. PMI emphasized the necessity of methodology to be specific to the industry.

PMI (2013) defined project management as a discipline that involves controlling, motivating, organizing, and planning resources for the purpose of meeting a stated and coherent goal. Methodologies can ensure the proper management of project and resources during the project life cycle. The goal of project management is not just the literal project objective but also the larger goal of achieving the project objective in a timely, effective, and efficient manner, delivering beneficial outcomes for all stakeholders. Unique skills and strategies are necessary, with the proper methodology facilitating that both goals and objectives are met (see Figure 4).

A review of the literature surveying various methodologies such as Prince2, Six Sigma, spiral model, extreme programming, crystal, feature-driven development (FDD), RUP, and RAD, indicates that agile and waterfall are the most commonly used methodologies in software development and information technology. Therefore, agile and waterfall were the selected methodologies for this research project. Most of the project methodology in software programming has moved from waterfall to agile.

However, there are various other methodologies intended to assist project managers in realizing project goals, such as scope, schedule, and budget. Specific methodologies (used interchangeably with model) are designed according to the unique project needs in planning, design, implementation, and evaluation of project goals (Charavat, 2003). In most cases, project models are designed to fit the needs at the industry level, such as agile in software development.

Figure 4. PMI project management methodology.

The goal is to select the ideal model given the project's qualities. Even in software development, there might be other methodologies besides the preferred agile or waterfall that are better suited to achieve such alignment. In addition, it might be possible to select some aspects of a certain methodology that have benefits on their own merits and add them to a different model. The goal should be to recognize models as neither inferior nor superior but rather as offering unique ways to solve project problems.

Prince2

Prince2 was created by the United Kingdom for use in government projects in 1996 (Hedeman & Seegers, 2009). This model is process-driven, creating stages with processes and plans as guidelines. Prince2 requires accreditation not necessarily due to the complexity, specificity, and sophistication of its eight levels of processes, but rather because of the interest of the government of the United Kingdom in ensuring uniform project management activities. Prince2 is also focused on the issues of project management developed from best practice evaluations of successful government projects in the United Kingdom. In addition, Prince2 has also developed best practices from private sector case studies.

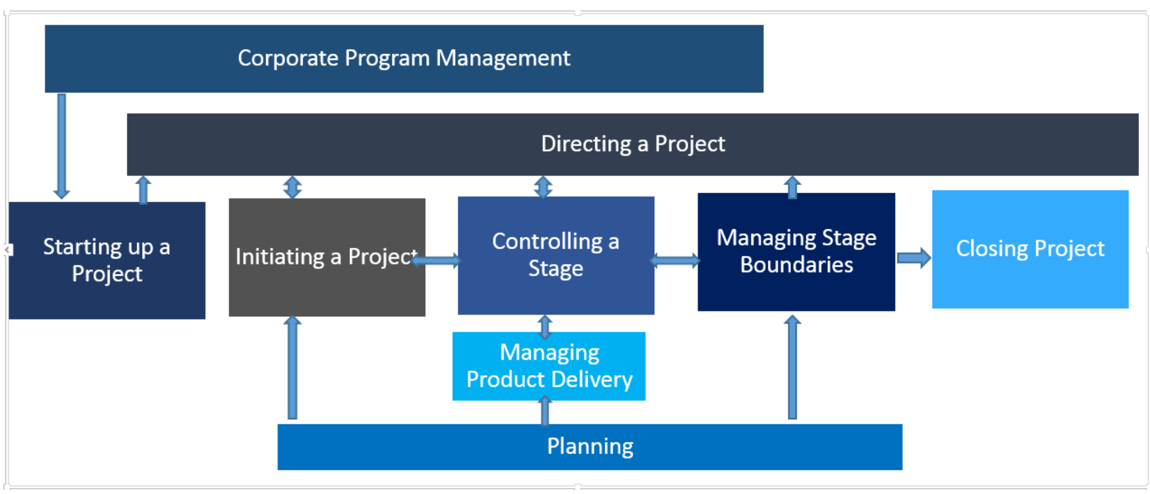


Figure 5. Prince2 process.

Prince2 is intended to be a generic model that can be applied to most government projects regardless of their differences (Hedeman & Seegers, 2009). "Because Prince2 is generic and based on proven principles, organizations can easily adopt the method as a standard and substantially improve their organizational capability to perform projects and deliver change" (Hedeman & Seegers, 2009, p. 7). Thus, there might be aspects of Prince2 applicable to software development programming as well.

Prince2 approaches project management from the perspective of Project Environment, Prince2 Processes, Prince2 Themes, and Prince2 Principles (Hedeman & Seegers, 2009). Each of these Prince2 perspectives is further broken down into seven domains. The seven principles include continued business justification, learning from experience, defined roles and responsibilities, managing by stages, managing by exception, focusing on products, and tailoring the process to suit the environment. The seven themes include organization, business case, plans, quality, risk, progress, and change. The seven processes include starting up a project (which should be differentiated from starting the project), initiating a project, directing a project, controlling a stage, managing stage boundaries, managing product delivery, and closing a project.

In many ways, Prince2 provides a generic framework for project management that reflects the goal of the United Kingdom in ensuring uniformity among its project managers (Hedeman & Seegers, 2009). Indeed, the Project Management Institute considers Prince2 compatible with the Project Management Professional (PMP) credential provided for project managers in the United States. Prince2 outlines the fundamentals of project management essentials to most methodologies.

There are several key goals of Prince2 (Hedeman & Seegers, 2009). First, the methodology seeks to ensure the efficient use of resources by establishing controls that monitor their use and organization. Secondly, the process attempts to provide a common language that all project participants can understand in achieving further efficiencies. Lastly, Prince2 seeks to specifically define various management roles and responsibilities in a manner that ensures both efficiency and broad application.

However, there are some disadvantages to the Prince2 methodology, particularly its focus on process-driven design (Hedeman & Seegers, 2009). For example, the methodology might be unsuitable for projects that are smaller and require constant adaptation in response to a changing environment. In addition, the high level of documentation required might defeat goals of efficiency in the private sector. In response to these criticisms, Prince2's accreditation organization in the United Kingdom emphasizes the fact that the methodology can be scaled down to fit the needs of specific organizations and industries if applied outside of the government environment.

Six Sigma

Six Sigma is the most popular type of management methodology for many business professionals. Motorola initiated the training program in 1986 as a statistically based method to reduce error in electronic manufacturing processes (Pyzdek, 2000). More currently, Six Sigma is referred to by BusinessBalls.com as a business performance methodology that applies to a variety of fields, including prisons, hospitals, government departments, and banks.

There are some ideas central to the Six Sigma method. For one, teams and team leaders are essential to the correct functioning of the methodology (Pyzdek, 2000).

Therefore, teams and team leaders are responsible for implementing the procedures of Six Sigma. Accordingly, these people need to be trained in Six Sigma's techniques, which include the use of the measurement and improvement tools, excellent communications, and relationship skills in order to include and meet the needs of customers and suppliers. Training is, therefore, a crucial element of Six Sigma (Pyzdek, 2000).

Six Sigma's success has led to the formalization of a Six Sigma culture. For instance, some people are referred to as black belts and others green belts to denote various degrees of expertise and knowledge and different responsibilities for putting Six Sigma into practice (Pyzdek, 2000). Six Sigma team leaders, for example, are called black belts, and Six Sigma teams use a variety of tools at each stage of the Six Sigma implementation process to manage people, teams, and communications processes, as well as measure, define, explain, analyze, and control change in process quality.

There is a significant amount of expertise required to implement Six Sigma (Pyzdek, 2000). When an organization decides to use this process, an executive team meets to decide on the strategy, often called an *improvement initiative*. This strategy should focus on the necessary processes required to meet the client's hopes. The executive team or black belts who are responsible for these processes are responsible for doing the following in order:

1. Identifying and comprehending these processes in detail.
2. Understanding the level of internal and external quality customers expect.
3. Finally, the performance of each stage of the Six Sigma process is assessed and measured by the number of defects per million operations.

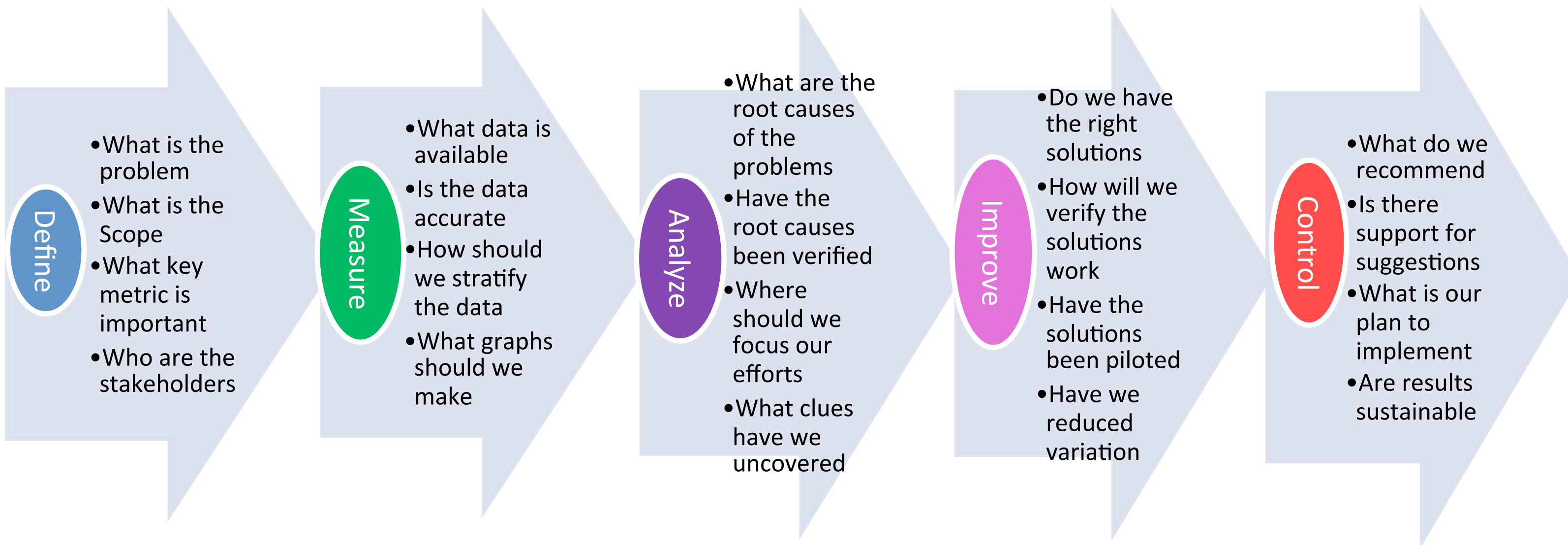


Figure 6. Six Sigma process model.

Most practitioners refer to the acronym DMAIC to describe Six Sigma:

- D-Define opportunity
- M-Measure performance
- A-Analyze opportunity
- I-Improve performance
- C-Control performance

Six Sigma can be thought of in a mathematical context, as a way to statistically define error by the number of defects per million operations (Pyzdek, 2000). For instance, it is thought that most ordinary businesses perform at a level of between two and three in sigma performance, which translate to 66,800 and 308,500 defects per million operations (DPMO), though this is considered to be an unmanageable level of performance. A measurement of four-sigma is equal to about 6,200 DPMO, or around 99.4% perfection, which is considered good enough for most ordinary business operations, such as a restaurant, but not good enough for a business such as an airline measuring passenger aircraft maintenance. An operation can be a manufacturing process or a process needed to understand a client's satisfaction, such as the process of dealing with a customer's complaint. As such, Six Sigma can be used in all service-related businesses and activities.

Six Sigma is used as a data driven approach in information technology, such as emphasizing quality of software from the beginning of an IT project. Most often the quality process is introduced at the end of the project cycle. Six Sigma is broadly used in software development project stages of concept and requirement gathering (Pyzdek, 2000).

Spiral

The spiral model was introduced because of flaws in the waterfall model and original models being used in software engineering. The spiral model is a combination of waterfall and prototype models. Like the name infers, the activities of software development proceed in a spiral manner. The entire software development process is broken down into small projects. The phases of the spiral model are as follows:

1. Planning phase
2. Risk analysis phase
3. Engineering phase
4. Coding and implementation phase
5. Evaluation phase (Satalkar, 2011).

The advantage of the spiral model is the disadvantage of the waterfall model in that changes to the software program can be done at any phase. The spiral model is considered a realistic model and is commonly used when developing large software applications. The spiral model employs a methodical approach and is integrated together through an iterative framework (see Figure 7), a process that helps ensure there are no problems in the resultant delivered software.

The disadvantages of the spiral model include the need to have a skilled expert in risk assessment on the team. The software can be a complete disaster and unusable if there is not correct risk assessment done during the project. As such, the client may have to spend a considerable amount of time with the software development team correcting problems that have emerged with the software. When the customer is too involved in this process, they may ignore certain risks and damage the product.

In the waterfall model, the customer is not involved in the software development process, commonly creating situations where the software is not being created according to the requirements of the customer. In the spiral process, the customer is involved in the software development process from the very beginning and is told all that is happening during development, ensuring quality is present and requirements are met.

In addition, once a phase is completed using the waterfall method, there is no way to go back and make changes. This often creates problems, especially during the coding phase. It is often seen that the software design is doable on paper, while in the implementation phase it may be difficult to code correctly (Harris, 2006). On the other hand, in the spiral model, design changes are much easier to implement based on the iterative delivery process. In the spiral model, the team can look back on the different phases of development as often as necessary during the entire project. This helps in revising and changing the developing code when necessary, not as feasible in the waterfall model.

Because the spiral model appears to be an intricate model, with much iteration, people commonly confuse the two. Unlike the waterfall model and its volumes of documentation, the spiral model has no documentation, simplifying some aspects of the spiral method. Some things to consider when deciding which model to use include the size and urgency of the project. Similarly, the available resources will have an important role in the software development process and approach.

The spiral model is relatively new model of software development and is designed to eliminate the problems of the waterfall model. The spiral model is also called Boehm's model. Spiral model includes four phases, each iteratively following one another in order to get rid of all potential problems that the waterfall model presents. Iterating or repeating stages can solve problems associated with any given phase. This also simplifies planning strategies. The four phases of the spiral model are as follows: Planning Phase, Risk analysis Phase, Engineering Phase, Evaluation Phase.

1. Planning Phase (requirement gathering)—In planning phase, the goals, changes, and limitations of the project are decided and documented. The goals and other specifications are fixed in order to determine which strategies are best to follow during the project's life cycle.
2. Risk analysis Phase—This phase is focused on identify possible project risk and evaluate and determine the possible alternative solutions.
3. Engineering Phase—Executing development plan phase iteratively, code and build the software.
4. Evaluation Phase—User involved in testing the software and providing suggestion and feedback.

More complete software is created as it moves iteratively through the four phases. The first phase is the most important as all possible risks, constraints, and requirements are determined. In the following iterations, all known strategies are decided upon and used to build the total software system. The major dimensions show a moving forward of the product towards a complete system (Agarwal, Tayal, & Gupta, 2010).

Nevertheless, the spiral model has its disadvantages. As this model was created to overcome and compensate for problems with the waterfall model, additional expenses in time and labor are incurred, as specialists in the area of planning, risk assessment, mitigation, development, and customer relations, to name a few, are required when using this model. In addition, more time is required due to the iterations, often repeated more than once (James, 2008).

The model recognizes the iterative aspect of creating prototypes and thus embraces an evolutionary design approach (Agarwal et al., 2010). At the same time, the model attempts to achieve traditional controls with the integration of the linear methodologies. The spiral methodology is conceived of as a metamodel for its capacity to accommodate process-development models. The methodology is intended to be used as a reference for selecting other methodologies or a hybrid that borrows some elements from each.

This model is desirable when the project has certain advanced features. These could include a project that is creating a new, large, and complex system without the ability to examine best practices in similar projects, a project wherein the user cannot provide straightforward requirements, a project where there are multiple users, functions, features, applications, and platforms, or a project with high requirements for integration, interfacing, data migration, and replication.

The model is intended to examine costs relative to project progress (Agarwal et al., 2010). Four quadrants attempt to define certain aspects of the project approach for each phase. The first quadrant establishes goals of the phase and alternative solutions. The second quadrant evaluates alternatives in the context of goals and barriers. Risk is paramount in this evaluation. The third quadrant develops strategies to address risks and other uncertainties. These strategies can include benchmarking, prototyping, and simulation. The fourth quadrant determines the goals necessary in the next development cycle, paying close attention to the overall goal to construct a complete system.

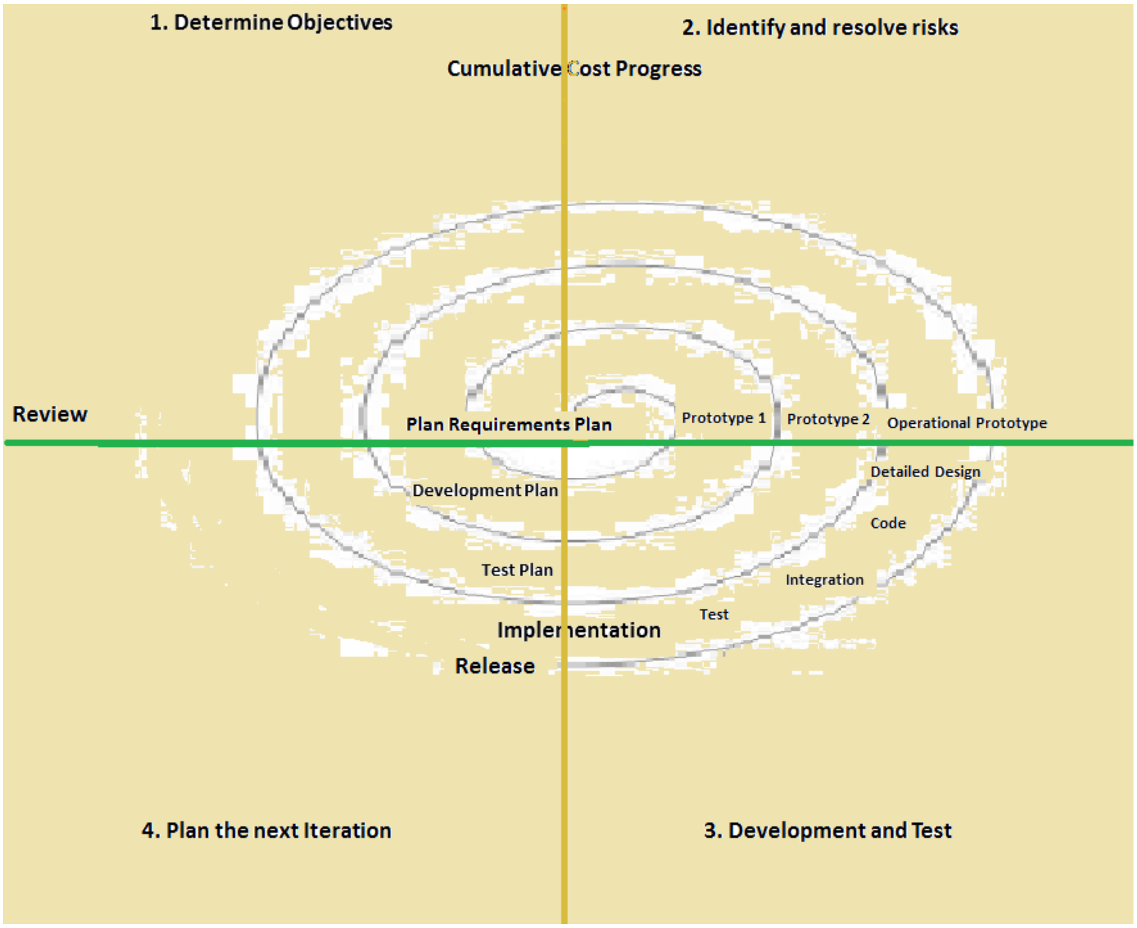


Figure 7. Spiral model.

As with all models, there are limitations, advantages, and disadvantages. Advantages of the spiral model include its ability to reduce risk, require fewer documentation requirements, and leveraging the use of prototyping and flexibility. Limitations include the lack of strict standards for software development and the lack of a linear or sequential ordering that includes a beginning, middle, and end. Disadvantages of the spiral model include its inability to be applied as a life cycle model. There is a deficiency of explicit process advisement in how goals, barriers, and alternatives are established. Risk assessment expertise is assumed rather than explicitly defined. Excessive flexibility might threaten efficiency for many projects (Agarwal et al., 2010).

Scrum

Scrum is a subcategory of agile methodology, specifically designed for software development projects (Cohn, 2009); however, does have application to product development outside the software industry, as well. Scrum avoids a linear or sequential approach to the project. Rather, the emphasis is on holistic, flexible strategizing within a team, referencing the rugby metaphor of a team of players constantly changing its individual and group dynamics around a common goal. Scrum was developed based on examination of best practices in a variety of organizations and industries that successfully brought products to market with quality and cost efficiency. Scrum is depicted in Figure 8.

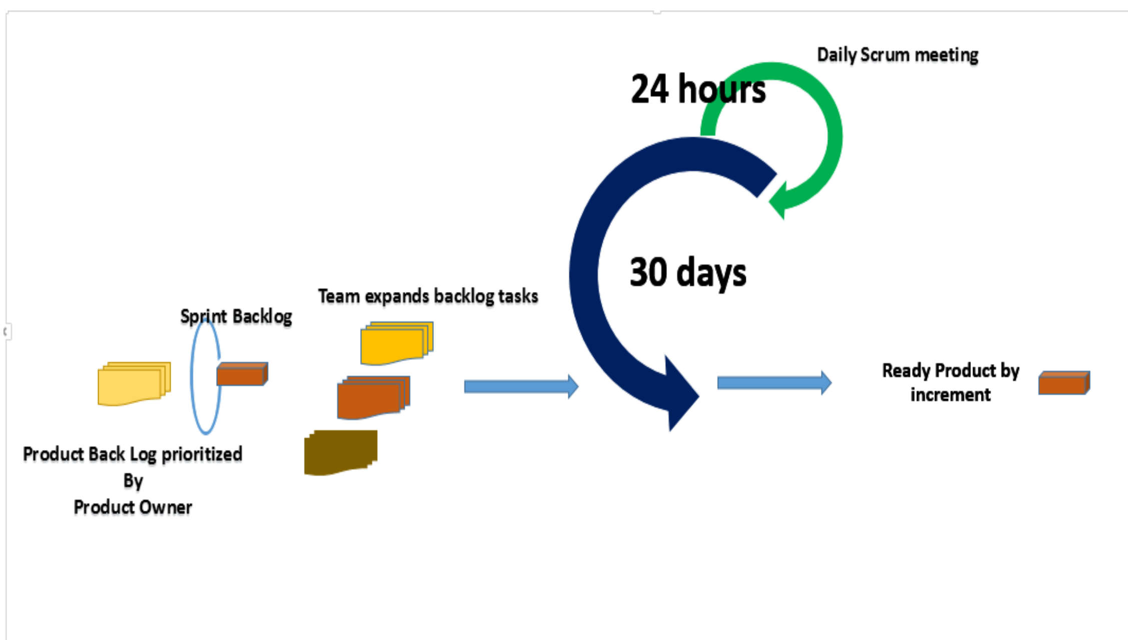


Figure 8. Scrum model.

The Scrum team consists of three core roles. The Product Owner is a single position that is responsible for voicing the customer's interests and goals in the project. The Product Owner should be conceived of as the representative of the primary stakeholders. The Product Owner prioritizes customer needs. Cohn (2009) emphasized the necessity of distinguishing the Product Owner from essential positions on the development team, such as the ScrumMaster.

The ScrumMaster essentially serves as a type of project manager, removing barriers to develop deliverables with quality and efficiency. However, the ScrumMaster is not the team leader. Rather, the ScrumMaster should be seen as an enforcer of rules and regulations, ensuring that the determined process is followed. Some theorists refer to the ScrumMaster as a servant-leader, noting the fact that two roles are played simultaneously. The key difference of project manager is that the ScrumMaster does not manage outside the specific roles noted earlier.

The other core role is the Development Team. The team has a variety of skills necessary to deliver the product or solution at the end of each Sprint, or project end. Most notably, the product development team under the Scrum approach is self-organizing. This is a relatively small group of typically three to nine individuals (Cohn, 2009).

Cohn (2009) noted that there are two other ancillary roles under the Scrum model. Stakeholders are the primary customers for the product. Notably, their involvement only occurs during Sprint reviews. Managers are those involved in establishing and managing controls within the project environment.

As noted, the Scrum process includes phases known as Sprints. The Sprint has specific timeframes, usually as short as a week or as long as a month. A planning meeting occurs before each Sprint, while an interim meeting is held to evaluate progress. The Sprint is organized by the product backlog established as the Product Owner prioritizes customer needs. The Product Owner is responsible for determining which needs are to be delivered after the specific Sprint. Once the backlog is delivered to the Development Team, that team itself has sole authority for development.

While this process sounds deliberate and sequential, in fact Scrum recognizes the ever-changing environment related to customer needs and project impediments. As a result, flexibility is built into the design. The Development Team's sole authority during the Sprint phase is intended to allow adaptability to changes (Cohn, 2009). This should be facilitated by the ScrumMaster's removal of impediments.

Rapid Application Development

The goals of RAD include timeliness and quality (Stair & Reynolds, 2009). A variety of software development tools have been developed for the RAD market, noting the particular competitive pressures of most software development projects. Most products utilizing rapid application development generate computer code specific to the project needs.

The goal of RAD is to minimize planning, since this activity adds costs that are not directly related to product development. Rather, rapid prototyping allows for timely writing of software, while necessary changes can be accommodated without significant and costly delays. Indeed, rapid application development emerged in response to the extensive planning of waterfall, which often resulted in projects being obsolete before delivery due to the rapidly changing user environment (Stair & Reynolds, 2009).

The four phases of rapid application development attempt to ensure timeliness and flexibility, while maintaining quality and efficiency (Stair & Reynolds, 2009):

- The Requirements Planning Phase integrates system planning with systems analysis. All stakeholders discuss needs, scope, barriers, and requirements.
- The User Design Phase involves interactions between users and system analysts to create models and prototypes necessary for inputs, outputs, and all system processes. However, this phase should be understood as a continuous process through the life of the project.
- The Construction Phase involves development tasks, while still utilizing the continued involvement of users. The user community constantly evaluates the project to ensure needs are met, making changes as necessary.
- The Cutover Phase involves testing, data conversion, changeover, and user training. These components occur at a much more rapid pace than under traditional waterfall methodology.

There are distinct variables that influence the success of rapid application development (Stair & Reynolds, 2009). These must be understood as either facilitators or barriers to timeliness and quality. There must be alignment between the project type and the specific rapid application development methodology or product utilized. There must be alignment between company culture and the flexible nature of rapid application development. For example, IBM avoids rapid application development for flagship projects, which follow more traditional methodologies such as waterfall, though refinements of the original product might use this methodology (Stair & Reynolds, 2009).

The disadvantages of rapid application development include its emphasis on iterative processes that might prevent development of an ideal new product (Stair & Reynolds, 2009). Rapid application development might prefer timeliness over a broader perspective that includes business goals and applications. In addition, the strong role of users might impede efficiency as micromanaging occurs.

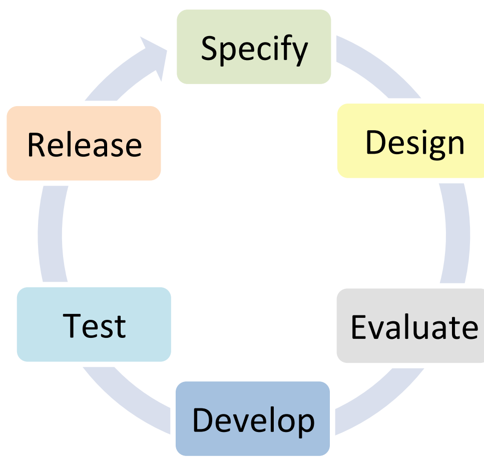


Figure 9. Rapid application development.

James Martin in 1991 first created a development process and practice of RAD (Petch & Reeve, 1999). RAD is a mostly used method in the development of high quality software. The goal of RAD is to create a total software solution in a time-effective manner.

RAD employs the best practices in gathering requirement, use of trained and skilled resources and effective management support to deliver high quality software product or service. The project life cycle can be summarized as follow:

- Software Requirement gathering
- Software Design, prototyping design verification
- Managing changes, integration of design changes
- Quick code review, testing, and build process.

The advantages of using RAD include the following:

1. Prototyping provides a visualization of the end product to customer. The customer has the opportunity to provide feedback and make changes.
2. Customer's change demands can easily applied as RAD process allows for quick updates by employing prototype methods.
3. Cost reduction of development effort and meeting the project schedule.

Crystal

The Crystal methodology avoids the process-driven emphases of most other methodologies (Cockburn, 2004). Rather, Crystal is used for small teams and emphasizes the interpersonal interactions that occur at this level, notably related to communications, psychology, and skills. Three factors are considered determinative of the methodology chosen: communications load, project priorities, and system criticality. Alistair Cockburn developed the Crystal family of methodologies with software development projects in mind by examining best practices.

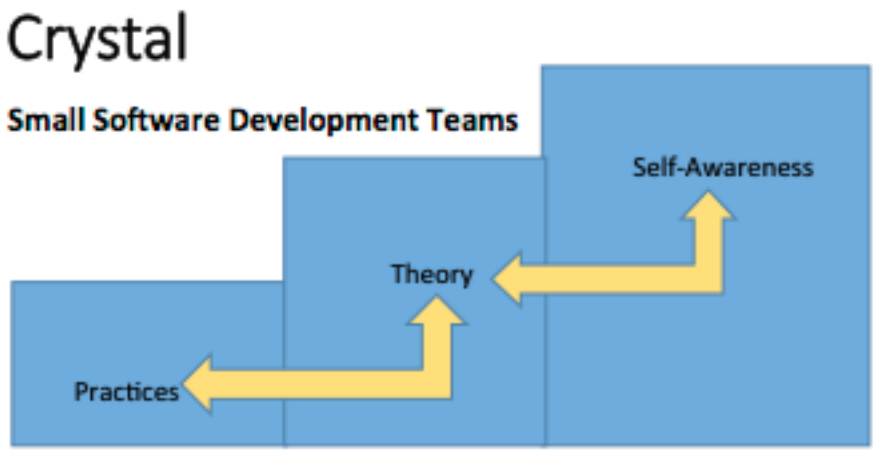


Figure 10. Crystal model.

Cockburn has identified seven properties of highly successful projects.

- *Frequent Delivery* requires critical feedback to sponsors on team progress, user opportunities to determine achievement of their requests, developer focus, breaking of deadlocks, debugging of development and deployment processes, and morale boosts from accomplishments.
- *Reflective Improvement* includes learning through mistakes in a formal evaluation process.
- *Osmotic Communication* involves constant, multidirectional flow of communication among all stakeholders.
- *Personal Safety* involves an environment free of fear of speaking one's mind about project progress.
- *Focus* involves clear objectives and the time to meet them.

- *Easy Access to Expert Users* includes up-to-date requirements, rapid feedback on design decisions, rapid feedback on product quality, and a place to deploy and test deliverables.
- *Technical Environment with Automated Tests*, *Configuration Management*, and *Frequent Interaction* emphasizes the importance of including automated rather than manual testing, configuration management that allows check in work asynchronously, and frequent interaction that ideally occurs at least once every other day and possibly several times a day.

Extreme Programming

The primary goal of Extreme Programming is to reduce the costs of requirement changes (*Extreme Programming Pocket Guide,* 2003). This methodology emphasizes customer input, programming in pairs, frequent testing, and communal ownership of code. Four activities provide the conceptual foundation for Extreme Programming.

- *Coding* reflects Extreme Programming's emphasis on the code. The goal is to create code that is clear and concise, avoiding potential for misinterpretation.
- *Testing* occurs at three levels. Unit tests involve automated tests intended to evaluate specific features. *Coding* is complete when testing does not break the code. Acceptance tests determine alignment of the requirements with the deliverable. System-wide integration testing determines overall stability.
- *Listening* involves both understanding what the customer wants and applying that to business realities. The *Planning Game* provides a process for facilitating proper listening between customer and programmer.

- *Designing* involves looking at the system as a whole. This will organize the various activities to avoid dependencies meaning changes in one activity will not impact other parts of the system.

In addition to these four basic principles, Extreme Programming also emphasizes five basic values. These include communication, simplicity, feedback, courage, and respect. The goal of communication is to rapidly distribute information coherently and accurately. The goal of simplicity is recognize the benefit of simple solutions that can incrementally add complexity as needed. The goal of feedback is to recognize different dimensions within the system, including feedback from the customer, the team, and the system. The goal of courage is to encourage programmers to take initiatives to make decisions facilitative of accuracy and efficiency (Lindstrom & Jeffries, 2004). The goal of respect is to encourage a collective solidarity rather than narrow self-interest.

Criticisms and disadvantages of Extreme Programming include the incremental nature of receiving requirements, the lack of a holistic up-front design, pairing of developers, attachment of a customer representative to the project, and the interdependency of Extreme Programming's elements. All of these aspects might result in a lack of focus and central management for projects under this methodology. Projects might have a tendency to drift aimlessly, significantly threatening efficiency and timeliness.

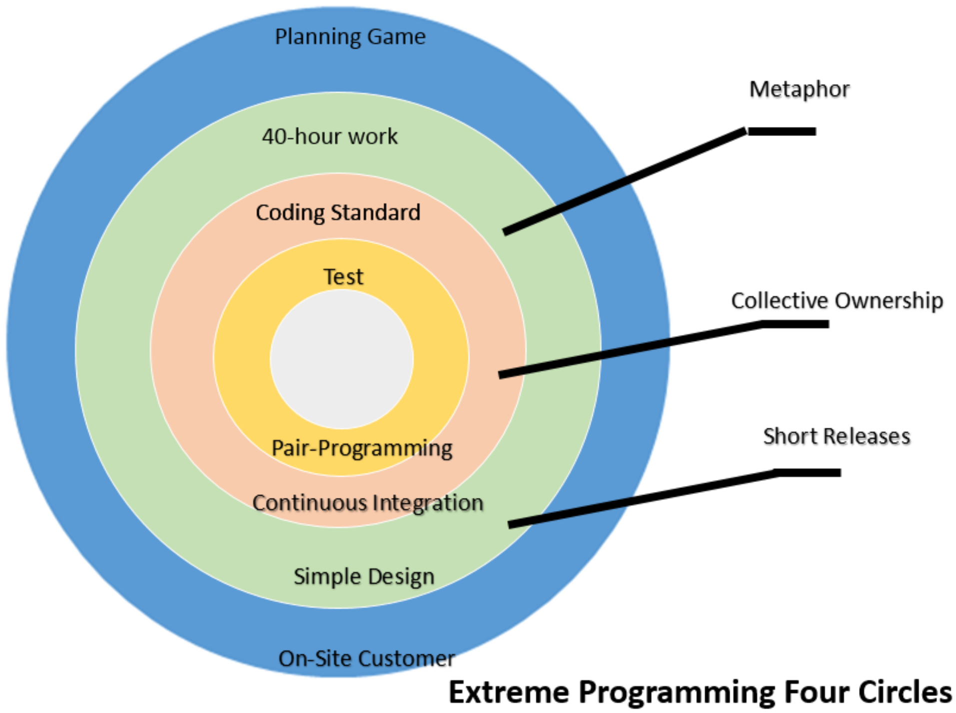


Figure 11. Extreme programming—four circles.

Rational Unified Process

The RUP attempts to unify the various software development methodologies, recognizing the best elements of each (Kruchten, 2004). Created by IBM as an iterative software development process, the methodology also emphasizes flexibility as teams and organizations determine what is best for their project. The rational unified process is constructed upon the building blocks of roles, work products, and tasks, or who, what, and how.

The four phases of the rational unified process recognize the life cycle of the project (Gibbs, 2006). Though there are similarities to the waterfall model, this methodology emphasizes the importance of recognizing development iterations existing within all the phases. A primary objective and milestone is to be achieved after each phase. Iterations include nine project development disciplines by IBM: business modeling, analysis and design, requirements, implementation, test, deployment, configuration and change, project management, and environment.

The four phases of RUP include the following:

- The Inception Phase involves scoping the system to determine costs. The project is designed from a business case perspective. This phase should develop stakeholder agreement, requirement understanding, cost and schedule credibility, depth of prototypes, and a baseline to compare actual versus budgeted expenditures.
- The Elaboration Phase is intended to reduce risk as the project becomes more refined and detailed. As a result of this phase, a use-case model should be developed that includes cases, actors, and descriptions. Software architecture and system development processes are described. The business case and risk list are revised. A development plan is completed. Prototypes are developed to reduce each technical risk identified.
- The Construction Phase creates the system, resulting in the first software release.
- The Transition Phase involves placing the system into production, resulting in release to the user community. Training occurs during this phase, as well as testing, and product release occurs if milestones are reached.

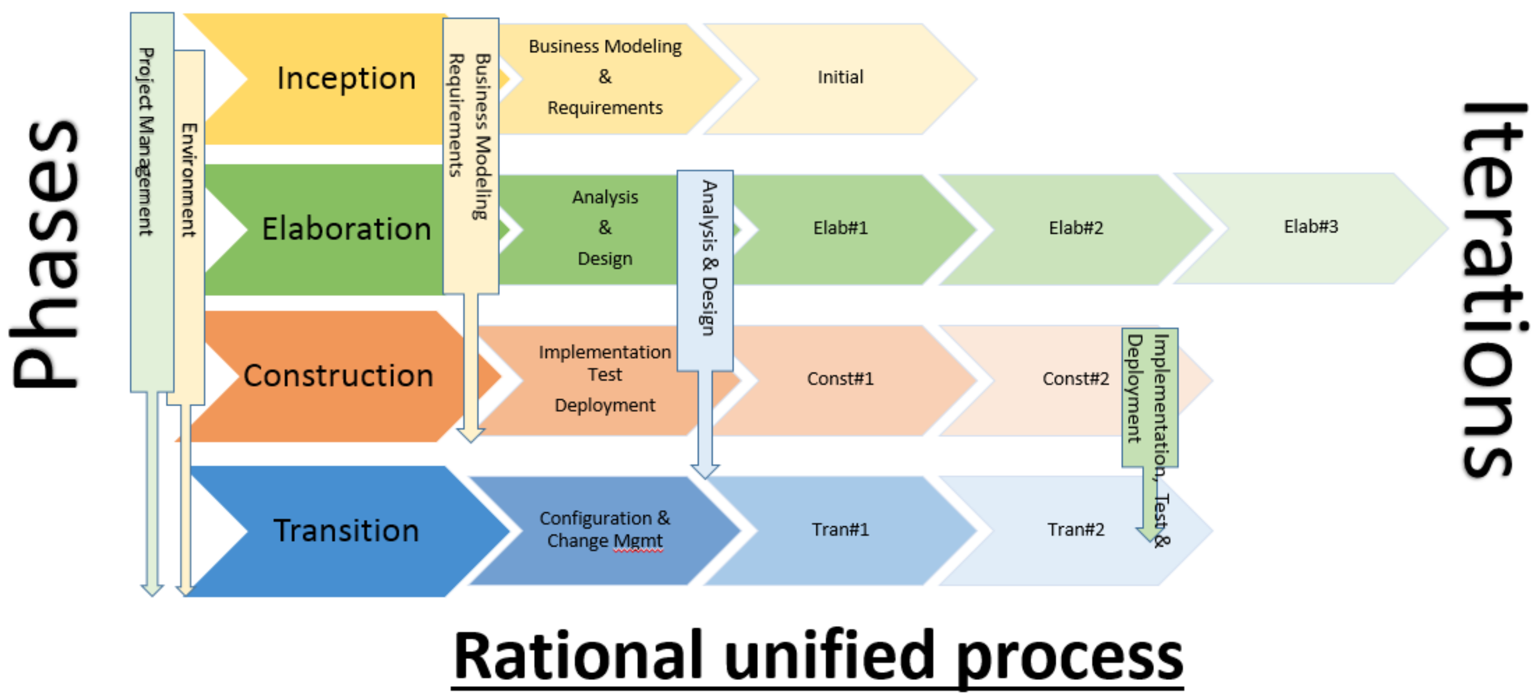


Figure 12. Rational unified process.

The rational unified process is the culmination of decades of experience in software and project development by IBM (Gibbs, 2006). These processes have been identified as iterative development of software, successful management of requirements, use of component-based architecture, visual modeling of software, continuous verification of software quality, and control of software changes. Of course, it should be noted that the rational unified process is an IBM product. Even though it outlines a basic software engineering process, the product is maintained and distributed by IBM. However, IBM touts rational unified process as flexible enough in its process and product forms to be applicable to all types of software development projects and organizations (Gibbs, 2006).

Waterfall

Software engineering includes software development as well as creating good software by using current knowledge with the assistance of methods and effective tools, such as the methods that make up the waterfall model. These different software approaches, or models, are also called Software Development Process models. They are used during the development course of a software program. In order to make sure that the course of the software development is successful, each process model follows a particular life cycle (James, 2008).

The waterfall method is one such model. The waterfall model was the first methodology developed for software development (*Waterfall Model,* 2013). Waterfall was formally introduced by Winston Royce as an idea in 1970 and termed a p*rocess model*. However, Royce ironically introduced it as an illustration of a flawed software development method that was weak because of its many inadequacies. Although he did not use the word *waterfall*, Royce constructed a process that contributed to the development of this metaphor. However, it should be noted that Royce also included nonsequential processes in his initial model that is attributed to the development of the waterfall model. Royce believed that any well-designed software model should allow for going back and forth between phases, which the standard waterfall model does not allow.

Nevertheless, many software companies have successfully implemented this model. The waterfall philosophy was taken from the hardware manufacturing and construction tactics used during the 1970s; hence it has a highly structured approach to software creation. It is called the waterfall model because the model develops downward, from one stage to another. As an example, after Stage I is complete, the process progresses to Stage II, and so on, with each stage being completed before moving on to the next. There is no way to turn back to a prior phase once the last one has been reached. There are five to seven stages that make up the waterfall model. The disparity in the number of stages is propagated by the numerous references to the process in literature today. Nevertheless, in the waterfall method (see Figure 13) the entire process of software development is divided into separate phases. These phases, described in order, include the following:

1. Requirement gathering
2. Software system design includes:
    a. High level basic design
    b. Technical detail design
3. Implementation and construction

4. Testing, deployment, and maintenance:
    a. Testing—Design verification, requirement tractability, design reliability and functionality
    b. Deployment— Deployment of tested software code into the production
    c. Maintenance—Maintenance of the system in production to ensure continues business operation.

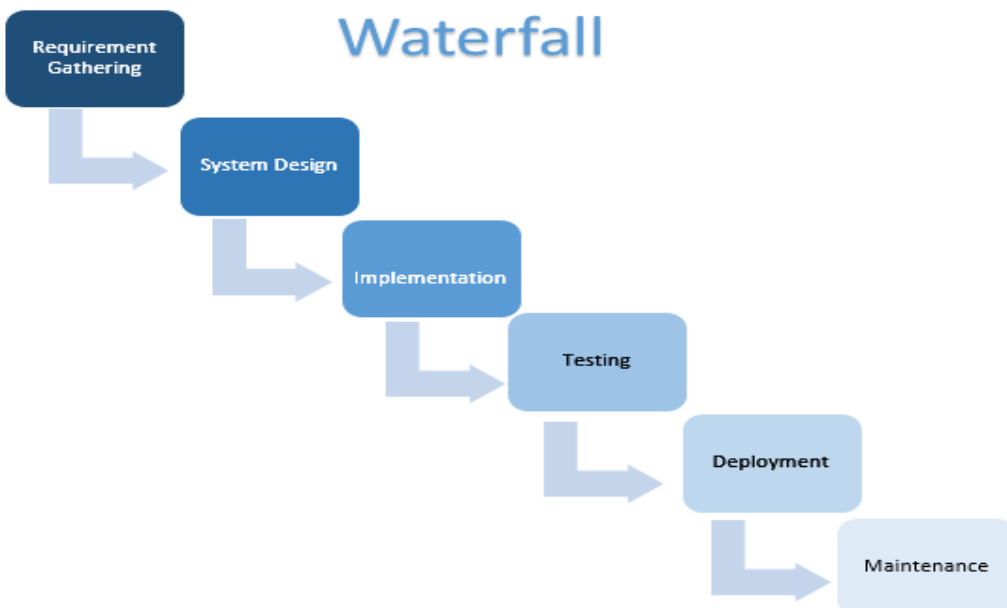


Figure 13. Waterfall process.

All of these phases are used in the order presented. That is, one only moves onto the next phase after a certain set of goals is reached, thus the name of the model. All the methods and processes used in the model can be seen more clearly (Winston, 1970).

The first phase, the requirement specifications phase, includes setting out all possible requirements of the process. Requirements are necessary system functionalities that are required for the end user to use the system in operation (Thomas & Fernandez, 2008). Software requirement specification is created outlining that is used as a guideline for the design phase (Morris, 2004).

The software system design phase is done after requirement gathering and before starting software coding. The system design details in identifying design of the system solution. The design specification document is an input to the construction and software-coding phase (Benington, 1983).

In the implementation and unit-testing phase, the work outlined in the documents from phase two is separated into different modules or units, and the actual coding of the software is begun. Before coding starts, the system is developed in small programs called units, which are incorporated in the following phase. Each small program, or unit, is created and then tested to make sure it works and serves the purpose it was developed for. This is termed unit testing. Unit testing basically confirms if the units meet their specified purposes (Benington, 1983).

After the system is separated into units, developed, and tested for their functions, the integration and system-testing phase begins where the units are incorporated into the overall, total system. The units are tested to make sure they all coordinate with each other and the total system operates per specifications. After the software is successfully tested, the system is given to the end user (Haughey, 2009).

The next phase, operations and maintenance, goes on indefinitely. Overall, problems with the system created that are not discovered during its development life cycle emerge after its use begins are solved after deployment of the system. This process is termed maintenance, since the problems with the system come up during its use and needs to be figured out as the system us used (James, 2008).

The advantages of the waterfall model include the following:

- Compartmentalization of the work involved through each phase makes it easier to set a specific time period for the tasks to be done.
- There is no overlapping of phases, as each phase must be completed before the next one can be started.
- Because the project requires that one phase be complete before moving on to the next, any errors in the software can be detected early and corrected.
- Because it is a linear progression, most managers prefer it. Due to its linear design, the associated costs are less than other models, which in turn can help reduce the cost of the overall project.
- Unlike the newer methods of software development, this method uses paperwork to document the different stages and development of the model. Therefore, it is easier for new workers entering the project to pick up where the previous worker left off.
- Testing is done after the development phase is completed, reducing the number of errors and maintaining the quality of the project (James, 2008).
- The waterfall model is a well-known model among software developers, so tends to be easier to use. It is also easier to create different types of software using this method in a short time period.

The disadvantages of the waterfall model include the following:

- As it is very important to collect all possible requirements during the first phase in order to correctly design the system, a problem exists when not all requirements are received at once. Typically, the requirements of the customer continue to be given after the first phase is completed. This has the potential to affect the system development process and its success.
- External factors can influence the project to a large degree. For instance, when/if a client changes the requirements of the project, you have to start from the beginning again, as the waterfall model does not allow for alteration of previous phases. Cost efficiency is, therefore, a negative facet of this model.
- The issues with one phase are never totally solved before the next one begins, often resulting in a system that is poorly structured.
- The project is not divided in flexible phases.
- As the customer continues to add requirements, not all of them are fulfilled, resulting in the development of a system that is potentially unusable. The cost of the system development increases when the requirements are implemented in the newer version of the system (Haughey, 2009).
- A large amount of time is potentially wasted on excessive documentation of the project.
- The testing of the software is untimely; that is, it occurs fairly late in the developmental process, therefore by the time errors are discovered, a lot of time and money might have already been wasted on the project.

Despite its disadvantages, the waterfall model is one of the most used software development processes, namely because the advantages outweigh the disadvantages in certain types of projects. In addition, if the team on the project does not have a lot of experience, this model perfectly serves its purpose (Haughey, 2009).

Agile

The agile model, like the waterfall model, is also a popular method used in the software development process. Waterfall model was developed in the 1970s; the agile model of software development was created during the 1990s when developers began to move away from highly traditional model with lack of ability to provide flexibility during software development lifecycle (Kruchten, 2004).

In 1974, Edmonds formally defined agile, also called the lightweight methods, in a research paper. Scrum, the adaptive software development and dynamic systems development methods was defined in 1995, Crystal Clear and Extreme Programming in 1996, and feature driven development. In 2001, a group of agile software developers declared the Agile Manifesto, a set of guidelines that were suggested as an all-encompassing framework for agile software development models (Lindstrom & Jeffries, 2004).

There are conceptual differences between the waterfall and agile models. The waterfall model works in a linear, sequential manner, moving from one phase to the next. The phases of the waterfall model move from requirement specification, analysis, design, coding, testing and installation, and finally maintenance. In this approach, the project team moves to the next phase of development only after the previous one has been completely finished. Software development companies that use this model spend a lot of time. The philosophy that underlies this model is that the amount of time spent on each phase of development corrects any problems in advance. Once the design phase is completed, it's implemented in the coding phase. Documentation of each phases are required for this model (Phatak, 2012).

The agile model, on the other hand, focuses on flexibility and adaptability during development. Instead of one very time-consuming, inflexible process of development, agile models involve several iterations, defined as a continuous repetition of an operation or procedure. Each iteration cycle goes through the steps of design, coding, and testing. agile design is often kept open to allow last-minute change and alterations (Lindstrom & Jeffries, 2004).

Teams work closely together and with the other teams that support the process. The design concept allows for the evolution of new concepts as they come along. The importance of documentation is not emphasized, with more focus on the speed of the delivery of the system. Customers are given demonstrations at the end of each iteration cycle and feedback from the reviewers helps decide the next course of program development. This iterative cycle keeps on going until the final product meets the specifications of the customer (Lindstrom & Jeffries, 2004).

The agile method has gained huge popularity and acceptance among software developers and IT managers since it was introduced in the 1990s. A Forrester survey (West, 2009) indicated that about 35% of its participants used the agile method. The agile method, despite its advantages such as cost savings for its consumers, does have its disadvantages. Agile processes are constructed on several key ideologies that are central to the methodology. Many companies hurry to implement agile methodology, ignoring the fundamental principles and technological and cultural aspects that are a part of implementing the process. The result is chaos and stress (Thillaisthanam, 2013).

One of the main principles of agile methodology is the idea of self-organizing teams, which presents a cultural challenge. Software engineering teams are enabled in unparalleled ways, lessening the role of project manager to one of facilitator. In cases of new implementation of the process, when project managers are given an agile project, they may use the same management style of command and control, decreasing and threatening the self-organizing ideology of the agile methodology (Augustine & Cuellar, 2006).

Software engineers who are used to project managers that exert strong leadership may be tested by the freedom that comes with agile. With agile, there is little documentation and planning, and once-a-day meetings operate well when there is sufficient teamwork and motivation. Thillaisthanam (2013) suggested agile is a frame of mind and every team member must assume a specific role to achieve success.

Agile methodology supports and encourages teamwork and collocation. Based on this assumption, then, is the idea that working software is more important than documenting the entire process (Northern, Matfield, Benito, & Casagni, 2010). The inferred assumption is that engineers associated with the project will communicate with each other as is necessary. Therefore, the underlying principle is informal communication, rather than formal meetings, documentation, and service level agreements.

It is also important to staff a project that suits the agile methodology. For instance, the agile model requires that all staff members be recruited from the beginning. This is different from the waterfall model in which staffing follows a normal curve, with staff entering and leaving projects at various points in the schedule. Many project managers who are used to using traditional methodologies in the past may undervalue the importance of correct staffing for the agile methodology, resulting in project delays.

Agile methodology is defined by goal completion. Agile demands that at every completed step, the product is shippable. This means that from the very first goal that is achieved, the customer has a usable product. However, this is another area that is misunderstood because many managers continue to say that features are complete, whose meaning in an agile respect is not evidently established.

The agile method suggests that within every sprint, or every step achieved, changes are well understood and created perfectly. The agile method does not say that change requirements can be done within a sprint. Instead, agile suggests that if there are serious changes within a sprint, it should be left and a new one started in its place (PMI, 2013). The basic flow of agile methodology is presented in Figure 14.

Mahanti (2006) has suggested a number of reasons that agile methodologies have attracted more attention in recent years. Some of the reasons for the interest in agile are: increased return on investment, early detection of any design problem that leads to higher quality, more project control and flexibility.

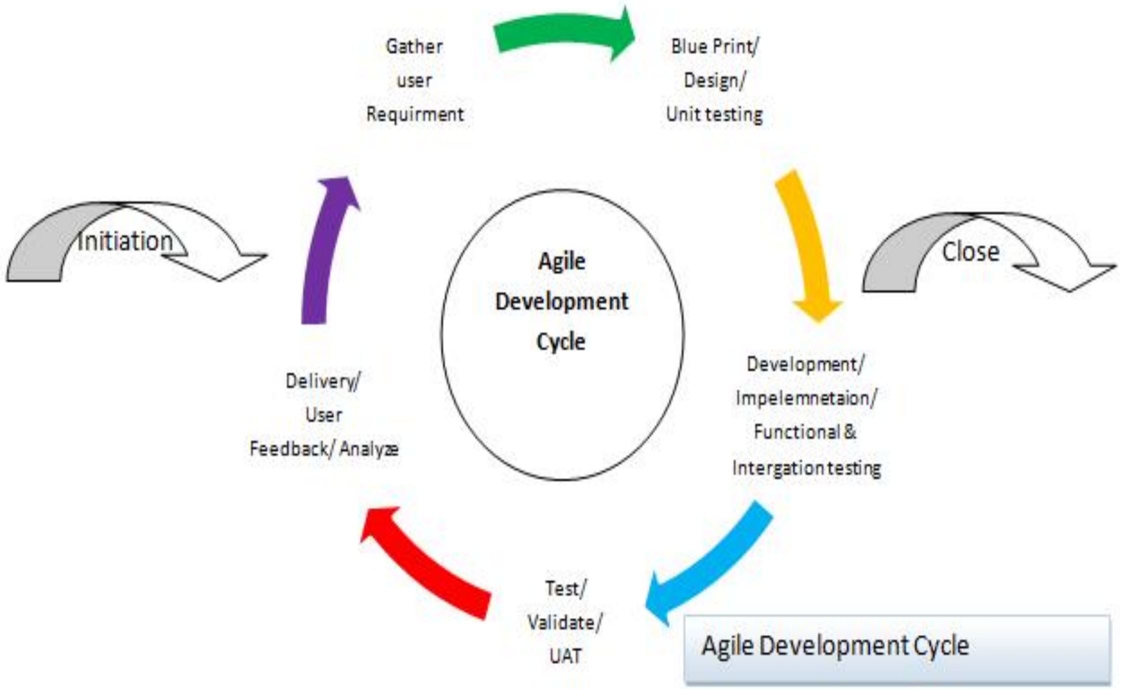


Figure 14. Agile process.

Project Success

Project success might seem like an obvious determination to make; however, success remains the focus of significant debate (Shenhar, Levy, & Dvir, 1997). Some scholars even note the possibility that project success can never be determined as a result of improper planning, management, and methodologies (Al-Ahmad et al., 2009). Therefore, it is vital that project managers establish identifiable and coherent criteria for success and the measures to evaluate success. These criteria and measures must be agreed upon by stakeholders (Watson, 2009).

Simplistic criteria for project success include time, cost, and the quality of the project outcome (Ika, 2009). However, scholars claim that these criteria represent a barrier to determining project success and the factors related to success. Rosenau (1984) goes so far as to identify these three criteria as the triple constraint. Likewise, Sommer (2004) emphasized the expansion of success criteria, identifying the factors as budget, schedule, and sponsor objectives, defined requirements of features and functions, and customer satisfaction.

Shenhar et al. (1997) developed a multidimensional model for project success using contingent organizational theory. Other scholars have credited this model as "the most important line of research" (Ika, 2009, p. 113) in determining project success. The multidimensional model establishes success on five criteria, mainly project efficiency, impact on the team, impact on the customer, business and direct success, and preparation for the future (Shenhar & Dvir, 2007). Regardless of the criteria used to determine project success, it is paramount that project managers consult with stakeholders and agree on clearly identified success criteria (Pinto & Slevin, 1989).

The Standish Group (2011) reported that agile projects are successful three times more often than nonagile projects. The report indicated, "The agile process is the universal remedy for software development project failure. Software applications using agile process have three times the success rate of the waterfall method and a much lower percentage of time and cost overruns" (p. 25). The Standish Group defined project success as on time, on budget, and with all planned features (requirements) implemented. The results are from projects conducted from 2002 through 2010.

Project Failure

Project failure has been the focus of significant study due to the significant repercussions on professional and organizational success. Project failure represents a significant opportunity for learning and should not always be characterized as organizational failure (Kerzner, 2006). By better understanding the causes of project failure, project success can be achieved through instituting preventative measures (Nelson, 2005).

The CHAOS report issued by the Standish Group (2008) identified cause of project failure in software development projects, noting the prevalence of failure as established by the terms “project challenged” (p. 4) and “project impaired” (p. 4). The study found a success rate of only 16%, with challenged projects representing 53% and impaired projects representing 31%. Despite this high rate of failure, significant learning is possible from mistakes, but is only possible when the project is subjected to evaluation (Preuss, 2006). As a result of the Standish Group’s emphasis on evaluating project failure, project challenge and impairment have been reduced since the 1995 study, though the majority of projects are still determined to fail (Johnson, 2006; Preuss, 2006).

Critical Success Factors

Identification of critical success factors can allow project managers to avoid faulty assumptions about software development (Cottmeyer, 2009). Accurate identification of critical success factors, and their implementation, can also increase certainty and improve market timing, which has become increasingly difficult now that project life cycles are much shorter than previous eras. Agile critical factors are particularly necessary to identify because they relate to the ever-changing nature of projects.

Cottmeyer (2009) established an agile project management value system of critical success factors unique to agile projects. The factors include

- *Self-organization* that involves both the project manager and the team. Both parties must be responsible for delivery and requires them to be self-organized since direct supervision is not always possible or even desirable.
- *Empowerment* related closely to self-organization. This is the result of the environment that is established by the project manager and allows for local decision making and autonomy as much as is feasible.
- *Trust* is essential for the project manager to facilitate the best performance out of subordinates. Trust given to worthy subordinates elevates their aspirations, and with it a feeling of respect comes responsibility. Trust also facilitates a healthy team culture of goodwill and positive interaction.
- In return for the autonomy that is given to agile project team members under this framework, *accountability* is necessary. This translates into frequent delivery on deadline throughout the life of the project.

Other critical success factors relate to encouraging potentially resistant stakeholders to agree to the agile framework. Sliger and Broderick (2008) identified the following critical success factors that make agile a good sell to stakeholders:

- Face-to-face meetings
- Gross-level estimating or estimates based on general guidelines more than fixed estimates

- Technical planning emerging from self-organizing teams (Real versus unrealistic feature sets)
- Long-term budgeting.

Some critical success factors relate to the environment that is cultivated by the project manager. Fewell (2009) claimed that project managers must "have a ruthless commitment to value delivery over performance to plan" (p. 5). Each deliverable should be provided based on its risk and value. After consideration by the client, the next highest priority deliverable, based on risk and value, should be delivered. The entire project should not be delivered without these prior deliverables based on performance value.

An emphasis on value delivery means that one of the critical success factors for agile projects is breaking the project up into these deliverables. Return on investment is improved when these critical success factors are implemented, yet it often stands in contrast to traditional project management that establishes fixed features and deadlines in a sequential order unrelated to risk and value (Boynlon & Zmud, 1984).

In addition, high-performance teams are perhaps the most important critical success factor that can be influenced by the project manager. This is not necessarily a subjective measure of talent, experience, and motivation. Rather, high-performing teams have a record of success at delivering high-value projects and have demonstrated their motivation and talent by delivering features that can be objectively measured. Project managers must inspire and evaluate performance.

Fewell, Jack, Prior, Rosado, and Tarne (2009) identified the establishment of a "project champion" (p. 3) as a critical success factor for agile projects. The project champion manages all of the stakeholders, product demonstrations, and business requirements. Full-time involvement is necessary, with the primary focus on business analysis. Therefore, this critical success factor might not necessarily be the project manager. The project champion operates from the perspective of business analysis, identifying and promoting the value of the project to the firm and client's businesses. In most cases, however, this role is filled by the project manager.

Another critical success factor distinguishes agile projects from traditional waterfall models. This critical success factor can be considered as selective use of traditional tools. "Project Management Professionals are taught to look at the standard as a large set of tools from which they should pick and choose the components that will serve them on their project and leave the rest" (Fewell et al., 2009, p. 2). This conceptualizes agile not as a rejection of traditional models but rather as the selective use of the standard tools of these models. Essential to this critical success factor is the accurate identification of those tools that are best suited to the project.

A low level of administrative oversight is also a critical success factor of agile projects. Fewell et al. (2009) noted the fact that low levels of bureaucracy are characteristic of agile projects, meaning the environment cultivates a distinct philosophy that contains several critical success factors. These can be best understood in the following phraseology: People matter more than process, deliverables matter more than documentation, collaboration matters more than contracts, and planning matters more than any given plan. Thus, lower administrative oversight should only be understood in light of these other demands of the agile process, not as an end unto itself.

The project manager must spend considerable time assuring the client that the agile process is the best way to complete a valuable project on budget and on time. This critical success factor is a direct response to the appearance of agile as unplanned, risky, and uncertain. Clients must consistently be reminded that agile is the best way to complete the project so that they maintain a commitment to the project and do not interfere with the team's work (Griffiths, 2004).

Even though agile attempts to distinguish itself from traditional methods, the fact remains that control processes are critical success factors of agile projects as well as those based on traditional methods. Griffiths (2004) noted the fact that an integrative approach requires a careful calibration of control and autonomy by the project manager. This integrative approach represents a critical success factor and has specific qualities that must be identified and implemented by the project manager.

For example, some projects might require control tactics, such as evaluating the iterations compared to the initial goals (Griffiths, 2004). In projects where the goal is accurate, this is desirable. However, in other cases when the goal should be refined given the findings that result from execution, it might be necessary to change the goal. This element of control is complex given the fact that agile does not necessarily stick to a fixed plan if the experiences warrant a change.

Adaptability is thus an important critical success factor of agile projects, including those that integrate traditional models. Adaptability must also be the subject of control. Applying the concept from biological evolution, adaptability occurs when new traits are presented to the environment and generate beneficial results. Adaptability doesn't mean merely changing for the sake of change. Adaptability must always test certain features with metrics that measure success (Standing, Guilfoyle, Lin, & Love, 2006).

Another control process that represents a critical success factor is to prioritize change requests and defect reports. Projects often fail because of scope creep, which occurs when change requests and defects accumulate over time, eventually overwhelming budgets and violating deadlines. Careful control of defects and change requests can prioritize these in light of the remaining features to be developed. New features might need to be abandoned as a result of scope creep and the need to remain on budget and deadline (Hong, Thong, Chaslow, & Dhillon, 2011).

Critical success factors for projects that suffer from potential resistance or at least lack of buy-in from necessary stakeholders include promoting the use of agile methodologies during a preproject stage (Hong et al., 2011). There is a theoretical foundation to this critical success factor based on an evolutionary perspective. Resistance and lack of buy-in is directly related to individual and group concerns about survival, either in the natural world or the organizational or industry environment. Fear and anxiety are characteristic of change, and agile methodologies demand change of individuals and groups accustomed to traditional project management methodologies. Incorporating a training and education component to the preproject effort can address fear and anxiety resulting from concerns about project or individual failure (Standing et al., 2006).

Moreover, resistance to agile methodologies might be the result of social norms that have evolved in the organization. When change is introduced, social norms become a tactic of resistance, resulting in the project and the project leaders being conceptualized as antagonistic to social norms. Since these norms are attributed to prior and current success, it is paramount for them to be dismantled, as they are associated with traditional project management methodologies. A preproject training and education component can assist in this effort. Such a component can have lasting impacts throughout the life of the project. A preproject education and training component is necessary because resistance can occur throughout the life of the project, not just upon its onset. Indeed, resistance might actually be greater after the project is enacted and participants' experience fear and anxiety as a result of new processes, policies, and demands contrary to their experience (Standing et al., 2006).

The goal of preproject training and education in the benefits of agile methodologies is to cultivate the culture of agile, not just wear down resistance (Nerur & Balijepally, 2007). This critical success factor becomes markedly more complex and sophisticated because it must confront norms, fears, anxieties, and other entrenched features of individual and organizational culture that supports traditional methodologies. The goal of preproject training and education is to encourage the individual to develop a sense of curiosity about agile and its various features throughout the project (Nerur & Balijepally, 2007).

Preproject training and education should continue to be referenced throughout the life of the project. This critical success factor cannot be conceptualized or applied as a one-time measure, rather it must continue to be enacted informally, particularly as the project manager recognizes resistance and its various permutations among participants. Training and education about agile can also be referenced informally as the project manager recognizes a new stage in the project that is particularly threatened by resistance (Nerur & Balijepally, 2007).

Theorists have developed more abstract notions of the critical success factors of agile methodologies. Nerur and Balijepally (2007) claimed that agile methodologies achieve success because they cultivate shifts in patterns of thought as participants engage in higher levels of inquiry and validation of basic knowledge. This shift in mentality ensures a problem-solving approach to project solutions rather than one that relies on rote methods. A mentality that emphasizes inquiry and validation of previous systems of knowledge is more likely to solve problems with novel solutions.

The agile methodology is thus conceptualized as a critical success factor in and of itself as it produces this culture and shift in mentality. Nerur and Balijepally (2007) claimed that strategic thinking is more likely to emerge as a result of the proper application of agile methodologies. This type of thinking is more holistic and thus more likely to accurately predict problems and devise their solutions. “The shift from a mechanistic perspective to a perspective that acknowledges the existence of environmental uncertainty and complexity is evident in today’s strategic management thinking” (Nerur & Balijepally, 2007, p. 80). The creative nature of agile methodologies represents a critical success factor, emerging organically from the process of agile project management.

In addition to the abstract theoretical critical success factors associated with agile methodologies, the literature is replete with evidence of more practical and applied critical success factors. Mumford (2010) noted the influence of the Scrum structure, which in itself contains at least several critical success factors associated with agile methodologies, including the following:

- Early and continuous delivery of project deliverables
- An intimate interdependent work environment
- Continuous evaluation of technical performance.

These critical success factors owe their success to distinct outcomes as a result of their implementation in the Scrum structure. Early and continuous delivery of project deliverables allows more time for testing. An intimate, interdependent work environment ensures more minds are applied to problem solving and evaluation. Continuous evaluation of technical performance results in a delivered final project that requires less revision.

Another critical success factor of agile methodologies relates to its inherent questioning of traditional methodologies for project management. Cottmeyer (2009) noted the fact that software development no longer allows for easy predictions that lend themselves to traditional methodologies, particularly the emphasis on planning. Fundamental assumptions about how the project will unfold will prove costlier as these assumptions fail to reflect reality. Thus, agile methodologies are conceptualized as inherently capturing the critical success factor of being adaptive and responsive to real problems rather than perceived, past, or theoretical assumptions.

The trade literature also emphasizes the practical and applied critical success factors of agile methodologies. Gregory (2009) emphasized the iterative and incremental nature of agile methodology, which breaks the project into its necessary components and emphasizes attention to detail at a level not realized through traditional methodologies. In addition, Gregory emphasized other critical success factors for agile projects that are not always considered in the scholarly literature, including

1. Looking at the big picture. This includes a holistic and strategic perspective that examines both external and internal elements.

2. Use of the whole team approach that involves including as many individuals as possible to develop a comprehensive and complete evaluation of the project. As opposed to a hierarchical structure promoted by traditional methodologies, the agile methodology facilitates a horizontal approach that seeks input from all project participants.
3. Automated regression testing that subjects unit tests to those with the greatest return on investment. Design for testability is encouraged as part of a team effort that involves selecting the proper tools.
4. Providing and obtaining feedback.
5. Collaborating with the customer, undertaken not just during the feedback stage but rather during the whole project.

The trade literature also emphasizes the importance of quantifying collaboration with customer, including customer goals. Haughey (2012) noted the necessity of obtaining measurable customer goals so that these can be built into the project rather than assumed. Measurable customer goals can also avoid scope creep, which involves the incremental expansion of the project beyond its initial scope. Scope creep threatens budget, deadlines, and performance, and it can be more easily avoided when each new element of the project is subjected to its relevancy to measurable customer goals (Nan & Harter, 2009).

While the literature is replete with claims of collaborative teamwork as a critical success factor, the dynamics of this collaborative team environment demand greater scrutiny. Bavani (2009) supported the development of initial documents or wiki pages that outline the scope, high-level requirements, user stories, architecture, designs, roadmaps, and milestones. While this appears related to standard methodologies, in the agile project these documents or wiki pages are subjected to constant change as the project unfolds. They are not the blueprint for the project but rather a template to be modified.

Collaborative teams must be structured to encourage interaction. Bavani (2009) noted the fact that geographically dispersed teams often are structured to report to a single team leader, discouraging interaction. The author recommended the dismantling of hierarchical structures that attempt to steer team contributions through a single site and source.

Of course, the critical success factor of team collaboration can threaten overall productivity if teams are left to pursue open-ended and undefined goals. Bavani (2009) emphasized the necessity of explicit delegation to teams. This will allow their collaborative energies to be properly channeled. In order to be explicit, milestones might be applicable, and these should only be changed as the team encounters new obstacles to the initial milestone. Delegation must recognize the "criticality, the background, the audience and the priority of task on hand" (Bavani, 2009, p. 2).

One of the more specific critical success factors Bavani (2009) emphasized is tool-driven query resolution. There are various web-based tools developed specifically for software developers that allow for query resolution in a manner that is superior to e-mails and other communications. These tools ensure the query follows specific parameters and is not subject to the ambiguous nature of human language.

The trade literature, including consulting reports, emphasized several key critical success factors that apply to all project management, but which must be specifically addressed in agile methodologies due to their relatively open-ended nature. Hirshfield (2010) noted the importance of developing realistic timeframes and expectations. Many agile project managers will base their assumptions about timeframes and expectations on an overly optimistic notion of agile project productivity.

Scholars emphasize the importance of identifying critical success factors based on the level of management and the phase of development. Kerzner (2003) claimed that the executive management acceptance phase must include the critical success factors of considering employee recommendations, recognizing the necessity of change, and understanding the executive role in project management. These critical success factors can be conceptualized as sequential, with employee recommendations prompting recognition of the necessity for change and change prompting the executive to take action. Participative executive leadership is the primary concept underlying critical success factors at this stage of development and level of management.

The line management acceptance phase includes the critical success factors of willingness to place company interest before personal interest, accept accountability, and see associates advance. These should also be conceptualized as interrelated. When line managers accept the primary status of company interest, they are more likely to accept responsibility for the project's success and to facilitate the success of their colleagues. Selflessness is the primary concept underlying critical success factors at this level of management and phase of development. When the needs of the organization and others are placed forefront, line managers serve the project rather than themselves and their own limited career objectives (Cao, 2006).

Some critical success factors are cross-organizational, but do change depending on the phase of the project. The growth phase should include the critical success factors of recognizing the need for a corporate-wide methodology, supporting uniform status monitoring and reporting and recognizing the importance of effective planning. However, these critical success factors might confront resistance as they are implemented throughout the organization. Likely critical failure factors include the perception among some members and groups that a standard methodology is a threat rather than a benefit, inability or unwillingness to understand the benefits of the project, and lip service paid to planning. Resistance becomes more problematic during the growth phase as the inclusion of more interest groups raises the likelihood of conflict (Fowler, 2007).

During the maturity phase, critical success factors include recognizing the intertwined nature of cost and schedule, tracking actual costs, and developing project management training. These cross-organizational factors are also likely to face resistance. The most complex form of resistance involves the belief that growth and success are equivalent. This form of resistance leads to a failure to properly track costs and significantly increases the likelihood of cost overruns (Pinto & Slevin, 1989).

Critical success factors should be the focus of preproject planning (Heldman, 2011). For example, the critical success factor of cultivating understanding and consensus by key stakeholders, the project team, management, and project manager requires a preproject effort that can identify potential resistance. Consider the project manager who attempts to cultivate these qualities after the project is underway. Significantly more resistance is likely as these key stakeholders and other interested parties perceive a lack of input into the project. Resentment is more likely. In addition, failure to gain understanding and consensus before the project will fail to identify potential deficiencies and possible improvements in the project. Key stakeholder and interest groups are more likely to offer recommendations for refinements that the project manager and team are unlikely to identify. Key stakeholder and interest groups are more likely to offer a comprehensive review of all existing and potential elements of the project than the project manager and team acting alone. As they gain input in project planning, stakeholder and interest group understanding and consensus for the project are likely to significantly increase.

Identification of critical success factors can be conceptualized by the identification of project elements necessary for success (Camilleri, 2011). With the knowledge of specific project elements, the factors necessary to achieve these can be identified as critical success factors that must be implemented as part of project planning. For example, in order to achieve some specific element of the project, interdivisional cooperation might be necessary. The critical success factor should not merely be encourage cooperation among divisions. Rather, the critical success factor should identify specific ways in which to achieve cooperation, including identification of barriers to cooperation. When critical success factors are specified rather than generalized, they are more likely to be realized. Too often, project managers espouse generalized critical success factors that provide a series of abstract conceptual statements.

Critical success factors might also be formulated by identifying customer requirements (Milosevic, 2003). This will require the project manager to interact with customers during the planning phase. In addition to customers, other external stakeholders can assist in identifying critical success factors (Milosevic, 2003).

Stockholders, governments, competitors, the general public, creditors, suppliers/vendors, unions, and local communities provide a comprehensive perspective of potential external stakeholders. Some scholars emphasize the benefit of a SWOT analysis to identify critical success factors. Such an analysis involves identifying both internal (strengths and weaknesses) and external factors (opportunities and threats) that can identify critical success factors. "Information about the environment, combined with the knowledge of the organization's capabilities, enables project teams to identify critical success factors" (Milosevic, 2003, p. 133). Such analyses must involve all parties internal and external to the organization to achieve the level of comprehensiveness and specificity necessary to make critical success factors specified rather than generalized (Glass, 2006).

While much of the literature emphasizes the necessity of project planning, other scholars emphasize the critical success factor of quickly implementing the project to generate early wins and fast failures (DeCarlo, 2010). The agile processes of project management are particularly suited to this concept of project development, which emphasizes adaptive and rapid movement, with learning occurring during the project. Rather than trying to identify and construct the project through planning, thus anticipating all potential setbacks and building strengths, this emphasis tolerates failure as it seeks immediate innovation. However, it should be noted that the identification of critical success factors by quickly implementing the project, and learning by error, is only possible for certain types of projects. DeCarlo (2010) emphasized this approach for projects that are relatively small scale and involve high levels of innovation, such as software development projects that are building upon previous systems. These projects lack the complexity and scope of projects that require extensive planning. In addition, these projects identify and solve problems using known skill sets and knowledge domains. For these projects, project planning consists primarily of assembling the team, distributing customer requirements, and providing a general framework for delivery. In this environment, rapid implementation and learning by error can be executed without the cost of failure for larger, more complex projects.

Critical success factors can also be conceptualized as what is necessary to meet the desired deliverables of the project (Kerzner, 2009). From this perspective, critical success factors are those that meet the minimum requirements for the project, not the optimal provision of deliverables from an efficiency and business value perspective. In order to diminish confusion between minimum and optimal critical success factors, Kerzner recommended distinguishing these as primary and secondary, respectively. Those factors necessary to meet the desired deliverables should be designated as primary while those necessary to optimize efficiency and business value should be designated as secondary. Kerzner took the position that customer requirements should always be designated as primary, since no organization can survive without first satisfying customers. It is not possible, in the long-term, to achieve efficiencies or business value while customer requirements are unmet. The philosophical debate between primary and secondary critical success factors can be avoided by clearly delineating between the two.

Over the last 30 years a number of researchers have done valuable CSF studies on user involvement as a critical success factor. Rockart (1979) and Ramaprasad and Williams (1998) found that critical success factors could have major impacts on the design and implementation of IT solutions. Leidecker and Bruno (1984) defined critical success factors as "those characteristics, conditions or variables that, when properly sustained, maintained, or managed, can have a significant impact on the success of a firm competing in particular industry" (p. 37).

Stakeholders must be organized at the beginning of the project to develop a shared understanding of project success. The project manager's first responsibility is to determine stakeholders and inquire as to their interests and expectations. Goals agreed upon by the stakeholders must be quantifiable and coherent and they should directly translate to success criteria. The literature documents a failing to fully develop all success factors (Nelson, 2005).

Chapter Summary

This literature review provided an extensive background of traditional or standard (e.g., waterfall) and agile software development methodologies, as well as hybrid methodologies that borrow from both models. Background was provided on the principles, elements, advantages, and disadvantages of these models. The waterfall philosophy was taken from the hardware manufacture and construction tactics during the 1970s and hence has a highly structured approach to software creation. In the waterfall method, the entire process of software development is divided into separate phases. The waterfall model is one of the most used software development processes, namely because the advantages outweigh the disadvantages in particular types of projects (Cao, 2006; Nelson, 2005; Pinto & Slevin, 1989).

The agile model of software development was created during the 1990s when developers decided to move away from models that were highly compartmentalized approaches and move towards models that would offer more flexibility (Phatak, 2012).

The agile model focuses on flexibility and adaptability in development. Agile models involve several iterative development schedules that look to improve the output at every iterative step. Teams work closely together and with the other teams. The design concept allows for it to evolve as new concepts come along. The importance of documentation is not as emphasized as the waterfall model and there is more focus on the speed of the delivery of the system.

The research design and methodological approach of this study focused on perceived differences in project success between project managers who use agile or waterfall methodologies. Additionally, the researcher sought to measure the extent to which project managers in each of the two groups think specific critical success factors (user involvement, communication and quality) are used by their methodology and if they contribute to project success.

CHAPTER 3. METHODOLOGY

Research Design

This study used a quantitative, nonexperimental, descriptive method to collect data using an online survey instrument that sought to understand the role that identified CSFs play in the delivery of successful software development projects in IT. A quantitative methodology is deemed ideal for a study examining relationships between methodologies and critical success factors. Creswell (2009) identified quantitative surveys as ideal to test such relationships. The three primary questions that should be asked in a research design are as follows:

- What knowledge claims does the researcher make?
- What strategies of inquiry determine the procedures?
- What methods of data collection and analysis should be utilized?

Creswell also identified the positivist assumptions that determine traditional quantitative research as a means for testing objective theories by examining the relationship among the variables. Data are numeric and measures quantities, and this requires the collection and analysis of numerical data to identify and explain phenomena. Quantitative research methods are ideal to examine large populations with descriptive data to allow for comparisons among groups, to predict data, and to build predictive models.

Surveys are useful for testing hypotheses, particularly quantitative surveys that use scales such as the Likert (Higgins, 2006). Such surveys allow for the use of sophisticated statistical analytical tools for data analysis. The purpose of such surveys is to produce statistical data. The quantitative method of allows the researcher to test relationships among variables categorized as independent, dependent, moderating, extraneous, intervening, and controlling (Cooper & Schindler, 2011). Moreover, variables influence measures used for the survey questions, including nominal, continuous, ordinal, or ratio (Trochim, 2001). Survey data can explain, predict, associate, compare, or predict relationships among variables.

This quantitative study utilized a validated instrument that contained 56 items as a questionnaire originally developed by Cao (2006). The survey instrument was adapted for use with the waterfall method because the original was based exclusively on agile methodology. Researcher received signed permission to use the instrument from Dr. Cao. Due to modifications to the instrument to include additional items, a field test was conducted using a small panel of subject matter experts. A sample of the questionnaire was sent to the panel and all comments and recommendations were used to improve the quality of the adapted survey. Adaptation involved making the instrument more inclusive of waterfall and critical success factors identified specifically for this study. The enhanced instrument resulted in a 50-item questionnaire specifically used for this research effort. Each item was rated based on a 7-point Likert scale. The instrument was used to collect data from IT project managers and IT functional management.

Data collection was conducted using the Zoomerang audience database, an online survey database of panelists (SurveyMonkey, 2013). An informed consent form for participants was obtained prior to conducting the research. A consent form is necessary to protect the rights of participants during data collection (Creswell, 2009).

An e-mail was sent to prospective participants via SurveyMonkey explaining the nature of the study and requesting their willingness to participate. Participants were anonymously directed to answer questions via the online, web-based questionnaire (SurveyMonkey, 2013). SPSS 21 software was used for data analysis in this research (Field, 2013).

Population and Sample

Population and Data Collection Method

One of the most problematic issues related to sampling is determining the sample size (Lenth, 2001). Scientific objectives must be ensured by gathering quantitative information prior to the study. Statistics and subject matter experts can be utilized in gathering such information. Impartial evaluation of information is essential to determine the validity and reliability of the study.

Indeed, Lenth (2001) noted the unfortunate fact that "sample size is not always determined based on noble scientific goals" (p. 10). In some cases, the sample size might be determined based on available resources, or even the knowledge and expertise of the researcher. However, even in such cases, evaluation of the sample size proposal should determine whether basic scientific standards are met.

One of the most significant issues in survey-based research relates to using a complete sample frame within the target population (Wang & Doong, 2010). Internet-based surveys, as a result of their nature, often include unknown populations. This threatens the generalizability of the study, as a biased result is possible if the sample frame includes certain attributes uncharacteristic of the population. Researchers can avoid this pitfall by ensuring that the Internet-based survey is only accessible to individuals from the desired population.

Another problematic issue related to sampling also relates to Internet-based surveys. Sampling method and sampling error can face problems, as Internet-based surveys might not use probability sampling. For quantitative research purposes, better generalizability is gained from a random sample; however, Internet-based surveys might not provide this possibility.

A sampling strategy must effectively balance needs for scientific standards, practicality, and ethics. The target population for this research study was IT project managers, program managers, and functional managers who had worked on agile or waterfall projects. The researcher utilized the services of Zoomerang, which is affiliated with the SurveyMonkey site. Zoomerang (2013) offered access to panels of 3 million registered respondents from a broad range in industries and job titles. The use of the ZoomPanel function improved the quality of participant recruitment and data collection via the TrueSample feature, a large-scale audience validation software application that reduces risk by ensuring the authenticity of sample sources and thereby improving the reliability of the research. This Internet-based survey was only accessible to individuals from the IT project management population.

An e-mail with an invitation describing the purpose and nature of the study was sent via SurveyMonkey to prospective Zoomerang participants. Interested participants were asked to click on a link redirecting them to the SurveyMonkey webpage to complete the questionnaire. This study's sampling strategy was to use nonprobability sampling to draw a convenience sample from the population of Zoomerang panelists who had worked as a manager on IT projects. Nonprobability samples are not based on chance but are selected based on the design of the research (Graziano & Raulin, 2009). The sample selection was drawn from registered participants with IT project management experience across varying industries or sectors using agile or waterfall methodologies for IT projects. To participate in the study, individuals were required to be 18 years of age or older, employed in information technology for more than 5 years, with at least 5 years of work experience as a project manager, and had used different project management methodologies.

Interested participants were asked to complete a brief screening questionnaire to ensure that they meet the inclusion criteria for the study. The screening questionnaire collected information on the prospective participants' age, experience in the IT field, experience as an IT project manager, whether or not they had experienced working with the agile or waterfall method, and their level of English proficiency. Participants who met the inclusion criteria continued to the informed consent page. The informed consent page included assurances regarding the confidentiality of all data provided. Once they clicked the link indicating their consent to participate in the study, they were directed to the actual study questionnaire.

Sampling Frame

In determining the sample, errors can occur when the sample is not representative of the population being studied, when an inappropriate sampling method is used, or when the sample is too small to generalize the results to a population (Fink, 2003). Lenth

(2001) proposed a power approach to determining the appropriate sample size. First, a sample size must be established that meets the concerns of scientific standards, practicality, and ethics. The a priori sample size calculator for multiple regression was selected to calculate the sample size. This calculator determines the minimum required sample size for a multiple regression study, given the desired probability level, the number of predictors in the model, the anticipated effect size, and the desired statistical power level (Soper, 2013). Minimum required sample size for this study was calculated as 103. The rationale for the proposed sample size was to gather data from enough respondents to provide a rich, detailed, and comprehensive account of project management dynamics, and specifically the relationship between IT project success and the implementation and use of different project management methodologies. The minimum sample size of 103 participants permitted extensive statistical analysis of the respondents' data.

The following parameters were considered in the calculation of the minimum sample size:

- The type of statistical analysis procedure that was conducted, namely ANOVA, correlation, and regression.
- By convention, effect sizes of 0.02, 0.15, and 0.35 are considered small, medium, and large, respectively. This particular study was aiming for a medium effect size, or 0.15.
- The desired statistical power level was 0.80.

- There were a total of seven predictors in the model, not including the regression constant. (waterfall or agile, user involvement, communication, quality, costs, timelines, and size).
- To claim statistical significance, the *p* value (or alpha level or Type I error rate), should be less than or equal to 0.05.

Based on these parameters, the minimum required sample size was recalculated to be 98 units for analysis.

Zoomerang (2013) provided the survey research panel that participated in the online survey instrument for this study. The use of the Zoomerang service was deemed ideal because it provided access to individuals dispersed geographically, provided access to individuals from target groups, and provided timely access to an adequate number of participants.

Instrumentation

Characteristics of a Survey

Survey data usually are made up of responses tallied via selection within a measurement scale. Researchers then translate this data into numbers through a coding process that readies the data for computer analysis (Cooper & Schindler, 2011). There are advantages and disadvantages with conducting a survey. Some of the advantages are that it allows for more employees to participate at more locations and it produces a large amount of data. It also provides an overview of opinions and attitudes. Surveys can produce quantitative data with multiple-choice questions and qualitative data with open-ended questions and written comments (Zammuto & Krakower, 1991). The main disadvantage of conducting employee surveys is cost. Surveys can be expensive and time consuming, requiring training when using internal administrators, while especially expensive when outsourced. Developing a survey requires considerable time and expertise to design the questionnaire.

A survey is an opportunity to get an overview of people's opinions or attitudes about various topics; it may also tell how widespread interest is in a certain topic or how many people have participated in a particular activity over a period of time (Lyytinen & Hirschheim, 1987). A survey may also provide a baseline of attitudes and opinions from which to measure changes in those attitudes or opinions. However, a survey should not and cannot be used to explain attitudes and opinions, just to show that they exist at a certain rate or level in the group being studied. A survey can be the basis for generalizations about a given population in certain areas (Fink, 2003).

Respondents will want to know what will happen with the results of the survey and what, if anything, they should expect to hear back. If respondents suspect that nothing will happen with the survey results, they will probably be less likely to answer questions honestly and thoughtfully, or they may not answer them at all. One of the most important issues for survey administrators to address is respondent confidentiality; the respondent should not be identified with any specific comment.

Survey Questionnaire Development

Before attempting to write a questionnaire, the researcher must have a clear idea of the objectives in conducting the survey. Writing the questions is one of the most difficult aspects of survey research. Each question must be clear and easy to understand.

Common, everyday language should be used to ask questions and for administering instructions as well.

The questionnaire should not switch from one topic to another and questions should not be repeated. For, example after employees are asked how often they use the health services, the next question could ask for what reasons they visit the health services. A shift in topic should be clearly indicated through headers or transition paragraphs, and any changes in the format or procedure should be clearly stated. When using ratings or ranking scale respondents should be offered an opportunity to distinguish as finely as possible between their choices (Church & Waclawski, 1998). For example, rather than a scale of *agree*, *neutral* and *disagree*, respondents can be offered to choose between *strongly agree*, *agree*, *neutral*, *disagree* and *strongly disagree*. These finer distinctions allow for more reliable correlation analysis.

The survey in this study was informed by questions developed by Cao (2006) related to agile methodologies. Categories of the survey relate to each methodology and specific critical success factors. To measure the importance of success factors, a 7-point Likert scale was used to reflect the level of perception of the question by the respondent. The scale is from 1 (*strongly disagree*) to 7 (*strongly agree*), with 4 as the neutral point, and a N/A selection for “don’t know” or “not applicable.” There were 50 questions corresponding to four success factor attributes as discussed in the perception of success section. In order to avoid ambiguity in terms of perception of success on the part of the respondent, the questions focused on one particular project of the respondent’s choice in case he/she had been involved in multiple agile or waterfall projects.

Zoomerang provided the sample selection. Zoomerang offers access to panels of 3 million registered respondents. Use of ZoomPanel will improve quality via the TrueSample feature, a large-scale audience validation technology that reduces research risk by ensuring the authenticity of sample sources and improving reliability of the research (Zoomerang, 2013). Zoomerang (SurveyMonkey) has an effective way of providing a platform for distributing both informed consent and online research survey instruments providing an online platform for capturing respondent views as well as identifying target population participants meeting research criteria.

This study used an online survey instrument to collect data on IT project success rates and methodology used for project delivery. This quantitative study utilized a validated instrument to collect data and survey was conducted by SurveyMonkey, an online survey database (SurveyMonkey, 2013). This research used survey services of SurveyMonkey for data collection. Zoomerang.com (a service within SurveyMonkey) is a web-based survey service. The surveys were sent to a broad target population of IT project managers (Zoomerang, 2013). This study employed a survey questionnaire to collect data from IT project managers and IT functional management. An e-mail was sent to prospective participants from the Zoomerang panelist service via SurveyMonkey explaining the nature of the study and requesting their willingness to participate.

The SurveyMonkey Internet site was used to allow the creation of a hyperlink to a location where the survey may be taken. As part of the service, the site controls security, provides anonymity for participants, and collects the raw data of participant responses. The response data can then be downloaded into an Excel spreadsheet and coded appropriately in preparation to conduct the data analysis. The privacy policy for

SurveyMonkey (2013) states, “Accounts which are SSL enabled ensure that the responses of survey respondents are transmitted over a secure, encrypted connection” (p. 3). The SurveyMonkey privacy policy also notes that “We safeguard respondents’ e-mail Addresses” and “We will keep your data securely” (p. 3). Participants were anonymously directed to answer questionnaires via SurveyMonkey. Participants were given informed consent prior to conducting the research. A consent form is necessary to protect the rights of participants during data collection (Creswell, 2009).

SPSS 21 software was used for data analysis in this research effort (Field, 2013). For the first research subquestions, the independent variable was defined as agile or waterfall project management methodology. The dependent variables were critical success factors such as communication, quality, user involvement performing user acceptance test (UAT) planning, and organization change management planning. For the second research subquestion, the independent variables were respectively project managers’ reports of critical success factors in agile or waterfall projects, while the dependent variables was project managers’ reports of project success as measured by completion of the project on time, within budget, and of high quality. The first research subquestion was analyzed using ANOVA. The second subquestion was analyzed using multiple regression.

Previous studies provided the basis for the adaptation of items in existing instruments in this study. Zoomerang participants received questions in a 70-item questionnaire originally developed by Cao (2006). Completion time was estimated at 30 minutes. Survey instruments have been adapted because they were based exclusively on agile methodology. Adaptations involved making the instrument more inclusive of waterfall and critical success factors for this study. To measure the perception of success of agile or waterfall projects by project managers, a 7-point Likert scale was utilized to reflect and measure the level of perception of the question by the participants. The scale is from 1 (*very unsuccessful*) to 7 (*very successful*), with 4 as the neutral point. There were 50 questions related to the perception of success by the respondent, corresponding to the four attributes of success as discussed earlier.

The original questionnaire was comprised of an introductory section, description section, purpose section, consent form, demographic section, and section inquiring about use of agile methodologies. The last section was adapted to inquire about critical success factors in projects. The first section gathers demographic information, including information about the participant's organizations. Inquiries about the nature of projects and use of methodologies were made to ensure individuals were reporting unique projects. The questionnaire asked the participant about the methodology used. The logic of the survey allowed participants that select "agile" to continue with the survey and answer questions related to agile. If the answer is "waterfall," the survey continues with questions related to waterfall. If the answer is "none," the logic of the survey will take the participant to the last page, thanking them for their participation and will end the survey. These relationships was measured utilizing multiple regression analysis, which Lam and Lee (2006) identified as an ideal procedure of constructing an equation to detail the studied phenomenon's behavior.

Key terms and instructions for completion represent the second section. Questions about critical success factors and perceptions of project success represent the third section. Prior to testing the proposed model, a reliability analysis test was conducted.

Cronbach's alpha was utilized to evaluate consistency between measurements of the variables.

The survey measured the dependent variable project success with 22 questions on the 7-point Likert scale, ranging from *very unsuccessful* to *very successful*, which was consistent with the scale of other similar studies (Hair, Black, Babin, Anderson, & Tatham, 2006; Neuman, 2011). Success included covariates of scope, timelines, and costs. This interval data quantified perceptions of success. Perceptions of organizational change management, communication, user involvement, quality, agile, and waterfall were also been measured as independent variables.

Data Collection

The researcher developed the study's survey on the SurveyMonkey platform and used the SurveyMonkey coordinator to deploy the survey to Zoomerang panelists within the database. A survey questionnaire conveyed via hyperlink on the SurveyMonkey website collected data from participants. The strategy used nonprobability sampling to draw a sample from the Zoomerang population of project managers who had worked on IT projects. Respondents who met the selection criteria were redirected to the informed consent page. Upon indicating informed consent, the participants clicked on a hyperlink directing them to the study instrument. They completed and returned the instrument anonymously via the SurveyMonkey.com portal. SurveyMonkey.com provided indications of missed or nonresponse information on the questionnaires. The finished questionnaires were maintained on a password-protected SurveyMonkey.com account until the data were downloaded to the researcher's computer for the analysis of data.

Only the researcher had access to the data. To participate in the study, individuals were required to be 18 years of age or older, must have had to work in information technology for more than 5 years, must have had at least 5 years of work experience as a project manager, and must have used different project management methodologies. Surveys were administered only in English.

The researcher collected data until up to 200 female and male project managers were selected to participate in this research. This Internet-based survey was only accessible to individuals from the IT project management population. Survey questions asked participants if they used agile or waterfall methods for their project. The study sought to survey an equal number of IT Project Managers from each group (agile or waterfall).

The researcher used a survey instrument based upon one originally developed by Cao (2006). The measurement tool was enhanced to address any changes in technology that occurred since 2006 and to address waterfall methodology. The plan was to make adjustments as necessary to the research instruments. The researcher performed content validation to check if research questions were representative of the variables being researched. To select measures with good internal reliability, the researcher used the Cronbach's alpha measurement, with a threshold of .70 or higher (Cronbach, 1951).

Data Analysis

The SPSS statistical software package was used for data analysis in this research effort. For the first research subquestions, the independent variable was defined as agile or waterfall project management methodology. The dependent variables were critical success factors such as communication, quality, user involvement performing UAT planning, and organization change management planning. For the second research subquestion, the independent variables were respectively project managers' reports of critical success factors in agile or waterfall projects, while the dependent variables were project managers' reports of project success as measured by completion of the project on time, within budget, and of high quality. The first research subquestion was analyzed using ANOVA. The second subquestion was analyzed using multiple regression.

This study's questionnaire tool utilized a summated rating scale that generated quantitative interval data. Thus, parametric analyses, such as ANOVA or regression analysis was used, as long as assumptions of homogeneity of variance were met; the dependent variable is normally distributed and the independent variables are related to the dependent variable in a linear fashion (Vogt, 2010). Descriptive statistical analysis was used to present means and standard deviations for the subscales and the total score. Coefficients were computed to determine the path and extent of the relations between project success and communication, quality, and user involvement. The significance level for all analysis procedures was set at $p = 0.05$.

Hypotheses 1–3 were tested using ANOVA to determine whether the extent of use of the identified CSFs varied with regard to the type of project management model used. If the preliminary data analysis revealed that the assumption of normality was not met, then a nonparametric test was performed instead of the ANOVA.

Hypotheses 4–9 were tested using correlation analysis. Correlation provides a single number that identifies the nature (direct or inverse) and magnitude of the relationship between the variables. The specific correlation test that was conducted was the Pearson product–moment correlation coefficient (Pearson's *r*). However, if the preliminary data analysis indicated that the data did not meet the assumption of normality required for correlations, the nonparametric equivalent of the Pearson's *r* correlation analysis was conducted instead.

In relation with the correlation testing for Hypotheses 4–9, multiple regression analysis was also conducted to determine the extent to which the CSFs affect or influence project success. Multiple regression analysis was utilized to determine the relationship between the dependent variable and the combination of predictor variables and provided information concerning the course and magnitude of the correlations (Sekaran & Bougie, 2010; Swanson & Holton, 2005). Regression coefficients show the quality of each indicator variable when predicting the outcome variable (Sekaran & Bougie, 2010). Specifically, multiple regression analysis was employed to analyze the relation and effect between the dependent or outcome variable (project success) and each of the independent variables (i.e., agile, waterfall, communication, user involvement and quality), while other involved variables were held constant (Cooper & Schindler, 2011; Vogt, 2010).

To compare responses from the two groups of agile and waterfall project managers, *t* test were used to calculate the difference experiences of project managers. Responses from each group are independent of those in the other group. Correlation allows the researcher to determine the probability of chance. A one-tailed correlation was utilized to determine the relationship between methodologies.

Validity and Reliability

Three criteria for evaluation of a measurement instrument are practicality, validity, and reliability (Cooper & Schindler, 2011). Validity determines whether the test measures what it is intended to measure. Reliability involves accuracy of the measurement instrument. Practicality determines the efficiency, convenience, and interpretability of the instrument. Cooper and Schindler (2011) noted the relationship among reliability and validity. Reliability is a ceiling on the level of validity.

Reliability

Reliability is referred to as internal consistency reliability and is computed using Cronbach's alpha (Cronbach, 1951). A coefficient alpha of 0.70 is considered large, and is widely acknowledged as the lowest alpha needed to justify grouping a set of items as a scale; an alpha of 0.80 or greater shows high internal consistency reliability (Doloi, 2009). Sekaran and Bougie (2010) noted that the reliability of a measure indicates consistency and stability and facilitates an evaluation of the measure's accuracy.

Research focused solely on agile with 12 critical success factors has shown five factors with a low alpha of .50 (Cao, 2006). Only the organizational and people factors provided results above .70. A second reliability analysis by Cao (2006) utilized Cronbach's alpha method on the reduced list of factors, resulting in nine out of 10 acceptable levels of reliability. Thus, a Cronbach's alpha value of .70 or higher for this study should predict a strong positive relationship.

Validity

One of the validity problems related to quantitative survey research methodology is that replication is often not undertaken. Shadish, Cook, and Campbell (2001) noted, "Multiple replication is more typical of science than a once and for all definitive experiment. Experiments really need replication and cross-validation at various times and conditions before the results can be theoretically interpreted with confidence" (p. 37). This concept is challenged by the fact that surveys are typically administered at one time and only for one time to the participants. Validity issues related to research design might therefore be constructed into the survey instrument.

Internal validity is achieved when the experimental treatment or condition makes a difference, and there is sufficient evidence to support this claim. Efficacy studies can address issues of internal validity. For example, in quantitative survey research of IT project management, the cause–effect relationship between project type and project success relates to internal validity. The ability of the measurement to make the cause–effect claims is in question. Internal validity problems relate to how the research design is constructed.

Validity indicates the test results and measurements produce the desired outcome (Cooper & Schindler, 2011). Criterion or convergent validity determines whether the goal number using the instrument is in fact an indication of external standards (Creswell & Plano Clark, 2011), and "asks whether the measure really predicts the dependent variable it is supposed to measure" (Swanson & Holton, 2005, p. 36).

Validity defines the degree to which a test measures what you want it to measure (Cooper & Schindler, 2011). Content validity implies that "the content of your measure matches the content of what you are trying to measure" (Swanson & Holton, 2005, p. 36). Secondly, criterion or convergent validity determines whether scores relate to an external standard (e.g., scores on a similar instrument; Creswell & Plano Clark, 2011), and "asks whether the measure really predicts the dependent variable it is supposed to measure" (Swanson & Holton, 2005, p. 36).

Bias was addressed by limiting respondents to only those who had participated in agile and waterfall projects. Cao (2006) conducted a pilot study of those who had used agile Alliance to receive feedback that could be used to update the instrument sent to the target participants. This study increased validity by conducting field testing with experts to review the survey before the actual data collection. A minimum level of Cronbach's alpha of .70 was desired since Pallant (2005) found that optimal mean correlation can range from .2 to .4. Furthermore, knowledge tests were used to address content and face validity, as recommended by Cao (2006).

Validity occurs in three types: content, criterion, and construct (Pellissier, 2008). Content validity involves utilization of experts in the subject area to evaluate the survey questions, format, and organization. A group of subject matter experts provided feedback on the proposed adapted instrument, with suggestions being implemented for the general data collection phase. Criterion validity involves prediction, which determines whether the survey predicts external constructs, and concurrency, which involves the capacity of the survey to agree with expected results. Construct validity involves whether the scores yield co-correlates with other constructs as predicted (Pellissier, 2008).

Content Validity

Critical success factors were gathered from the literature study and industry experts. A group of three experts in project and program management were used to evaluate the survey. In addition, a presurvey including a small sample was distributed and evaluated by this group to determine face validity, with necessary adjustments made before release of the survey to the entire sample.

Construct Validity

This was established through the use of industry standards published by PMI (2013) specifically for surveys.

Criterion Validity

This was determined through statistical tests, including parametric, one-way ANOVA, which examined the relationships between the dependent variables and the null hypothesis (Cooper & Schindler, 2011). Levene's test validated the equal variances assumptions for the dependent variables using a one-way ANOVA. A Tukey multiple comparison test determined whether means were different based on the result of Levene's test signaling a relationship (Cooper & Schindler, 2011). A Tukey test assumes normal distribution, a common standard deviation for all groups, and yields a matrix that compares pairs with a *p* value.

Pearson's correlation coefficient was used to evaluate relationships among the independent variables, as well as linear multiple regression analysis, beneficial in measuring was not supported project success of specific methodologies. Validity was determined if the Pearson's correlation coefficient values showed the normality assumption is supported (Mirabella, 2008) and the Spearman Rank correlation test was utilized.

Potential relationships of nominal and ordinal variables were evaluated using the chi-square test of independence, which Mirabella (2008) supported in conjunction with ordinal variables when using the Likert scale. First, the researcher of this study determined that the observations of the collected data were independent. A *p* value of .05 determined rejection of the null hypothesis and the value for lambda showed the level of predictability between ordinal variables, with Gamma indicating the direction of relationship (Mirabella, 2008).

Ethical Considerations

Ethical considerations are based on the principles developed in the Belmont Report of 1979, including justice, beneficence, and respect for persons (U.S. Department of Health & Human Services [USDHHS], 1979). The data compiled in this study were not regarded as sensitive; therefore, the principle of respect for persons was supported. The purpose of the study was fully and clearly represented to participants (Cooper & Schindler, 2011). Participants should know exactly what the study seeks to achieve so they can decide whether their own ethical, professional, and practical needs are facilitated or impaired by their participation. Among potential benefits, the study's findings might be utilized to improve project performance.

Participants were asked to provide informed consent. Responses were anonymous and confidential, with no threats to participant privacy. No names or other contact information was collected presenting a minimal risk to participants. There was minimal risk and comparable benefits for all participants.

The Belmont principle of beneficence (USDHHS, 1979) was also facilitated insofar as this research design did not present any significant threat to privacy or wellbeing. Specifically, individuals were not contacted to take surveys in the workplace.

Participants were those who agreed to take surveys through Zoomerang on their personal time.

Respect for persons must also ensure that job security and performance are not threatened by the study. Participation was voluntary and no incentives were provided to encourage participation.

The principle of justice was recognized through the administration of the research process in a reasonable, nonexploitative, and fair manner to all (Swanson & Holton, 2005). It was also paramount that participants engage in full disclosure of their experiences and beliefs and was only made possible if their confidentiality was guaranteed.

In addition, the justice (equity) principle provided participants an equal likelihood of being selected to take part in the research (Creswell, 2009). The principle prescribes fair processes for research participant selection (USDHHS, 1979) and says all participants must share equally in the risks or burdens and any expected benefits of the research. Since all eligible IT managers accessible through Zoomerang were contacted for this study, the principle of justice was respected by the study methodology. Researchers have a position of power over their participants, and this power comes with an ethical obligation to pursue the principle of transparency so that participants know exactly what the study seeks to achieve through their participation. Only with this knowledge can participants know whether their own ethical, professional, and practical needs were facilitated or impaired by their participation. The following chapter presents the analysis of the data collected.

CHAPTER 4. RESULTS

Introduction

The objective of this study was to relate the use of the agile or waterfall methodologies and specific CSFs to IT project success for a sample of IT project managers who have used the agile and waterfall methodologies. This study was conducted by using a 56-item questionnaire originally developed by Cao (2006). Statistical analyses of the data were conducted using ANOVA, Pearson's correlation test, and multiple linear regression models. The following research questions and hypotheses guided the analysis:

RQ 1: What is the difference between the extent of use of the identified CSFs in the agile and waterfall model?

RQ 1.1: To what extent do project managers in IT for each methodology, agile and waterfall, report using effective communication?

H01: There is no significant difference between the extent of use of effective communication between the agile model and the waterfall model.

Ha1: There is significant difference between the extent of use of the effective communication between the agile model and the waterfall model.

RQ 1.2: To what extent do project managers in IT for each methodology, agile and waterfall, report using user involvement?

H02: There is no significant difference between the extent of user involvement between the agile model and the waterfall model.

Ha2: There is significant differences between the extent of user involvement between the agile model and the waterfall model.

RQ 1.3: To what extent do project managers in IT for each methodology, agile and waterfall, report using a quality plan?

H03: There is no significant difference between the extent of the use of a quality plan between the agile model and the waterfall model.

Ha3: There is significant difference between the extent of the use of a quality plan between the agile model and the waterfall model.

RQ 2: Which among the identified CSFs are correlated with successful projects in the agile and waterfall model?

RQ 2.1: In the waterfall project management model, which among the identified CSFs are correlated with successful projects?

H07: Project success is not significantly associated with effective communication in the waterfall model.

Ha7: Project success is significantly associated with effective communication in the waterfall model.

H08: Project success is not significantly associated with user involvement in the waterfall model.

Ha8: Project success is significantly associated with user involvement in the waterfall model.

H09: Project success is not significantly associated with the use of a quality plan in the waterfall model.

Ha9: Project success is significantly associated with use of a quality plan in the waterfall model.

RQ 2.2: In the agile project management model, which among the identified CSFs are correlated with successful projects?

H04: Project success is not significantly associated with effective communication in the agile model.

Ha4: Project success is significantly associated with effective communication in the agile model.

H05: Project success is not significantly associated with user involvement in the agile model.

Ha5: Project success is significantly associated with user involvement in the agile model.

H06: Project success is not significantly associated with the use of a quality plan in the agile model.

Ha6: Project success is significantly associated with use of a quality plan in the agile model.

RQ 2.3: In the waterfall project management model, how influential are the identified CSFs on the success of the project?

RQ 2.4: In the agile project management model, how influential are the identified CSFs on the success of the project?

This chapter begins with frequency tables to summarize information for the whole sample of IT project and functional management. Following that, the Cronbach's alpha measure to evaluate consistency between measurements of the variables will be presented. The descriptive statistics of the study variables are also summarized. Finally, the results of the ANOVA, Pearson's correlation test, and multiple linear regressions to address the research questions are presented.

Description of Sample

Demographic Information

The sample consisted of 106 respondents. The 106 samples consisted of 53 IT project managers and IT functional management who have applied waterfall methodology for project management and another 53 IT project managers or functional managers who have applied agile methodology for projects. The frequency and percentage breakdown of different demographic information are summarized in Tables 3–6.

Table 3. *Age of Managers*

Age	Agile	Waterfall	Total
Missing	12	22	34
18–29	1	7	8
30–44	25	18	43
45–60	15	6	21
Total	53	53	106

Table 4. *Household Income of Managers*

Household income	Agile	Waterfall	Total
Missing	12	22	34
$25,000–49,999	1	0	1
$50,000–99,999	19	18	37
$100,000–149,999	21	8	29
$150,000	0	5	5
Total	53	53	106

Table 5. *Education Level of Managers*

Education level	Agile	Waterfall	Total
Missing	12	22	34
Associate or bachelor degree	25	29	54
Graduate degree	12	2	14
Some college	4	0	4
Total	53	53	106

Table 6. *Location of Managers*

Manager's location	Agile	Waterfall	Total
East North Central	2	8	10
Middle Atlantic	9	9	18
Mountain	2	2	4
New England	3	3	6
Pacific	23	7	30
South Atlantic	0	2	2
West South Central	2	0	2
Total	53	53	106

The frequency and percentage breakdown of the different project profile information of the respondents is summarized in Tables 7–11.

Table 7. *Projects Handled*

Project	Agile	Waterfall	Total
Airline	0	1	1
Aprimo	0	2	2
Asset management software	1	0	1
Audit/sox—financial	0	1	1
Bank-customer support work flow	0	1	1
Billing, HRM	1	0	1
Computer asset management system	0	2	2

Table 7. *Projects Handled* (*continued*)

Project	Agile	Waterfall	Total
Contract management	1	1	2
CRM	1	1	2
Documentation management	0	1	1
e-Commerce—online sales	0	1	1
Education	1	1	2
Employee benefit, training	0	2	2
Enterprise MRM application	1	1	2
Financial	5	2	7
Food services	0	1	1
Geolocation survey application	1	0	1
HelpDesk application	0	1	1
Human resource application	2	1	3
Insurance—membership	0	1	1
Joint business/retail operations	0	1	1
Knowledge management	0	1	1
Lab monitoring	0	1	1
Law enforcement	0	1	1
Legal application	1	0	1
Marketing	5	2	7
Medical coding	0	1	1
Mobile application	1	0	1
New savings account product	0	1	1
Online baking	0	1	1
Online courses-application	1	0	1

Table 7. *Projects Handled* (*continued*)

Project	Agile	Waterfall	Total
Online sales, marketing, and supply chain	5	2	7
Oracle	1	3	4
Order management	0	2	2
Order to cash	0	1	1
Patient billing/care/records	3	1	4
People soft	0	1	1
Pricing engine project	1	0	1
Project regarding financial services	1	0	1
Quality system, PLM	1	0	1
Remedy software	0	1	1
Resource management	1	0	1
Restaurant management	1	0	1
Retail	2	4	6
Sales and marketing	10	2	12
SAP	1	2	3
School book publishing software	0	1	1
SharePoints	1	0	1
Software development	2	0	2
STAR data warehouse	0	1	1
Training application	1	0	1
UI + data filters	1	0	1
Vendor management	0	2	2
Warehouse/electronic	1	0	1
Total	53	53	106

Table 8. *Size of the Project (Number of Project Team Members)*

No. team members	Agile	Waterfall	Total
0–4	20	3	23
5–10	23	27	50
11–20	4	10	14
21–50	1	3	4
51–100	4	4	8
> 100	1	6	7
Total	53	53	106

Table 9. *Location of Project*

Project location	Agile	Waterfall	Total
Asia	1	1	2
Europe	1	0	1
North America	50	50	100
South America	1	2	3
Total	53	53	106

Table 10. *Level of Experience With Projects*

Experience (years)	Agile	Waterfall	Total
1–3	7	0	7
3–7	30	33	63
7–10	14	13	27
> 10	2	7	9
Total	53	53	106

Table 11. *Number of Agile Projects Involved With*

No. agile projects	Agile	Waterfall	Total
1–3	16	2	18
3–7	19	12	31
7–10	15	6	21
> 10	3	33	36
Total	53	53	106

The frequency and percentage breakdown of the company or organization project profile of the respondents are summarized in Tables 12 and 13.

Table 12. *Company Size*

Company size	Agile	Waterfall	Total
< 1,000	21	4	25
1,001–5,000	4	0	4
5,001–10,000	6	1	7
10,001–20,000	17	36	53
> 20,000	5	12	17
Total	53	53	106

Table 13. *Company Industry*

Industry	Agile	Waterfall	Total
Financial	4	2	6
Healthcare	7	6	13
Manufacturing	2	6	8
Others	20	11	31
Retail	4	3	7
Services	4	19	23
Technology	12	6	18
Total	53	53	106

Reliability Measure of the 56-Item Survey

The reliability of the survey results of the 56-item questionnaire, originally developed by Cao (2006), was measured to test the internal consistency and instrument reliability. The Cronbach's alpha was computed as the reliability measure, testing each of the constructs in the Cronbach point-biserial correlation. These included the measures for CSF and project success for both the agile and waterfall methodologies. Table 14 summarizes the Cronbach's alpha reliability statistic. Based on Table 14, it can be observed that the statistic for each of the constructs of CSF (agile; $\alpha = 0.86$), project success (agile; $\alpha = 0.79$), and CSF (waterfall; $\alpha = 0.71$) were greater than the generally acceptable minimum value of 0.7, implying that these measures were acceptable, reliable, and internally consistent in measuring the study variables. However, the Cronbach's alpha statistic for the construct of project success (waterfall) of 0.42, was less than the minimum acceptable value of 0.7, implying that these measures had a questionable reliability. This will be considered a limitation of the study and will be further discussed in the succeeding chapter.

Table 14. *Cronbach's Alpha Reliability Statistics*

Variable	Cronbach's α	No. items
CSF (agile)	0.86	37
Project success (agile)	0.79	4
CSF (waterfall)	0.71	37
Project success (waterfall)	0.42	4

Descriptive Statistics of Study Variables

The descriptive statistics presented in this section include the statistics of mean and standard deviation. First, the descriptive statistics of the responses on the 37-item CSF measures were obtained and are summarized in Table 15.

Table 15. *Descriptive Statistics of 37 Items of CSF Measures*

CSF	Agile			Waterfall		
	M	*N*	*SD*	*M*	*N*	*SD*
Received strong executive support	6.16	51	1.12	4.26	53	2.31
Committed sponsor or a committed organization manager	6.09	53	0.84	5.19	53	1.73
Cooperative culture instead of hierarchal	6.00	53	1.11	5.02	53	1.54
Oral culture placing high value on fluid (face-to-face communication style)	5.91	53	1.10	4.81	53	2.11
Agile methodology was universally accepted in the organization	4.81	53	1.74	4.58	53	1.23
Reward system that was appropriate for agile behavior	5.32	53	1.45	4.58	53	1.74
Project team was collocated	5.98	53	1.10	5.19	53	1.35
Worked in a facility with proper agile-style work environment	5.25	53	1.66	5.19	53	1.57
High technical competence and expertise team members	5.68	53	1.36	5.49	53	1.05
Project team members had great motivation and were committed to the project success	5.43	53	1.85	5.00	53	1.40
Project management was knowledgeable in agile principles and processes	5.68	53	1.40	6.15	53	1.68
Project management had light-touch and/or adaptive management style	5.52	52	1.28	3.75	53	1.62
Worked in a coherent, self-organizing teamwork manner	5.57	53	1.67	5.21	53	2.12
Good relationship with the customer	5.43	53	1.28	4.26	53	1.00
Well-defined project scope and objectives	4.96	53	1.34	4.79	53	1.03
Agile-oriented requirement process	4.98	53	1.72	5.75	53	1.04
Agile project management style	5.55	53	1.19	5.17	53	1.07
Agile-oriented configuration management process	6.00	53	1.69	5.74	53	0.92
Agile-friendly progress tracking mechanism	5.91	53	1.36	5.60	53	0.93

Table 15. *Descriptive Statistics of 37 Items of CSF Measures* (*continued*)

CSF	Agile			Waterfall		
	M	*N*	*SD*	*M*	*N*	*SD*
Strong communication focus and rigorous communication schedule	5.43	53	1.47	4.60	53	1.61
Project honored regular working schedule	5.25	53	1.36	5.96	53	1.66
Strong customer commitment and presence	5.60	53	1.31	5.45	53	1.34
Customer representative had full authority and knowledge to make decisions on-site	5.96	53	1.78	5.42	53	1.71
Well-defined coding standards up front	5.17	53	1.55	6.23	53	1.01
Pursued simple design	5.21	53	0.99	4.19	53	0.96
Pursued vigorous refactoring activities	5.40	53	1.10	5.32	53	1.03
Maintained right amount of documentation	6.02	53	1.20	5.96	53	0.52
Followed continuous and rigorous unit and integration testing strategy	5.26	53	1.36	4.15	53	0.91
Delivered working software regularly within short periods of time	5.36	53	1.13	4.42	53	1.76
Delivered most important features first	5.85	53	1.06	5.68	53	1.17
Employed proper platforms, technologies, and tools suitable	5.96	53	1.18	5.17	53	0.83
Provided appropriate technical training to team	5.26	53	1.37	5.53	53	1.07
Project nature was a non-life-critical software project	5.70	53	1.23	6.25	53	1.22
Well defined scope upfront with solid requirements	5.64	53	1.51	5.87	53	0.52
Dynamic, accelerated schedule	6.04	53	1.18	6.43	53	0.77
Up-front, detailed cost evaluation	4.68	53	1.65	6.04	53	1.21
Up-front risk analysis using agile method	5.17	53	1.58	5.74	53	0.79

This measured the independent variables of communication, user involvement, and the use of a quality plan. Scores for these variables were collected using a 7-point Likert scale, ranging from 1 (*strongly disagree*) to 7 (*strongly agree*).

The descriptive statistics of the CSF measures for the IT project and functional managers who had applied agile or waterfall methodologies and reported project success were analyzed. Those that had used agile methods for their projects showed a range of responses for the 37-item critical success factors between 4.68 and 6.16, being associated with *somewhat agree* to *strongly agree* on the scale. The results indicated that the IT companies represented by these respondents used good to excellent communication skills, encouraged user involvement, and used a quality plan in their project activities. The mean responses for waterfall respondents ranged between 3.75 and 6.43, or reporting a neutral to strongly agreement to the CSF measures. The comparison of the mean responses between managers using agile and waterfall methodologies showed that the managers using agile project management practiced the various CSFs more consistently or more often than managers using waterfall methodologies.

The descriptive statistics of the responses on the project success measures were obtained and summarized in Table 16. These statistics measured the dependent variable of perceptions of success for the project managers that participated in the study. Participants responded to the statements pertaining to project success using a 7-point Likert scale, ranging from 1 (*very unsuccessful*) to 7 (*very successful*). Project success was measured in terms of the success in areas of quality, scope of project, timeliness, and cost. For the purpose of this study, the average score of the four areas of project success determined overall project success.

Based on the mean comparison, the overall project success for the managers who applied agile methodologies (*M* = 5.38) was higher as compared to the overall project success for the managers who used waterfall methodologies (*M* = 4.25). The project success rate for agile managers was in the range of *somewhat successful* to *successful*. On the other hand, project success for managers who applied waterfall methodologies was in the range of *somewhat unsuccessful* to *somewhat successful*, which indicated that projects involving waterfall methodology were unsuccessful in some areas of quality and cost.

Table 16. *Descriptive Statistics of Measures of Project Success*

	Agile			Waterfall		
Project success	*M*	*N*	*SD*	*M*	*N*	*SD*
Project success (quality)	5.49	53	1.15	3.96	53	0.94
Project success (scope)	5.81	53	1.14	4.91	53	1.54
Project success (timeline)	5.17	53	0.91	4.40	53	1.51
Project success (cost)	5.06	53	1.06	3.74	53	1.15
Overall project success	5.38	53	0.84	4.25	53	0.79

Test of Normality and Outlier

First, the test of normality was conducted by investigating the skewness and kurtosis of the data, as summarized in Table 17. Skewness statistics greater than a positive or negative three and kurtosis statistics between 10 and 15 indicated nonnormality (Kline, 2010). The skewness of the CSF measures and the project success measures ranged between –1.3 and –2.23, less than the unacceptable limit of three. The results indicated that the normality distribution was inherent in the data of the CSF measures and project success measures.

Table 17. *Minimum, Maximum, Skewness, and Kurtosis Statistics*

				Skewness		Kurtosis	
CSF	*N*	Min	Max	Statistic	*SE*	Statistic	*SE*
Received strong executive support	104	1	7	–1.15	0.24	–0.14	0.47
Committed sponsor or a committed organization manager	106	1	7	–1.78	0.23	2.64	0.47
Cooperative culture instead of hierarchal	106	1	7	–1.61	0.23	1.78	0.47
Oral culture placing high value on fluid (face-to-face communication style)	106	1	7	–1.13	0.23	–0.15	0.47
Agile methodology was universally accepted in the organization	106	1	7	–0.44	0.23	0.04	0.47
Reward system that was appropriate for agile behavior	106	1	7	–0.32	0.23	–0.91	0.47
Project team was collocated	106	1	7	–1.70	0.23	3.25	0.47
Worked in a facility with proper agile-style work environment	106	1	7	–1.69	0.23	1.89	0.47
High technical competence and expertise team members	106	2	7	–1.88	0.23	3.14	0.47
Project team members had great motivation and were committed to the project success	106	1	7	–1.19	0.23	0.96	0.47
Project management was knowledgeable in agile principles and processes	106	1	7	–1.85	0.23	2.34	0.47
Project management had light-touch and/or adaptive management style	105	1	7	–0.56	0.24	–0.46	0.47
Worked in a coherent, self-organizing teamwork manner	106	1	7	–0.99	0.23	–0.12	0.47

Table 17. *Minimum, Maximum, Skewness, and Kurtosis Statistics* (*continued*)

CSF	*N*	Min	Max	Skewness Statistic	Skewness *SE*	Kurtosis Statistic	Kurtosis *SE*
Good relationship with the customer	106	1	7	–0.04	0.23	–0.33	0.47
Well-defined project scope and objectives	106	1	7	–1.20	0.23	1.99	0.47
Agile-oriented requirement process	106	1	7	–2.21	0.23	4.02	0.47
Agile project management style	106	3	7	–0.19	0.23	–1.43	0.47
Agile-oriented configuration management process	106	1	7	–2.12	0.23	4.86	0.47
Agile-friendly progress tracking mechanism	106	1	7	–1.62	0.23	2.88	0.47
Strong communication focus and rigorous communication schedule	106	1	7	–1.30	0.23	0.68	0.47
Project honored regular working schedule	106	2	7	–0.66	0.23	–0.85	0.47
Strong customer commitment and presence	106	1	7	–2.07	0.23	3.71	0.47
Customer representative had full authority and knowledge to make decisions on-site	106	1	7	–1.14	0.23	0.17	0.47
Well-defined coding standards up front	106	1	7	–0.93	0.23	0.19	0.47
Pursued simple design	106	2	7	0.10	0.23	–0.18	0.47
Pursued vigorous refactoring activities	106	1	7	–1.98	0.23	5.43	0.47
Maintained right amount of documentation	106	1	7	–2.59	0.23	9.01	0.47
Followed continuous and rigorous unit and integration testing strategy	106	1	7	–0.29	0.23	0.00	0.47
Delivered working software regularly within short periods of time	106	2	7	–0.79	0.23	–0.58	0.47
Delivered most important features first	106	2	7	–1.73	0.23	3.99	0.47
Employed proper platforms, technologies, and tools suitable	106	2	7	–0.51	0.23	0.74	0.47

Table 17. *Minimum, Maximum, Skewness, and Kurtosis Statistics* (*continued*)

CSF	*N*	Min	Max	Skewness Statistic	Skewness *SE*	Kurtosis Statistic	Kurtosis *SE*
Provided appropriate technical training to team	106	1	7	–2.23	0.23	4.62	0.47
Project nature was a non-life-critical software project	106	1	7	–1.90	0.23	4.57	0.47
Well defined scope upfront with solid requirements	106	2	7	–1.89	0.23	4.21	0.47
Dynamic, accelerated schedule	106	3	7	–1.34	0.23	1.17	0.47
Up-front, detailed cost evaluation	106	1	7	–0.82	0.23	–0.04	0.47
Up-front risk analysis using agile method	106	1	7	–2.18	0.23	4.42	0.47
Project success	106	2	7	–0.31	0.23	–0.15	0.47

ANOVA Results for Differences in the Extent of Use of CSFs

A one-way ANOVA was conducted to determine whether the extent of use of the identified CSFs of effective communication, user involvement, and use of a quality plan varied with regard to the type of project management model utilized. The independent variable was defined be the type of project management model, while the dependent variables were the CSFs of effective communication, user involvement, and use of a quality plan. A level of significance of 0.05 was used in the statistical test, implying there would be a statistically significant difference in the CSFs between the two methods when the *p* values (sig.) measured less than or equal to the level of significance value of 0.05. This analysis will address Hypotheses 1–3. The ANOVA results are summarized in Table 18. The analysis revealed there were significant differences in 20 out of the 37 CSF items. These included 11 items on effective communication, three items in user involvement, and seven items in use of a quality plan, having *p* values less than or equal to the level of significance of 0.05.

For the 11 items on effective communications, there was significant difference in the following CSFs between the agile and waterfall method of project management:

1. Project management had a light-touch and/or adaptive management style ($F[1] = 38.47$; $p = 0.00$). Agile method ($M = 5.52$) had a better rating for this effective communication item than waterfall method ($M = 3.75$).
2. The organization had a cooperative culture instead of hierarchal. A cooperative culture is one that fosters ad hoc teams driven by the needs of the job at hand, while a hierarchal culture is one that has clear divisions of responsibility and authority ($F[1] = 14.19$; $p = 0.00$). Agile method ($M = 6.00$) had a better rating for this effective communication item than the waterfall method ($M = 5.02$).
3. The organization had a reward system that was appropriate for agile behavior ($F[1] = 5.60$; $p = 0.02$). Agile method ($M = 5.32$) had a better rating for this effective communication item than waterfall method ($M = 4.58$).
4. The organization had an oral culture placing high value on a fluid, face-to-face communication style ($F[1] = 11.20$; $p = 0.00$). Agile method ($M = 5.91$) had a better rating for this effective communication item than the waterfall method ($M = 4.81$).

5. The project delivered working software regularly within short periods of time (F[1] = 10.80; p = 0.00). Agile method (M = 5.36) had a better rating for this effective communication item than waterfall method (M = 4.42).
6. The project had a committed sponsor or a committed organization manager (F[1] = 11.73; p = 0.00). Agile method (M = 6.09) had a better rating for this effective communication item than waterfall method (M = 5.19).
7. The project had a dynamic, accelerated schedule (F[1] = 4.20; p = 0.04). Waterfall method (M = 6.43) had a better rating for this effective communication item than the agile method (M = 6.04).
8. The project had strong communication focus and a rigorous communication schedule (F[1] = 7.67; p = 0.01). Agile method (M = 5.43) had a better rating for this effective communication item than the waterfall method (M = 4.60).
9. The project honored a regular working schedule (i.e., 40-hour work week) and no overtime (F[1] = 5.91; p = 0.02). Waterfall method (M = 5.96) had a better rating for this effective communication item than did agile method (M = 5.25).
10. The project received strong executive support. "Executive" may mean the whole Board of Directors or the CEO, CFO, CIO, and so forth, who influenced the decision making (F[1] = 28.01; p = 0.00). Agile method (M = 6.16) had a better rating for this effective communication item than waterfall method (M = 4.26).
11. The project team was collocated (F[1] =11.02; p = 0.00). Agile method (M = 5.98) had a better rating for this effective communication item than waterfall method (M = 5.19).

For the six CSF items related to the use of a quality plan, there were significant differences in the following CSFs success between the agile method and the waterfall method of project management:

1. The project employed proper platforms, technologies, and tools suitable for agile practice ($F[1] = 16.12$; $p = 0.00$). Agile method ($M = 5.96$) had a better rating for this use of quality plan item than waterfall method ($M = 5.17$).
2. The project followed the agile-oriented requirement process ($F[1] = 7.90$; $p = 0.01$). Waterfall method ($M = 5.75$) had a better rating for this use of quality plan item than agile method ($M = 4.98$).
3. The project had up-front risk analysis done and evaluated for using agile method ($F[1] = 5.46$; $p = 0.02$). Waterfall method ($M = 5.74$) had a better rating for this use of quality plan item than agile method ($M = 5.17$).
4. The project had up-front, detailed cost evaluation done and approved ($F[1] = 23.39$; $p = 0.00$). Waterfall method ($M = 6.04$) had a better rating for this use of quality plan item than agile method ($M = 4.68$).
5. The project imposed a well-defined coding standard up front ($F[1] = 17.21$; $p = 0.00$). Waterfall method ($M = 5.17$) had a better rating for this use of quality plan item than agile method ($M = 6.23$).
6. The project pursued simple design ($F[1] = 28.95$; $p = 0.00$). Agile method ($M = 5.21$) had a better rating for this use of quality plan item than waterfall method ($M = 4.19$).

For the three CSF items on user involvement, there were significant differences in the following CSFs success between the agile method and the waterfall method of project management:

1. The project followed continuous and rigorous unit and integration testing strategy during each and every iteration ($F[1] = 24.55$; $p = 0.00$). Agile method ($M = 5.26$) had a better rating for this use involvement item than the waterfall method ($M = 4.15$).
2. The project nature was a non-life-critical software project, although it could be business mission-critical software ($F[1] = 5.26$; $p = 0.02$). Waterfall method ($M = 6.25$) had a better rating for this use involvement item than agile method ($M = 5.70$).
3. Project management had a good relationship with the customer ($F[1] = 27.46$; $p = 0.00$). Agile method ($M = 5.43$) had a better rating for this use involvement item than waterfall method ($M = 4.26$).

In general, the results of ANOVA showed there were significant differences between the extent of use of effective communication between the Agile and Waterfall model; between the extent of user involvement between the Agile and Waterfall model; and between the extent of the use of a quality plan between the Agile and Waterfall model. The null hypotheses for hypotheses 1–3 were rejected.

Table 18. *ANOVA Results of Difference of CSF Measures Between Agile and Waterfall Project Management*

CSF		*SS*	*df*	*MS*	*F*	Sig.
Received strong executive support	Between groups	93.11	1	93.11	28.01	0.00*
	Within groups	339.05	102	3.32		
	Total	432.15	103			
Committed sponsor or a committed organization manager	Between groups	21.74	1	21.74	11.73	0.00*
	Within groups	192.64	104	1.85		
	Total	214.38	105			
Cooperative culture instead of hierarchal	Between groups	25.51	1	25.51	14.19	0.00*
	Within groups	186.98	104	1.80		
	Total	212.49	105			
Oral culture placing high value on fluid (face-to-face communication style)	Between groups	31.74	1	31.74	11.20	0.00*
	Within groups	294.64	104	2.83		
	Total	326.38	105			
Agile methodology was universally accepted in the organization	Between groups	1.36	1	1.36	0.60	0.44
	Within groups	236.98	104	2.28		
	Total	238.34	105			
Reward system that was appropriate for agile behavior	Between groups	14.35	1	14.35	5.60	0.02*
	Within groups	266.42	104	2.56		
	Total	280.76	105			
Project team was collocated	Between groups	16.64	1	16.64	11.02	0.00*
	Within groups	157.09	104	1.51		
	Total	173.74	105			
Worked in a facility with proper agile-style work environment	Between groups	0.08	1	0.08	0.03	0.86
	Within groups	271.92	104	2.61		
	Total	272.01	105			
High technical competence and expertise team members	Between groups	0.94	1	0.94	0.64	0.42
	Within groups	152.79	104	1.47		
	Total	153.74	105			
Project team members had great motivation and were committed to the project success	Between groups	4.99	1	4.99	1.86	0.18
	Within groups	279.02	104	2.68		
	Total	284.01	105			
Project management was knowledgeable in agile principles and processes	Between groups	5.90	1	5.90	2.47	0.12
	Within groups	248.34	104	2.39		
	Total	254.24	105			

Table 18. *ANOVA Results of Difference of CSF Measures Between Agile and Waterfall Project Management* (*continued*)

CSF		*SS*	*df*	*MS*	*F*	Sig.
Project management had light-touch and/or adaptive management style	Between groups	81.72	1	81.72	38.47	0.00*
	Within groups	218.79	103	2.12		
	Total	300.51	104			
Worked in a coherent, self-organizing teamwork manner	Between groups	3.41	1	3.41	0.94	0.34
	Within groups	377.74	104	3.63		
	Total	381.14	105			
Good relationship with the customer	Between groups	36.26	1	36.26	27.46	0.00*
	Within groups	137.32	104	1.32		
	Total	173.58	105			
Well-defined project scope and objectives	Between groups	0.76	1	0.76	0.53	0.47
	Within groups	148.64	104	1.43		
	Total	149.41	105			
Agile-oriented requirement process	Between groups	15.86	1	15.86	7.90	0.01*
	Within groups	208.79	104	2.01		
	Total	224.65	105			
Agile project management style	Between groups	3.77	1	3.77	2.96	0.09
	Within groups	132.60	104	1.28		
	Total	136.38	105			
Agile-oriented configuration management process	Between groups	1.85	1	1.85	1.00	0.32
	Within groups	192.30	104	1.85		
	Total	194.15	105			
Agile-friendly progress tracking mechanism	Between groups	2.42	1	2.42	1.78	0.19
	Within groups	141.21	104	1.36		
	Total	143.62	105			
Strong communication focus and rigorous communication schedule	Between groups	18.26	1	18.26	7.67	0.01*
	Within groups	247.70	104	2.38		
	Total	265.96	105			
Project honored regular working schedule	Between groups	13.62	1	13.62	5.91	0.02*
	Within groups	239.74	104	2.31		
	Total	253.36	105			
Strong customer commitment and presence	Between groups	0.60	1	0.60	0.35	0.56
	Within groups	181.81	104	1.75		
	Total	182.42	105			

Table 18. *ANOVA Results of Difference of CSF Measures Between Agile and Waterfall Project Management* (*continued*)

CSF		*SS*	*df*	*MS*	*F*	Sig.
Customer representative had full authority and knowledge to make decisions on-site	Between groups	7.93	1	7.93	2.60	0.11
	Within groups	316.79	104	3.05		
	Total	324.73	105			
Well-defined coding standards up front	Between groups	29.58	1	29.58	17.21	0.00*
	Within groups	178.75	104	1.72		
	Total	208.34	105			
Pursued simple design	Between groups	27.51	1	27.51	28.95	0.00*
	Within groups	98.83	104	0.95		
	Total	126.34	105			
Pursued vigorous refactoring activities	Between groups	0.15	1	0.15	0.13	0.72
	Within groups	118.23	104	1.14		
	Total	118.38	105			
Maintained right amount of documentation	Between groups	0.08	1	0.08	0.10	0.75
	Within groups	88.91	104	0.85		
	Total	88.99	105			
Followed continuous and rigorous unit and integration testing strategy	Between groups	32.84	1	32.84	24.55	0.00*
	Within groups	139.09	104	1.34		
	Total	171.93	105			
Delivered working software regularly within short periods of time	Between groups	23.58	1	23.58	10.80	0.00*
	Within groups	227.06	104	2.18		
	Total	250.64	105			
Delivered most important features first	Between groups	0.76	1	0.76	0.61	0.44
	Within groups	130.34	104	1.25		
	Total	131.10	105			
Employed proper platforms, technologies, and tools suitable	Between groups	16.64	1	16.64	16.12	0.00*
	Within groups	107.40	104	1.03		
	Total	124.04	105			
Provided appropriate technical training to team	Between groups	1.85	1	1.85	1.22	0.27
	Within groups	157.51	104	1.51		
	Total	159.36	105			
Project nature was a non-life-critical software project	Between groups	7.93	1	7.93	5.26	0.02*
	Within groups	156.98	104	1.51		
	Total	164.92	105			

Table 18. *ANOVA Results of Difference of CSF Measures Between Agile and Waterfall Project Management* (*continued*)

CSF		*SS*	*df*	*MS*	*F*	Sig.
Well defined scope upfront with solid requirements	Between groups	1.36	1	1.36	1.07	0.30
	Within groups	132.26	104	1.27		
	Total	133.62	105			
Dynamic, accelerated schedule	Between groups	4.16	1	4.16	4.20	0.04*
	Within groups	102.94	104	0.99		
	Total	107.10	105			
Up-front, detailed cost evaluation	Between groups	48.91	1	48.91	23.39	0.00*
	Within groups	217.47	104	2.09		
	Total	266.38	105			
Up-front risk analysis using agile method	Between groups	8.49	1	8.49	5.46	0.02*
	Within groups	161.77	104	1.56		
	Total	170.26	105			

*Significant difference a level of significance of 0.05.

Pearson's Correlation Test Result of Relationship of Extent of CSFs

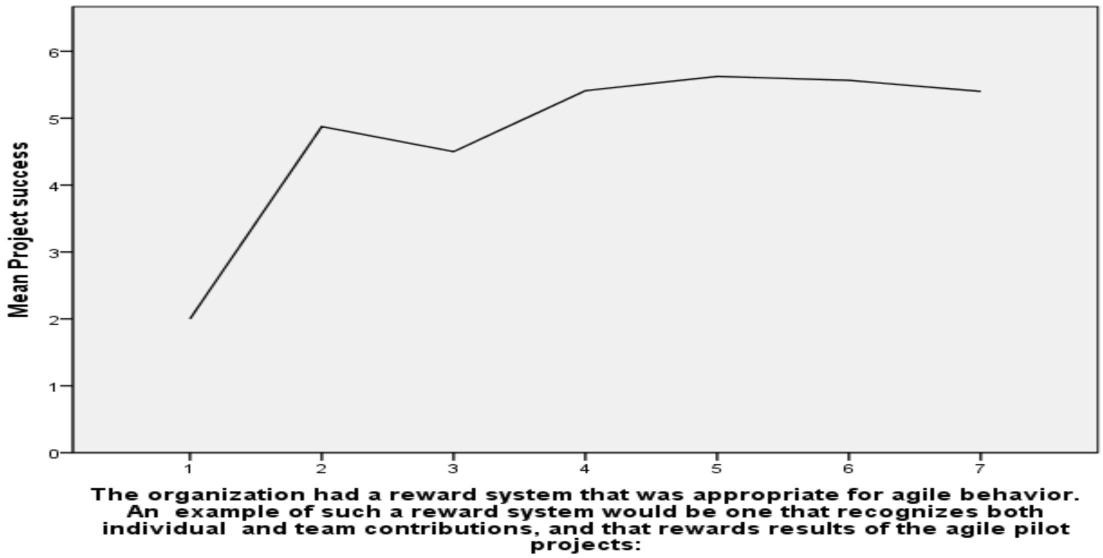
The following analysis investigated the correlation of the different CSFs of effective communication, user involvement, and use of a quality plan in relation to project success. Separate correlation tests were conducted on the different project management models (agile versus waterfall). A level of significance of 0.05 was also used in the hypothesis testing. A significant relationship existed once the p value was less than or equal to the level of significance value. The Pearson's correlation test also investigated the degree of the correlation (positive or negative) and the strength of the correlation.

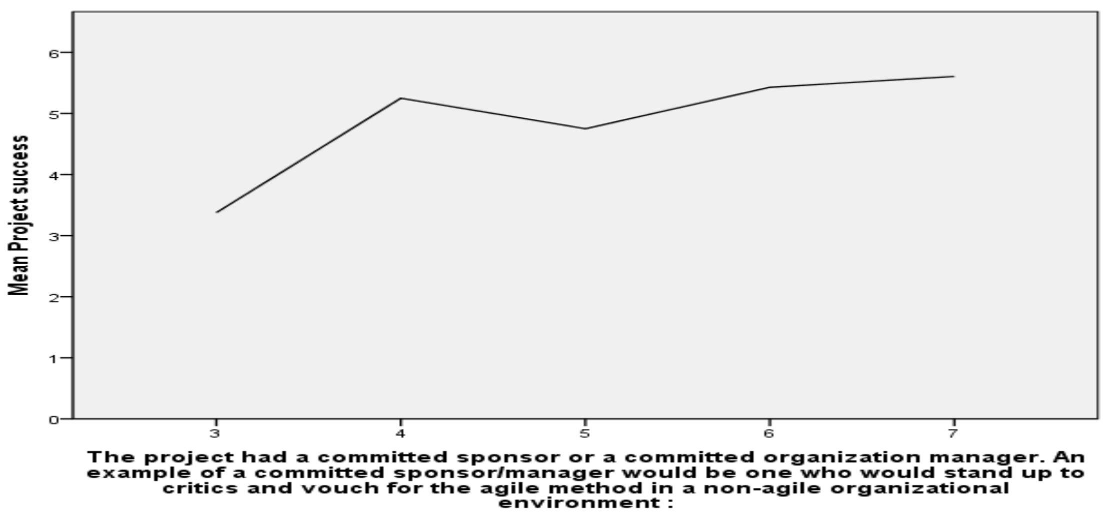
The results of the Pearson's correlation test to determine which among the identified CSFs are correlated with successful projects in the agile project management model are summarized in Table 19. This analysis addressed Hypotheses 4–6. The Pearson's correlation test result showed the existence of significant positive relationships in 13 of the 37 CSF items against the variable, project success. These include three items on effective communication, two items on user involvements, and eights items on use of a quality plan.

For the three CSF items on effective communications in agile project models, project success was significantly, positively related to the following effective communication items:

1. The organization had a reward system that was appropriate for agile behavior ($r[51] = 0.35$, $p = 0.01$).
2. The project had a committed sponsor or a committed organization manager ($r[53] = 0.44$, $p < .0001$).
3. The project received strong executive support. "Executive" may mean the whole Board of Directors or the CEO, CFO, CIO, and so forth, who influenced the decision making ($r[51] = 0.40$, $p < .0001$).

The positive correlations were evident in Table 19 that graphically illustrated the correlations of project success in agile projects with the three CSF items on effective communications. This was because the graphs showed an increasing trend indicating that project success in agile will increase when each of the three identified items on effective communications also increase.





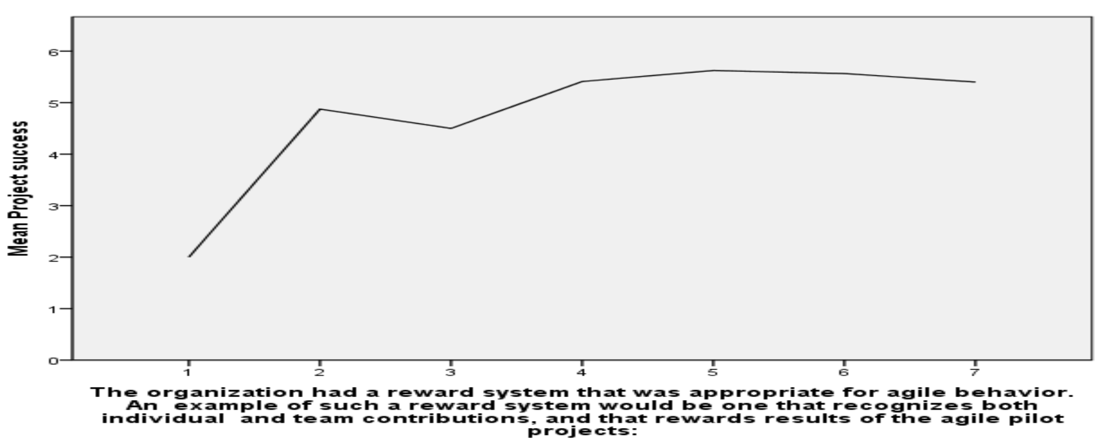


Figure 15. Correlation of project success of Agile projects with CSF items on effective communications.

For the two CSF items on user involvement in agile project models, project success was significantly, positively related to the following user involvement items:

1. Project management had a good relationship with the customer ($r[51] = 0.30$, $p = 0.03$).
2. The project had strong customer commitment and presence ($r[51] = 0.28$, $p = 0.05$).

The positive correlations were evident in Figure 16 that graphically illustrated the correlations of project success in agile projects with the two items on user involvement. This was evident because the graphs showed an increasing trend indicating that project success in agile projects will increase when each of the two identified items on user involvement also increase.

For the eight CSF items on use of a quality plan in agile project models, project success was significantly, positively related to the following use of quality plan items:

1. Project management was knowledgeable in agile principles and processes ($r[53] = 0.34$, $p = 0.01$).
2. The project employed proper platforms, technologies, and tools suitable for agile practice ($r[53] = 0.48$, $p < .0001$).
3. The project followed agile-oriented configuration management processes ($r[53] = 0.39$, $p < .0001$).
4. The project maintained the right amount of documentation for agile purpose (i.e., not too focused on producing elaborate documentation as milestones, but not ignoring documentation altogether either; $r[53] = 0.34$, $p = .01$).

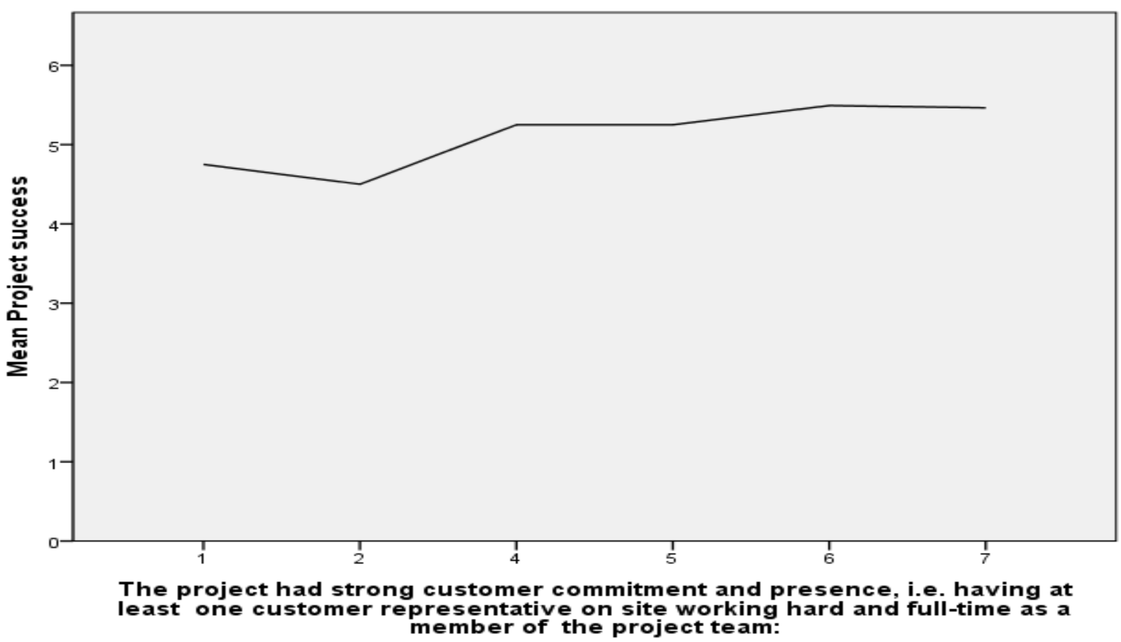


Figure 16. Correlation of project success of Agile projects with CSF items on user involvement.

5. The project provided appropriate technical training to teams, including training on subject matter and agile processes ($r[51] = 0.32$, $p < .0001$).
6. The project pursued vigorous refactoring activities to ensure the results are optimal and to accommodate all changes in requirements ($r[51] = 0.35$, $p = .01$).
7. The project scope and objectives were well defined ($r[51] = 0.40$, $p < .0001$).
8. The selected project team members had high technical competence and expertise (problem solving, programming, subject matter; $r\ (53) = 0.55$, $p < .0001$).

The positive correlations were evident in the Figure 17 that graphically illustrated the correlations of project success in agile projects with the eight CSF items on use of a quality plan. This was demonstrated in the graphs that showed an increasing trend indicating that project success in agile projects will increase when each of the eight identified items on use of a quality plan also increase.

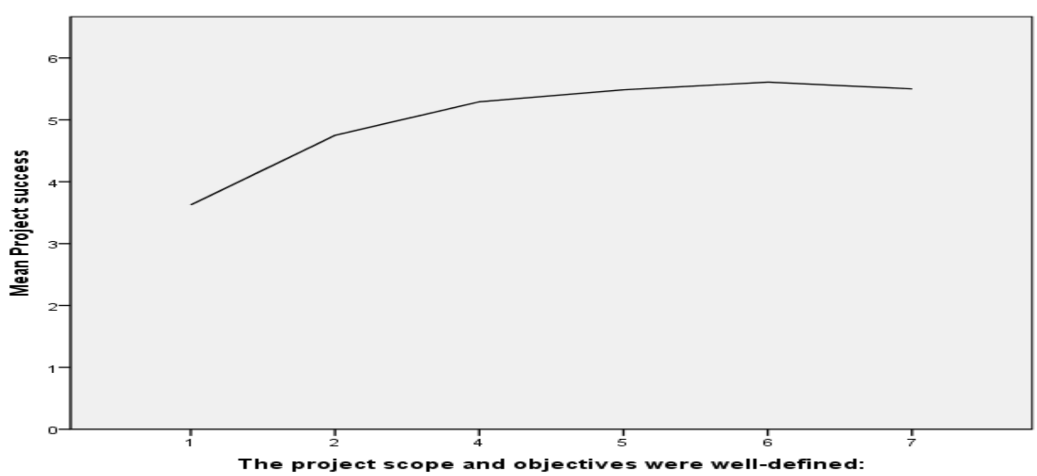


Figure 17. Correlation of project success of Agile projects with CSF items on use of a quality plan.

As a summary, the results of the Pearson's correlation tests showed that project success was significantly associated with effective communication, user involvement, and use of a quality plan in the agile model. The null hypothesis for Hypotheses 4–6 were rejected. The correlations were all positive, implying that a higher extent of the use of CSFs of effective communication, user involvements, and use of a quality plan in agile project management would result in better project success. It should be noted that the strengths of the correlations were all moderate, since all correlation coefficients were between 0.3 and 0.7, in the moderate strength range.

Table 19. *Pearson's Correlation Test Results of Correlation Between CSF Measures and Project Success in Agile Project Management*

CSF		Project success
Received strong executive support	Pearson correlation	0.40*
	Sig. (2-tailed)	0.00
	N	51.00
Committed sponsor or a committed organization manager	Pearson correlation	0.44*
	Sig. (2-tailed)	0.00
	N	53.00
Cooperative culture instead of hierarchal	Pearson correlation	0.24
	Sig. (2-tailed)	0.08
	N	53.00
Oral culture placing high value on fluid (face-to-face communication style)	Pearson correlation	0.04
	Sig. (2-tailed)	0.78
	N	53.00
Agile methodology was universally accepted in the organization	Pearson correlation	0.24
	Sig. (2-tailed)	0.09
	N	53.00
Reward system that was appropriate for agile behavior	Pearson correlation	0.35*
	Sig. (2-tailed)	0.01
	N	53.00

Table 19. *Pearson's Correlation Test Results of Correlation Between CSF Measures and Project Success in Agile Project Management* (*continued*)

CSF		Project success
Project team was collocated	Pearson correlation	0.14
	Sig. (2-tailed)	0.31
	N	53.00
Worked in a facility with proper agile-style work environment	Pearson correlation	0.18
	Sig. (2-tailed)	0.21
	N	53.00
High technical competence and expertise team members	Pearson correlation	0.55*
	Sig. (2-tailed)	0.00
	N	53.00
Project team members had great motivation and were committed to the project success	Pearson correlation	–0.01
	Sig. (2-tailed)	0.94
	N	53.00
Project management was knowledgeable in agile principles and processes	Pearson correlation	0.34*
	Sig. (2-tailed)	0.01
	N	53.00
Project management had light-touch and/or adaptive management style	Pearson correlation	0.11
	Sig. (2-tailed)	0.43
	N	52.00
Worked in a coherent, self-organizing teamwork manner	Pearson correlation	0.25
	Sig. (2-tailed)	0.07
	N	53.00
Good relationship with the customer	Pearson correlation	0.30*
	Sig. (2-tailed)	0.03
	N	53.00
Well-defined project scope and objectives	Pearson correlation	0.40*
	Sig. (2-tailed)	0.00
	N	53.00
Agile-oriented requirement process	Pearson correlation	0.04
	Sig. (2-tailed)	0.77
	N	53.00
Agile project management style	Pearson correlation	0.25
	Sig. (2-tailed)	0.07
	N	53.00

Table 19. *Pearson's Correlation Test Results of Correlation Between CSF Measures and Project Success in Agile Project Management* (*continued*)

CSF		Project success
Agile-oriented configuration management process	Pearson correlation	0.39*
	Sig. (2-tailed)	0.00
	N	53.00
Agile-friendly progress tracking mechanism	Pearson correlation	0.22
	Sig. (2-tailed)	0.11
	N	53.00
Strong communication focus and rigorous communication schedule	Pearson correlation	0.10
	Sig. (2-tailed)	0.48
	N	53.00
Project honored regular working schedule	Pearson correlation	0.22
	Sig. (2-tailed)	0.12
	N	53.00
Strong customer commitment and presence	Pearson correlation	0.28
	Sig. (2-tailed)	0.05*
	N	53.00
Customer representative had full authority and knowledge to make decisions on-site	Pearson correlation	0.21
	Sig. (2-tailed)	0.13
	N	53.00
Well-defined coding standards up front	Pearson correlation	0.21
	Sig. (2-tailed)	0.13
	N	53.00
Pursued simple design	Pearson correlation	0.09
	Sig. (2-tailed)	0.50
	N	53.00
Pursued vigorous refactoring activities	Pearson correlation	0.42*
	Sig. (2-tailed)	0.00
	N	53.00
Maintained right amount of documentation	Pearson correlation	0.34*
	Sig. (2-tailed)	0.01
	N	53.00
Followed continuous and rigorous unit and integration testing strategy	Pearson correlation	0.25
	Sig. (2-tailed)	0.07
	N	53.00

Table 19. *Pearson's Correlation Test Results of Correlation Between CSF Measures and Project Success in Agile Project Management* (*continued*)

CSF		Project success
Delivered working software regularly within short periods of time	Pearson correlation	0.25
	Sig. (2-tailed)	0.07
	N	53.00
Delivered most important features first	Pearson correlation	0.25
	Sig. (2-tailed)	0.07
	N	53.00
Employed proper platforms, technologies, and tools suitable	Pearson correlation	0.48*
	Sig. (2-tailed)	0.00
	N	53.00
Provided appropriate technical training to team	Pearson correlation	0.35*
	Sig. (2-tailed)	0.01
	N	53.00
Project nature was a non-life-critical software project	Pearson correlation	0.08
	Sig. (2-tailed)	0.59
	N	53.00
Well defined scope upfront with solid requirements	Pearson correlation	0.14
	Sig. (2-tailed)	0.30
	N	53.00
Dynamic, accelerated schedule	Pearson correlation	–0.01
	Sig. (2-tailed)	0.92
	N	53.00
Up-front, detailed cost evaluation	Pearson correlation	–0.18
	Sig. (2-tailed)	0.21
	N	53.00
Up-front risk analysis using agile method	Pearson correlation	0.25
	Sig. (2-tailed)	0.07
	N	53.00

*Significant correlation a level of significance of 0.05.

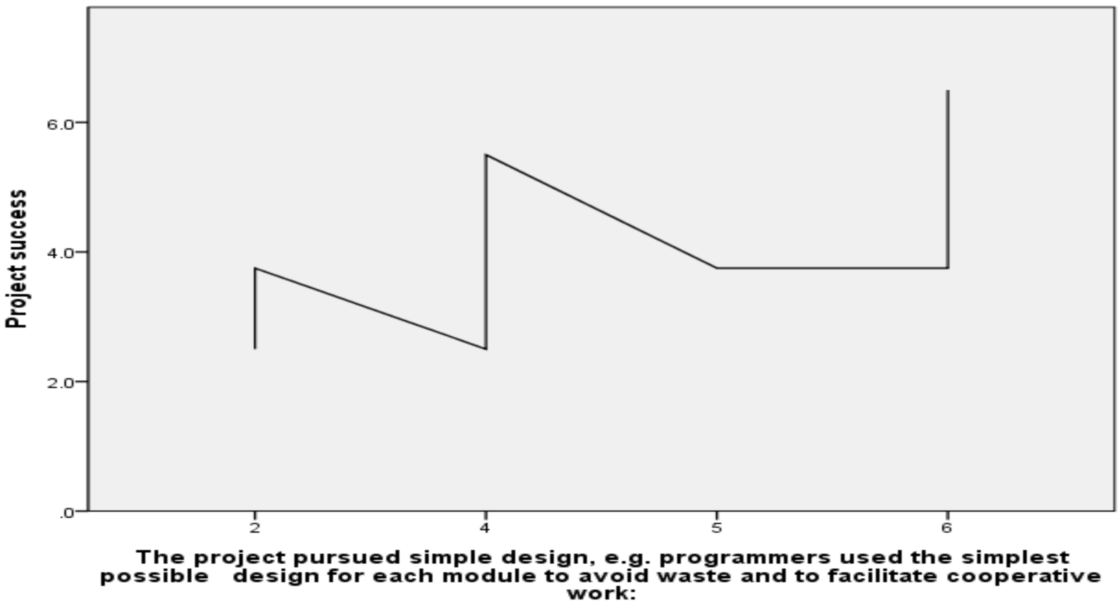
The results of the Pearson's correlation test to determine which among the identified CSFs were correlated with successful projects in the waterfall project management model are summarized in Table 20. This analysis addressed Hypotheses 7–9.

The Pearson's correlation test result showed the existence of a significant positive relationship in four out of the 37 CSF items compared to the variable, project success. These included one item on effective communication and three items in use of a quality plan.

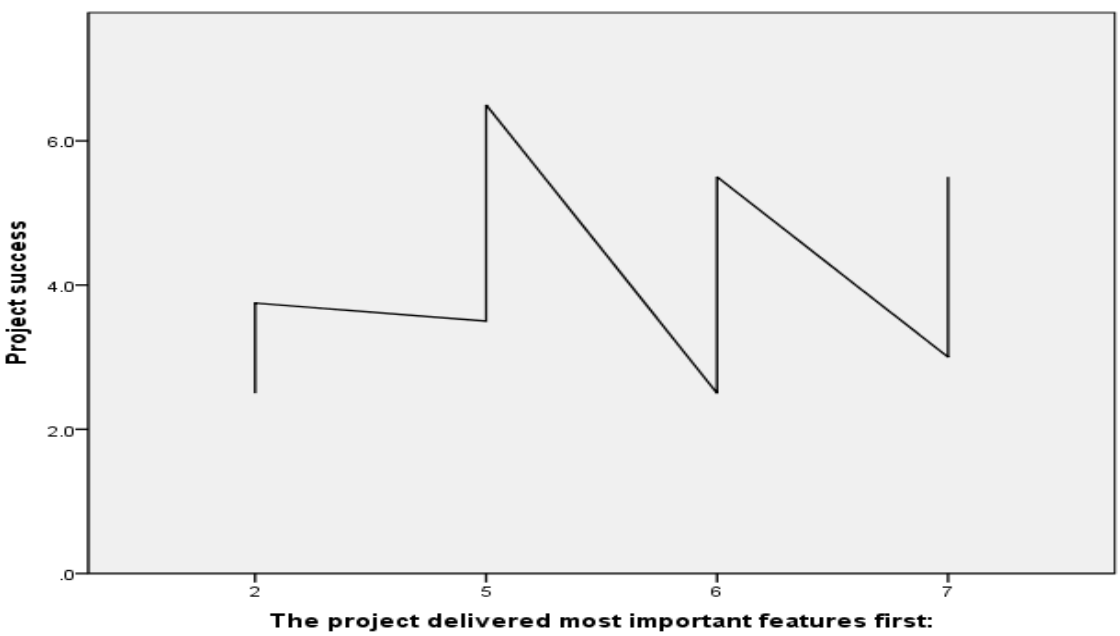
For the one CSF item on effective communications in waterfall project management, project success was significantly, positively related to the following effective communication item of "The organization had an oral culture placing high value on fluid, face-to-face communication style" ($r[53] = 0.39$, $p < .0001$). For the three CSF items on use of a quality plan in waterfall project management, project success was significantly, positively related to the following use of quality plan items:

1. The project pursued simple design ($r[53] = 0.47$, $p < .0001$).
2. The project manager followed an agile-friendly progress tracking mechanism ($r[53] = 0.30$, $p = 0.03$).
3. The project delivered most important features first ($r[53] = 0.31$, $p = 0.02$).

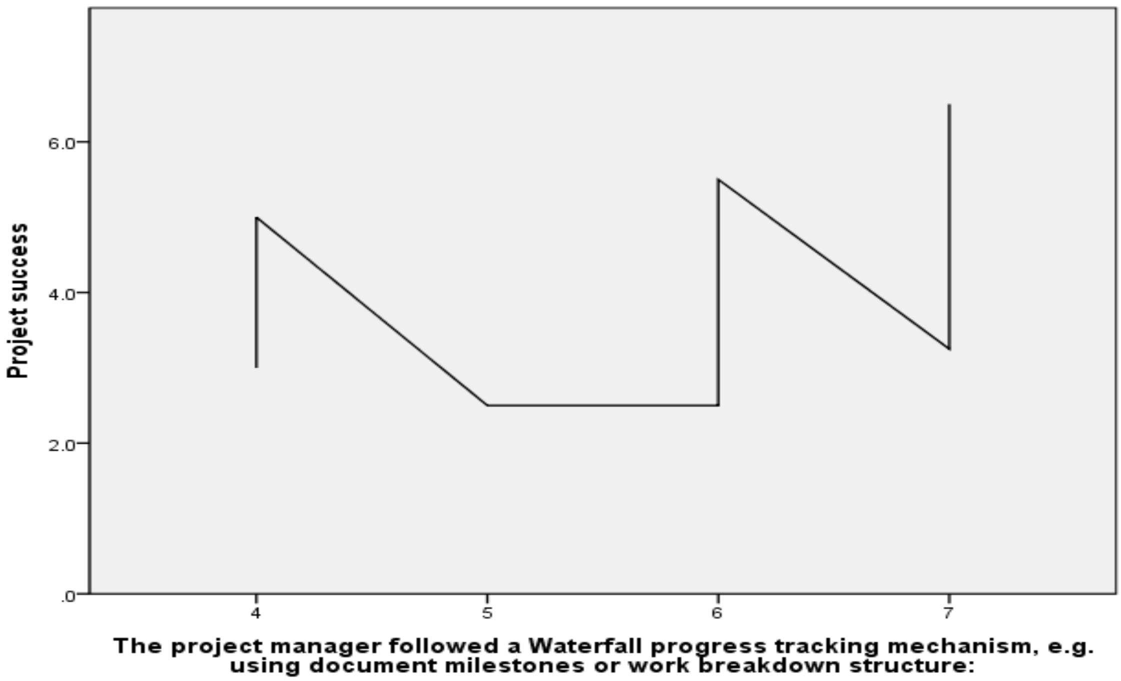
The positive correlations were graphically illustrated in Figure 18. The graph did not show a linear positive trend toward positive correlation of project success with the CSF items on effective communication and use of a quality plan. The graph showed an erratic trend. The positive correlation was not evident in the graph since the strength of the correlations was only moderate. The positive correlation is more evident in the line graph if the r coefficient is close to the value of 1, suggesting a strong correlation.



Project success
6.0
4.0
2.0
.0
2
4
5
6
The project pursued simple design, e.g. programmers used the simplest possible design for each module to avoid waste and to facilitate cooperative work:



Project success
6.0
4.0
2.0
.0
2
5
6
7
The project delivered most important features first:



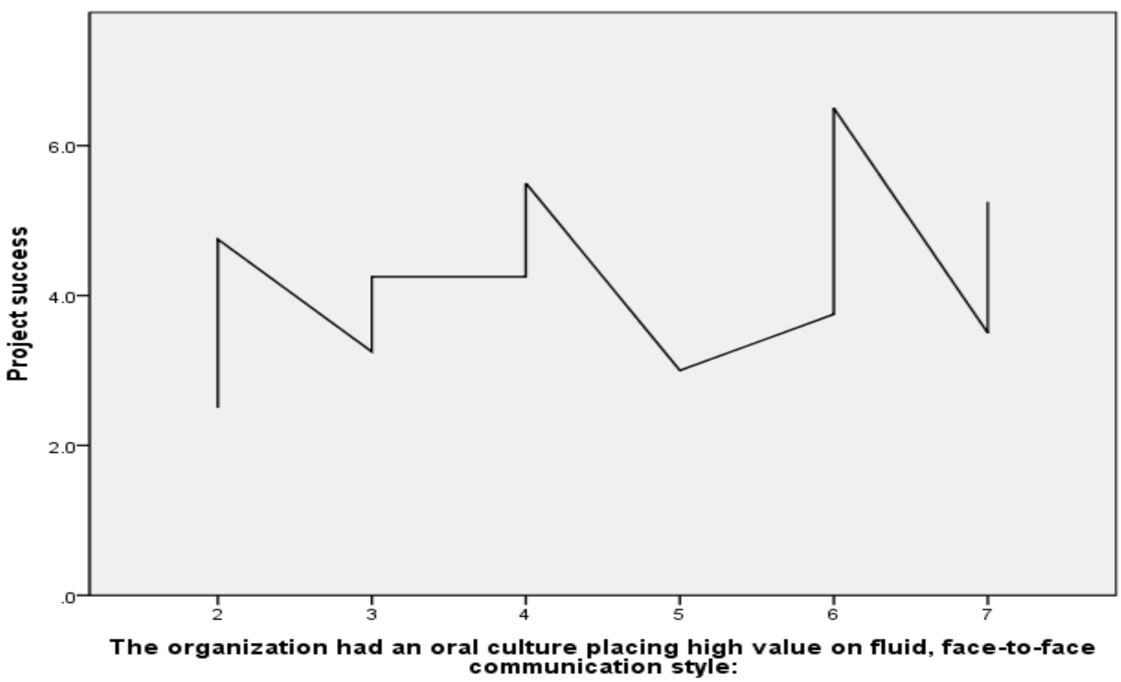


Figure 18. Correlation of project success in Waterfall projects with CSF items on effective communication and use of a quality plan.

As a summary, the results of the Pearson's correlation test showed that project success is significantly associated with effective communication and use of a quality plan in the waterfall model. The null hypothesis for Hypotheses 7 and 9 were rejected. The correlations were positive, implying that higher extent of use of CSFs of effective communication and use of quality plan would result in better project success of projects involving waterfall project management. In addition, the strengths of correlations were all moderate since the *r* correlation coefficients were between 0.3 and 0.7, the moderate strength range. However, Null Hypothesis 8 was not rejected, since the results showed that project success was not significantly associated with user involvement in the waterfall model.

Table 20. *Pearson's Correlation Test Results of Correlation Between CSF Measures and Project Success in Waterfall Project Management*

CSF		Project success
Received strong executive support	Pearson correlation	0.12
	Sig. (2-tailed)	0.39
	N	53.00
Committed sponsor or a committed organization manager	Pearson correlation	0.12
	Sig. (2-tailed)	0.38
	N	53.00
Cooperative culture instead of hierarchal	Pearson correlation	0.15
	Sig. (2-tailed)	0.27
	N	53.00
Oral culture placing high value on fluid (face-to-face communication style)	Pearson correlation	0.39*
	Sig. (2-tailed)	0.00
	N	53.00
Agile methodology was universally accepted in the organization	Pearson correlation	0.08
	Sig. (2-tailed)	0.55
	N	53.00

Table 20. *Pearson's Correlation Test Results of Correlation Between CSF Measures and Project Success in Waterfall Project Management* (*continued*)

CSF		Project success
Reward system that was appropriate for agile behavior	Pearson correlation	0.07
	Sig. (2-tailed)	0.64
	N	53.00
Project team was collocated	Pearson correlation	0.06
	Sig. (2-tailed)	0.65
	N	53.00
Worked in a facility with proper agile-style work environment	Pearson correlation	0.14
	Sig. (2-tailed)	0.33
	N	53.00
High technical competence and expertise team members	Pearson correlation	–0.14
	Sig. (2-tailed)	0.32
	N	53.00
Project team members had great motivation and were committed to the project success	Pearson correlation	0.10
	Sig. (2-tailed)	0.50
	N	53.00
Project management was knowledgeable in agile principles and processes	Pearson correlation	–0.03
	Sig. (2-tailed)	0.84
	N	53.00
Project management had light-touch and/or adaptive management style	Pearson correlation	0.10
	Sig. (2-tailed)	0.47
	N	53.00
Worked in a coherent, self-organizing teamwork manner	Pearson correlation	0.06
	Sig. (2-tailed)	0.65
	N	53.00
Good relationship with the customer	Pearson correlation	0.18
	Sig. (2-tailed)	0.21
	N	53.00
Well-defined project scope and objectives	Pearson correlation	–0.13
	Sig. (2-tailed)	0.35
	N	53.00
Agile-oriented requirement process	Pearson correlation	–0.02
	Sig. (2-tailed)	0.87
	N	53.00

Table 20. *Pearson's Correlation Test Results of Correlation Between CSF Measures and Project Success in Waterfall Project Management* (*continued*)

CSF		Project success
Agile project management style	Pearson correlation	–0.06
	Sig. (2-tailed)	0.69
	N	53.00
Agile-oriented configuration management process	Pearson correlation	0.22
	Sig. (2-tailed)	0.11
	N	53.00
Agile-friendly progress tracking mechanism	Pearson correlation	0.30*
	Sig. (2-tailed)	0.03
	N	53.00
Strong communication focus and rigorous communication schedule	Pearson correlation	0.26
	Sig. (2-tailed)	0.06
	N	53.00
Project honored regular working schedule	Pearson correlation	0.04
	Sig. (2-tailed)	0.76
	N	53.00
Strong customer commitment and presence	Pearson correlation	0.01
	Sig. (2-tailed)	0.95
	N	53.00
Customer representative had full authority and knowledge to make decisions on-site	Pearson correlation	–0.02
	Sig. (2-tailed)	0.90
	N	53.00
Well-defined coding standards up front	Pearson correlation	–0.07
	Sig. (2-tailed)	0.64
	N	53.00
Pursued simple design	Pearson correlation	0.47*
	Sig. (2-tailed)	0.00
	N	53.00
Pursued vigorous refactoring activities	Pearson correlation	0.01
	Sig. (2-tailed)	0.93
	N	53.00
Maintained right amount of documentation	Pearson correlation	0.00
	Sig. (2-tailed)	1.00
	N	53.00

Table 20. *Pearson's Correlation Test Results of Correlation Between CSF Measures and Project Success in Waterfall Project Management* (*continued*)

CSF		Project success
Followed continuous and rigorous unit and integration testing strategy	Pearson correlation	0.10
	Sig. (2-tailed)	0.47
	N	53.00
Delivered working software regularly within short periods of time	Pearson correlation	0.15
	Sig. (2-tailed)	0.29
	N	53.00
Delivered most important features first	Pearson correlation	0.31*
	Sig. (2-tailed)	0.02
	N	53.00
Employed proper platforms, technologies, and tools suitable	Pearson correlation	0.25
	Sig. (2-tailed)	0.07
	N	53.00
Provided appropriate technical training to team	Pearson correlation	0.12
	Sig. (2-tailed)	0.39
	N	53.00
Project nature was a non-life-critical software project	Pearson correlation	0.17
	Sig. (2-tailed)	0.21
	N	53.00
Well defined scope upfront with solid requirements	Pearson correlation	0.09
	Sig. (2-tailed)	0.51
	N	53.00
Dynamic, accelerated schedule	Pearson correlation	–0.05
	Sig. (2-tailed)	0.74
	N	53.00
Up-front, detailed cost evaluation	Pearson correlation	–0.10
	Sig. (2-tailed)	0.50
	N	53.00
Up-front risk analysis using agile method	Pearson correlation	–0.05
	Sig. (2-tailed)	0.70
	N	53.00

*Significant correlation a level of significance of 0.05.

Multiple Regression Analysis of Extent CSFs

The results of the multiple linear regression analysis to determine how influential the identified CSFs on the success of the project using agile project management model and using waterfall project management model are summarized in Tables 21 and 22, respectively. This analysis addressed Research Questions 2.3 and 2.4. The results of the regression model showed that none of the CSFs of effective communication, use of quality plan, and user involvement significantly influenced the success of the project using both the agile and waterfall project management models, since all the *p* values were greater than the level of significance value of 0.05.

Table 21. *Regression Results of Extent of Influence of CSF in the Project Success in Agile Project Management*

Model		Unstandardized coefficients B	Unstandardized coefficients *SE*	Standardized coefficients β	*t*	Sig.
1	(Constant)	1.57	5.78		0.27	0.79
	Received strong executive support	0.17	0.16	0.22	1.02	0.33
	Committed sponsor or a committed organization manager	0.35	0.48	0.35	0.72	0.49
	Cooperative culture instead of hierarchal	–0.16	0.55	–0.21	–0.29	0.78
	Oral culture placing high value on fluid (face-to-face communication style)	–0.10	0.39	–0.13	–0.25	0.81
	Agile methodology was universally accepted in the organization	0.02	0.24	0.04	0.09	0.93
	Reward system that was appropriate for agile behavior	0.15	0.27	0.25	0.54	0.60

Table 21. *Regression Results of Extent of Influence of CSF in the Project Success in Agile Project Management* (*continued*)

Model		Unstandardized coefficients		Standardized coefficients		
		B	*SE*	β	*t*	Sig.
1	Project team was collocated	0.16	0.30	0.21	0.53	0.60
	Worked in a facility with proper agile-style work environment	–0.18	0.39	–0.35	–0.45	0.66
	High technical competence and expertise team members	0.29	0.33	0.47	0.89	0.39
	Project team members had great motivation and were committed to the project success	0.00	0.22	0.01	0.02	0.98
	Project management was knowledgeable in agile principles and processes	–0.22	0.39	–0.36	–0.55	0.59
	Project management had light-touch and/or adaptive management style	0.21	0.22	0.31	0.94	0.36
	Worked in a coherent, self-organizing teamwork manner	0.33	0.52	0.65	0.64	0.53
	Good relationship with the customer	0.14	0.33	0.21	0.42	0.68
	Well-defined project scope and objectives	0.37	0.39	0.56	0.95	0.36
	Agile-oriented requirement process	–0.01	0.56	–0.02	–0.01	0.99
	Agile project management style	0.43	0.56	0.59	0.77	0.46
	Agile-oriented configuration management process	0.14	0.18	0.28	0.77	0.46
	Agile-friendly progress tracking mechanism	–0.38	0.78	–0.57	–0.49	0.63
	Strong communication focus and rigorous communication schedule	0.07	0.40	0.13	0.19	0.86
	Project honored regular working schedule	–0.18	0.20	–0.28	–0.89	0.39

Table 21. *Regression Results of Extent of Influence of CSF in the Project Success in Agile Project Management* (*continued*)

Model		Unstandardized coefficients		Standardized coefficients		
		B	*SE*	β	*t*	Sig.
1	Strong customer commitment and presence	–0.07	0.72	–0.09	–0.09	0.93
	Customer representative had full authority and knowledge to make decisions on-site	–0.73	0.61	–1.39	–1.19	0.26
	Well-defined coding standards up front	0.16	0.42	0.29	0.38	0.71
	Pursued simple design	–0.26	0.77	–0.30	–0.34	0.74
	Pursued vigorous refactoring activities	0.32	0.37	0.41	0.85	0.41
	Maintained right amount of documentation	0.15	0.72	0.18	0.21	0.84
	Followed continuous and rigorous unit and integration testing strategy	–0.10	0.69	–0.15	–0.14	0.89
	Delivered working software regularly within short periods of time	–0.61	0.59	–0.82	–1.03	0.32
	Delivered most important features first	–0.28	0.63	–0.33	–0.45	0.66
	Employed proper platforms, technologies, and tools suitable	0.23	0.50	0.31	0.46	0.65
	Provided appropriate technical training to team	0.00	0.37	0.00	0.00	1.00
	Project nature was a non-life-critical software project	0.50	0.60	0.74	0.83	0.42
	Well defined scope upfront with solid requirements	–0.16	0.39	–0.29	–0.42	0.68
	Dynamic, accelerated schedule	–0.19	0.61	–0.22	–0.31	0.76
	Up-front, detailed cost evaluation	–0.13	0.17	–0.25	–0.78	0.45
	Up-front risk analysis using agile method	0.31	0.26	0.57	1.20	0.25

Note. $F(37, 12) = 1.61$, Sig. = 0.19, $R^2 = 0.83$, $N = 49$.
a. Dependent Variable: Project Success (agile). b. Predictors: CSF (agile).

Table 22. *Regression Results of Extent of Influence of CSF in the Project Success in Waterfall Project Management*

		Unstandardized coefficients		Standardized coefficients		
Model		B	*SE*	β	*t*	Sig.
1	(Constant)	0.00	0.00		0.00	1.00
	Received strong executive support	0.00	0.00	0.00	0.00	1.00
	Committed sponsor or a committed organization manager	0.00	0.00	0.00	0.00	1.00
	Cooperative culture instead of hierarchal	0.00	0.00	0.00	0.00	1.00
	Oral culture placing high value on fluid (face-to-face communication style)	0.00	0.00	0.00	0.00	1.00
	Agile methodology was universally accepted in the organization	0.00	0.00	0.00	0.00	1.00
	Reward system that was appropriate for agile behavior	0.00	0.00	0.00	0.00	1.00
	Project team was collocated	0.00	0.00	0.00	0.00	1.00
	Worked in a facility with proper agile-style work environment	0.00	0.00	0.00	0.00	1.00
	High technical competence and expertise team members	0.00	0.00	0.00	0.00	1.00
	Project team members had great motivation and were committed to the project success	0.00	0.00	0.00	0.00	1.00
	Project management was knowledgeable in agile principles and processes	0.00	0.00	0.00	0.00	1.00
	Project management had light-touch and/or adaptive management style	0.00	0.00	0.00	0.00	1.00

Table 22. *Regression Results of Extent of Influence of CSF in the Project Success in Waterfall Project Management* (*continued*)

Model		Unstandardized coefficients B	Unstandardized coefficients *SE*	Standardized coefficients β	*t*	Sig.
1	Worked in a coherent, self-organizing teamwork manner	0.00	0.00	0.00	0.00	1.00
	Good relationship with the customer	0.00	0.00	0.00	0.00	1.00
	Well-defined project scope and objectives	0.00	0.00	0.00	0.00	1.00
	Agile-oriented requirement process	0.00	0.00	0.00	0.00	1.00
	Agile project management style	0.00	0.00	0.00	0.00	1.00
	Agile-oriented configuration management process	0.00	0.00	0.00	0.00	1.00
	Agile-friendly progress tracking mechanism	0.00	0.00	0.00	0.00	1.00
	Strong communication focus and rigorous communication schedule	0.00	0.00	0.00	0.00	1.00
	Project honored regular working schedule	0.00	0.00	0.00	0.00	1.00
	Strong customer commitment and presence	0.00	0.00	0.00	0.00	1.00
	Customer representative had full authority and knowledge to make decisions on-site	0.00	0.00	0.00	0.00	1.00
	Well-defined coding standards up front	0.00	0.00	0.00	0.00	1.00
	Pursued simple design	0.00	0.00	0.00	0.00	1.00
	Pursued vigorous refactoring activities	0.00	0.00	0.00	0.00	1.00

Table 22. *Regression Results of Extent of Influence of CSF in the Project Success in Waterfall Project Management* (*continued*)

Model		Unstandardized coefficients		Standardized coefficients		
		B	*SE*	β	*t*	Sig.
1	Maintained right amount of documentation	0.00	0.00	0.00	0.00	1.00
	Followed continuous and rigorous unit and integration testing strategy	0.00	0.00	0.00	0.00	1.00
	Delivered working software regularly within short periods of time	0.00	0.00	0.00	0.00	1.00
	Delivered most important features first	0.00	0.00	0.00	0.00	1.00
	Employed proper platforms, technologies, and tools suitable	0.00	0.00	0.00	0.00	1.00
	Provided appropriate technical training to team	0.25	0.00	0.30	4100132.19	0.00
	Project nature was a non-life-critical software project	0.25	0.00	0.49	14165895.45	0.00
	Well defined scope upfront with solid requirements	0.25	0.00	0.48	31612882.57	0.00
	Dynamic, accelerated schedule	0.25	0.00	0.36	9095779.62	0.00

Note. $F(35, 12)$ = Sig. = a, $R^2 = 1.00$, $N = 52$. Up-front, detailed cost evaluation. Up-front risk analysis using agile method.
a. Dependent Variable: Project Success (waterfall). b. Predictors: CSF (waterfall).

Chapter Summary

In summarizing the results of this study, analysis of the ANOVA showed that there were significant differences between the extent of use of effective communication, user involvement, and use of a quality plan between the agile and waterfall models. The results of the Pearson's correlation test showed that project success is significantly, positively associated with effective communication, user involvement, and use of a quality plan in projects using agile project management. Results of the Pearson's correlation test showed that project success is significantly and positively associated with only effective communication and use of a quality plan in projects using waterfall project management.

CHAPTER 5. DISCUSSION, IMPLICATIONS, RECOMMENDATIONS

Introduction

Previous studies have identified the importance of project management methodologies to achieve project success. The understanding of the nature of the project's inherent risks is equally important in preventing project failure. The success or failure of any project can be influenced by many factors. According to the Project Management Institute (PMI, 2013), projects consist of five phases, termed initiating, planning, executing, monitoring and controlling, and closing. Mismanagement at any of these phases can result in the failure of the entire project. Failure can occur in different phases due to project management practices in areas such as risk management, quality management, procurement management, communication, or user training. Thus said, effective communication and meeting the stakeholder's objective are essential tools in project management practices.

The success of a project has different, and sometimes conflicting, definitions. For instance, Lewis (2006) stated that project success is achieved by meeting the required expectations of the project stakeholders and business requirements goals for project. On the other hand, Shenhar and Peerasit Patanakul (2011) argued that while project success is always determined based on time and cost criteria, this does not necessarily apply for all projects.

However, regardless of all possible definitions, project managers need to understand what are the success factors for their projects. Project stakeholders should define the requirements for project success in the early phase of project. Standard success criteria should be considered to measure the successful implementation and delivery of any projects. In line with this view, the purpose of this quantitative, nonexperimental descriptive study was to relate the use of the agile or waterfall methodologies and specific CSFs to IT project success for a sample of IT project managers who have used the agile and waterfall methodologies.

To achieve this purpose, an online survey instrument was used to collect data from a nonprobability sample of participants with IT project management experience. The online survey collected data on IT project success rates and the methodology used for project delivery. In addition, data were collected to determine the role that is played by CSFs in the delivery of successful IT projects through the use of the agile and waterfall project management methodologies. Statistical analysis of the data was done using ANOVA, Pearson's correlation test, and multiple linear regression.

This chapter summarizes the previous four chapters of the study, including the results of the data analysis. The chapter includes a discussion of whether the goal and objectives of the study were achieved. Accordingly, suggestions are also made to enable further research to be conducted where it is required. The chapter also contains a summary of the assumptions on which the study was based and the limitations that could affect the validity of the study results.

Summary of Results

Two main research questions provided the basis for seven investigative questions. The first research question investigated the difference in the extent of use of various CSFs in the agile and waterfall models. This research question was further dissected into the three specific CSFs that were of importance to the study, namely effective communication, user involvement, and the use of a quality plan. To resolve this first research question, a series of ANOVAs was conducted to determine whether the agile and waterfall project management methodologies significantly differed with regards to the extent of use of the following CSFs: effective communication, user involvement, and the use of a quality plan. The results of the ANOVA indicated that for 20 out of the 37 CSF items, the two methods differed significantly. This included 11 items on effective communication, three items related to user involvement, and six items in use of a quality plan. Based on these results, the three null hypotheses under the first main research question were rejected. It is, therefore, concluded that the agile and waterfall methods differed significantly with regards to the extent of use of the following CSFs: effective communication, user involvement, and the use of a quality plan.

The second research question provided the framework for determining which among the three identified CSFs (effective communication, user involvement and the use of a quality plan) were correlated with successful projects using the two project management methodologies (agile and waterfall). To resolve this research question, a Pearson's correlation analysis was conducted to determine the existence and nature of a significant relationship between the study variables. The results of the Pearson correlation analysis indicated that in the agile model, project success had a moderate, direct correlation to three items on effective communication, two items on user involvement, and eight items on use of a quality plan. This direct correlation demonstrates that under the agile model, the higher extent of use of effective communication, user involvement, and use of a quality plan was correlated with higher chances of project success. Based on these results, Null Hypotheses 4–6 were rejected. A separate set of Pearson's correlation analysis used data on the waterfall model. The results indicated that project success was directly correlated with effective communication and the use of a quality plan. These results were the basis for rejecting Hypotheses 7 and 9. Hypothesis 8, which deals with the relationship between project success and user involvement in the waterfall model, was not rejected. The results indicated that a higher extent of use of effective communication and use of a quality plan in the waterfall method was associated with higher project success scores.

The correlation analysis identified the existence and nature of the relationship between the variables. However, the results of a correlation analysis cannot be used as the basis for determining causal relationships between variables. Therefore, two regression models were generated to determine the effect of communication, user involvement, and use of a quality plan on project success. For both agile and waterfall models, effective communication, user involvement, and use of a quality plan did not significantly influence project success.

Discussion

This research study set out to explore the critical success factors used in agile and waterfall software development projects. The study utilized a quantitative approach. The data were collected from 53 project managers who had worked with the agile model and another 53 project managers who had worked with the waterfall model. These project managers were recruited from a diverse group of organizations of various sizes, from various industries, and geographical locations. The data collected from this sample provided enough empirical information for the statistical analysis procedures on which the conclusions of this study are based.

The dynamic nature of the IT industry has contributed to the difficulty of defining the concept of project success. While in some cases cost, time, and quality are considered to be the predominant criteria, some researchers have suggested that project success is more complex than this simplistic definition (Rosenau, 1984; Shenhar et al., 1997; Sommer, 2004). The aim of this study was to determine the factors that are critical to the success of a project in order to decrease the chances of project failure. In relation to this aspect, some factors were identified as common causes of project failure, such as missed project deadlines, quality and high cost of final project, lack of upfront project planning, including risk management, and applying change control processes, among others. But a review of the literature identifies other causes of project failure, such as the inability to meet the stakeholder’s requirements, which underscores the importance of identifying the criteria for success before commencing a project. In contrast, tools to avoid project failure include project planning, applying risk management, developing realistic budget estimates, providing proper breakdown of development and implementation into manageable steps, using effective communication with stakeholders and project teams, developing valid and realistic business needs for new solutions, conducting user working groups to understand user requirements, and aligning between the project and the organization's strategic goals and priorities. Critical factors such as requirements, statement of work to be achieved, and budget must be aligned at the initiation of the project. There is a need to develop a proper plan, and it is advisable to have it reviewed by external parties. Similarly, these plans must be flexible enough to accommodate the changes that may need to be made through the course of the project.

While most technological breakthroughs are rooted in new software development, the process of software engineering is often problematic (Sommer, 2004). Project managers have to deal with problems like cost overruns, schedule delays, and poor quality of final deliverables or output, leading to the need for project managers to develop a process to address these problems. In light of this, it was the goal of this study to contribute to increased project success rates by highlighting the relationship of project success to three specific CSFs: effective communication, user involvement, and the use of a quality plan. While this study considered two project management models, the review of existing literature indicated that the waterfall model is a flawed software method that has many inadequacies. These inadequacies include the model's failure to allow going back and forth between phases. This can lead to a problem with cost efficiency, since any changes made in the requirements of the project require starting from the beginning because the waterfall model does not easily allow for alteration of previous phases. Hence, this may point to the need for a project management model that is flexible, yet cost efficient.

Recently, a new class in the software development process entitled agile methods has become prevalent (Phatak, 2012). A review of the literature indicates there has been a shift from the waterfall to agile model. Compared to other project management models, the agile model is gaining in popularity because it facilitates the timely, within-budget delivery of software products. This premise is what this study was based upon.

In line with the findings of this study, the three CSFs of effective communication, user involvement, and the use of a quality plan were directly correlated with project success under the agile model. Under the agile process, teams work closely together in support of the project. Likewise, communication with the client continues throughout the process, operating as an iterative cycle. Customers are given demonstrations after each iteration and feedback is used as the basis for the next course of action. This repeats until the product meets the specifications of the customer (Phatak, 2012). Based on this, it is apparent that communication between the members of the project team and the customer is a key factor in the success of the agile project management model. Also, the flexibility afforded by the agile model may be a challenge for software engineers who are used to project managers who exert strong leadership. In light of this, the exercise of effective communication becomes vital so that every member of the project team is informed about any potential changes in the project and is able to respond in a timely fashion to prevent any potential issues or major problems. In addition, it gives team members a chance to be well prepared during discussions with stakeholders, giving them a better idea of what questions to ask clients. Similarly, Sliger and Broderick (2008) found that face-to-face meetings with clients are one of the critical success factors that make the agile model a good sell to stakeholders.

The described relationship between the project team and the client in the agile model is a manifestation of user involvement, which was directly correlated to success for projects completed under the agile model. The strong presence and commitment of the client in the completion of the project was a CSF that was associated with project success. Under the agile model, user involvement is part of the repeated iteration of the process to continually improve the product based on feedback and updated requirements from the client.

Lastly, while the flexibility or adaptability of the agile method is one of its greatest strengths, project managers using this model should be cautioned against merely changing for the sake of change. Applying the concept of adaptability from biological evolution, adaptation should occur when the changes presented to the environment are those that generate beneficial results. In line with this, all changes should still be made within the context of the overall project plan. Similarly, a review of literature revealed that there are several key CSFs that apply to all project management models, but should be specifically addressed in agile methodologies due to their relatively open-ended nature. Hirshfield (2010) noted the importance of developing realistic timeframes and expectations, based on a thorough and well-presented plan.

The PMI (2013) recommended the use of a specific project management methodology because one common cause of unsuccessful projects is the failure to employ a coherent methodology or the application of the wrong process (Al-Ahmad et al., 2009; Johnson, 2006; Milosevic, 2003). It should also be noted that the choice of which project management methodology to implement should also take into consideration not just the unique needs and qualities of the organization (Kendrick, 2009) but also the industry or domain that the project is classified under (PMI, 2013). In light of this, while the conclusions of this study encourage the use of the agile model for most projects, there may be industries wherein the waterfall model is the most appropriate choice. For commercial products and services that have no connection or impact on patient safety and welfare, the use of the agile method is recommended because of its quick process of development. This is especially true in nonregulated software development environments. This project management model is recommended for fast-growing software development industries, such as e-commerce, online marketing, and social networking (e.g., Facebook, Google, eBay, Yahoo, LinkedIn). On the other hand, the waterfall model is recommended for highly regulated environments or industries, such as health care, aviation software, patient safety software, medical device software, or biotechnological applications. Software that is geared towards these industries requires step-by-step validation, which is more appropriate for the waterfall model. The waterfall method is also appropriate for science and technology-based industries, such as drug manufacturing, biomedical research and development, and information technology management in the healthcare domain. These industries require massive compliance testing, validation, and compliance with regulatory bodies, such as the U.S. Food and Drug Administration, and therefore cannot be completed with haste.

Limitations and Assumptions

The results of the study are limited by the assumptions described in Chapter 1. Specifically, the results of the study were reached on the assumption that the field test provided evidence of the validity and reliability of the measures used, as well as the relevance of the constructs of the study. This assumption was already validated, as discussed in the subsequent paragraph, by the results of the Cronbach's alpha analysis for reliability and internal consistency. Secondly, it was also assumed that since the survey methodology was used with a large sample, the findings of the study could be generalizable to the target population. Thirdly, it was assumed that the study's web-administered instrument would provide anonymity for the participants, the capacity to conduct many surveys in a shorter period of time, and to have a shorter turnaround time for data collection (Cooper & Schindler, 2011), which should have contributed to the validity of the data collected. Lastly, it was assumed that research using a critical factors approach can help managers of software development and IT organizations determine which project management methodology is most applicable for their organization's individual needs.

The limitations of this study include the time constraints. Similarly, the reliability test conducted for the study instrument and the individual study variables indicated that the Cronbach's alpha for Project Success (Waterfall) measured 0.42, less than the minimum acceptable value of 0.70. The minimum value is generally considered a requirement to ensure the acceptability, reliability, and internal consistency of the constructs. Internal consistency refers to the ability of the items in an instrument to measure a single construct, and the failure to meet this minimum required value indicates that the items for the variable of Project Success (Waterfall) may not accurately reflect the quantification of this variable. Therefore, this is considered another limitation of the study that could potentially affect the validity of the conclusions derived from the results of the data analysis. Aside from this, the results are limited by the fact that only English speakers participated in this study. The subject that was investigated is of interest to organizations around the world. The results could have been affected or enriched by data contributed by non-English-speaking IT project managers. This is an aspect of interest for future researchers and will be discussed in the Recommendations section of this chapter.

Recommendations for Future Research

This exploratory research responded to the need for more empirical evidence on the benefits of using the agile software development methodology. It adds to the current existing knowledge on the critical success factors of agile projects, which has largely been composed of anecdotal evidence. Considering the limitations encountered in this study, modifications need to be made in future research studies in order to improve the validity of the results of the quality of findings and conclusions that are based on these results. Several suggestions are also made for future areas of exploration to help expand the current body of knowledge on the subject.

It has been recommended that while project management models that are based on the agile method share common characteristics, there may be unique features that differentiate it from specific agile models. It is recommended that future researchers collect data from practitioners of specific agile methods to identify features that are unique to each. Based on the findings of this study, an empirical comparison of these models, particularly with regards to critical success factors, can be conducted to add to the body of knowledge on the subject.

In relation to the limitation of the study to English-speaking project managers, it is recommended that this study be replicated to include other languages. A suggested method would be to have the instrument translated into a variety of languages and administered in other geographical locations where English is not the primary language.

Also, given that software development is a global industry, the agile method is likely to be implemented in other countries by individuals with unique technological and cultural backgrounds. In light of this, it is recommended that studies on the process of software development be expanded to include other aspects or elements of the process, such as the human, social, organizational, and logistical elements. Further research in this particular area could contribute to the creation of a more comprehensive and well-rounded perspective on the success of the agile model.

It is also suggested that future researchers investigate the individual elements of the study through a phenomenological perspective. The study can be conducted by recruiting participants from organizations that implement agile. The suggested focus for the study could be the underlying motivations, assumptions, and practices applied and used in project management. Participants could be interviewed on how the effective communication, user involvement, and quality plans contribute to the success of projects managed using agile models. The findings that result from the suggested study can be used to support or enrich the findings from this study.

Also, further studies are recommended on the outcomes of projects and programs using other methodologies, such as Prince2. This study could also be replicated for projects from other industrial sectors, such as healthcare management, drug manufacturing, or biomedical research. The manufacturing and education industries are other areas of interest. It was suggested that the agile method might not be the most appropriate project management methodology for the unique needs and attributes of these industries; therefore, an empirical study would support this assertion. In connection with this recommendation, it is also suggested that future researchers consider failed projects that used the agile method in these industries in identifying factors that contributed to the failures. Another avenue for future research could be replicating this same study, but focusing on other CSFs, such as support from management, risk management, or human resource management (Cao, 2006).

Conclusion

The purpose of this quantitative, nonexperimental descriptive study was to relate the use of the agile or waterfall methodologies and specific CSFs to IT project success for a sample of IT project managers who had used the agile and waterfall methodologies. Based on the data collected and the subsequent data analysis, the results of the study indicated that the agile and waterfall project management methods significantly differed with regards to the extent of use of the three CSFs used in the study, namely effective communication, user involvement, and use of a quality plan. The analysis of the results also revealed that for the agile model, the three CSFs had a direct, moderate correlation with project success. For the waterfall model, only effective communication and use of a quality plan had a direct moderate correlation with project success. No predictive relationships were found between any of the variables for both project management models.

Through the literature review, it was found that the waterfall model is a flawed method with many inadequacies and that the agile model was rapidly responding to a need for a more flexible and cost-efficient project management model. This descriptive study adds to existing knowledge on the critical success factors of agile projects, by providing empirical or quantifiable evidence. It also revealed various areas of study that still need to be explored, as detailed in the Recommendations section. In conclusion, it is asserted that while the findings of this study add to what is already known, there are still more questions that need to be answered by further research and exploration on the subject.

REFERENCES

Adams, J. R., & Lovedahl, K. G. (1983, August). *Project Management Quarterly* special report: Ethics, standards, and accreditation. *Project Management Quarterly, 14,* 68.

Agarwal, B. B., Tayal, S. P., & Gupta, M. (2010). *Software engineering and testing.* Sudbury, MA: Jones & Bartlett.

Al-Ahmad, W., Al-Fagih, K., Khanfar, K., Alsamara, K., Abuleil, S., & Abu-Salem, H. (2009). A taxonomy of an IT project failure: Root causes. *International Management Review, 5*(1), 93.

Atkinson, R. (1999). Project management: Cost, time, and quality. *International Journal of Project Management, 17*(6), 337–342.

Augustine, S., & Cuellar, R. (2006). The lean-agile PMO: Using lean thinking to accelerate project delivery. *Cutter Consortium*, *7*(10), 1–24.

Baumgartner, J. S. (1963). *Project management.* Homewood, IL: Irwin.

Bavani, R. (2009, February). *Critical success factors in distributed agile for outsourced product development.* Paper presented at the Computer Society of India International Conference on Global Software Engineering, Mumbai.

Benediktsson, O., Dalcher, D., & Thorbergsson, H. (2006). Comparison of software development life cycles: A multiproject experiment. *IET Journals and Magazines, 153*(3), 87–103.

Benington, H. D. (1983). Production of large computer programs. *IEEE Annals of the History of Computing, 5*(4), 350–361. doi:10.1109/MAHC.1983.10102

Bennington, P., & Baccarini, D. (2004). Project benefits management in IT projects: An Australian perspective. *Project Management Journal, 35*(2), 20–30.

Boynlon, A. C., & Zmud, R. W. (1984). An assessment of critical success factors. *Sloan Management Review, 25*(4), 17–27.

Camilleri, E. (2011). *Project success: Critical factors and behaviors.* New York, NY: Gower.

Cao, D.-B. (2006). *An empirical investigation of critical success factors in agile software development projects.* Doctoral dissertation, Capella University, Minneapolis, MN.

Charavat, J. (2003). *Project management methodologies: Selecting, implementing, and supporting methodologies and processes for projects.* Hoboken, NJ: Wiley.

Church, A. H., & Waclawski, J. (1998). *Designing and using organizational surveys.* Hampshire, England: Gower.

Cleland, D. I., & King, R. W. (1983). *Industrial project management: System analysis and project management.* New York, NY: McGraw-Hill.

Cockburn, A. (2004). *Crystal clear: A human-powered methodology for small teams.* Boston, MA: Addison-Wesley.

Cohen, S., Dori, D., & Haan, U. (2010). A software system development life cycle model for improved stakeholders communication and collaboration. *International Journal of Computers, Communications, & Control, 5*(1), 20–41.

Cohn, M. (2009). *Succeeding with agile: Software development using scrum.* Boston, MA: Addison-Wesley.

Cook-Davies, T. (2002). The "real" success factors in projects. *International Journal of Project Management, 20*(3), 185–190.

Cooper, C. R., & Schindler, P. S. (2011). *Business research methods.* Boston, MA: McGraw-Hill.

Cottmeyer, M. (2009). Agile PMP: Managing software projects in the face of uncertainty. *Global Congress Proceedings.* Retrieved from http://ebookee.org/dl/Agile-PMP-Managing-software-projects-in-the-face-of-uncertainty/

Creswell, J. W. (2009). *Research design: Qualitative, quantitative, and mixed methods approaches.* Thousand Oaks, CA: Sage.

Creswell, J. W., & Plano Clark, V. L. (2011). *Designing and conducting mixed methods research.* Thousand Oaks, CA: Sage.

Cronbach, L. J. (1951). Coefficient alpha and the internal structure of tests. *Psychometrika, 16*(3), 297–334.

DeCarlo, D. (2010). *Extreme project management: Using leadership, principles and tools to deliver value in the face of volatility.* New York, NY: Wiley.

Doloi, H. (2009). Relational partnerships: The importance of communication, trust and confidence and risk management in achieving project success. *Construction Management and Economics, 27*(1), 1099–1109.

Edmonds, E. A. (1974). A process for the development of software for nontechnical users as an adaptive system. *General Systems, 19,* 215–218.

Extreme programming pocket guide. (2003). San Francisco, CA: O'Reilly Media.

Fewell, J. (2009). *Looking for an edge: Agile is emerging as the new weapon in the battle against economic slump.* Retrieved from http://www.pmi.org/

Fewell, J., Jack, M., Prior, D., Rosado, P., & Tarne, B. (2009). Challenges in implementing agile project management. *Global Congress Proceedings.* Retrieved from http://marketplace.pmi.org/Pages/ProductDetail.aspx?GMProduct =00101191700

Field, A. (2013). *Discovering statistics using SPSS.* Thousand Oaks, CA: Sage.

Fink, A. (2003). *The survey handbook* (2nd ed.). Thousand Oaks, CA: Sage.

Fowler, J. J. (2007). Are information systems' success and failure factors related?: An exploratory study. *Journal of Organizational and End User Computing, 19*(2), 1–22.

Gibbs, R. D. (2006). *Project management with the IBM rational unified process: Lessons from the trenches.* Upper Saddle River, NJ: IBM Press.

Glass, R. L. (2006). Looking into the challenges of complex IT projects. *Communications of the ACM, 49*(11), 15–17. doi:10.1145/1167838.1167850.

Gray, P. S., Williamson, J. B., & Karp, D. A. (2008). *The research imagination: An introduction to qualitative and quantitative methods.* England: Cambridge University Press.

Graziano, A. M., & Raulin, M. L. (2009). *Research methods: A process of inquiry* (7th ed.). Boston, MA: Allyn & Bacon.

Gregory, J. (2009). *Agile testing*: *A practical guide for testers and agile teams.* Boston, MA: Addison-Wesley.

Griffiths, M. (2004). *Utilizing agile principles alongside: A guide to the project management body of knowledge for better project execution and control in software development projects.* Newtown Square, PA: Project Management Institute.

Griffiths, M. (2007). *Developments in agile project management.* Retrieved from http://leadinganswers.typepad.com/files/developments-in-agile-project-management---mike-griffiths.pdf

Hair, J., Black, W., Babin, B., Anderson, R., & Tatham, R. (2006). *Multivariate data analysis* (6th ed.). Upper Saddle River, NJ: Pearson Prentice Hall.

Harpum, P. (2004). *Portfolio, program, and project management in the pharmaceutical and biotechnology industries.* Hoboken, NJ: Wiley.

Harris, J. (2006). Managing change in IT improvement initiatives. *Government Finance Review, 22*(1), 36–40. Retrieved from http://www.amazon.com/Managing-change-improvement-initiatives-Government/dp/B000EYK6VU

Haughey, D. (2009, September). *Waterfall v agile: How should I approach my software development project?* Retrieved from http://www.projectsmart.com/articles/waterfall-v-agile-how-should-i-approach-my-software-development-project.php

Haughey, D. (2012). *Project management jumpstart.* New York, NY: Wiley.

Hedeman, B., & Seegers, R. (2009). *PRINCE2™—A pocket guide.* Zaltbommel, Netherlands: Van Haren.

Heldman, K. (2011). *Project management jumpstart 3.* San Francisco, CA: Wiley.

Higgins, J. (2006). *The radical statistician: A beginners guide to unleashing the power of applied statistics in the real world* (5th ed.). Sacramento, CA: Author.

Hirshfield, M. (2010). *Critical success factors for healthcare software implementations.* Retrieved from http://docsfiles.com/pdf_critical_success_factors_for_healthcare_software_implementations.html

Hoffer, J. A., George, J. F., & Valacich, J. S. (2011). *Modern systems analysis and design.* Boston, MA: Prentice Hall.

Hong, W., Thong, J. Y. L., Chaslow, L. C., & Dhillon, G. (2011). User acceptance of agile information systems: A model and empirical test. *Journal of Management Information Systems, 28*(1), 237–272.

Ika, L. (2009). Project success as a topic in project management journals. *Project Management Journal, 40*(4), 6–19.

International Council on Systems Engineering. (2010). [Home page.] Retrieved from http://www.incose.org/

James, K. L. (2008). *Software engineering.* New Delhi, India: PHI Learning.

Johnson, J. (2006). *My life is failure: 100 things you should know to be a successful project leader.* West Yarmouth, MA: Standish Group International.

Johnston, A. K. (1995). *A hacker's guide to project management.* Oxford, England: Butterworth-Heinemann.

Johnstone, D., Huff, S., & Hope, B. (2006). IT projects: Conflict, governance, and systems thinking. *System Sciences, 8,* 197b.

Kendrick, C. (2009). *Project success factors and information technology project success* (Doctoral dissertation). Available from ProQuest Dissertations and Theses database. (UMI No. 3387347)

Kerzner, H. (1982). *Project management for executives.* New York, NY: Wiley.

Kerzner, H. (2003). *Project management: A systems approach to planning, scheduling and controlling.* New York, NY: Wiley.

Kerzner, H. (2006). *Advanced project management: Best practices on implementation.* New York, NY: Wiley.

Kerzner, H. (2009). *Project management: A systems approach to planning, scheduling, and controlling.* Hoboken, NJ: Wiley.

Kline, R. B. (2010). *Principles and practice of structural equation modeling* (3rd ed.). New York, NY: Guilford.

Kruchten, P. (2004). *The rational unified process: An introduction to the rational unified process.* Boston, MA: Addison-Wesley.

Kuruppuarachchi, P. R., Mandal, P., & Smith, R. (2002). IT project implementation strategies for effective changes: A critical review. *Logistic Information Management, 15*(2), 126–137.

Lam, J. C. Y., & Lee, M. K. O. (2006). Digital inclusiveness—Longitudinal study of Internet adoption by older adults. *Journal of Management Information Systems, 22*(4), 177–206.

Laudon, K. C., & Laudon, J. P. (2002). *Management information systems activebook.* Upper Saddle River, NJ: Prentice Hall.

Leffingwell, D. (2007). *Scaling software agility: Best practices for large enterprises.* Boston, MA: Addison-Wesley.

Legris, P., & Collerette, P. (2006). A roadmap for IT project implementation—Integrating stakeholders and change management issues. *Project Management Journal, 27(*5), 64–75.

Leidecker, J., & Bruno, A. (1984). Identifying and using critical success factors. *Long Range Planning, 17,* 23–32.

Lenth, R. V. (2001). Some practical guidelines for effective sample-size determination. *American Statistician, 55*(3), 187–193.

Lewis, J. P. (2006). *Fundamentals of project management.* New York, NY: AMACOM.

Lindstrom, L., & Jeffries, R. (2004). Extreme programming and agile software development methodologies. *Information Systems Management, 21*(3), 41–52.

Lyytinen, K., & Hirschheim, R. (1987). Information systems failures—A survey and classification of the empirical literature. *Oxford Surveys in Information Technology, 4,* 257–309.

Mahanti, A. (2006). Challenges in enterprise adoption of agile methods—A survey. *Journal of Computing and Information Technology, 3,* 197–206.

Milosevic, D. Z. (2003). *Project management toolbox: Tools and techniques for the practicing project manager.* New York, NY: Wiley.

Mirabella, J. (2008). *Hypothesis testing with SPSS: A non-statistician's guide and tutorial.* Retrieved from http://www.drjimmirabella.com/ebook/

Morris, P. W. G. (2004, May). Science, objective knowledge, and the theory of project management. *ICE Civil Engineering, 150,* 82–90. doi:10.1680/cien.2002.150.2.82

Mumford, E. (2010). *How is QA different for waterfall, agile, and XP?* Retrieved from http://www.websoftwareqa.com/2010/08/how-is-qa-different-for-waterfall-agile-and-xp/

Nan, N., & Harter, D. E. (2009). Impact of budget and schedule pressure on software development cycle time and effort. *IEEE Transactions on Software Engineering, 35*(5), 624–637.

Nardi, P. M. (2005). *Doing survey research: A guide to quantitative methods.* New York, NY: Pearson/Allyn & Bacon.

Nelson, R. (2005). Project retrospectives: Evaluating project success, failure, and everything in between. *MIS Quarterly Executive, 4*(3), 361–372. Retrieved from http://www2.commerce.virginia.edu/cmit/Research/MISQE%209-05.pdf

Nerur, S., & Balijepally, V. (2007). Theoretical reflections on agile development methodologies. *Communications of the ACM, 50*(3), 78–84.

Neuman, W. L. (2011). *Basics of social research: Qualitative and quantitative approaches* (3rd ed.). Whitewater: University of Wisconsin.

Northern, C., Matfield, K., Benito, R., & Casagni, M. (2010, December 15). *Handbook of implementing agile in Department of Defense information technology acquisition.* Retrieved from http://www.mitre.org/sites/default/files/pdf/11_0401.pdf

Oisen, R. P. (1971). Can project management be defined? *Project Management Quarterly, 2*(1), 12–14.

Pallant, J. (2005). *SPSS survival manual.* New York, NY: McGraw-Hill.

Pellissier, R. (2008). *Business research made easy.* Cape Town, South Africa: Juta.

Petch, J., & Reeve, D. (1999). *GIS, organizations and people: A socio-technical approach.* Philadelphia, PA: CRC Press.

Phatak, O. (2012). *Waterfall vs. agile model comparison.* Retrieved from http://www.buzzle.com/articles/waterfall-model-vs-agile.html

Pinto, J. K., & Slevin, D. P. (1989). Critical success factors in R&D projects. *Research Technology Management, 32*(1), 31–35.

Praeg, C. P., & Spath, D. (2009). Perspectives of IT-service quality management: A concept of life-cycle based quality management of IT services. In A. Cater-Steel (Ed.), *Information technology governance and service management: Frameworks and adaptations* (pp. 381–407). Hershey, PA: IGI Global.

Preuss, D. H. (2006). *Interview: Jim Johnson of the Standish Group.* Retrieved from http://www.infoq.com/articles/Interview-Johnson-Standish-CHAOS

Project Management Institute. (2013). *A guide to the* Project Management Body of Knowledge*.* Retrieved from http://www.pmi.org/PMBOK-Guide-and-Standards.aspx

Pyzdek, T. (2000). *The Six Sigma handbook.* New York, NY: McGraw-Hill.

Ramaprasad, A., & Williams, J. (1998). The utilization of critical success factors: A profile. Paper presented at the 29th annual meeting of the Decision Sciences Institute, Las Vegas, NV.

Rockart, J. (1979). Chief executives define their own information needs. *Harvard Business Review, 25,* 81–92.

Rockart, J., & Bullen, C. (1981). *A primer on critical success factors*. Cambridge: Massachusetts Institute of Technology, Center for Information Systems Research.

Rodriguez-Repiso, L., Setchi, R., & Salmeron, J. L. (2007). Modeling IT projects success: Emerging methodologies reviewed. *Technovation, 27*(10), 582–594.

Rosenau, M. D. (1984). *Project management for engineers.* New York, NY: Lifetime Learning.

Royce, W. (1970). Managing the development of large software systems. *Proceedings of IEEE WESCON*. Retrieved from http://www.cs.umd.edu/class/spring2003/cmsc838p/Process/waterfall.pdf

Satalkar, B. (2011). *Comparison between waterfall model and spiral model.* Retrieved from http://www.buzzle.com/articles/waterfall-model-vs-v-model.html

Schwalbe, K. (2010). *Information technology project management.* Independence, KY: Cengage.

Sekaran, U., & Bougie, R. (2010). *Research methods for business: A skill building approach.* Hoboken, NJ: Wiley.

Shadish, W. R., Cook, T. D., & Campbell, D. T. (2001). *Experimental and quasi-experimental designs for generalized causal inference.* Independence, KY: Cengage.

Shenhar, A. J., & Dvir, D. (2007). Project management research: The challenge and opportunity. *Project Management Journal, 38*(2), 93–100.

Shenhar, A. J., Levy, O., & Dvir, D. (1997). Mapping the dimensions of project success. *Project Management journal, 28*(2), 5–14.

Shenhar, A. J., & Peerasit Patanakul, P. (2011). What is really project strategy: The fundamental building block in strategic project management. *Project Management Journal, 43*(1), 4–20. doi:10.1002/pmj.20282

Sliger, M., & Broderick, S. (2008). *The software project manager's bridge to agility.* Boston, MA: Addison-Wesley.

Somers, T. M., & Nelson, K. (2001, January). *The impact of critical success factors across the stages of enterprise resource planning implementations.* Paper presented at the 34th Hawaii International Conference on System Sciences, Honolulu.

Sommer, D. (2004, October). *Project failure—12 mistakes to avoid.* Paper presented at the PMI Global Congress North America, Atlanta, GA.

Soper, D. (2013). *Statistical calculator: A-priori sample size calculator for multiple regression.* Retrieved from http://www.danielsoper.com/statcalc3/calc.aspx?id=1

Stair, R. M., & Reynolds, G. W. (2009). *Principles of information systems.* Independence, KY: Cengage.

Standing, C., Guilfoyle, A., Lin, C., & Love, P. (2006). The attribution of success and failure in IT projects. *Industrial Management & Data Systems, 106*(8), 1148–1165.

The Standish Group. (2008). *The trends in IT value.* Yarmouth, MA: The Standish Group International.

The Standish Group. (2010). *CHAOS.* Retrieved from http://www.standishgroup.com

The Standish Group. (2011). *CHAOS.* West Yarmouth, MA: The Standish Group International.

Stuckenbruck, L. C. (1981). *The implementation of project management: The professional's handbook.* New York, NY: Addison-Wesley.

SurveyMonkey. (2013). [Home page.] Retrieved from http://www.surveymonkey.com/

Swanson, R. A., & Holton, E. F. (2005). *Research in organizations: Foundations and methods in inquiry.* San Francisco, CA: Berrett-Koehler.

Thillaisthanam, N. (2013). *Agile methodology: When processes clash with principles.* Retrieved from http://www.siliconindia.com/

Thomas, G., & Fernandez, W. (2008). Success in IT projects: A matter of definition? *International Journal of Project Management, 26*(7), 733–742.

Trochim, W. M. K. (2001). *The research methods knowledge base* (2nd ed.). Cincinnati, OH: Atomic Dog.

United States Air Force, Department of Defense. (1969). *Military standard systems engineering management.* Washington, DC: Author.

U.S. Department of Health & Human Services. (1979). *The Belmont report: Ethical principles and guidelines for the protection of human subjects of research.* Retrived from http://www.hhs.gov/ohrp/humansubjects/guidance/belmont.html

Verner, J., & Evanco, W. (2005). In-house software development: What project management practices lead to success? *IEEE Software, 22*(1), 86–93.

Vogt, W. P. (2010). *OM8020: Quantitative research methods for professionals.* Boston, MA: Pearson.

Wang, H. C., & Doong, H. S. (2010). Nine issues for Internet-based survey research in service industries. *Service Industries Journal, 30*(14), 2387–2399.

Waterfall model. Advantages, examples, phases and more about software development. (2012). Retrieved from http://www.waterfall-model.com/

Watson, P. (2009). A methodology for attaining and assessing project success for rehabilitation projects. *Journal of Building Appraisal, 4*(3), 181.

West, D. (with Gilpin, M., Leganza, G., & D'Silva, D.). (2009, July 10). *Balance architecture and agility to sustain value delivery report.* Retrieved from http://www.forrester.com

When is a model agile? (2013). Retrieved from http://www.agilemodeling.com/essays/whenIsAModelAgile.htm

Wideman, M. R. (1986). *A framework for project and program management integration.* Newtown Square, PA: Project Management Institute.

Winston, R. (1970, August). Managing the development of large software systems. *Proceedings of IEEE WESCON, 26,* 1–9.

Zammuto, R. F., & Krakower, J. Y. (1991). Quantitative and qualitative studies of organisational culture. *Research in Organisational Change and Development, 5,* 83–114.

Zoomerang. (2013). [Home page.] Retrieved from www.zoomerang.com