

# Software Project Management and Its Related Factors

Humphery Nwachukwu, Sayeed Sajal  
Department of Computer Science  
Minot State University  
500 University Ave W, Minot, ND, 58707  
Sayeed.Sajal@minotstateu.edu

## Abstract

In view of the rapid demand for software and the rapid growth of technology, change is inevitable. Software development has gone from just a one-man working on a piece of software with just a few believers into an industry with more than one person working on a piece of software with many believers. Most of our devices has gone from analog to digital, we are becoming more and more connected. Internet of Things (IoT) is now the way, stay connected is now a common catchphrase for tech advertisers. From our watches to our phones, our businesses are now online, the next generation will never know a world where they exist no software. With this rapid growth, one might be thinking that all software projects started were completed but after series of research, we discovered that not only that some software projects weren't completed but tons of money where lost in the process of creating software's that failed. In this paper, we will be discussing some important factors to be considered in software project management. Some of this factor include project management methodologies (agile, waterfall and spiral); scheduling in software project management; and cost estimation.

## 1. Overview

All software projects regardless of their size need a leader called a project manager and a management methodology. Big organizations like Apple and Microsoft needs it the most. Project management has been in existence for quite a long time but just as everything changes so does software project management. In our world today, the need for software is relentless and the idea of the future cannot be accomplished without it. Improved methodologies [1] of software project management are evolving to suit our high-end tech demands, time, quality, and cost of making software is serially improving as we learn more about better ways to manage software projects. Project management covers all that is to be done to ensure that a finished product is produced. Choosing an accurate project management methodology [5] that is suitable for the project at hand is closely as important as the delivery date and quality of the product because it defines how the iterative processes involved in software development will be carried out. Scheduling is a big factor in project

management, the effects of scheduling are liking having your goods delivered when you don't need them anymore. Most poor scheduling occurs due to a failure in some project management iterative processes not necessary just not delivering the project when needed. Cost estimation, cost estimation is like making an educative guess on what building software will cost and if gone wrong will lead to the failure of the software project because running a project highly depends on money.

## 2. Project Management Methodologies

### A. Agile Project Management Method

This method is adapted for changing projects like software projects. Customers are constantly bringing in new ideas, and this consistent behavior of constant change is one of the reasons why agile is suitable for most software projects. According to H. Frank [2], the four core principles of agile project management were; individual and interaction over process and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, responding to change over following a plan. The listed principles can be obviously observed in most projects using agile methodology. Agile project methodology emphasis on the need for risk minimization using short iterations of clearly defined deliverable and direct communication with partners during the development process in the project, according to H. Frank Carvone [2]. Just like they are many different projects, so they are many ways of applying agile methods to most projects. Some of the common ways as a state by H. Frank Carvone [2] are “Scrum, extreme project management, adaptive project management, and dynamic project management method. Through thorough inspection of agile projects, one could conclude that agile is most suited for changing or dynamic project. In figure 1, the agile thought process diagram [6] is shown.

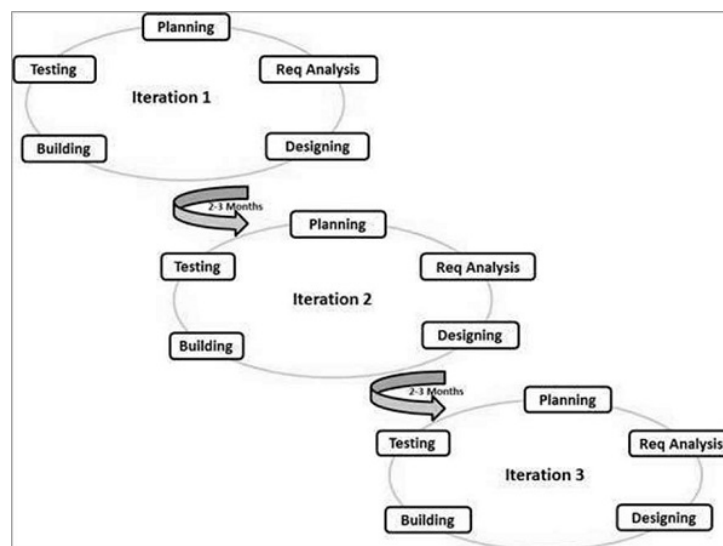


Figure 1: Agile thought Process

The whole project is broken down into iterations which helps minimize risk, at every iteration, working software is built and with increasing iteration, new working features are added to the build, so the final iteration has all the need features for complete working software.

## B. Waterfall Project Management Method

The idea of waterfall methodology as proposed by Dr. Royce is based on strict documentation and plan following. You plan completely then you execute hoping that things don't change, unlike the agile project management methodology that gives enough room for changes. Making changes in the waterfall model can be extremely expensive as the only place where changes can be implemented is within the feedback loop leading from one stage or iteration to the next. A waterfall diagram is shown in Figure 2.

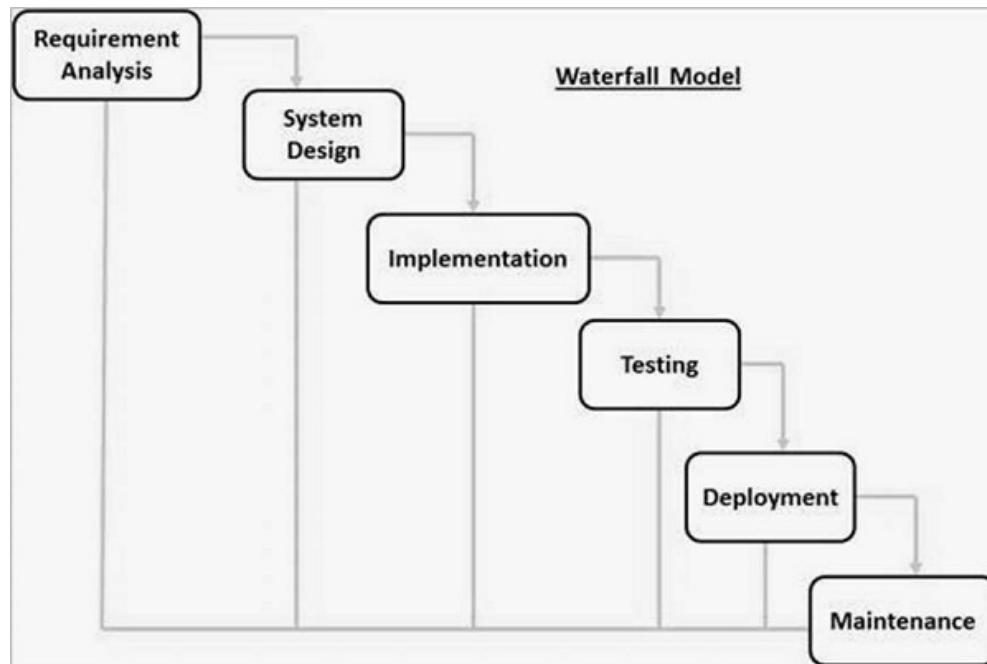


Figure 2: Process in waterfall method [6]

## C. Spiral project Management Method

This methodology is a product of the combination of ideas from both waterfall and agile methodology. It incorporated the sequential idea running a software project and agile iterative ideal i.e., it is paying attention to the risks involved in project development like a waterfall and it iteratively releases products like agile. A diagram for the spiral method is shown in Figure 3.

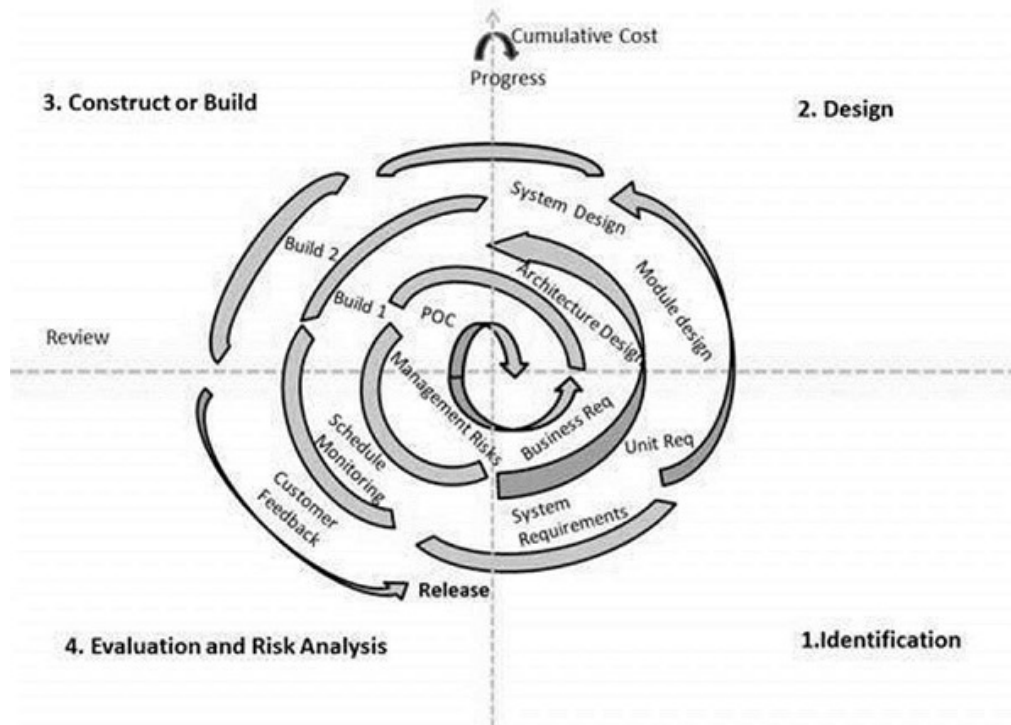


Figure 3: Process in the spiral method [6]

**Identification Phase:** In this phase, information relating to business requirements, identification requirements, subsystem requirement, and unit system requirement are gathered.

**Design Phase:** In this phase all the design starts, the architectural design, module design, and system design.

**Construct /Build Phase:** This phase executes the production of the software in every spiral phase.

**Evaluation and Risk Analysis:** This phase risk monitoring, schedule monitoring management risk are checked. This is the final phase so thorough checking is carried out.

### 3. Scheduling in Software Project Management.

Project scheduling [3] refers to the allocation of time to all the activities involved in project development. This project's activity is divided into tasks and milestones with serious attention to activities in the critical path because those activities make a big difference in the scheduling process.

Scheduling problems are classified into three categories

- i. The field  $\alpha$ : resource-oriented category
- ii. The field  $\beta$ : activity-oriented category
- iii. The field  $\gamma$ : performance-oriented category

Scheduling sometimes might not be accurate since other external factors like organizational issues could influence the outcome of the project.

## 4. Cost Optimization in Software Project Management

In a generally simplified term, it is making an exaggerated guess of what building a software might cost. Accurate cost estimate for a piece of software is always difficult because to some extent it's like predicting the future from experience and future projection. They have been cases in the past like the 1989 Dutch police automated identifying system failure where cost estimation has been a real kicker, the cost of building the Dutch police automated identifying system was estimated to cost \$21million but after series of audit it was found to cost around \$43million and the system did not meet the need it was meant for. Cost estimation is a big factor to consider in software development, here are some of the important techniques according to [4] through which cost estimation is made.

- i. Estimation based on available capacity
- ii. Estimation based on price to win
- iii. Estimation based on parametric module
- iv. Estimation by expert
- v. Estimation based on reasoning by analogy

Several cost estimations modules these includes

**Constructive Cost Module:** This is recorded as one of the best and most transparent modules currently in use. According to [4], it's focused on estimation is made after making an estimation of 15 cost drivers in a project.

**The Function Point Analysis Module:** This is based on the function software is meant to serve.

**Before You Leap Module:** According to [4], is a link-up between FPA and COCOMO. BYL works by calculating the net function points and having them translated into source codes, with the language in consideration. By focusing on the 15 COCOMO cost drivers cost and time scale is estimated.

## 5. Conclusion

Project management methodology, scheduling, and cost estimation are Some important influence on software project management. Having a better understanding of these topics will help us to know what to do to improve software project management. The growth and demand for software are at an exponential rate, having a sophisticated software project management while help us meet up with the worlds demand for technology.

## References

- [1] A. U. Rehman and R. Hussain, “Software Project Management Methodologies/Frameworks Dynamics ‘A Comparative Approach,’” 2007 International Conference on Information and Emerging Technologies, 2007.
- [2] Cervone, H. (2011). Understanding agile project management methods using Scrum. OCLC Systems & Services: International Digital Library Perspectives, 27(1), 18-22. doi: 10.1108/10650751111106528
- [3] Ghaeli, M., & Sadi-Nezhad, S. (2019). A scientometrics survey on project scheduling. Journal Of Project Management, 165-176. doi: 10.5267/j.jpm.2019.1.006
- [4] Heemstra, F. (1992). Software cost estimation. Information And Software Technology, 34(10), 627-639. doi: 10.1016/0950-5849(92)90068-z
- [5] C. J. West. A Comparison of Software Project Architectures: Agile, Waterfall, Spiral, and Set-Based. B.Sc. Computer Science Utah State University, 2007.
- [6] Why Software Fails. (2019). Retrieved from. Source: <https://spectrum.ieee.org/computing/software/why-software-fails> Access Date: March 25, 2019