What is Software Project Management (SPM)?

Will Lin[*]
Zhejiang Normal University
Software Project Management

Abstract

This report will explain the definitions of Project, Project Management and Software Project Management in detail. The relationship between these three conceptions will be clarified, the purposes why we need use these in our daily work will also be mentioned.

Keywords: software engineering, project management

1 Introduction

software project management first came out as a way to manage new development efforts of software production. Its methods focused on matching user requirements to delivered products, since some confusions often occurred in the gray zone between the user specifications and the delivered software. This method is known now as the waterfall model.

This article aims to explain some details about Software Project Management (SPM), including its definition, process and characteristics. But before that, the conceptions of Project and Project Management need to be mentioned, because they closely relate to SPM.

2 Preconditions

Project: We usually use ’project’ to describe a set of interrelated tasks need to be executed over a fixed period and within certain cost and other limitations. To be more cautious, project can be defined as a one-off, temporary activity with a clear start and a clear end; it has full or part-time resources clearly assigned to it; it has a temporary management hierarchy that takes precedence over the company hierarchy; and it sets out to deliver something unique.

Project Management: It is the discipline of planning, organizing, and managing resources on time and within budget to bring about the successful completion of specific project goals and objectives. It often requires several skills, such as leadership, communication, problem solving, negotiating, influencing the organization, mentoring, process and technical expertise.The relationship of SPM and Project Manager can be illustrated in a graph of concentric circles.(e.g., see Figure 1)

3 Overview

A software project is a complex undertaking by two or more persons within the boundaries of time, budget, and staff resources that produces new or enhanced computer code that adds significant business value to a new or existing business process. In Software Project Management, software projects need to be planned, monitored and controlled.

3.1 Project Planning, Monitoring and Control

Project Planning: Its purpose is to identify the scope of the project, estimate the work involved, and create a project schedule. Project planning begins with requirements that define the software to be developed. The project plan is then developed to describe the tasks that will lead to completion.

Project Monitoring and Control: The purposes of them are to keep the team and management up to date on the project’s progress. If the project deviates from the plan, then the project manager can take action to correct the problem. Project monitoring and control involves status meetings to gather status from the team. When changes need to be made, change control is used to keep the products up to date.

4 Types of SPM

Many of the techniques of general project management are applicable to software project management, but Fred Brooks pointed out that the products of software projects have certain Characteristics that make them different. One way of perceiving software project manager is as process of making visible that which is invisible.

Invisibility: When a physical artefact such as a bridge or road is being constructed the progress being made can actually be seen. With software, progress is not immediately visible.

Complexity: Per dollar, pound or euro spent, software products contain more complexity than other engineer artefacts.

Flexibility: The ease with which software can be changed is usually seen as one of its strengths. However, this means that where the software system, it is expected that, where necessary, the software will change to accommodate the other components rather than vice versa. This means the software system are likely to be subject to a high degree of change. [Hughes and Cotterell 1999]

4.1 Characteristics

A successful software project has many characteristics, these four of which are the most important.

Low Complexity The project really is simple, this will often be attributable to the fact that the software development project rings of familiarity. It might be a straightforward application of established business rules and therefore take advantage of existing designs and coding. To the developer it might look like a cut-and-paste exercise. In such cases integration and testing will be the most challenging phases of the development project.

Well-Understood Technology Infrastructure A well-understood technology infrastructure is one that is stable and has been the foundation for many software development projects in the past. That means that the accompanying skills and competencies to work with the technology infrastructure are well-grounded in the development teams.

[*] e-mail:1073318082@qq.com

The article is based on work to support Software Project Management.
Software Project Management (2016/29)
Author Name: Will Lin
University: Zhejiang Normal University
Title: What is Software Project Management (SPM)?
Supervisor: Dr. Kenwright

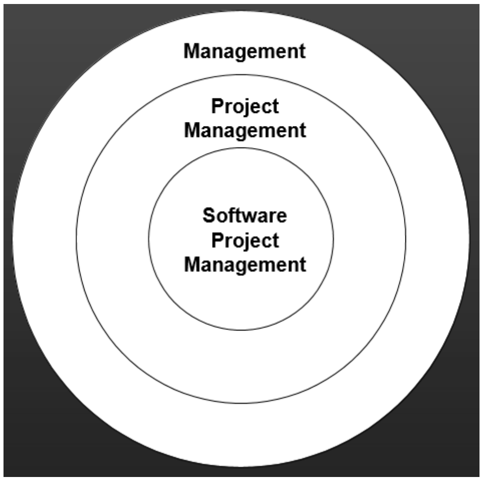

Figure 1: *Software Project Management is a sub-discipline of project management Project Management*

Low Risk The total environment for development projects in this quadrant is that it is known. All that could happen to put the project at risk has occurred in the past, and you have well-tested and well-used mitigation strategies in place. Experience has rooted out all of the mistakes that could be made. The customer is confident that it has done a great job identifying requirements, functions, and features, and they are not likely to change. Except for acts of nature and other unavoidable events, the project is protected from avoidable events.

Experienced and Skilled Developer Teams Past projects have been good training grounds for the teams. They have had opportunities to learn or to enhance their skills and competencies. [Wysocki 2006]

5 Conclusion

To make a great software project, well organized plan and efficient management are needed, and those who have the ability to make full use of time, money and other resources to manage software project will surely earn a success.

References

HUGHES, B., AND COTTERELL, M. 1999. *Software Project Management (Second Edition)*. McGraw-Hill Publishing Company. 1

WYSOCKI, R. K. 2006. *Effective Software Project Management*. Wiley Publishing, Inc. 2