Software Project Management

Author name Luo Wenjie
Zhejiang Normal University
Software Project Management

Abstract

The article gives a brief and concise explanation to the high valuable name of software project management (SPM). This article has been formatted, so whether or not you are currently a software project manager, actively working on a project team or completely dark in a mysterious field, this article has something good for you. If you are experienced, you'll find this article a light refresher to the subject.
Keywords: software engineering, project management, planning, risk management.

1. Instruction

Software Project management is the art and science of planning and leading software projects. [1] It is a sub-discipline of project management in which software projects are planned, monitored and controlled.

2. History of Software Project management

In the 1970s and 1980s, the software industry grew very quickly, as computer companies quickly recognized the relatively low cost of software production compared to hardware production and circuitry. To manage new development efforts, companies applied the established project management methods, but project schedules slipped during test runs, especially when confusion occurred in the gray zone between the user specifications and the delivered software. To be able to avoid these problems, software project management methods focused on matching user requirements to delivered products, in a method known now as the waterfall model. As the industry has matured, analysis of software project management failures has shown that the following are the most common causes:[2][3][4]

Insufficient end-user involvement
Poor communication among customers, developers, users and project managers
Unrealistic or unarticulated project goals
Inaccurate estimates of needed resources
Badly defined or incomplete system requirements and specifications
Poor reporting of the project's status
Poorly managed risks
Use of immature technology
Inability to handle the project's complexity
Sloppy development practices
Stakeholder politics (e.g. absence of executive support, or politics between the customer and end-users)
Commercial pressures

The first five items in the list above show the difficulties articulating the needs of the client in such a way that proper resources can deliver the proper project goals. Specific software project management toolsare useful and often necessary, but the true art in software project management is applying the correct method and then using tools to support the method. Without a method, tools are worthless.
Since the 1960s, several proprietary software project management methods have been developed by software manufacturers for their own use, while computer consulting firms have also developed similar methods for their clients. Today software project management methods are still evolving, but the current trend leads away from the waterfall model to a more cyclic project delivery model that imitates a software development process.

3. Overview

From the article below, we can know that software

project management is very important while creating a project. It can help us work more efficient.

4. Project planning, monitoring and control

Project planning

Its purpose is to identify the

1. Scope of the project,
2. Estimate the work involved,
3. 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 purpose

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.

Project Monitoring and Control

The purpose

1. To keep the team and management up to date on the project's progress.
2. 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.

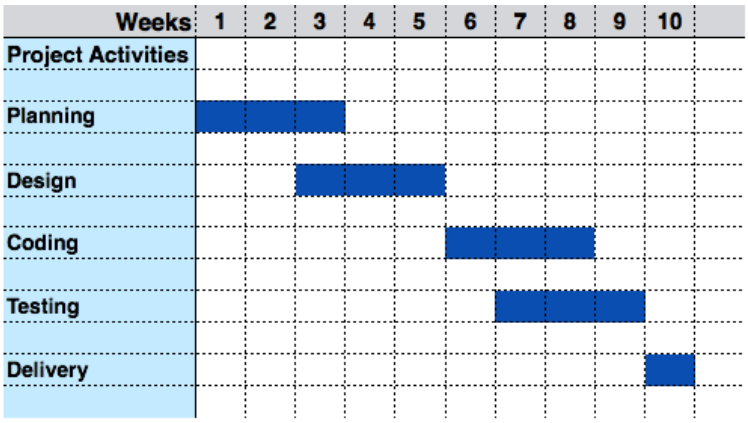
5. Project Management Tools

The risk and uncertainty rises multifold with respect to the size of the project, even when the project is developed according to set methodologies.

There are tools available, which aid for effective project management. A few are described -

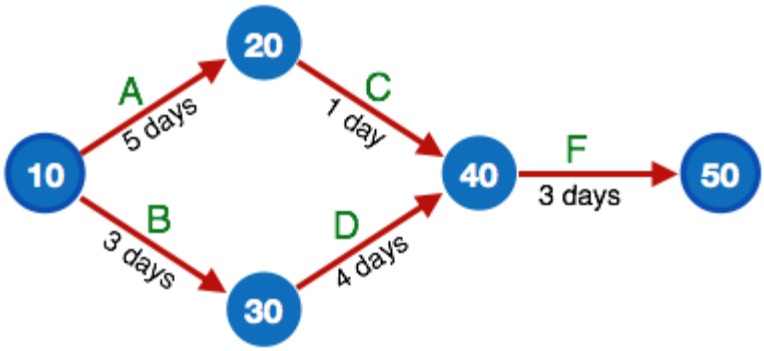
Gantt Chart

Gantt charts was devised by Henry Gantt (1917). It represents project schedule with respect to time periods. It is a horizontal bar chart with bars representing activities and time scheduled for the project activities.



PERT Chart

PERT (Program Evaluation & Review Technique) chart is a tool that depicts project as network diagram. It is capable of graphically representing main events of project in both parallel and consecutive way. Events, which occur one after another, show dependency of the later event over the previous one.

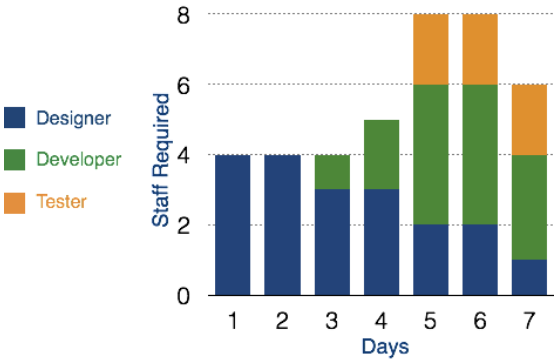


Events are shown as numbered nodes. They are connected by labeled arrows depicting sequence of tasks in the project.

Resource Histogram

This is a graphical tool that contains bar or chart representing number of resources (usually skilled staff) required over time for a project event (or phase). Resource Histogram is an effective tool for staff planning and coordination.

Staff	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
Designer	4	4	3	3	2	2	1
Developer	0	0	1	2	4	4	3
Tester	0	0	0	0	2	2	2
Total	4	4	4	5	8	8	6



6. Conclusion

Software project management is different from other project management, it has many particularities. The software is a special field, far from building engineering and other fields, so standardized software has great development space, experience plays a very important role in project management.

Acknowledgements

Thanks to my classmates and my teacher, I already made it.

References

1. *Stellman, Andrew; Greene, Jennifer (2005). Applied Software Project Management. O'Reilly Media. ISBN 978-0-596-00948-9.*
2. *"Why Software Fails", in IEEE Spectrum*
3. *^ Jump up to: a b Producing Open Source Software: How to Run a Successful Free Software Project (e-book, freely downloadable), by Karl Fogel*
4. *^ Jump up to: a b Robert Frese and Vicki Sauter, "Improving your odds for software project success," IEEE Engineering Management Review, Vol. 42, No. 4, Fourth Quarter, Dec 2014*