Problems with managing software development

Software Project Management is the process of planning, monitoring and controlling software projects. Managing the development of software is no easytask. It is a great skill which a project manager can only master after experience with several projects. Each software project is distinct from one another and needs to be handled specially. If the development activities are not managed properly, a number of things can go wrong. Following are some of the common problems faced by project managers while managing software development.

1. Poor requirements definition

The basis of successful development lies in correct and complete requirement analysis. If all the requirements have been gathered and have been put down in an appropriate manner, the designers and developers can better understand the software and hence produce a better and reliable product. On the other hand if the software has not been defined properly, it is likely that it will not be able to meet the customer’s requirements and will contain a lot of bugs. This is another problem the managers need to consider during development.

2. Overlooking hardware deficiencies

Technology plays an important role in the process of software development. The development work is highly related to the hardware equipment being used. If, during the development process there are hardware deficiencies, it can be a huge nightmare for the developers. It can affect the quality and reliability of the output. Moreover, malfunctioning of hardware can cause delays in development schedule.

3. Deadlines are unrealistic

It is a standard process that the project deadlines are determined considering the development time it should take. However, in order to win over a project, many estimators put up unrealistic deadlines. This becomes a problem for the development team and it becomes extremely difficult to manage development. What ends up happening is that either the team totally ignores the deadlines and carries development on their own routine, or in order to meet the deadlines, they create panic and end up making a poor quality, unreliable product.

4. Failure to establish a functional team of vendors, experts, and end users

Composition of a software team is very important and a crucial factor in the success of a project. The project team needs to have individuals who have the right skills required for the project. If a functional team of vendors, experts, and end users doesn't exist, the project cannot successfully complete. This is another critical problem faced during development when the team does not have the resources to complete the tasks.

5. Lack of user involvement

User is the key person in the software process. He is the ultimate person who will be using the software and hence need to be the center of all effort. In an ideal environment, user should be actively involved by giving his feedback and opinions. Often the user is not involved during development and the final product may fail to satisfy the user. This can result in a total failure of the project.

6. Software People don't understand the customer's needs

The software is designed to satisfy the customer's needs. This should always be the focus of the entire development process. Often what happens is that the team loses this focus in their mind and instead begins to concentrate on making reliable, efficient software. In many cases, the final product may be software of very quality but it may fail to meet the customer’s requirements. This happens when the development people have failed to realize what the customer had initially wanted. For instance, a customer may have asked for a simple program which should add two numbers. The software development team ended up making a complicated scientific calculator. This does not meet the customer’s needs.

Javelin is an anti-tank missile with automatic self-guidance and many other advanced capabilities. It was proposed to be acquired by the US Government but The General Accounting Office (GAO) published a report which questioned the adequacy of Javelin testing. The report, titled “ Army Acquisition – Javelin Is Not Ready for Multiyear Procurement”, opposed full-rate production of Javelin and emphasized the need for more operational tests due to the many redesigns undergone.

Development Test and Evaluation (DT&E) is usually conducted in large governmental projects to ascertain that the engineering design and development process is complete. It is used to qualify and validate the design, reduce risk, and ensure that the product is ready for government acceptance.

The DT&E results evaluate whether the design risks have been minimized and the system will be able to meet specifications. It performs a critical task in reducing the risks of development by testing high-risk features, components or subsystems. DT&E is the tool used to help the government by confirming that the system will perform as technically specified and that the system is capable of being used for field testing.

The above excerpt from a real-life project illustrates the problems with software development. If there are several revisions in the project, the end-product is not considered stable and has to be tested several times over to ensure performance.

References

Net Resources International. Javelin Anti-Armour Missile. 2009. http://www. army-technology. com/projects/javelin/ (accessed November 23, 2009).

Pressman, Roger S. Software Engineering - A Practitioners Approach. NY: McGraw-Hill, 2001.