

# CS 410/510 - Software Engineering

## Project Management

Reference: Sommerville, Software Engineering, 9 ed., Chapter 22

### The big picture

---

**Software project management** is concerned with activities involved in ensuring that **software is delivered on time and on schedule and in accordance with the requirements** of the organizations developing and procuring the software. Project management is needed because **software development** is always subject to **budget and schedule constraints** that are set by the organization developing the software.

Software project **success criteria**:

- Deliver the software to the customer at the agreed time.
- Keep overall costs within budget.
- Deliver software that meets the customer's expectations.
- Maintain a happy and well-functioning development team.

**Software engineering is different** from other types of engineering in a number of ways that make software management particularly challenging. Some of these differences are:

#### The product is intangible

Software cannot be seen or touched. Software project managers cannot see progress by simply looking at the artifact that is being constructed.

#### Many software projects are 'one-off' projects

Large software projects are usually different in some ways from previous projects. Even managers who have lots of previous experience may find it difficult to anticipate problems.

#### Software processes are variable and organization specific.

We still cannot reliably predict when a particular software process is likely to lead to development problems.

#### Software project management activities:

##### Project planning

Project managers are responsible for planning, estimating and scheduling project development and assigning people to tasks.

##### Reporting

Project managers are usually responsible for reporting on the progress of a project to customers and to the managers of the company developing the software.

##### Risk management

Project managers assess the risks that may affect a project, monitor these risks and take action when problems arise.

##### People management

Project managers have to choose people for their team and establish ways of working that leads to effective team performance.

##### Proposal writing

The first stage in a software project may involve writing a proposal to win a contract to carry out an item of work. The proposal describes the objectives of the project and how it will be carried out.

### Risk management

---

Risk management is concerned with **identifying risks** and drawing up **plans to minimize their effect** on a project. A risk is a **probability that some adverse circumstance will occur**:

- **Project risks** affect schedule or resources;
- **Product risks** affect the quality or performance of the software being developed;
- **Business risks** affect the organization developing or procuring the software.

### The risk management process:

#### Risk identification

Identify project, product and business risks. May be a team activity or based on the individual project manager's experience. A checklist of common risks may be used to identify risks in a project:

- Technology risks
- People risks
- Organizational risks
- Requirements risks
- Estimation risks

#### Risk analysis

Assess probability and seriousness of each risk. Probability may be very low, low, moderate, high or very high. Risk consequences might be catastrophic, serious, tolerable or insignificant.

#### Risk planning

Consider each risk and develop a strategy to manage that risk.

- **Avoidance** strategies: the probability that the risk will arise is reduced
- **Minimization** strategies: the impact of the risk on the project or product will be reduced
- **Contingency** plans: if the risk arises, contingency plans are plans to deal with that risk

#### Risk monitoring

Assess each identified risks regularly to decide whether or not it is becoming less or more probable. Also assess whether the effects of the risk have changed. Each key risk should be discussed at management progress meetings.

## Managing people

---

**People are an organization's most important assets.** The tasks of a manager are essentially people-oriented. Unless there is some understanding of people, management will be unsuccessful. Poor people management is an important contributor to project failure.

#### People management factors:

- **Consistency:** team members should all be treated in a comparable way without favorites or discrimination.
- **Respect:** different team members have different skills and these differences should be respected.
- **Inclusion:** involve all team members and make sure that people's views are considered.
- **Honesty:** you should always be honest about what is going well and what is going badly in a project.

An important role of a manager is to **motivate the people** working on a project. Motivation means organizing the work and the working environment to encourage people to work effectively. If people are not motivated, they will not be interested in the work they are doing. They will work slowly, be more likely to make mistakes and will not contribute to the broader goals of the team or the organization. **Motivation is a complex issue** but it appears that there are different types of motivation based on:

- Basic needs (e.g. food, sleep, etc.);
- Personal needs (e.g. respect, self-esteem);
- Social needs (e.g. to be accepted as part of a group).

In software development groups, basic physiological and safety needs are not an issue. Here's how to satisfy other types of needs:

- **Social:** provide communal facilities; allow informal communications e.g. via social networking
- **Esteem:** recognition of achievements; appropriate rewards
- **Self-realization:** training - people want to learn more; responsibility

Motivation should also take into account different **personality types**:

- **Task-oriented:** the motivation for doing the work is the work itself
- **Self-oriented:** the work is a means to an end which is the achievement of individual goals - e.g. to get rich, to play tennis, to travel etc.;

- **Interaction-oriented:** the principal motivation is the presence and actions of co-workers. People go to work because they like to go to work.

## Teamwork

---

Most **software engineering is a group activity**. The development schedule for most non-trivial software projects is such that they cannot be completed by one person working alone. A good group is cohesive and has a team spirit. The people involved are motivated by the success of the group as well as by their own personal goals. Group interaction is a key determinant of group performance. Flexibility in group composition is limited: managers must do the best they can with available people.

In a **cohesive group**, members consider the group to be more important than any individual in it. The **advantages** of a cohesive group are:

- Group **quality standards** can be developed by the group members.
- Team members **learn from each other** and get to know each other's work; inhibitions caused by ignorance are reduced.
- **Knowledge is shared**. Continuity can be maintained if a group member leaves.
- Refactoring and **continual improvement is encouraged**; group members work collectively to deliver high quality results and fix problems, irrespective of the individuals who originally created the design or program.

Three generic **factors that affect team effectiveness**:

### The people in the group

You need a mix of people in a project group as software development involves diverse activities such as negotiating with clients, programming, testing and documentation.

### The group organization

A group should be organized so that individuals can contribute to the best of their abilities and tasks can be completed as expected.

### Technical and managerial communications

Good communications between group members, and between the software engineering team and other project stakeholders, is essential.

A manager or team leader's job is to create a **cohesive group** and organize their group so that they can work together effectively. This involves creating a group with the **right balance of technical skills and personalities**, and organizing that group so that the members work together effectively. It may not be possible to appoint the ideal people to work on a project:

- Project budget may not allow for the use of highly-paid staff;
- Staff with the appropriate experience may not be available;
- An organization may wish to develop employee skills on a software project.

Managers have to work within these constraints especially when there are shortages of trained staff.

Group composed of members who share the same motivation can be problematic:

- **Task-oriented** everyone wants to do their own thing;
- **Self-oriented** everyone wants to be the boss;
- **Interaction-oriented** too much chatting, not enough work.

An effective group has a balance of all types. This can be difficult to achieve software engineers are often task-oriented. Interaction-oriented people are very important as they can detect and defuse tensions that arise.

The way how **group organization** affects the decisions that are made by that group, the ways that information is exchanged and the interactions between the development group and external project stakeholders. Key questions include:

- Should the project manager be the technical leader of the group?
- Who will be involved in making critical technical decisions, and how will these be made?
- How will interactions with external stakeholders and senior company management be handled?
- How can groups integrate people who are not co-located?
- How can knowledge be shared across the group?

**Small** software engineering groups are usually organized **informally** without a rigid structure. For **large** projects, there may be a **hierarchical structure** where different groups are responsible for different sub-projects. **Agile** development is always based around an **informal group** on the principle that formal structure inhibits information exchange.

An **informal group** acts as a whole and comes to a consensus on decisions affecting the system. The group leader serves as the external interface of the group but does not allocate specific work items. Rather, work is discussed by the group as a whole and tasks are allocated according to ability and experience. This approach is successful for groups where all members are experienced and competent.

Good **communications** are essential for effective group working. Information must be exchanged on the status of work, design decisions and changes to previous decisions. Good communications also strengthens group cohesion as it promotes understanding. The effectiveness and efficiency of communications is influenced by:

- **Group size:** the larger the group, the harder it is for people to communicate with other group members.
- **Group structure:** communication is better in informally structured groups than in hierarchically structured groups.
- **Group composition:** communication is better when there are different personality types in a group and when groups are mixed rather than single sex.
- **The physical work environment:** good workplace organization can help encourage communications.