Lesson-2: Managing Software Projects

CS 438
Ali Aburas PhD

Today’s Goals

- **Managing Software Projects**
  - ○ **What is it?**
  - ○ **Who does it?**
  - ○ **Why is it important?**
  - ○ **Project Management Activities**

HOW'S YOUR PROJECT GOING?
DO YOU MEAN THE ONE THAT HAS...
Dilbert.com DilbertCartoonist@gmail.com
...NO MANAGEMENT SUPPORT, AMBIGUOUS GOALS, NO BUDGET, AND AN ANGRY TEAM OF OVERWORKED PEOPLE WHO WANT IT TO DIE?
11-24-12 ©2012 Scott Adams, Inc./Dist. by Universal Uclick
NO, THE OTHER ONE.
SOMETIMES THERE ISN'T AN "OTHER ONE."

Project Management

- **Project management** involves the **planning, monitoring, and control** of the *people, process, and events* that occur as software evolves from a preliminary concept to an operational implementation.

- **Why is it important?**
  - Building computer software is a complex undertaking, particularly if it involves many people working over a relatively long time. That’s why software projects need to be managed.

- **The success criteria for project management**

  1. Deliver the software to the customer at the agreed time.
  2. Keep overall costs within budget.
  3. Deliver software that meets the customer’s expectations.
  4. Maintain a happy and well-functioning development team.
  5. Provide visibility about the progress of a project

Project Management Activities

- **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. It usually includes cost and schedule estimates and justifies why the project contract should be awarded to a particular organization or team.
- **Project planning.**
  - Determining a schedule, estimating the required resources such as people and funds, and assigning people to tasks.
- **Risk management**
  - Anticipation of possible risks that may affect a project, monitor these risks, and take action when problems arise (risk management).
- **Progress tracking**
  - Reconfirming people's expected performance, monitoring actions taken and results achieved, addressing problems encountered, and sharing information with people interested in the project.
- **Final analysis**
  - Analysis of project and extract lessons learned for a new project.

Project Planning

- **Project planning** involves breaking down the work into parts and assign these to project team members, anticipate problems that might arise and prepare tentative solutions to those problems.
- **The planning process**
  - **Assess** the constraints affecting the project.
    - These constraints are the required delivery date, staff available, overall budget, available tools, and so on
  - **Identify** the project milestones and deliverables.
    - **Milestones** are points in the schedule against which you can assess progress, for example, the handover of the system for testing.
    - **Deliverables** are work products that are delivered to the customer (e.g., a requirements document for the system).
  - **Decide** when you will do each thing (scheduling) and how long will it take? (estimation)
    - Draw up an estimated schedule for the project and the activities defined in the schedule are initiated.
  - **Review** progress and note discrepancies from the planned schedule.

Estimation

- Estimation involves working out how much effort/time is required to complete each activity and calculating the total cost of activities.

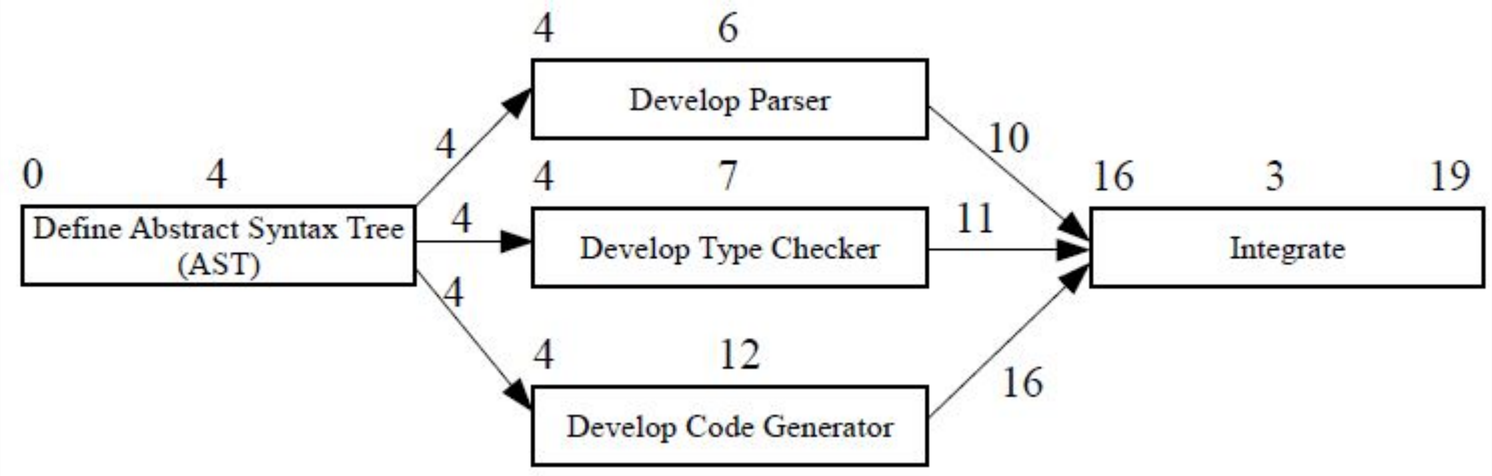
- Estimation techniques:
    - Compare with other tasks you have done
    - If too hard to estimate, break into smaller steps
    - Get several “expert” people to estimate

- Why is software project disaster so common?
    - Estimation techniques are poor and assume things will go well
    - Schedule progress is poorly monitored
    - When schedule slippage is recognized, the natural response is to add more developers
    - Inaccurate cost estimation (i.e., effort cost, hardware and software costs, travel and training cost)

Project scheduling

- **Project scheduling** is the process of deciding how the work in a project will be organized as separate tasks, and when and how these tasks will be executed.
- **Project scheduling activities**
  - Split project into tasks and estimate time and resources required to complete each task.
  - Organize tasks concurrently to make optimal use of workforce.
  - Minimize task dependencies to avoid delays caused by one task waiting for another to complete.

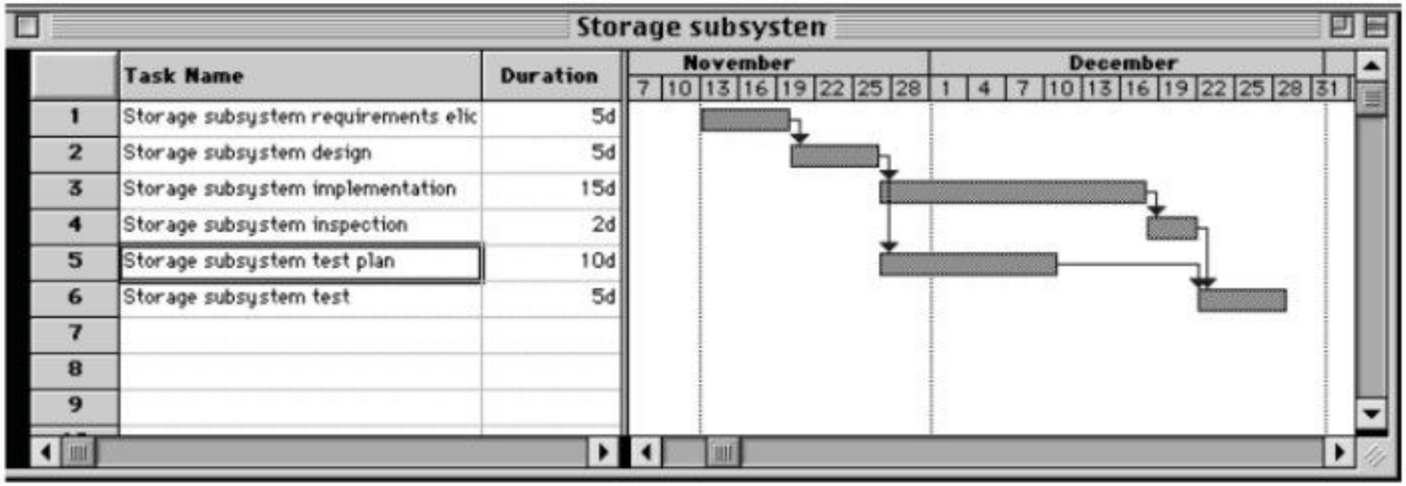
Scheduling Presentation

- Program Evaluation and Review Technique (PERT) chart represents a schedule as an acyclic graph of tasks.
  - The chart shows the tasks, the time expected to complete these tasks, and the task dependencies.
  - 

- The **‘critical path’** is the longest sequence of dependent tasks. This defines the project duration.

Scheduling Presentation

- **Bar charts**, which are calendar-based, show who is responsible for each activity, the expected elapsed time, and when the activity is scheduled to begin and end. Bar charts are sometimes called ‘Gantt charts’, after their inventor, Henry Gantt.
- **Gantt Charts** is a bar graph on which the horizontal axis represents time and the vertical axis lists the different tasks to be done.



Storage subsysten

	Task Name	Duration
1	Storage subsystem requirements elic	5d
2	Storage subsystem design	5d
3	Storage subsystem implementation	15d
4	Storage subsystem inspection	2d
5	Storage subsystem test plan	10d
6	Storage subsystem test	5d
7		
8		
9		

Project Tracking

- **Project tracking** is used to keep track of the progress of the project and compare actual and planned progress and costs.
  - checks that the development is on time and within budget.
- **Progress tracking**
  - Keep track of when each task is finished
  - Tasks are either finished or not
  - Tasks are either finished or not
  - Regular modification of the plan
  - Watch for “feature creep”

Feature Creep

- **Feature/Scope Creep** : Gradual accumulation of features over time.
- Feature/Scope Creep can happen for many reasons:
  - Users might suddenly understand the potential of the new system and realize new functionality that would be useful.
  - Developers might discover interesting capabilities to which they become very attached.

Risk Management

- **Risk management** is concerned with identifying risks and drawing up plans to minimise their effect on a project.
- **Types of Risks**
  1. ***Project risks*** affect the project schedule or resources.
     - E.g., experienced staff will leave the project before it is finished.
  2. ***Product risks*** affect the quality or performance of the software being developed.
     - E.g., software tools that support the project do not perform as anticipated.
  3. ***Business risks*** affect the organization developing the software.
     - E.g., a competitive product is marketed before the system is completed

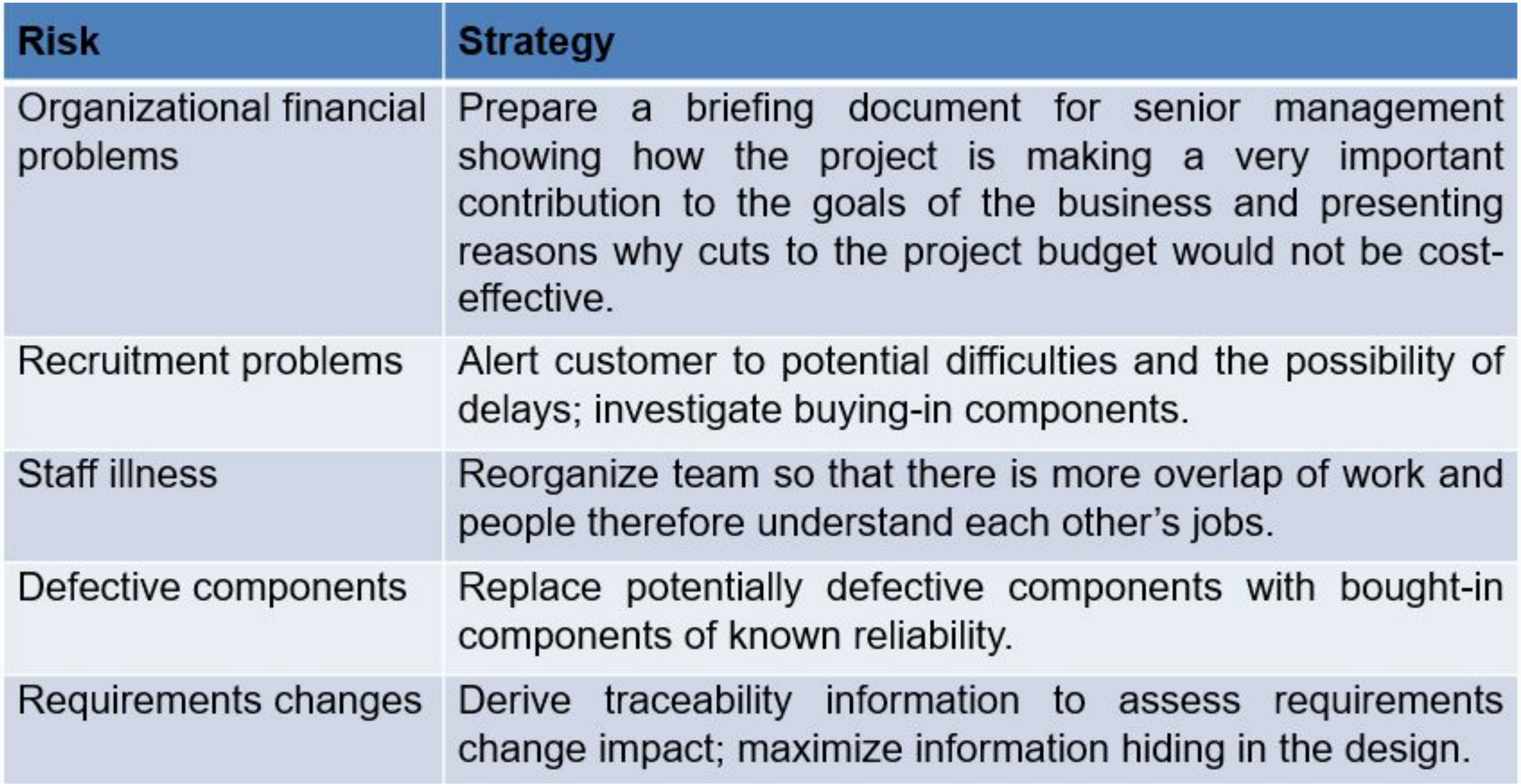
The risk management process

- **The risk management process**
  - **Risk identification**
    - Identify project, product and business risks;
  - **Risk analysis**
    - Assess the likelihood/probability and consequences of these risks;
      - Probability may be very low, low, moderate, high or very high.
      - Risk consequences might be catastrophic, serious, tolerable or insignificant.
  - **Risk planning**
    - Draw up plans to avoid or minimise the effects of the risk;
  - **Risk monitoring**
    - Monitor the risks throughout the project;

Risk types and examples

Risk	Probability	Effects
Organizational financial problems force reductions in the project budget.	Low	Catastrophic
It is impossible to recruit staff with the skills required for the project.	High	Catastrophic
Key staff are ill at critical times in the project.	Moderate	Serious
Faults in reusable software components have to be repaired before these components are reused.	Moderate	Serious
Changes to requirements that require major design rework are proposed.	Moderate	Serious
The organization is restructured so that different management are responsible for the project (6).	High	Serious
The time required to develop the software is underestimated	High	Serious

Strategies to help manage risk



Risk	Strategy
Organizational financial problems	Prepare a briefing document for senior management showing how the project is making a very important contribution to the goals of the business and presenting reasons why cuts to the project budget would not be cost-effective.
Recruitment problems	Alert customer to potential difficulties and the possibility of delays; investigate buying-in components.
Staff illness	Reorganize team so that there is more overlap of work and people therefore understand each other's jobs.
Defective components	Replace potentially defective components with bought-in components of known reliability.
Requirements changes	Derive traceability information to assess requirements change impact; maximize information hiding in the design.

Key points

- Good project management is essential if software engineering projects are to be developed on schedule and within budget.
- Project scheduling involves the creation of various graphical representations of part of the project plan. Bar charts, which show the activity duration and staffing timelines, are the most commonly used schedule representations.
- A project milestone is a predictable outcome of an activity or set of activities. At each milestone, a formal report of progress should be presented to management. A deliverable is a work product that is delivered to the project customer.
- Risk management involves identifying and assessing project risks to establish the probability that they will occur and the consequences for the project if that risk does arise. You should make plans to avoid, manage or deal with likely risks if or when they arise.