

Project Management

Lecture #16
Software Engineering and
Project Management

Instructed by Steven Choy on Feb 26, 2007



1

What we will discuss

- Software Project Management
 - Role of a Project Manager
 - Project Development Phrases
 - Project Development Activities
 - Project Organization
 - Risk Management



2

Role of Project Manager



3

Are you Ready for a PM Post?

- Project Management Skill
 - Communication
 - Leadership
 - Negotiation
 - Problem Solving
 - Multitasking
 - Influencing your team and organization
 - Mentoring/Coaching
 - Technical Expertise
 - Business/Domain Knowledge



4

Job Functions for a PM



- Define project scope
- Identify/Deal with stakeholder and client/decision maker
- Develop task list (Break down a big project into manageable task)
- Build schedule
- Build the team and choose appropriate team member
- Identify resources and decide budget
- Setup communication policy
- Evaluate/negotiate project requirements
- Identify and evaluate risk (Prepare for contingency)
- Track milestone
- Secure necessary resources
- Develop/Manage software development process
- Motivate team members and retain them
- Report project status to upper management
- Meeting, meeting and meeting

5

Software Project Management Overview



6

Project Management Overview



- Ensure the delivery of a **quality** system **on time** and **within budget**
- Motivate a group of developers/engineers to work efficiently & happily towards the same goal
- The success factors:
 - Quality
 - Money (i.e. Cost)
 - Time
- Quality, money and time → Trade Off

7

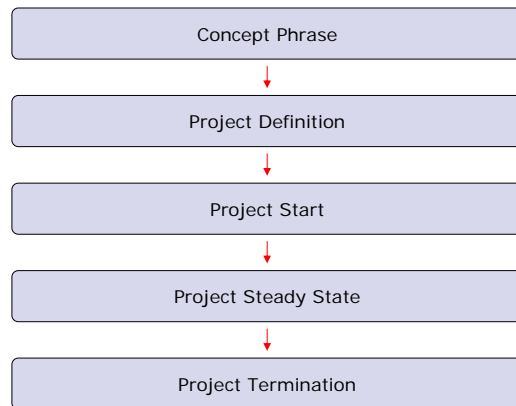
What makes a Successful Project?



- Upper management's Support
- Reliable estimate (both cost and time)
- Experienced Project Management
- User Involvement
- Clear Business Objectives/Requirements
- Well-chosen Development Methodology
- Well-chosen team structure/members
- Standard Software Architecture

8

Software Project Phrases



9

Concept Phrase



- The “WHY” Phrase
- The idea comes out! Collect them!
- Analyze and Justify the project
 - Cost-benefit Analysis
 - Return-On-Investment
 - Feasibility Study
- Initial project planning and analysis
- Identify project sponsor
 - The authority for approval and decision making

10

Project Definition Phrase



- The “WHAT” Phrase
- Define the scope of the system in terms of functionalities, constraints and deliverables
- Project manager, clients and software architect (or team leader) are involved
- Agree on acceptance criteria and delivery date
- Potential Deliverables:
 - Software project management plan
 - Requirement Specification Document
 - Requirement Approval & Sign-off

11

Project Start Phrase



- The “Kick-Off” Phrase
- Analysis and Design Phrase
- The project manager sets up the project infrastructure and the team
 - **Project infrastructure**: communication channel, meeting arrangement, coding standard, configuration management policy, development workflow, etc.
 - **Skill identification**: Identify the skills required
 - **Team assembly**: Assign participants to teams, select team leader, and assign roles & responsibilities
 - **Kick-off**

12

Project Start Phrase



- Potential Deliverables:
 - Functional Specification
 - Software Design Document
 - Detailed Software Design Document
 - User Interface Specification
 - Data Model/Design Document
 - Prototype
 - Updated Project Management Plan with latest estimates

13

“Project Steady State” Phrase



- The “Development” Phrase
- Team leaders take over some management functions and ensures the tasks assigned are on track
- Design/Coding/Unit Testing/Integration Testing
- Project manager arranges regular status meeting
- Project manager is responsible to deal with client, renegotiating resources and deadlines

14

Project Termination Phrase



- The deliverables are delivered to customer
- Delivery: Acceptance test & Installation
- Postmortem: Collect the history of the project and learn from it. Avoid to repeat the same mistakes and better the project development process

15

Software Project Activities



16

Project Management Activities



- Planning
 - Specify what to be achieved and work out a plan
 - Plan for schedule, resources including manpower and budget
- Organizing
 - Organize your project team in terms of role and responsibilities
- Controlling
 - Monitor the progress and ensure everything is on track
- Terminating
 - The project comes to an end, that the software system is accepted by the client and installed
 - Review and learn from the project

17

Project Planning Activities



- Clearly state the system to be achieved
 - Identify the scope and objectives
- Estimate cost, resource, effort, staffs, etc.
- Project scheduling
- Identify resources
 - How many team members are required?
 - What types of skill sets you need?
 - Any training plan?

18

Planning Documentation



- Software development plan
- Software configuration management plan
- Software quality assurance plan
- Risk management plan/Contingency plan
- Software process improvement plan
- Operations plan
- Etc...

19

Software Development Plan



- Some consider it as "The Master Document"
- Fundamental Sections
 - Project overview
 - Deliverables
 - Project organization
 - Managerial processes
 - Technical processes
 - Budget
 - Schedule

20

Project Scheduling



- How long will you take to deliver the system?
- How many man-days will it involve?
- Prediction is hard, especially for the future and something unknown

21

Project Scheduling - Estimate



- **How long will it take to finish this project?**
→ Pretty hard question!
- Why it's so hard to give an exact answer?
- How do you make the prediction? On what basis?
- Estimation is hard, especially for something you do not know about and familiar with!
- Most software estimations are off by 25-100%

22

Estimate Your Schedule



“The single most important task of a project: setting realistic expectations. Unrealistic expectations based on inaccurate estimates are the single largest cause of software failure.”

Futrell, Shafer, Shafer, “Quality Software Project Management”

23

How to give a Better Estimate #1



- Divide & Conquer (Break into smaller pieces)
 - How long will it take for your project? → Very Difficult
 - How long will it take to deliver a subsystem of the project? → Still difficult but I can give a better estimate
 - Okay, how long will take to deliver a module of a subsystem? → More obvious and easy answer, though it's still a rough estimate
- Ask the Expert
 - Based on expert's past experience
 - Based on past projects with similar scope

24



How to give a Better Estimate #2

- Estimate Iteratively!
- Give your best estimates in each phrase and refine them gradually
- Take resource experience or skill set into account
 - Often needed more time for staffs with less experience
 - Unfamiliar with a new technology or tool
- Don't forget those "non-project" time & common tasks
 - Meetings, phone calls, sick days, urgent leave, ad-hoc tasks
- How you present an estimate makes huge impact
 - Range – 8 to 9 months
 - Coarse Date – 2006 Q4

25



Divide & Conquer by WBS

- Break the whole project into small manageable units and predict the time required
- Mark the end of a group of activities with a milestone: a significant point of software development project
- Minimize task dependency, so that tasks can be done concurrently
- Make a good and accurate estimates is not an easy task. It depends on the project manager's experience

26



Project Scheduling Tool: Work Breakdown Structure

- A hierarchic decomposition or breakdown of a project into successive levels

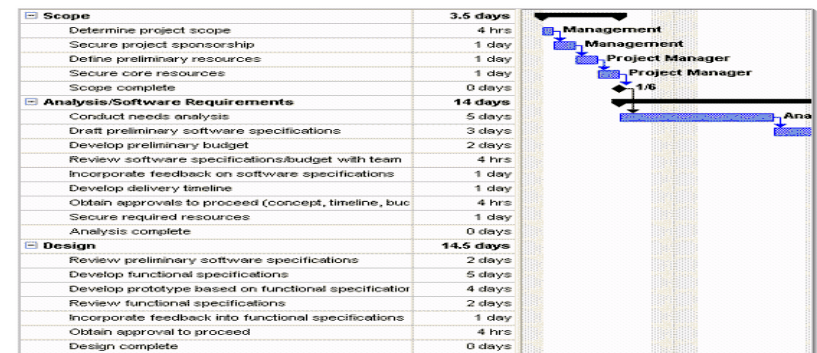
WBS number	Task Description
1.0	Project initiation
1.1	Draft project plan
2.0	Analysis phase
2.1	Plan user interviews
2.2	Schedule users interviews
3.0	Examination and test
4.0	Design
5.0	Test
6.0	Implementation
7.0	Post implementation review

27



WBS & Gantt Chart

- Show high-level view of project activities over time
- Notation:
 - Bar – show task duration
 - Line with arrow – show dependency



28

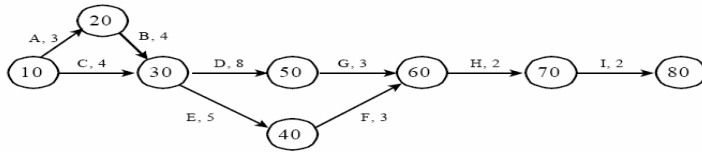
Source: Gantt chart from Microsoft Office Website

Project Scheduling Tool: PERT Chart



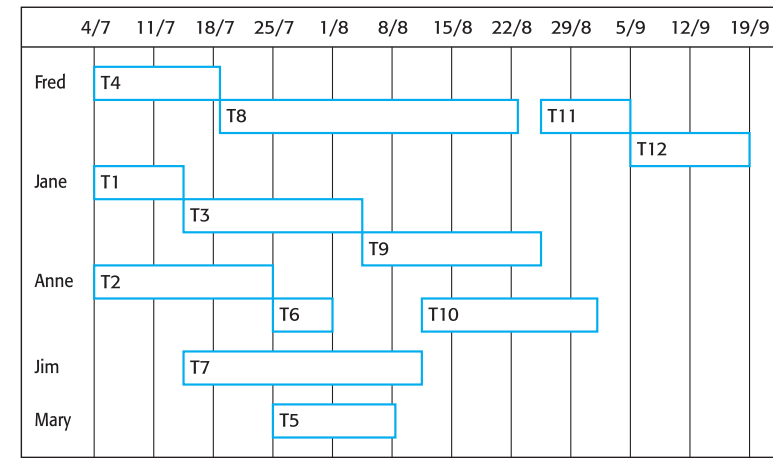
- PERT chart shows precedence activities that must be completed before the next activities may be started, i.e. task dependency

A	Conduct Interviews	None	3
B	Questionnaires	A	4
C	Read Reports	None	4
D	Analyze Data Flows	B, C	8
E	Introduce Prototypes	B, C	5
F	Observe Reactions	E	3
G	Perform Cost/Benefit	D	3
H	Prepare Proposal	G	2
I	Present Proposal	H	2



29

Team Member Allocation



30

Assembling your Team



31

Project Organization



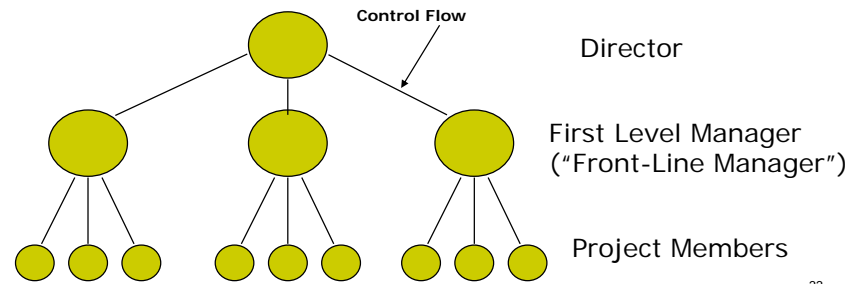
- Functional
 - Hierarchical form
 - Grouped by disciplines and specialties
 - Organizational units are commonly known as departments
- Project-based
 - Dynamic in nature and created on request
- Matrix
 - A mix of functional and project-based organization

32

Functional Organization

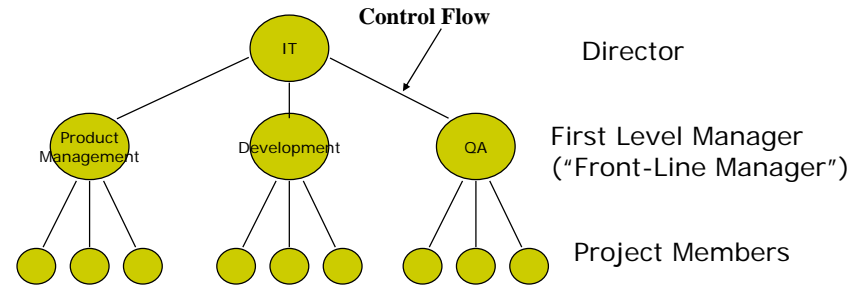


- Characteristics:
 - Technical expertise deeper
 - People within the same department get to know each other well
 - Service-oriented



33

Sample Functional Organization

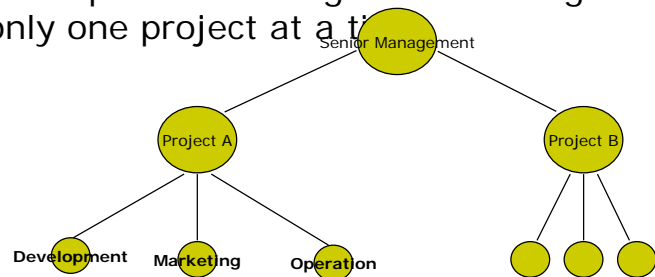


34

Project-based Organization



- Assembled dynamically, created on request and terminated when the project is accomplished
- Participants are assigned to the organization only one project at a time



35

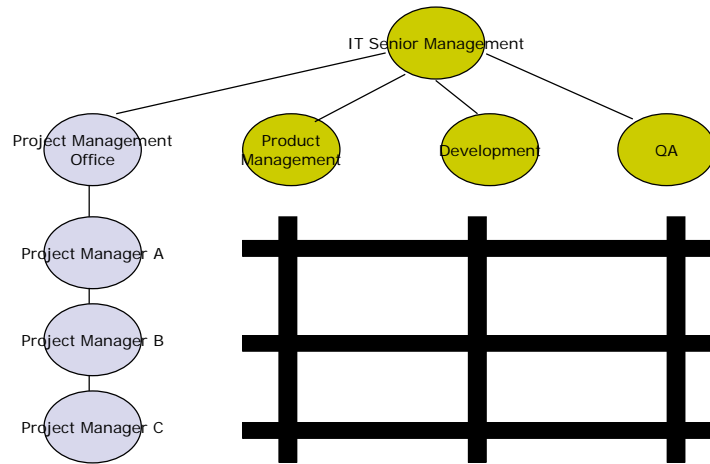
Matrix Organization



- Combines the characteristics of functional and project-based organization
- People from different departments are temporarily "borrowed" to work on a project
- When the project is finished, they are returned to their own department
- Project members that are assigned may work on part-time basis for the project
- Team leader is the principal authority for the project

36

Matrix Organization



37

Risk Management



38

Risk Management



- A risk is a potential problem that may lead to project failure
- Risk Management
 - Identify risk
 - "Attack" the risk

39

Common Software Project Risks



- Staff turnover
- Unrealistic schedule and budget
- Everlasting requirement changes
- Management/Corporate direction Change
- Corporate technology policy change
- Size of the system underestimate
- External vendor delays

40

Risk Types



- Technology Risks
 - E.g. The web server used in the system cannot handle the expected number of request
- People Risks
 - Staff turnover
 - Recruitment problem
- Organizational Risks
 - Reorganization of departments
 - Corporate direction changes, budget reduction
- Requirement Risks
 - Changes of requirement in major feature
 - Everlasting change
- Estimation Risks
 - The schedule is underestimated
 - Budget/Manpower is underestimated
 - The size of system is underestimated

41

Risk Analysis



- Assess probability and seriousness of each risk
- Probability: very low, low, moderate, high, very high
- Seriousness: critical, significant, tolerable or insignificant

42

Risk Planning



- Develop strategy to manage the risks
- Avoidance Strategies
 - Avoid the risk
- Minimization Strategies
 - Minimize the probability of the risk
- Potential Deliverable
 - Contingency Plan
 - If the risk arises, how do we handle it?

43

Project Documentation



Software Project Management Plan (SPMP)

1. Introduction
 - 1.1 Project Overview
 - 1.2 Project Deliverables
 - 1.3 References
 - 1.4 Definition and Acronyms
2. Project Organization
 - 2.1 Process Model
 - 2.2 Organizational Structure
 - 2.3 Project Responsibilities
3. Managerial Process
 - 3.1 Management objectives and priorities
 - 3.2 Assumptions, dependencies and constraints
 - 3.3 Risk management
 - 3.4 Monitoring and controlling mechanism
4. Technical Process
 - 4.1 Methods, tools and techniques
 - 4.2 Software Documentation
 - 4.3 Project Support Functions
5. Work elements, schedule and budget

44

Project Controlling



45

Project Control Tools



- Regular status meeting with internal team member to update project status
- Meeting with client to review the system
- Software configuration management system
- Issue Tracking System
- Regular Build & Review with client/user

46

A few words about meeting



- Meeting is expensive!
- Meeting is a waste of time?
 - No agenda
 - No preparation
 - No conclusion come up
- Advices for meeting:
 - Announce the agenda in advance stating clearly
 - Purpose of meeting, people involved, date/time, things to discuss
 - Write minutes
 - Keep track with the action items

47

Measuring Progress



- We need metrics to determine our project progress
- Example of metrics:
 - Number of defects in the system
 - Number of line of code developed
 - Number of use case accomplished
 - Number of change request
- Analyze the metric with respect to time
 - Example:
 - Number of change request during Alpha period
 - Number of defects over time

48

Project Termination



49

Terminating the Project



- Client Acceptance Test
- System Deployment/Installation
- Postmortem Analysis

50

Acknowledgement

The slides were originally authored by **Simon Ng**.



51