Software Project Management
Objectives

- Introduce students to a variety of approaches and techniques in SPM
- Use current SPM tools
- Develop new SPM ideas
- Document reading, experiences and ideas
- Improve writing and presentation skills
Today

- Introduction to SE 362
- Fundamentals of Project Management
- Introduction to Software Project Management
Course Outline

• Introduction to PM and SPM
• Software models and process improvement
• Project planning, scheduling, estimation
• Personnel and project organization
• Change management, monitoring & control
• Protocols and standards
• Software QA and risk management
Workload and Evaluation

- Six assignments (30%)
- Midterm (20%)
- Final Exam (50%)
- In-class presentations, critiques, discussion
- Groupwork for three of the assignments
Textbooks

- Recommended Text (Free, in the library!)
  Quality Software Project Management, Futrell & Shafer
- Lots of Other Texts (Widely varying content)
  Rapid Development, Steve McConnell
  Information Technology Project Management, Kathy Schwalbe
  IT Project Management, Jack Marchewka
  Software Engineering Project Management, edited by Richard H. Thayer
  Software Project Survival Guide, Steve McConnell
Format

- Essentials of software project management
- Research and textbook readings and writing
- Real-world case studies
- Industry visitors
- Highly interactive
My Background

- Civil Engineering
- Information Systems Management
- 15 years each in industry and academia
- 1 year with KPMG, IT Consulting
- Projects of all shapes and sizes – health, construction, government
PM Profession

- Professional Organizations
  - Project Management Institute (PMI) (pmi.org)
  - Software Engineering Institute (SEI)
  - IEEE Software Engineering Group

- Certification
  - PMI PMP

- The “PMBOK” – PM Body of Knowledge
Project Management Skills

- Leadership
- Communications
- Problem Solving
- Negotiating
- Influencing the Organization
- Mentoring
- Process and technical expertise
Software Project Management
Some PM History

- 1970’s: Early PM ideas and software
  - military, defense, construction industry
  - Fred Brooks, *The Mythical Man-Month*
- 1980’s: SPM, TQM
- 1990’s: Large shift to PM-based models
  - 1990-93: Re-engineering, self-directed teams
  - 1996-99: Risk mgmt, project offices
- 2000’s: Global projects
Project Management

- What’s a project?
- PMI definition
  - *A project is a temporary endeavor undertaken to create a unique product or service*
- Progressively elaborated
  - With repetitive elements
- A project manager
  - Analogy: conductor, coach, captain
Project vs. Program Management

- What’s a ‘program’?
- Mostly differences of scale
- Often a number of related projects
- Longer than projects
- Definitions vary
As a PM, who do you interact with?

Project Stakeholders
- Project sponsor
- Executives
- Team (developers and maybe others)
- Customers
- Contractors (as needed)
- Functional managers
PM Tools: Software

- **Low-end**
  - Basic features, tasks management, charting
  - MS Excel, Milestones Simplicity

- **Mid-market**
  - Handle larger projects, multiple projects, analysis tools
  - MS Project (approx. 50% of market)

- **High-end**
  - Very large projects, specialized needs, enterprise
  - AMS Realtime
  - Primavera Project Manager
Tools: Gantt Chart

<table>
<thead>
<tr>
<th>Task Name</th>
<th>Duration</th>
<th>Start</th>
<th>Jan 2002</th>
<th>July 2002</th>
<th>August 2002</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Beginning of Project</td>
<td>0 days</td>
<td>Sat 6/15/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Project launch/feasibility study</td>
<td>26 days</td>
<td>Mon 6/17/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Key project decision-maker identified</td>
<td>5 days</td>
<td>Mon 6/17/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Vision statement created, reviewed, and baselined</td>
<td>2 days</td>
<td>Mon 6/24/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Business case for the software established</td>
<td>4 days</td>
<td>Wed 6/26/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. Preliminary effort and schedule targets created, reviewed, and baselined</td>
<td>5 days</td>
<td>Tue 7/2/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Technical approach developed</td>
<td>2 days</td>
<td>Tue 7/8/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8. Change Control Plan created, reviewed, and baselined</td>
<td>4 days</td>
<td>Thu 7/11/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9. Intel Top 10 Risks List created, reviewed, and baselined</td>
<td>3 days</td>
<td>Wed 7/21/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10. Software Project Log started</td>
<td>1 day</td>
<td>Mon 7/22/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11. Project launch/feasibility study complete</td>
<td>0 days</td>
<td>Mon 7/22/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12. Preliminary requirements development</td>
<td>24 days</td>
<td>Wed 7/29/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13. GA lead on board</td>
<td>2 days</td>
<td>Thu 7/2/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14. Documentation lead on board</td>
<td>2 days</td>
<td>Thu 7/2/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15. Key users identified and interviewed</td>
<td>3 days</td>
<td>Thu 7/29/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16. Simple user-interface prototype created, reviewed by users</td>
<td>2 days</td>
<td>Thu 8/1/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17. User Interface Style Guide created, reviewed, and baselined</td>
<td>5 days</td>
<td>Mon 8/9/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18. First project estimate (accurate to ±100%, ±5%) created, reviewed, and baselined</td>
<td>2 days</td>
<td>Tue 8/7/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19. Preliminary Software Development Plan created, reviewed, and baselined</td>
<td>3 days</td>
<td>Thu 8/14/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20. Top 10 Risks List updated</td>
<td>2 days</td>
<td>Thu 8/20/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21. Software Project Log updated</td>
<td>2 days</td>
<td>Thu 8/20/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22. Preliminary requirements development complete</td>
<td>0 days</td>
<td>Fri 8/23/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23. Detailed requirements development</td>
<td>18 days</td>
<td>Mon 9/2/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24. Detailed user interface prototype created, reviewed, and baselined</td>
<td>5 days</td>
<td>Mon 8/18/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25. User Manual/Specification created, reviewed, and baselined</td>
<td>4 days</td>
<td>Thu 9/5/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26. Software Quality Assurance Plan created, reviewed, and baselined</td>
<td>3 days</td>
<td>Wed 9/14/02</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27. Detailed Software Development Plan created, reviewed, and baselined</td>
<td>2 days</td>
<td>Wed 9/20/02</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Tools: Network Diagram
PMI’s 9 Knowledge Areas

1. Project *integration* management
2. Scope
3. Time
4. Cost
5. Quality
6. Human resource
7. Communications
8. Risk
9. Procurement
Strategy

- Classic Mistake Avoidance
- Development Fundamentals
- Risk Management
- Schedule-Oriented Practices
Four Project Dimensions

- People
- Process
- Product
- Technology
Trade-off Triangle

- Fast, cheap, good. Choose two.
Trade-off Triangle

- Know which of these are fixed & variable for every project

- Scope
- Performance

Product

Time

Schedule

Resources

Cost
People

“It’s always a people problem.”
Gerald Weinberg, The Secrets of Consulting

Developer productivity: 10-to-1 range

- Improvements:
  - Team selection
  - Team organization
  - Motivation
Other success factors

- Matching people to tasks
- Career development
- Balance: individual and team
- Clear communication
Process

- Is process stifling?
- Two Types: Management & Technical
- Development fundamentals
- Quality assurance
- Risk management
- Lifecycle planning
Process ...2

- Customer orientation
- Process maturity improvement
- Rework avoidance
- Avoid abuse by neglect
Product

- The “tangible” dimension
- Product size management
- Product characteristics and requirements
- Feature creep management
Technology

- Often the least important dimension
- Language and tool selection
- Value and cost of reuse
Planning

- Determine requirements
- Determine resources
- Select lifecycle model
- Determine product features strategy
Tracking

- Cost, effort, schedule
- Planned vs. Actual
- How to handle when things go off plan?
Measurements

- Actual (To date) and Projected
  - Cost
  - Schedule
  - Effort
  - Product features
Measurements ...2

- Alternatives
  - Earned value analysis
  - Defect rates
  - Productivity (ex: SLOC)
  - Complexity (ex: function points)
Technical Fundamentals

- Requirements
- Analysis
- Design
- Construction
- Quality Assurance
- Deployment
Project Phases

- All projects can be divided into phases
- Project Life Cycle is all phases together
- Each phase marked by Deliverables
- Same for software project phases
Lifecycle Relationships

Business Life Cycle

Policy Planning | Needs Identification | Project Conception | Realization | Product in Service | Disposal

Product Life Cycle

Feasibility | Acquisition | Operations | Disposal

Project Life Cycle

Concept | Development | Implementation | Termination
Software Project Management

- Key Issues: Cost, Quality, Time
- Software Project Planning
- Software Project Control
Software Project Planning

Software Requirements Engineering

1. Users’ needs, goals, expectations
2. Prototyping strategy
3. Software requirements specifications
4. Requirements traceability method
Software Project Planning

Project Plan Development

1. Activities, tasks
2. Cost estimation
3. Schedule, finish date
Software Project Planning - 3

Risk Management

1. Anticipate potential problems
2. Mitigate or avoid the problems
3. Track existing and potential problems
Incremental Release Process Model

1. Provide periodic demonstrations
2. Reach short-term goals
3. Check progress towards long-term goals
Software Project Control

Work Package Development

1. Work Breakdown Structures (WBS)
2. Work Package Specifications
3. Binary Tracking Technique
4. Earned Value Tracking Technique, comparing to planned values
Software Metrics

1. Product metrics
2. Process metrics

Necessary for planning and control
Software Project Control - 3

Software Configuration Management

1. Identify software components
2. Track component changes
3. Track documentation
   - Control software evolution
Software Project Control - 4

Software Quality Assurance

1. Use a SQA process
2. Use engineering standards

Peer Reviews

1. Walkthroughs
2. Inspections
Software Project Control - 5

Verification and Validation

1. Quality
2. Quantity

Software Engineering Assessment

1. Validate organizational capability
2. Identify software process improvements
Software Project Phases

- Software Concept
- Requirements Analysis
- Architectural Design
- Detailed Design
- Coding and Debugging
- Systems Testing
- Deployment & Maintenance
Classic Mistakes

- Seductive Appeal
- Types
  - People-Related
  - Process-Related
  - Product-Related
  - Technology-Related
People-Related Mistakes

- Undermined motivation
- Weak personnel
  - Weak vs. Junior
- Uncontrolled problem employees
- Heroics
- Adding people to a late project
People-Related Mistakes ...2

- Noisy, crowded offices
- Customer-Developer friction
- Unrealistic expectations
- Politics over substance
- Wishful thinking
People-Related Mistakes …3

- Lack of effective project sponsorship
- Lack of stakeholder buy-in
- Lack of user input
Process-Related Mistakes

- Optimistic schedules
- Insufficient risk management
- Contractor failure
- Insufficient planning
- Abandonment of plan under pressure
Process-Related Mistakes …2

- Wasted time during fuzzy front end
- Shortchanged upstream activities
- Inadequate design
- Shortchanged quality assurance
Process-Related Mistakes …3

- Insufficient management control
- Frequent convergence
- Omitting necessary tasks from estimates
- Planning to catch-up later
- Code-like-hell programming
Product-Related Mistakes

- Requirements gold-plating
  - Gilding the lily
- Feature creep
- Developer gold-plating
  - Beware the pet project
- Push-me, pull-me negotiation
- Research-oriented development
Technology-Related Mistakes

- Silver-bullet syndrome
- Overestimated savings from new tools and methods
  - Fad warning
- Switching tools in mid-project
- Lack of automated source-code control
References

- ACM Conferences
- ICSE
- SEI
- IEEE Computer Society Press
- IEEE Computer
- Journal of Systems and Software Sciences
Standards

- IEEE 1058, Standards for SPM Planning
- ACM
- CSA
- ISO
- SEI’s CMM
- National Bureau of Standards
Software

- PERT/Gantt/CPM Scheduling
- Work Breakdown Structure (WBS)
- COCOMO II Model Definition Manual
- Sim SE
- Problems and Programmers
Homework

- Assignment 1
  - Reading 56 pages
  - Thinking
  - Writing one page
- Choose group partners
- Start thinking about Assignment 2