Subject Objective

* This subject introduces best practice in project management in the context of software development.
* Emphasis is placed on:
  – Appreciating the difficulty of managing projects that develop systems based on computer software
  – The process of project management
  – The art of balancing project resources against product quality

Role of Managers

* “To get work done through other people”
* Typical activities:
  – Planning
  – Organising
  – Communicating
  – Monitoring
  – Controlling
  – Motivating (or avoiding De-motivating)

Role of “Process”

* Process re-use and process improvement rather than process invention.
* We need to use:
  – Experience
  – Reference models
  – Standards
* Good managers:
  – Put people first, but must
  – Have effective and efficient processes that give them the time to “put people first”

Costs of Poor Project Management

* Client dissatisfaction
  – Late delivery
  – Poor quality
  – Inability to plan
* Staff dissatisfaction
  – Frustration
  – Impact on family and personal life
* Waste of precious resources
Why is it all so hard?

- Immature and volatile technology
- Intangible products
- Ill-defined processes
- Hero complexes
- Client expectations
- Ill-defined and volatile requirements
- Inexperienced and volatile project teams
- Psychology of software developers

What can we do about it?

- Accept that it’s hard
- Expect the unexpected
- Identify risks and manage them
- Stabilise what can be stabilised
  - Choose mature technology
  - Define processes
  - Use standards
- Make work products tangible and measurable
- Collect and use experiences

About CS3710

- Some Theory and much Practice
- It won’t make you a Project Manager
- It will help you to participate in projects

CS3710 in 2003: A major change

- Team took over this subject two years ago
- Responding to student criticisms
- Realigning the subject with its goals
- Increasing the practical elements
- Decreasing the theoretical elements
- Exposing students to industrial size problems
- Programming exercises replaced by design exercises

Lectures and Seminars by Week

1: Subject Outline +
   Processes for Project Management – Planning
2: Project Management Tool
   Personal Software Process
3: Project Scheduling + quiz
4: Processes for Project Management – Monitoring
5: Integrated and Collaborative projects + quiz
6: No lecture and no formal tutorials
7: Seminar – An invited speaker from industry
8: Subject Review
9: Exam

Tutorial Exercise: Manage a mini-project

- Begins in Week 2, finishes in Week 8
- The project will be to design a web-based project management tool based on a given set of requirements
- You will manage that mini-project
  - Identify the design activities
  - Identify project risks
  - Estimate the design activities
  - Plan the design activities
  - Monitor and adjust the plan when necessary
- You will use a project management tool to help you to plan and control your mini-project
Tutorial Exercise: Desired Outcomes

After participating in the Tutorial Exercise, students will:
1. Have planned, monitored and revised a mini-project.
2. Have attempted to apply best practice in Project Management and Monitoring.
3. Have understood some of the requirements for a project management tool to support the Project Management and Monitoring processes.
4. Have been confronted by the particular problems presented by multi-organisational, collaborative system development.

Tutorial Exercise: You will be both a manager and a worker

- Why? Because you need to experience a project from both perspectives
- You need to appreciate the joint responsibilities for the success of the project
  - Responsibility of a worker: to provide data on progress on work products so that the project can be controlled
  - Responsibility of a manager: to provide the worker with adequate time and resources to do their job

Tutorial Exercise: Management deliverables

- Project Plan (week 2)
- Risk analysis (week 2)
- Project Status Report (each week)
- Revised Project Plan (when necessary)
- Project Review Report (at the end of the project)

Tutorial Exercise: Worker deliverables

- Project measures – deliver to the project manager
- Design of modules for Planning and Controlling
- Process Design
  - DFDs, Structured English descriptions and data dictionary for data flows
- Input/Output Design
  - Graphics that illustrate how the human-computer interface works
- Data Storage Design
  - Definition of a relational database that can store the data for planning and tracking

Tutorial Exercise: The Project Context

- Your organisation has been commissioned by a client, System Integrators Pty. (SIP), to design a web-based project management tool
- SIP manages the “Farm Cheese” project
- The “Farm Cheese” project involves 13 autonomous organisations that must somehow collaborate
- SIP wants a project management tool that is suitable for managing the “Farm Cheese” project
- If your organisation impresses SIP, you will get a contract to build the tool and make lots of money
Tutorial Exercise: What will impress SIP?

- SIP has had a lot of experience with managing multi-organisational projects
- SIP is convinced that reducing process variability and avoiding heroic efforts is the key to successful projects
  - They therefore seek suppliers who have demonstrated higher levels of capability maturity
  - They demand from their suppliers regular status reports that are based on objective evidence
  - They expect suppliers to conform to best practice as exemplified in ISO/IEC standards and the CMMI process models

Tutorial Exercise: The Farm Cheese Project

A pilot project to establish the viability of producing high-quality, high-value cheeses in remote dairies
1. Establish the dairies at five selected dairy farms
2. Establish internet connection at each of the farms
3. Install SCADA (supervisory control and data acquisition) RTUs (Remote Terminal Units) in each dairy
4. Establish the central control system
5. Establish subsidiary control systems at the premises of some key participants
6. Develop an expert system for managing the cheese-making process

Tutorial Exercise: What is SIP’s problem?

SIP must manage a complex project involving a large number of autonomous collaborating participants
- SIP has no powers of coercion
- Each participant has their own way of doing things
- SIP needs data from each participant to monitor the project
- Participants need data from each other
- The “Farm Cheese” system being developed is highly novel
- Most participants have not worked together before
- Participants have different technology and skills
- Participants have different goals that may be in conflict
- Government involvement requires higher level of reporting

Tutorial Exercise: SIP Requirements for the Project Management Tool

- Support mutual knowledge amongst participants
- Support consultation between participants
- Fair distribution of workload and risks
- Actively work to:
  - Reduce project uncertainty
  - Identify and manage project risks
  - Increase mutual trust and commitment to the project
  - Minimise coupling between participants’ project activities
- Exploit the common factor of participants’ IT
- Capture experience and support learning

Tutorial Exercise: PM Tool Requirements

- Support traceability between project objectives and project activities
- Support maximum autonomy for participants in their assigned area of responsibility
- Automated alerts when status data are overdue and when scheduled events are missed
Tutorial Exercise: Schedule by week

1: No tutorials – look at documents at ~cs3710
2: Initial Planning of your mini-project
3: Work on design for Planning Module of the PM Tool
4: Deliver design for Planning Module of the PM Tool
5: Work on design for Monitoring Module of PM Tool
6: No formal tutorial – revise plan for your mini-project
7: Deliver design for Monitoring Module of PM Tool
8: Deliver Project Review, Design change exercise
9: No tutorials

CS3710: Subject Assessment

- Tutorial Exercise: project management deliverables and design deliverables 40%
- Two Quizzes (multiple choice) 10% each
- Final Exam (multiple choice) 40%

Processes for Project Management

CMMI: Capability Maturity Model
Integrated

What is CMMI?

- CMMI is a reference model for systems engineering based on best practice
  - 30+ organisations developed the model
  - 40+ organisations reviewed the model
- Identifies the necessary processes for effective and efficient systems engineering
- Includes the management and control processes known as project management

CMMI Project Participants

- Computer Sciences Corporation
- Defense Logistics Agency
- DERA Systems
- Internal Revenue Service
- National Reconnaissance Office
- Federal Aviation Administration
- NASA
- General Dynamics
- Lockheed Martin
- THALES
- Northrop-Grumman Corporation
- Harris Corporation
- Local-7 Packard/Company
- Pacific Bell
- Honeywell Corporation
- Q-Labs Inc.
- Raytheon
- Rockwell Collins
- Science Applications International Corporation
- Software Engineering Institute
- EnerQual, Inc.

CMMI Reviewers

- Computer Sciences Corporation
- Defense Logistics Agency
- DERA Systems
- Internal Revenue Service
- National Reconnaissance Office
- Federal Aviation Administration
- NASA
- General Dynamics
- Lockheed Martin
- THALES
- Northrop-Grumman Corporation
- Harris Corporation
- Local-7 Packard/Company
- Pacific Bell
- Honeywell Corporation
- Q-Labs Inc.
- Raytheon
- Rockwell Collins
- Science Applications International Corporation
- Software Engineering Institute
- EnerQual, Inc.
Why have a Process Definition?

- A Defined Process:
  - can be managed
  - can be evaluated
  - is repeatable
- Processes are defined in sufficient detail that all activities and tasks are known
- The degree to which people follow with the defined process can be monitored and deviations analysed
- The Organisation can become less dependent on an individual’s skills

A CMMI Process Area Definition

- Purpose
- Introductory Notes
- Related Process Areas
- Specific Goals
- Generic Goals – what you need to achieve to be assessed at a particular capability maturity level
- Practice-to-Goal Relationship Table
- Specific Practices by Goal
- Generic Practices by Goal – what you need to do for a particular capability maturity level

Specific goals – an example

SG 1 Establish Estimates
Estimates of project planning parameters are established and maintained.

SG 2 Develop a Project Plan
A project plan is established and maintained as the basis for managing the project.

SG 3 Obtain Commitment to the Plan
Commitments to the project plan are established and maintained.

Practice-to-Goal Relationship Table – an example

<table>
<thead>
<tr>
<th>SG 1 Establish Estimates</th>
<th>SP 1.1-1 Estimate the Scope of the Project</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SP 1.2-1 Establish Estimates of Work Product and Task Attributes</td>
</tr>
<tr>
<td></td>
<td>SP 1.3-1 Define Project Life Cycle</td>
</tr>
<tr>
<td></td>
<td>SP 1.4-1 Determine Estimates of Effort and Cost</td>
</tr>
</tbody>
</table>

Specific Practices – an example

SP 1.5-1 Define Project Life Cycle
Define the project life-cycle phases upon which to scope the planning effort.
The determination of a project’s life-cycle phases provides for planned periods of evaluation and decision making. These are normally defined to support logical decision points at which significant commitments are made concerning resources and technical approach. Such points provide planned events at which project course corrections and determinations of future scope and cost can be made.

For Software Engineering
The determination of project phases for software typically includes selection and refinement of a software development model to address interdependencies and appropriate sequencing of software project activities.

For Systems Engineering
Identify the major product phase (e.g., concept exploration, development, etc.) for the current state of the product, expected future phases, and the relationships and effects among phases. Adjust planning parameters to account for relationships and effects among phases.
**Project Planning**

Reference:
Capability Maturity Model® Integration (CMMI), Version 1.1, for Systems Engineering and Software Engineering (CMMI-SE/SW, V1.1) Continuous Representation. CMU/SEI-2002-TR-001, ESC-TR-2002-001

---

**SP 1.1.1 Estimate the Scope of the Project**

1. Establish a top-level work breakdown structure (WBS) based on the product architecture
2. Identify the work packages in sufficient detail to specify estimates of project tasks, responsibilities, and schedule
   - Work package = units of work that can be separately assigned, performed, and tracked
   - Outcome of a work package is one or more work products
3. Identify work products (or components of work products) that will be externally acquired
4. Identify work products that will be reused

---

**SP 1.2.1 Establish Estimates of Work Product and Task Attributes**

1. Determine the technical approach for the project
2. Select the attributes of the work products and tasks that will be used to estimate the resource requirements (e.g., size, complexity, performance requirements)
3. Estimate the selected attributes of the work products and tasks
4. Estimate the labour, machinery, materials, and methods that will be required by the project

---

**SP 1.3.1 Define Project Life Cycle**

- Decompose the project into phases
  - Provide for planned milestones at which evaluation and decision making occur
- **For Software Engineering**
  - Project phases typically based on a software development model that considers interdependencies and appropriate sequencing of software project activities
- **For Systems Engineering**
  - Project phases typically based on a product development model that considers the current state of the product (e.g., concept, static prototype, working model), expected future phases, and the relationships and effects among phases

---

**SP 1.4.1 Determine Estimates of Effort and Cost**

1. Select the models and/or historical data that will be used to transform the attributes of the work products and tasks into estimates of the labour hours and cost
   - Historical coding productivity is 10 lines of Java per hour, historical labour productivity is 75%
2. Include supporting infrastructure needs when estimating effort and cost
   - Eg. Computing resources and software engineering facilities
3. Estimate effort and cost using models and/or historical data

---

**SP 1.4.1 – Inputs to Estimation methods**

- Judgemental estimates provided by an expert or a group of experts (e.g., Delphi Method)
- Risks, including the extent to which the effort is unprecedented
- Critical competencies and roles needed to perform the work
- Product and product-component requirements
- Technical approach
- WBS
- Size estimates of work products and anticipated changes
- Cost of externally acquired work products
- Selected project life-cycle model and processes
- Life-cycle cost estimates
- Capability of tools provided in engineering environment
- Skill levels of managers and staff needed to perform the work
- Knowledge, skill, and training needs
- Facilities needed (e.g., office and meeting space and workstations)
- Engineering facilities needed
- Capability of manufacturing processes
- Travel
- Level of security required for tasks, work products, hardware, software, personnel, and work environment
- Service-level agreements
- Direct labor and overhead
**SP 2.1-1 Establish the Budget and Schedule**
1. Are there milestones that must be in the schedule?
2. Identify any assumptions about the schedule
3. Identify constraints that limit flexibility of management options
4. Identify task dependencies
5. Define the budget and schedule
6. Establish corrective action criteria
   - What would be a significant deviation from the plan?

**SP 2.1-1 Input to Definition of Budget & Schedule**
- Defining activities of appropriate duration
- Defining milestones of appropriate time separation
- Defining a management reserve based on the confidence level in meeting the schedule and budget
- Using appropriate historical data to verify the schedule
- Defining incremental funding requirements
- Documenting project assumptions and rationale

**SP 2.2-1 Identify Project Risks**
1. Identify risks
2. Document the risks
3. Review and obtain agreement with relevant stakeholders on the completeness and correctness of the documented risks
4. Revise the risks as appropriate

**SP 2.3-1 Plan for Project Data Management**
- Various forms of documentation are required to support a project in all of its areas (e.g., management, software engineering, configuration management, quality assurance)
1. Establish requirements and procedures to ensure privacy and security of the data
2. Establish a mechanism to archive data and to access archived data
3. Determine the project data to be identified, collected, and distributed

**SP 2.4-1 Plan for Project Resources**
- Top-level WBS from SP1.1-1 is expanded by decomposing the top levels into work packages
1. Determine management process requirements
2. Determine staffing requirements
3. Determine facilities, equipment, and component requirements

**SP 2.5-1 Plan for Needed Knowledge and Skills**
- Staffing requirements are dependent on the knowledge and skills available to support the execution of the project
1. Identify the knowledge and skills needed to perform the project
2. Assess the knowledge and skills available
3. Select ways to provide needed knowledge and skills
4. Incorporate acquisition of needed knowledge and skills into the project plan
SP 2.6-1 Plan Stakeholder Involvement

- Stakeholders are the people and functions needing representation in the project
- Their relevance and the degree of interaction for specific project activities must be identified and incorporated

1. Include stakeholder participation in the plan

SP 2.7-1 Establish the Project Plan

- Produce and distribute a documented plan
  - Addresses all relevant planning items to achieve the mutual understanding, commitment, and performance of individuals, groups, and organizations that must carry out or support the plan
- The plan ties together in a logical manner:
  - project life-cycle considerations; technical and management tasks; budgets and schedules; milestones; data management, risk identification, resource and skill requirements; and stakeholder identification and interaction.

SG 3 Obtain Commitment to the Plan

SP 3.1-1 Review Other Plans that Affect the Project
  Eg. Quality Assurance Plan, Measurement Plan
SP 3.2-1 Reconcile Work and Resource Levels
  Typically accomplished by lowering or deferring technical performance requirements, negotiating more resources, finding ways to increase productivity, outsourcing, adjusting the staff skill mix, or revising all plans that affect the project or schedules
SP 3.3-1 Obtain Plan Commitment

Tailoring the Process

- The CMMI model is a “Best Practice” model
- Every task in the process needs to be considered by you
- Not every step needs to be carried out by you
- You must decide which tasks will add value to your management of your mini-project
- You have limited time and resources, where do you want to spend them?

COMP 3710 Software Project Management S2 2003
Lecture 1 – The End

Professor Ross Jeffery
K17, 405, 9385 6182,
rossj@cse.unsw.edu.au
and Mike Berry
mberry@cse.unsw.edu.au