Universitas Esa Unggul

Smart, Creative and Entrepreneurial

CPL230-PENGEMBANGAN PERANGKAT LUNAK (PERTEMUAN-14)

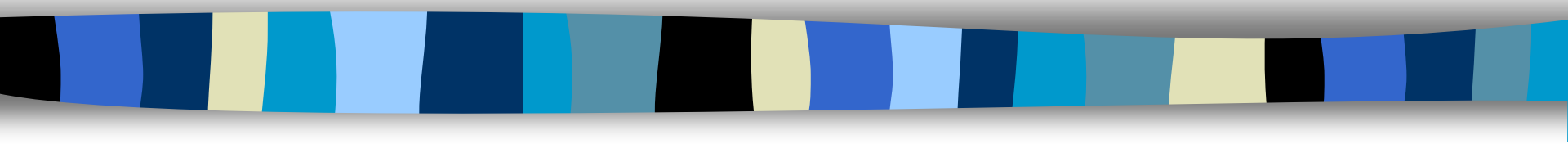
www.esaunggul.ac.id

Dosen Pengampu :

5165-Kundang K Juman

Prodi Teknik Informatika Fakultas Ilmu Komputer

Software Project Management Concepts



Infsy 570

Dr. R. Ocker

project management activity encompasses

- measurement and metrics
- estimation
- risk analysis
- schedules
- tracking and control

consider management techniques required to

- plan
- organize
- monitor
- control

sw projects

these activities are discussed in Part Two of text

Effective SW project management focuses on 3 P's:

- people
- problem
- process

1. people

- must be organized into effective teams
- motivated to do high-quality work
- coordinated to achieve effective communication and results

2. problem

- must be communicated from customer to developer
- decomposed into its parts
- positioned for work by SW team

3. process

- must be adapted to the people and problem
- to get the SW developed:
- common process framework is selected
- appropriate SWE paradigm is applied
- set of work tasks chosen

People

3.2 People

- very important component to success of SW project
- players:
  - senior managers - define business issues that impact project
  - project (technical) managers - must plan, motivate, organize and control the project team
  - customers - specify requirements
  - end users - use the sw

3.2 People

- team leaders:
  - lead the sw development team
- MOI model of leadership
  - motivation
  - organization - of processes
  - ideas or innovation - encourage creativity

3.2 People

- Another view - 4 key traits of effective project manager
  - problem solving skills
  - managerial identity - take charge of the project
  - achievement - reward initiative and controlled risk taking
  - influence and team building - ability to read people

Organization of SW teams

A. Mantei(81): team structures

- SW teams can be organized into number of different team structures
- appropriate team structure depends on type of problem task
- 3 generic team organizations

1.democratic decentralized (DD) (fig. 1)

- no permanent leader
- task coordinators appointed for short time and then replaced
- decisions made by group consensus
- horizontal communication

2.controlled decentralized (CD) (fig. 3)

- leader who coordinates tasks
- secondary leaders responsible for subtasks
- group problem solving
- horizontal communication among subgroups
- vertical communication along control hierarchy

3. controlled centralized (CC) - i.e. chief programmer team (fig. 2)

- top-level problem solving and team coordination managed by team leader
- vertical communication between leader and team members

7 project structures to consider when planning structure of swe teams:

1. difficulty of problem to be solved
2. size of programs (LOC and function points)
3. lifetime of team
4. degree to which problem can be modularized (and structured)
5. required quality and reliability of system being built
6. rigidity of delivery date
7. degree of communication required for project

Project structures

- table 3.1 in text - summarizes impact of project characteristics on team structure (Mantei81)
- centralized structure completes tasks faster - better at handling simple problems
- decentralized structure - generates more and better solutions so better at more difficult problems

B. Constantine organization structures

- discusses four **organizational** paradigms for SWE teams
- Projects or other tasks can be coordinated by:

1. Traditional Hierarchy of Authority

- closed paradigm
- standards and rules
- stability valued (no deviation from norm allowed)
- pyramid or hierarchical organizational structures
- collective interests come first
- demonstrate loyalty and defer to group
- examples: military or government

2.Innovative Individualism

- random paradigm (opposite of closed paradigm)
- independent individual initiative
- innovation and change through creative autonomy
- individual has freedom to create and act independently
- individual more important than group
  - examples: breakthrough project teams developing new technology

3. Adaptive Collaboration

- open paradigm
- integrates innovation with stability and individual with collective
- interests through negotiation and discussion
- roles and responsibilities are flexibly shared

4. Harmonious Alignment

- synchronous paradigm (antithesis of open paradigm)
- alignment with a common vision or direction (channeless communication)
- unified, parallel action through agreement and shared knowledge
- example: Amish barn raising

Strengths and Weaknesses (Refer to Table 2)

- 1. Traditional hierarchies
  - strengths: stability and predictable performance
  - weaknesses: lack of innovation
- 2. Random paradigm organizations
  - strength: creative invention
  - weakness: not highly stable or efficient

Strengths and Weaknesses (Refer to Table 2)

- 3. Open paradigm organizations
  - strengths: complex problem solving (sharing of diverse opinions)
  - weakness: slow due to debating issues.
- 4. Synchronous paradigm
  - strength: efficiently perform established procedures
  - weakness: may not be highly responsive or adaptive to change.

Team Building (Refer to Table 3)

- activities to build group cohesive
- Effective team building helps a team establish an appropriate organization and work culture
- means of increasing performance levels
- activities should be selected based on the organization, management, and culture of the team

Want to achieve a cohesive team:

- objective for every project team
  - synergy, jelled team (DeMarco and Lister)
- jelled teams:
  - more productive and motivated
  - share common goal and culture
  - sense of eliteness

Project Leadership

- ideally, style of leadership should fit team paradigm

Characteristics of managers by type of team:

- 1. Random teams - a respected member of the team; a
- charismatic leader; does not give orders
- 2. Open teams - supply structure that helps keep team focused;
- team players but also facilitators

Characteristics of managers by type of team:

- 3. Closed teams - strong leaders who give clear directions;
- manage by results
- 4. Synchronous teams - visionary leaders; observe and monitor
- performance and watch for changing trends

None of the paradigms is ideal for software development.

- SW development requires
  - complex analysis
  - innovation
  - predictable, routine tasks

Structured Open teams are:

- a hybrid of team paradigms
- a combination of closed (formal, fixed, or hierarchical) and open (shared, flexible, egalitarian) paradigms
- uses formal structures to promote flexibility and efficient problem solving
- catalog of essential team roles
- formal specification of functional roles
- default assignment of roles to assure essential functions are performed

Structured Open teams are:

- rotation of roles
- organized continuous record of what the group does (structured,
- externalized group memory)
- clear and simple external accountability
- technical consensus building
- promotion of personal ownership

Problems in SW development

A. Curtis et al. (88) Problems associated with large SW projects

- A Layered Behavioral Model of Software Development Processes
- developed model that emphasizes factors that affect psychological, social, and organizational processes to illustrate how they affect productivity and quality.

layered behavioral model consists of the following levels:

- **1. Individual** - analyzed as an intellectual task, subject to the effects of motivational and cognitive processes.
- **2. Team** - social processes interact with cognitive and motivational processes.
- **3. Project** - several teams integrating their work on different parts of the same system.
- **4. Company** - analyzing how company goals, corporate politics, culture and procedures affect the project.
- **5. Business Milieu** - looking at the overall business environment such as other corporations, co-contractors, customers, etc.

A. Sample and Study Procedures

- study consisted of 17 field study projects
- Structured interviews were conducted which were approximately one hour long with system engineers, senior software designers and project managers

B. Analysis of the Interviews

- three primary problems
- **a. the thin spread of application domain knowledge**
- **b. fluctuating and conflicting requirements**
- **c. communication and coordination breakdowns**

A. The Thin Spread of Application Domain Knowledge

- lack of deep integration of various knowledge domains required to integrate a complex system.

Thin spread of knowledge

- Specification mistakes often occurred when designers lacked sufficient application knowledge to interpret customer intentions.
- **c.** Exceptional designers stood out in the study as a scarce resource. Exceptional designers performed broader roles than design and were recognized as the intellectual core of projects.

B. Fluctuating and Conflicting Requirements

- The design team often negotiated to reduce conflicts and limit requirements to those that could be implemented within schedule, budget and technical constraints.
- Individuals often created a hidden source of requirements fluctuations as they added unrequired enhancements.

C. Communication and Coordination Breakdowns

- Large projects required extensive communication that wasn't reduced by documentation.
- Barriers to communication (often political) created a need for individuals to span team boundaries and create informal communication networks.

Implications for Software Tools and Practices

1. Increase the amount of application domain knowledge across the entire software development staff.

2. Development tools and methods must accommodate change as an ordinary process

3. Any software development environment must become a medium of communication to integrate people, tools, and info.

Implications for Project Management

- Recruiting and training must be coupled with team building to translate individual talent into project success.

Implications for Software Process Models

- Developers were frustrated when project conditions would not let them follow the process model.
- Models that only prescribe a series of development tasks provide no help in analyzing how much new information must be learned by a project staff.

B. Kraut & Streeter (95) Coordination and communication issues

- Researchers studied 65 projects involving hundreds of staff.

Project coordination techniques used:

- Formal information exchange procedures were used more when the projects were certain and were in the planning stages.
- Informal -interpersonal communication was used frequently regardless of project size, certainty, or life cycle stage.
- Electronic communication was used when projects were heavily dependent on input from other groups in the

Implications

- Personal communication is important for successful coordination, but it may be too expensive to be an effective communication mechanism.
- Software engineers must acquire information from those who are remote
- **communication tools for conferences or distributed meetings** are likely to prove more beneficial than FtF meetings

3.3 Problem

- dilemma of project manager
- at beginning of project, quantitative estimates and a project plan are needed
- no detailed analysis of requirements at this point to base these on
- need to examine the problem to establish scope and boundaries of problem

determine software scope

- first project management activity
- to determine scope, make a high-level surveillance of:
  - context - how does sw fit into the larger context of systems or business, what are the constraints
  - information objectives - what data objects are required for input and produced as output (analysis of inputs and outputs)
  - functionality of sw

project decomposition

- decompose complex problem into smaller pieces
- trying to put some structure to the problem in order to make estimates and complete high-level planning

Process

3.4 Process

- must select the process model that is appropriate for the project at hand
- refer to your process model

project manager

- decide on process model
- define preliminary plan based on process model
- decompose process (add details to the plan)

Melding the problem and process (cookbook approach)

- choose process model
- adapt common process framework (CPF) to process model
- CPF defines framework activities that apply to all sw projects
- task sets - collection of:
  - sw tasks, project milestones & deliverables
- adapt framework to sw project and team

Example:

- framework activities might include:
  - customer communication
  - planning
  - risk analysis
  - construction and release
  - customer evaluation

Example:

- then create matrix (fig. 3.2) -
  - major problem function X framework activities
  - cells - swe tasks
- project manager must estimate resource requirements for each cell, start and end dates for tasks, deliverables to be produced etc. (more in chps. 5 & 7)

Unit Review: Software Project Management

- Using the Process Model Comparison chart you developed, create a model/framework that addresses and incorporates the information obtained in the readings associated with the *Organization of SW teams* and *Problems in SW development.*

Unit Review

- This model/framework should identify, organize, and relate the various variables that influence the selection and application of a given process model to characteristics of the sw development task and situation in light of issues and problems identified in the readings.

Unit Review

- This exercise requires that you synthesize the material contained in the readings/chapter for this unit, and apply this knowledge to the previous material on processes.
- This is a group activity. Each group will make a 20-25 minute class presentation. Your group will be graded on both content and clarity of presentation. All group members will receive the same grade.

Hints:

- Review the models contained in the readings/chapter.
- Relate the models together to form a cohesive model. This model may have several levels (e.g. macro to micro).
- Incorporate/apply to the process models.