

Mod - 2

2.1 ✓ SW Estimation

✓ LOC (line of code)

✓ FP (Function point)

✓ Basic COCOMO model

✓ SW Project management plan (SPMP)

2.2 ✓ Scheduling

✓ Work breakdown structure

✓ Gantt chart

✓ Tracking the schedule

2.3 ✓ Risk Identification

✓ Risk Assessment

✓ Risk projection

✓ RMM plan

2.4 ✓ SW Configuration items

— SCM process

✓ Identification version control

✓ Change control

— Configuration Audit

✓ Status Reporting

Software Estimation

- The process of predicting the effort, time & resources required for developing a software project using object oriented principles.
- Analysing the factors such as project scope, requirements, available resources, capabilities to provide estimate for project planning.

Steps

- ① Define scope → understand the scope of project, features, functionalities.
- ② Gather requirements → details about what is needed from stakeholders & other relevant people, what is expected from project.
- ③ Identify tasks → breakdown project into smaller tasks and try achieving them.
- ④ Estimation technique → based on proj. characteristics, used estimation techniques → function point analysis (FPA), use case points (UCP), etc.
- ⑤ Assign Effort → considering complexity, determine effort for each task.
- ⑥ Estimate duration → timeline for completing each task, based on resources and team.
- ⑦ Risk assessment → identify the potential risks & their impact.
- ⑧ Review & validate → review the estimates with stakeholders, including project sponsor, developers & end-to-end users to match project goals.

- ⑨ Document Estimates → document the estimation process, assumption made.
- ⑩ Monitor & update → continuously monitor the progress & compare that to estimated timeline

* SLW cost Estimation

- for new slw project, we should need to know the cost to develop the project (to be done before dev.)
- Project scope, slw metrics for evaluation, shld be done
- project shld be broken into pieces when estimating.
- Delay estimation.

* Estimation Models

- a) static single variable model → when model uses only single variable to estimate / calculate values of cost, time, effort, etc
- b) static multi-variable model → use multiple factors to estimate time and cost in slw development.
Take into consideration various env. eqns.

LOC → line of code (estimation technique).

- Simplest among all to estimate the cost of project.
- Projects size is calculated by counting no. of source instruction in developed program.

- while counting the number of source instruction, comments are ignored.
- when project is completed, doing LOC is easy but at start, the estimation is hard.
→ for estimating LOC at start, project is divided into that many parts until each part can be called for fixed LOC.

ADV

- simple & straightforward metric
- basic project size can be known

DISADV

- names significantly based on language used.
- complexity of proj. is ignored.
- only quantity & no quality.

Function Point (FP)

- measures the size of application based on functional view of system
- size is determined by counting no. of inputs, outputs, queries, internal & external files & adjusting the total for functional complexity of system

* objectives

- Accurate estimate → Effort, time and resources for development and maintenance.
- Improved tracking → Monitor & manage projects, track progress.
- Enhanced costing → estimate development & maintenance cost based on complexity.

- Risk identification → potential issues due to complexity
- standardize measures → common metric for project tracking
- Better communication → discussion on projects scope & complexity
- Technology agnostic → compares project irrespective of language

Types

- a) External input (EI) ⇒ user supplied data entering the system (eg → username, password).
- b) External output (EO) ⇒ Information delivered to the user (eg displaying amount).
- c) External inquiry (EI) ⇒ user questions answered without modifying any data (eg → account balance).
- d) Internal logical file (ILF) ⇒ internal data structure managed by system (eg → customer dataset).
- e) External Interface File (EIF) ⇒ Data exchanged with another system (eg → sending info to supplier).

Benefits

- Technological independence (language)
- Better estimation
- Improved interaction
- Early recognition of changes in scope.

COCOMO (constructive Cost Model)

- slow cost estimation model → used to estimate effort, time and cost required to develop a project based on size & other factors.
- widely used for early stage planning.
- can predict efforts and schedule based on size of project.

Three types / categorized :

a) Organic ⇒

- A small group work in a familiar environment to develop well understood application, less "innovation and constraints", deadlines are less & dev. env. is suitable.

b) Semi detached ⇒

- Project team is a mix of experienced & inexperienced staff.
- Medium size proj & innovation is required.
- constraints & deadlines are moderate & dev. env. is fluid.

c) Embedded ⇒

- The S/W is strongly coupled to h/w, eg: ATC, ATM, weapon system.
- Project team is large, lots of innovation, tight deadlines & constraints.
- Dev has many complex phases.

* COCOMO SUM

SPMP (Software Project Management plan)

- SPM is a document that outlines how a software project has to be managed from start to end.
- This discusses the approaches, strategies for managing software development project.
- SPM is a crucial component for projects documentation and serves the roadmap for project team.

SPMP covers :

a) Introduction

- (1.1) Project Overview ⇒ description of purpose of project, expected delivery date.
- (1.2) Project deliverables ⇒ list of all functionalities & estimated date of delivery for each.

a) Project Organisation

- (2.1) SILC process Model ⇒ describe SPM used
- (2.2) Roles and responsibilities ⇒ identify and set roles for team members, have proper communication in them.
- (2.3) Tools & techniques ⇒ specify methods, prog. lang., techniques, tools to be used in project.

2) Project Management plan

(2.1) Task → identify tasks involved in executing the project
task should be specified clearly.

(2.1.1) Task-n → unique identifier for each task.

(2.1.1.1) Description → description in detail of each task.

(2.1.1.2) Milestones & delivery

(2.1.1.3) Resources needed for task → to complete its execution

(2.1.1.4) Dependencies → identify dependencies required based on the constraints.

(2.1.1.5) Risk & contingency → risk in completing task on time
plans to cope up risk.

(3.2) Assignments → task assignment to team members.

(3.3) Timetable → estimated start & completion of each task
(Gantt chart).

* Budget, Approvals, Quality assurance & communications are also taken care of.
(One liners on each).

2.2

#

Scheduling

- Defining the timeline of completing the project tasks, taking in consideration duration dependences & deadline.
- Planning and organising the tasks and activities required to develop a software project using OO principles

Scheduling process →

- a) Task identification → identify all tasks and requirements to complete those tasks.
- b) Estimation → estimate effort, time & resources required may use estimation techniques like FA.
- c) Task sequencing → order of execution for each task based on dependencies & priorities.
- d) Resource allocation → Assign resources including human resources, equipments for each task.
- e) schedule development → develop a schedule that gives timeline for each task.
- f) Milestones → identify milestones of deliverables to mark significant pts in development.
- g) Monitoring & controlling → continuously monitor progress against the timeline & make adjustments acc.

b) Communication \Rightarrow keep stakeholders informed about project progress and if any updates or changes are done.

Work breakdown structure (WBS)

* The process of dividing complex projects to simple & manageable work items is called WBS.

- * Displayed as a tree, root = name of project
 - & each node is recursively decomposed into smaller sub activities, until leaf node cannot be further divided.

- ~~tree diagram~~
steps
- ① identify major activity
 - ② divide into sub activity.
 - ③ Repeat division until independent activities are only left.

Uses of WBS

- * Cost estimation \Rightarrow precise cost estimation of each task.
- * Time estimation \Rightarrow time taken for each task.
- * Easy management & organisation.

Grantt Chart

- * A visual representation of project schedule, that helps in planning, managing & tracking tasks throughout development cycle.
- * Main objective is to provide a clear & organised view of project progress, timelines & dependencies etc.

\Rightarrow consists of \Rightarrow left column = project tasks (list).

\Rightarrow horizontal bar against each task to depict timeline.

* Components

- 1) Tasks → each task in project is listed in the chart.
- 2) Timeline → horizontal axis ~~is~~ divided on month, day, week
shows the duration of task.
- 3) Bars → to ~~depict~~ depict the start & end of a task,
length indicates this.
- 4) Dependencies → arrow / line connecting tasks indicate
dependencies, sequence of execution.
- 5) Milestones → significant milestones like delivery are marked
on the chart.
- 6) Resource allocation → assignment of no. of team members,
etc.

ADV

- Visual rep. of project ~~at~~ timeline, resources, etc.
- Easy planning & scheduling tasks
- Comm. among stakeholders.
- Tracking the milestones
- Resource management
- Easy decision making
- coordination b/w team members.

Tracking schedule

- A roadmap that defines the tasks and milestones that are to be tracked and controlled as project proceeds.
- Imp. activity for project managers, thus they can control timelines and monitor development.
- We have tracking & scheduling tools that automatically update plans as info. is inputted/entered.
- Tracking can be done via periodical project status meetings, evaluating the reviews, setting tentative deadlines, meeting with resources informally,

Features of project scheduling tools.

- Time management \Rightarrow tries to keep project on its timeline.
- Resource allocation \Rightarrow provides proper resource allocations
- Team collaboration \Rightarrow proj scheduling tool improves comm.
- user friendly interface.
- Time boxing \rightarrow when manager comes to know that project is not meeting the timeline & he needs to make a decision on the project scheduling
 \rightarrow It identifies as, whole project won't be delivered on time. hence, incremental slow process is used & schedule is proceeded for each incremental delivery.

2.3

#

RISK IDENTIFICATION

- Identifying potential risk that may impact projects objectives.
 - If failure occurs in identifying any specific or particular risk, then other steps in risk management are not applicable to this.
- To identify risk, project team should review scope, estimate cost, schedule, technical maturity, etc.
- Risk can be internal or external and can arise from many sources such as ~~technical~~ technical or organisations.
- * Two distinct types of risks
- generic risk → potential threat to any type of risk.
 - product specific risk → can be identified by only those who have deep knowledge about tech & project.

Types of risks

- Technology, people,
- tools, requirement,
- estimations, organisational risks.

Methods for identifying risks

- 1) checklist analysis → predefined tasks of prone risks.
- 2) Brainstorming → open participation of team members to spontaneously gather risk in creative environment.

- 3) Causal mapping \Rightarrow reviews failure, factors & causal cause of effect to facilitate learning & post project eval.
- 4) SWOT analysis \Rightarrow analyses internal strengths, weaknesses & opportunities & threats to identify risks - within org.
- 5) Flowchart method \Rightarrow represents process activities graphically to identify any potential risk.

Risk Assessment

- (prioritize risk)
- In case, risk assessment involves evaluating & prioritizing potential risk that could impact successful completion of a s/o project.

*Methods in Risk assessment

- 1) Prob. matrix & Impact matrix \Rightarrow assessing each identified risk based on its prob. of occurrence & impact on process.
- 2) Risk severity assessment \Rightarrow risk is evaluated based on seriousness of project impact on project.
- 3) Risk exposure analysis \Rightarrow involves calculating overall exposure of each risk by multiplying its prob. of occurrence with its potential impact = exposure value.
- 4) Risk ranking & scoring \Rightarrow risks are ranked based on certain criteria like impact, likelihood etc.

5) Expert Judgement → consulting domain experts, stakeholders to assess & prioritize risk.

* Steps in Risk Assessment

- 1) Identify the risk → list all risks & their impacts on project.
- 2) Evaluate risk → Assess & evaluate risk based on impact, likelihood.
- 3) Prioritize risk → based on severity, exposure & ranking.
- 4) Develop mitigation strategies → to manage & prioritize risk and measure to reduce the impact of ~~risk~~.
- 5) Monitor & review → monitor proj. environment for changes that may affect identified risk
Review & update risk assessment regularly.
→ This identify and prioritize and address potential risks.

#

Risk Projection

- Risk projection meaning estimation of the risk.
(~~risk~~)
- ⇒ two ways to ~~do~~ → Prob. that the risk is real
→ consequences of prob. associated with risk.

- The project planner, tech staff, project manager performs some steps for risk projection!

- steps for risk projection
 - A scale to indicate prob. of risk being real.
 - consequences of risk
 - impact of risk on project & product.
 - overall accuracy of risk projection to get clear understanding of SIC to be built.
- These steps help in prioritizing risk & thus making it easy to allocate resources to handle them.

RMMM Plan

- Use of a risk management strategy in SIC project plan can be divided into Risk mitigation, Risk Monitoring, & Risk Management plan.
- The RMMM plan, documents all the work done as a part of risk analysis & is an integral component of overall project plan.
 - ⇒ SIC team sometimes do not prefer RMMM doc, instead the risks are documented individually in Risk Info Sheet (RIS)
 - ⇒ Risk mitigation is called as problem avoidance activity.
 - ⇒ Risk monitoring is considered as project tracking activity
 - ↳ to assess if predicted risk do even occur.
 - ↳ make sure risk avoidance steps are applied correctly.
 - ↳ gather data for future risk analysis.

* Risk mitigation

- Activity to avoid risks .
 - ↳ finding out risk
 - ↳ remove the reason for risk creation
 - ↳ conducting timely review to speed up work.

* Risk Monitoring

- Activity for project tracking .
 - ↳ check if predicted risks occurs or not .
 - ↳ ensure proper application of risk avoidance step .
 - ↳ collect data for future risk analysis .

* Risk Management & planning

- Assume that mitigation failed & risk is a reality . Thus this task is done .
- If project manager effectively uses project mitigation to remove risk , then it is easy to manage the risk .
- Main obj. of risk management plan is risk register , that describes & focuses on predicted threats to slow projects .

20N

Software Configuration Management (SCM)

- In SW dev. changes during maintenance are common & must be managed effectively.
- A team of individuals produces work products, module versions, test data forming a SW configuration that grows rapidly as SW dev. progresses.
- The SCM process involves systematically managing & controlling changes to SW throughout the SW development cycle.
→ SCM → identifies changes, monitors & controls changes, ensures proper implementation of change, auditing the change.
- SCM defines a no. of tasks.

① Identification →

- Involves identifying & documenting all components in SW configuration. (version, component, release).

② Version Control →

- V.C. manages changes to SW configurations by tracking diff. versions of the components used in SW.
- ensure that dev. works on correct version of SW
→ provides history of changes.

SLW config

③ Change Control

- Regulates the process of making modifications to the SLW configuration.
- Includes procedure for submitting, reviewing, approving & implementing changes.
- Change control ensures that changes are properly evaluated, documented to avoid any issue.

④ Configuration auditing

- Verify & validate the SLW configuration & complies with its specifications, requirements etc.
- Examines all components versions & documentation to ensure consistency & completeness.
→ may be conducted periodically.

⑤ Status Reporting

- A regular update on status of SLW configuration, including progress of development, changes implemented, and any issues.
- Helps stakeholder stay informed about current state of SLW system & helps in decision making.

By performing these, SCM process helps understand SLW, ensures integrity, stability & quality of SLW configuration throughout time.