

Software Engineering

Why system fails?

- Customer change their requirement continuously
- communication gap

Software characteristics

- No spare parts

CS: main problem solve करने की समस्या

SE: which option is best among all option

SDLC

A structured set of activities required to develop a software system.

- 1. Requirement analysis [business objective वाले
एवं user एवं location
के बारे में
functionality culture
wise differences]
- 2. Designing / modeling

* As is flowchart (Activity Diagram)

* To be " (After confirm, draw class diagram)

- 3. Coding / Development

4. Testing (class)

5. Implementation / Integration phase
- (Delivery) 6. Operation / Maintenance (new feature add, bug remove, ...)
7. Documentation (testing & for error cases, new req. change QAT? ...)

Good & Bad Software

Good software is maintained
Bad u. discarded

When a hardware component wears out it is replaced by a spare part. There are no software spare part

Maintenance

1. corrective: bug removal for user [about 20%]
2. enhancement: new feature add [u 80%]



Cost of detection & correction of a fault

Requirement stage \rightarrow fault \rightarrow cost low

Requirements \rightarrow 1

specification \rightarrow 3

Design \rightarrow 4

Implementation \rightarrow 10

Integration \rightarrow 30-52

Maintenance \rightarrow 200-368

Sunday → quiz 1 → Lec 1, 2, 3

Embedded software → Hardware related

* * * Software myths [short question, broad question]

* Software Process

* set of structured process

Model: abstract representation

* Waterfall model

* It is a linear process

* Next step → अपेक्षित इनपुट की step must complete
प्राप्ति हो।

* only one testing
V-model

- subset of waterfall model, Linear process
- No backtracking
- Testing important [प्रारंभिक phase के testing नहीं]

Prototyping model

software → कोड page for user requirement
कोड तृप्ति हो।

customer

Evolutionary Development

5-9

1) Exploratory

Two types of proto-type

1) throw-away : अंत में तो क्या होगा
चाहे इस विचार से नहीं बदला जाए

2) Exploratory : Requirements विभिन्न ड्रॉग और ड्रॉप
करके विचार करते हुए लेबल देते हुए

2) Incremental Development

subset of waterfall.

waterfall → पूरी प्रोजेक्ट को एक बार नहीं होता

• Failure का chance कम होता है

• Increment 0 की तरफ एक बार नहीं होता

• Large Project

* Early release

लाइन प्रिंटिंग

प्रोजेक्ट का एक बार नहीं होता

प्रोजेक्ट का एक बार नहीं होता

Waterfall Model

Isom-V.

Feasibility study → Requirement analysis and specification →
Design → coding and unit testing → Integration →
maintenance

Advantage

- ① Bare model
 - ② Simple and easy
 - ③ Small project

disadvantage

- 1) No feedback
 - 2) No experiment
 - 3) No parallelism
 - 4) High risk
 - 5) Efforts maintenance

Project category: It is used for projects with clear and straightforward objective.

- Well-defined requirements, sequential nature, strong documentation, limited flexibility

size: small

*No backtracking

step by step ready
when the whole system is designed

V-Model

- * Known as ~~validation~~ verification and validation model.
- * subset/extension of waterfall model
- * Testing is associated with every phase of life cycle.
- * verification phase (req. analysis, system design, Arch., module)
- * validation

Advantage

time saving

every component must be testable
Progress can be tracked easily, early defect detection & correction

Disadvantage

No feedback

Risk analysis not done

Not good for big or object oriented projects

Rapid Action Development

Parallel ~~the~~ increment ~~of~~ ~~the~~ requirement related ~~to~~ ~~the~~.

* Big Budget. * 90 days

* incremental

* Plan Based Process model → subset of waterfall model

Component Based Development model
for code reuse

RUP

subset of incremental

model	Category	size	duration	Team size	Complexity
Waterfall	Well-defined requirements		Long (months to years)	Large	Simple
V-model	sequential	Large	Long	Large	Simple
Prototype					High
Incremental	Incremental	Medium to large	Medium to long	Medium	Moderate
Iterative					
Agile					

fuzzy \rightarrow compiled

■ Iterative Process model [incremental model]

* timebox fixed

[\Rightarrow req. at same stage, duration move 2nd]

* SFA increment [\Rightarrow duration same] [increment

* a single iterative duration \rightarrow 3 to 6 month [\Rightarrow better]

* incremental time differ off,

Timebox: Length of an iteration

each iteration like is like an individual project

■ Agile

• subset of iterative.

• time box fixed

* iteration \rightarrow 1-4 weeks [a single iteration]

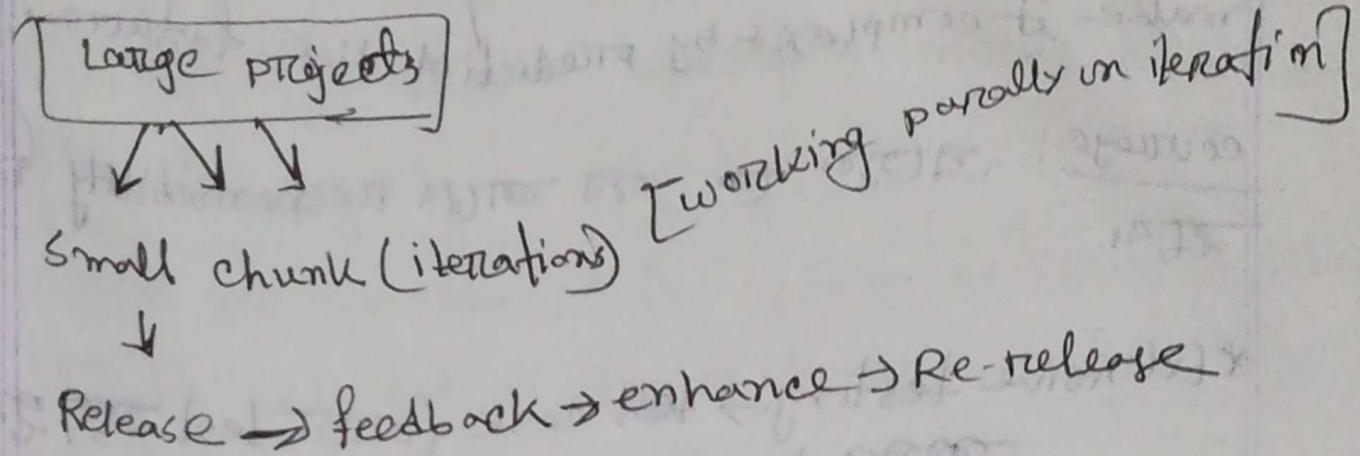
* low failure rate

* complex project \Rightarrow Agile

Agile proponents Believe

Requirement change \Rightarrow possibility high. Define customer interaction

Agile (more quality)



Advantage

- 1) frequent delivery, face to face communication with client, changes, time constant

Disadvantage

- Less documentation, maintenance problem

Extreme Programming (XP)

- XP use for small to medium size project.
- team size small, process iterative, customer intervention, important flexible.

XP values

Communication, simplicity, Feedback, courage, respect

Simplicity! Design immediately / requirement. Don't make it complex by predicting the future.

coulege: প্রযোজন করে তার জন্যে responsibility দেব।

* Extensive testing → test case (ক্ষেত্র test এর পর্যায়) → অধিগ্রহণ development → test base development

XP Process

1) Exploration: user ও requirement gathering এবং প্রযোজন করণ করা হয়।
few weeks to a few month

2) Planning: stories কে লেখা হয়। Estimate effort

3) Iteration to Release } of working with the user stories.

4) Productionizing

5) Maintenance and Death

Iteration to release phase:

Coding Policy: Pair programming (2 programmers One team)

* Time of one iteration: one to 4 weeks to implement

* customer decide iteration stories for iteration
Productionizing

check performance of the system before the system can be released to the customer

* new changes if they are in current phase

* three to one week

P:

(small release)

maintenance phase

update releases

* team size is till approx 225 iteration for

team size adjust approx 225

Roles and Responsibility

D. customer: write story, functional test and decide if req. is satisfied. set implementation priority for the req.

Programmer: write simple code and definite

Tester: help customer [functional test]

run test regularly, broadcast test result
and maintain testing tools.

Tracker: give feedback. Traces iteration.

Coach: responsible for the process as a whole.
Guide other members.

Consultant: external member. Possessing the
specific technical knowledge needed.

Manager (Big Boss): make decision.

[]

team: small to medium size

experience team member

people based rather plan based

team located in one place.

Scrum

- * enough structure
- * lightweight, iterative and incremental model
- * breakdown the development phase into stages or cycles called **sprints**.

* One sprint duration: ~~one~~ two weeks to one month

Product Backlog: requirement store ~~and~~ Q3, requirement prioritize ~~Q3~~

* communication on the daily basis

Scrum master

Product owner

Advantage

- Freedom & Adaptation, High quality, low risk product
- Reduce development time upto 40%.
- Reviewing the current sprint before moving to new one, customer satisfaction

Disadvantage

- No changes in the sprint, efficient for small team size.

[not sprint
change ~~Q3~~ ~~2010~~]

Process

- Pre-game, development game, post game

1) Pre-game

D Planning

(2) Architecture

Design → Design ~~review~~ meeting (I)

Black Box: We don't need to know logic, we need to know how it works.

Pre-game & Iteration 1

2) Game Phase

iteration 1 अंत तक 225, based on the num. of req. we have collected.

Iteration 2 की length का sprint तक 225

One sprint is planned to last from one week to one month.

3) Post-game Phase

iteration 2 की Development की मात्रा 225

③ Retrospective

Roles

Scrum master: Product backlog থেকে sprint backlog

১) copy করে রखি।

interact with project team, customer and management.

Product Owner: can be anyone

make final decisions.

② Daily scrum meeting: throughout the project, every iteration ৬ শাহুম, previous day এ কী করেছেন
রিপোর্ট, তা next day এ discuss করে রখ।

③ Sprint planning meeting

1) Customer & scrum master team মিলে কোটি meeting
করে phase ৬ ফর ফর রয়ে / কোন কোন req. মিলে,

2) scrum master team decide কী করবে কোটি কোটি

④ Sprint review meeting: last day কি meeting, কোটি sprint
করে আগু কোম ইআইস সাবে customer কি product দেখাবে
২য়, কোটি কোটি কোম কোম, তারে অক্ষ কোটি feedback দেয়।

Product retrospective meeting: Total feedback 241

Final Term

Design

wireframe work: Business analyst

Properties

- 1) Firmness: No bug
- 2) Commodity: Purpose should be sure
- 3) Delight: Customer should be get satisfaction after use.

Modularity: ~~কো~~ software for related function
অন্তর্ভুক্ত কো কো আগে আগে এবং এবং

Monolithic: ~~কো~~ page / ~~কো~~ module

Functional Independence

Cohesion

High cohesion important

Coupling

2 modules have strong relation

Module A, module B independent, low coupling

High cohesion, low coupling

Aspect

Some requirements are dependent requirement
depend on ex: user should be 18 yrs old, he/she
would be able to register

Refactoring

Program algo change to 23, outfit

आजाद राज़ी

internal logic change, but
not design.

Modularity: Trade off

num of module तो फैल, cost ज्यादा हो

integration cost ज्यादा हो

We should find out optimal num of module
so that cost should be balanced. SLOC is
minimal.

cost: development cost + integration cost

UI Design

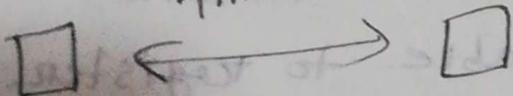
user should not memorize the step

Typical Design errors

Golden rule

time to reach target

Fitt's Law:



Analyze layout element

top left → bottom right

No about: think about

get user to focus first on slabs to move

get why user interested.

where do user moving two hands slide on

one hand and hands free last or

bottom

Q) Software Testing

D) Static testing: logic right or wrong? reviewed

→ test 221

D) Dynamic testing: code runs ~~the~~ from? proper output same from?

Developer → white box testing, unit testing

MCQ Verification: functionality following ~~the~~ form?

Smoke test: whether system is ready for testing

System testing: before delivery final testing from developer

Alpha test: ^{for} user access ~~the~~ 221 for ~~the~~ form

Beta test: Access ~~the~~ 221 for software license through 221 customer free PC installation

15. ChWT

Performance testing → stress testing [load across system for behavior exp]

Low "

H.W: White box testing vs Black Box testing

Debugging

- developer द्वारा करते हैं।

Debugging Technique

Brute force testing: least efficient but most used.

Backtracking:

Cause elimination,

Lather

\$ quality attribute

Availability

* Efficiency: Given setup के लिए software का नियन्त्रण

performance [memory, bandwidth, power ...]

Flexibility: system को ^{feature} add करने की capability

* Integrity: unauthorized access नियन्त्रित करना

Interoperability: Only software आपसी software

जो जानकारी दें, ex: ms word → pdf

g	t	s
5	10	t
9	h	3
2	5	4
sums		

$$87 = 829$$

$$h = 9 \times 6$$

$$-71 = 6 \times 9$$

$$E1B \rightarrow 5$$

$$E2S \rightarrow 3$$

$$I1T \rightarrow 1$$

$$E \leftarrow 3$$

$$E \leftarrow 2$$

E1B
E2S

I1T
I2S

O1S
O2S

For a complex project

Gymnastics: + center, & (3) the same

Athletic music

from now on - Gymnastic competition should be

a single mode - Gymnastic competition

for each year group

Rebustness:

The year groups follow the system

Maintenance

* SMI → Software modification establishe?

how you determine the stability of software?

M_T = num of module in the current release

F_C = ~~n n n n - - -~~ that have been changed

F_A = ~~- - - - -~~ added

F_D = ~~current release (not) delete to current next release (not) n1.~~

$$SMI = [M_T - (F_A + F_C + F_D)] / M_T$$

$$F_A = 2 \quad F_C = 4 \quad M_T = 12 \quad F_D = 1$$

$$SMI = (12 - 7) / 12$$

1 is not stable [0.85] 0.75 = 1

$$IS = 0.85$$

$$SI = 0.82$$

LB

Baseline

* SCM Repository
What is SCM?

what are the SCM layers,
content, change control process

b14

Make-Buy Decision

$$0.30 \times 380,000 + 0.70 \times 450,000 \\ = 429,000$$

Reuse

Major

$$(0.20 \times 310,000 + 0.80 \times 49000) \times 0.60$$

$$+ 0.90 \times 275,000$$

b15

why are project late?

* Effort and delivery time

effort for first development time work.

Resource for the 20,000 cost for 2874,

* Effort Allocation (40-20-40 Rule)

40% - 50% time \rightarrow modeling, req gathering, analysis,
~~efforts~~ collaborate with customer

15% - 20% \rightarrow coding

30% - 40% \rightarrow testing and installation

EVA

project total progress 70%

$$BCWP = 12 + 15 + 13 + 8 + 9.5 + 18 + 10 + \dots - 14$$

$$= 126.50 \quad [\text{Schedule complete no time wasted}]$$

$$BCWS = 126.50 + 16 + 6 + 8$$

$$= 156.50 \quad [\text{Schedule complete no waste}]$$

$$ACWP = 12.5 + 11 + 17 + \dots + 14.5$$

$$= 127.50$$

$$BAC = 582 \quad [\text{Total time}]$$

L-R

Risk Overview

- unpredictable

Proactive - আগে যেতে প্রয়োজন ২২%

Reactive - পরি সহজে কৈবল্য করা?

Impact

Critical \rightarrow high risk

Catastrophic software failure দুর্ভাব চance ২%

RRL

RE before = 15% chance, cost 15,000

RE after = 5%, cost 1500

$$\begin{aligned} \text{RRL} &= (0.15 \times 15000) - (0.05 \times 15000) \\ &= 1500 / 1500 \\ &= 1 \end{aligned}$$

RRL should be greater than 1. so we do not use the system.

DISK

Bohemian 10.1

mcg

Sud-12