

Software Development Life Cycle

1. Communication : Requirement analysis
2. Planning : features, functionalities
3. Modeling : design a model
4. Construction : code, test
5. Deployment : release, maintain

L-2

- Customer know what they want
- Small customer base

↳ waterfall model.

steps : 1. Requirement analysis

2. Design

3. Coding

4. Test

5. Deploy

6. Maintain

Sequential manner

TOPIC NAME: _____

DAY: _____

TIME: _____

DATE: / /

1. Requirement analysis:

a) Take from customer

- ↳ i) address problem in
- ii) Feasibility analysis
- iii) Finalize list

2. Design:

- logical (Database design), physical (server, deployment) designs.

3. Coding:

- i) design is split into blocks and converted in code.

4. Test:

- i) Deploy to people and fix code

5. Deploy, maintain:

- i) user environment deploy
- ii) maintain fix issues

Never go back.

TOPIC NAME : _____

DAY : _____

TIME : _____

DATE : / /

When to us :

- 1) Well known req, not changeable
- 2) small scale project, short term.
- 3) resources - time, human etc available
- 4) Technology - databas, language - not changeable

Advantages :

1. simple, easy to understand, manage, use
2. Each phase has deliverables
3. Results well documented
4. No confusion, by customer as can not change
5. No change in everything

Disadvantages :

1. No parallel activity
2. No change in design
3. No more features
4. No code reuse
5. Not much reflection
6. Risk
7. Not for complex proj
8. working software at the end.

TOPIC NAME: _____

DAY: _____

TIME: _____

DATE: / /

V-model

- Extension of waterfall model
- because of detailed software testing

1. Req Analysis : fix requirements

↳ Acceptance test case design
for req. verification

2. Architecture design : Hardware, language, database etc

↳ Test design of Architecture

3. Module design : Divide components

↳ Test design of component merge

4. Code generation :

↳ Unit test design : test scenario

↳ down to top

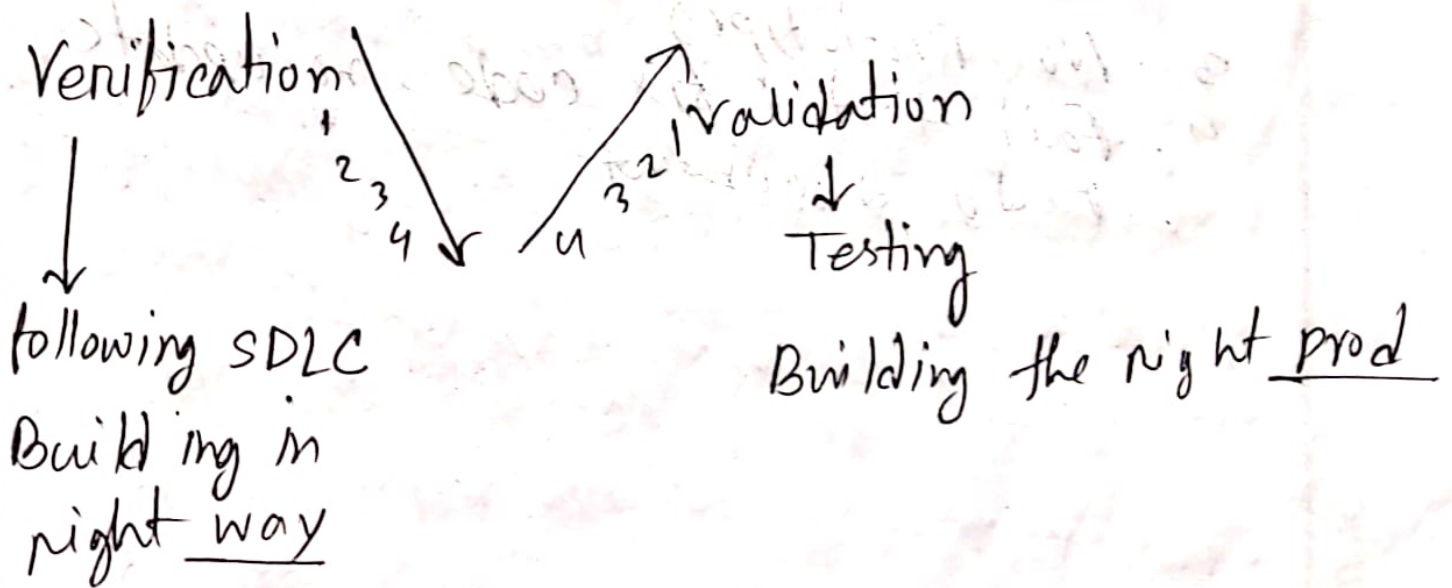
TOPIC NAME : _____

DAY : _____

TIME : _____

DATE : / /

4. Unit testing : test all blocks of code
3. Integration test : Components interacting w each other. So merged codes work or not
2. System testing : Software works with external entities or not. ex: database, hardware, print etc.
1. Acceptance test : Provide to customer and get feedback.



TOPIC NAME: _____

DAY: _____

TIME: _____

DATE: / /

Advantages:

1. verification, validation.
2. Test acts are planned before testing, so ensures better testing.
3. saves time: higher chance of success.

Disadvantages:

1. changes are not welcome
2. Final result in the end.
3. No Prototypes
4. fail test: fix code, no update redo document

~~Incremental~~ Process Model

- Customer don't know what they want

↳ Evolutionary process model

- req may change
- go back prev stage
- Not sequential
- iterative process
- Better than waterfall

1. Incremental : (customer well know n of req)

- customer wants to use software - in beginning
- software will be delivered in increments.

In each increment : SDLC cycle all steps

— n time increment.

— Parallelly work in increment 2 when completing 1.



TOPIC NAME : _____

DAY: _____

TIME: _____

DATE: / /

• When to use:

- fixed increments
- customer want prototypes
- parallel work
- prioritize cus req in each increment
- small → med size proj
- well known requirements

Iterative process Model

- customer not sure about requirements
- Dev not sure about design
- no fixed time as, quality matters
- customer feedback; Long time, complex proj

Iteration 1 → iteration 2

- customer feedback
- fix func

- Not necessary to add more function
- Rework on same func acc to customer till he accepts

TOPIC NAME : _____

DAY : _____

TIME : _____

DATE : / /

Spiral model

1. Unlimited budget, time
2. Risk factor
3. Multiple process model adapt
4. No. exp worker
5. complex

Steps :

1. Objectives identify
2. Risk reduction
3. Dev and validation
4. Planning.

Usages : Unlimited time, budget

- User not sure of reqs
- Req complex
- changes are expected

CMMI :

- ① Initial
- ② managed
- ③ Defined
- ④ Q. managed
- ⑤ optimize

TOPIC NAME: _____

DAY: _____

TIME: _____

DATE: / /

Agile philosophy

- Adaptation: changing requirements, needs, deployment while using software at the same time

Agile - combination of incremental, iterative process

- adaptable → rapid delivery
- while working product launched already

Agile Manifesto:

- focus on people more than process/tool
- working software over document
- customer collaborate over negotiate
- Respond to change over a plan

TOPIC NAME : _____

DAY : _____

TIME : _____

DATE : / /

1. Extreme programming XP

- + New version in a day
- every 2 week increment deliver
- Test builds before deliver

Principles

- incremental
- simple design - good functionality
- collaboration team
- sustainable pace
- customer involve
- pair programming : navigator, observer
 - └ better code
 - └ support
 - └ learning
- Unit testing : testing each method
 - └ better code quality
 - └ write test first then dev
 - └ T First develop
- Refactoring : restructure a code (collaboration benefit)
 - └ reducing duplicate code
 - └ long method → small methods
 - └ variable naming

TOPIC NAME : _____

DAY: _____

TIME: _____ DATE: / /

Agile Unified process (RUP)

4 phase :

1. Identify

req
scope
solution

2. Elaboration

Design archi
More req collect

3. Construction

Iterative implement
→ unit test

4. Transition

deploy - feedback

Disciplines:

project manage — Directing
— coordination

Configuration management — source code track

Environment — proper tools allocate
wide line process

Scrum

3 elements

1. Roles 2. Events 3. Artifacts/charts

Roles : customer - user

stakeholders - who tell dev team

prod owner ← manager/sponsor
Decide-features

└ dates

└ priority

└ cost

- Scrum master — Assign tasks

- Team — 5-10 mem
— types of members

— membership change between sprints
it needed.

④

stakeholder meet owner → requirement analysis is backlog

product backlog

— list of requirements → priority points

— only those that will be delivered
in next iteration in sprint backlog

1
2
3
4
5
6
7
8
9
10

TOPIC NAME : _____

DAY: _____

TIME: _____

DATE: / /

② Dev team select high priority task

③ - sprint backlog done in a sprint

break down tasks
To do, test, progress, Done

④ Every 24 hours :

Daily meeting — scrum meeting
yesterday, today, obstacle

⑤ Burndown chart — work progress chart

⑥ Deploy and plan next sprint

sprint review
sprint retrospective
problems identify