

→ Classical Waterfall Model

* Advantage

- ① Base Model.
- ② Simple and Easy.
- ③ Small projects.

disadvantages

- ① No feedback, Rigid
- ② No Experiment
- ③ No parallelism [phase overlapping]
- ④ High Risk
- ⑤ 60% efforts maintenance

[60% ← 60%]

Page No.: PANKAJ

Date: / /

Iterative Waterfall method = [modified version of classic] feedback occurred

[Feasibility Study]

Requirement Analysis
and Specifications

Design

Coding and
Unit Testing

System testing
and Integration

Maintenance

→ Iterative Waterfall model =

* Advantage

- ① Small projects
- ② Easy and Simple
- ③ Feedbacks
- ④ Base models

* disadvantages:- ① No parallelism ② No Intermediate delivery
③ Rigid (overlapping) ④ Customer Interaction [Loss]

verification and

Vee-model [V-models] ① Also Known as Validation model. [Step-wise step checks]
② Extension of waterfall model. ③ Testing is associated with
Every phase of life cycles.

[Library Management System, Bus Management etc]

Project
definition

Concepts of
operation

Requirements and
Architecture

Detailed
Design

Coding
Implementation

operation
and
maintenance

System Verification
and Validation

Integration, Test
and Verification

Project
Test
and
Integration

Advantage :- ① Time saving

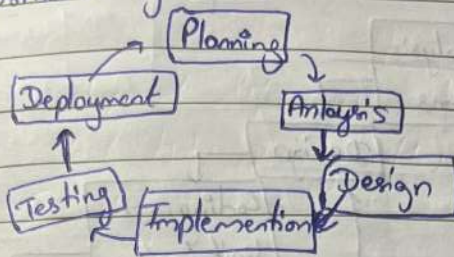
- ② Good Understanding of Projects in the beginning
- ③ Every concept must be testable
- ④ Progress can be tracked easily.
- ⑤ Proactive defect tracking.

disadvantages :- ① No feedback so less scope of Time
Changes

③ Not good for object oriented projects.

- ③ Software Test
 ① Bit Manipulation :- See often that (Some information)

SDLC [Software development life cycle] also referred to as the application development life cycle, is a process for planning, creating, testing and deploying an information systems.



Real world applications

- ① Data Science ② Big tech
- ③ Health care ④ Finance
- ⑤ IoT ⑥ System development
- ⑦ Cloud.

SWE ① Introduction ② Software development life cycle [SDLC]

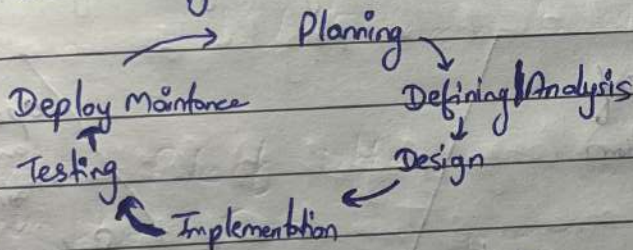
- ③ Requirement Analysis ④ Software Project Management ⑤ Software design
- ⑥ Coding and Testing ⑦ Maintenance ⑧ Quality Management and Review.

SWE :- Systematic disciplined, cost-effective techniques for software development.

Engineering Approach to develop a Software.

[Step-by-Step Approach (Goal Achievement)]

SDLC :- Software development life cycle, [structured way]. Systematic Approach



1) Customer and Service Provider :-

① Planning phase [Initial time (idea)]
 [Formal - Informal Communication]

② Defining/Analysis :- Shared with team -> Estimated Source Cost

SRS [documentation] -> Written
 Changes can be done.

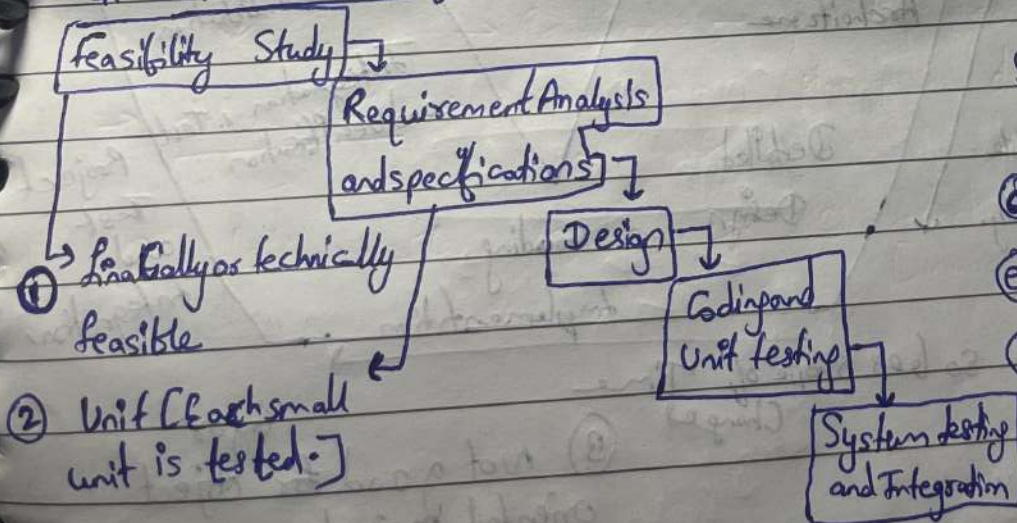
③ Designing phase ④ Coding/Testing

⑤ Testing ⑥ Deploy -> delivery.

⑦ Technical Support [Manual]

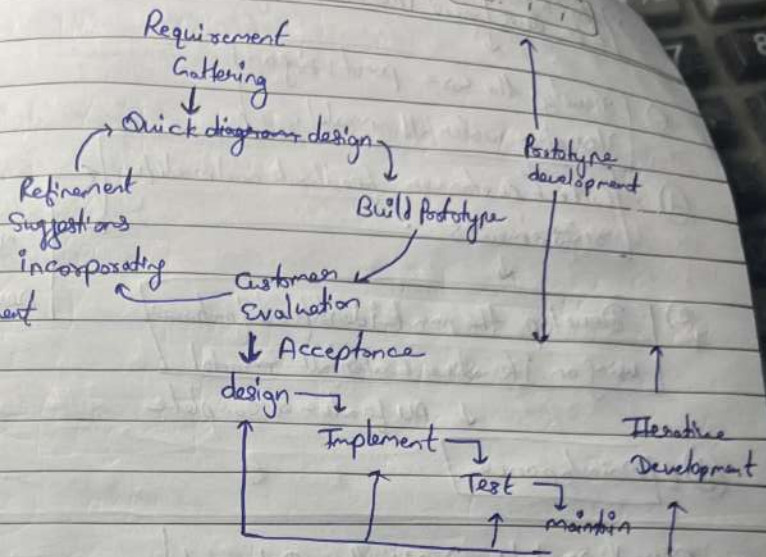
Maintenance

Classical Waterfall Model :- [1970]



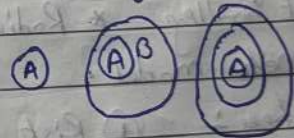
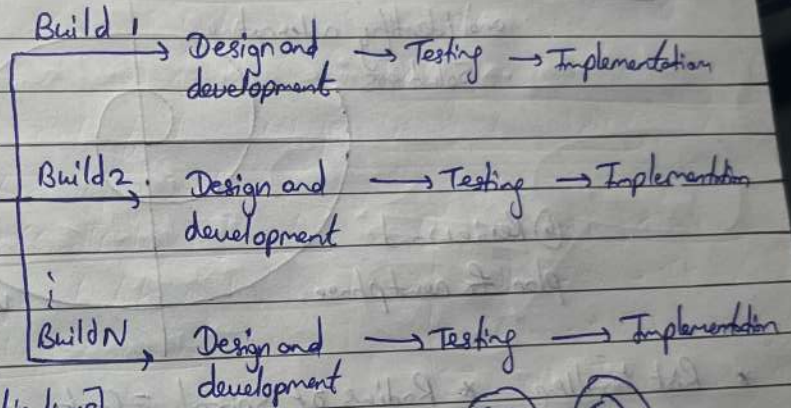
Prototyping Model :-

- ① Customer Not clear with idea
- ② Throw away model
- ③ good for technical and requirement risks
- ④ Increase in cost of development



Incremental Model :-

- * Module by Module Working
- * Customer Interaction Maximum
- * Large projects [web applications]
- * Early Release Product Demand
- * Flexible to changes.



Evolutionary Model :-

- ① Combination of Iterative and Incremental Model
- ② Incremental model first implement a few basic features and deliver to the Customers.
- ③ "Design a little, build a little, test a little, deploys a little model."

Advantage

- ① Customer requirements are clearly specified
- ② Risk Analysis is better.
- ③ It supports changing environment
- ④ Initial operating time is less / Suited for large mission-critical Projects.

disadvantages

- ① Not Suitable for Smaller projects
- ② Cost
- ③ Highly skilled resources

1. Rough Requirements Specifications
2. Identify the core and the dropouts to be developed incrementally

③ Develop the core product using iterative waterfall model.

④ Collect customer feedback and modify requirements

⑤ Develop the next identified feature using an iterative waterfall model

⑥ Maintenance

⑦ All features complete

⑧ Delivery of the next version to the customer.



⑨ Review and plan for next phase.

⑩ develop next version of product

⑪ Identify and resolve risks

⑫ Review and plan for next phase.

⑬ develop next version of product

⑭ Identify and resolve risks

⑮ Review and plan for next phase.

Part 2

① One of the most popular agile methodologies

② Scrum is a light weight, iterative and incremental framework

③ Scrum team has scrum master and product owner with cross functional roles on the daily basis.

④ Scrum master, Product Owner, Sprint, Backlog

⑤ Daily Scrum, ① Freedom and adaptation ② High quality, low risk product

⑥ Reduce the development time upto 40%

⑦ Scrum customer satisfaction is very important.

⑧ Reviewing the current sprint before moving to new one.

⑨ Disadvantages: ① More efficient for small team size.

② No changes in the sprint.

③ No changes in the sprint.

④ No changes in the sprint.

⑤ No changes in the sprint.

⑥ No changes in the sprint.

⑦ No changes in the sprint.

⑧ No changes in the sprint.

⑨ No changes in the sprint.

Classical	Iterative	Prototype	Incremental	Evolutionary
Waterfall	Waterfall	Model	Model	Model
Basic, rigid	Basic	User Requirement		
Inflexible, Not for Real Project	Problem's not clear, Well understood	Costly, No Early look on Requirements		

Today's task: ① SUE ② SDC ③ models ④ Testing - UML

Agile

① Need and Requirement Analysis: ① functional and ② Non-functional Requirements

③ Goals and objective

④ Expected outcome

⑤ Outage (unplanned)

⑥ Usability

* Design a robot for robotic Surgery : Planning → Requirement Analysis → Testing is imp.
 ↳ XYZ [Robotic Surgery] ↳ Medical Company.
 ① SWE ② SDLC and Model ③ SRS Documentation ④ UML ⑤ Testing and Types

PANKAJ
 Page No.:
 Date: / /

* Needs → Requirement/Need of Stakeholders [Who are stakeholders] according to case study surgeons, patients, hospital Administrations, Regulatory Authority.
 * Functional → Robot Control Interface, Haptic feedback, Surgical Instrumentation.

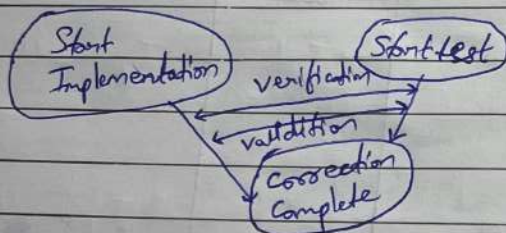
UML : Unified Modeling Language [Creating very simplified Visualisation]

⇒ Software testing : Consists certain set of activities.
 ↳ Part of Software development Process

[Bugs] Errors → Failure → Built → defects
 leads to Fault occur

① Verification and Validation V&V process ② Error → Failure → Built
 ③ Test Case ④ Testing = V+V

Test Strategy → Determine objectives and scope → Type of test → Internal & External test



Process ← Plain Environment After strategy

Structural Techniques

→ Stress
 → Execution
 → Recovery
 → Operation
 → Compliance

Functional

→ Requirement
 → Regression
 → Error-handling
 → System Testing

Software Testing Techniques

→ Unit Testing
 → Integration Testing
 → Function Testing
 → Usability Testing
 → Reliability Testing
 → Performance Testing

validation testing activities

→ Security
 → Black box and white box testing
 → Testing tools.
 → Complete system testing

* Core dump Analysis

① Static testing tools and dynamic testing tools. code inspectors

Static → Analyzers, ↑

Dynamic → Coverage analyzers.