Project Title:

Comparative Analysis of Waterfall Testing and V-Model Testing in Software Development

Project By:

Name: Sahil Ramesh Dhak
Role: QA Manual Tester Intern
Technologies Used: MS Word, Excel, Flowcharts, and Test Documentation
Duration: 3 days
Type: Manual Testing Documentation Project

Project Objective

The goal of this project is to understand and document the Waterfall Model and the V-Model (Verification and Validation Model) used in software testing. The project aims to simulate both models in a controlled testing scenario and compare their strengths, weaknesses, and real-world application.

Model 1: Waterfall Testing Model

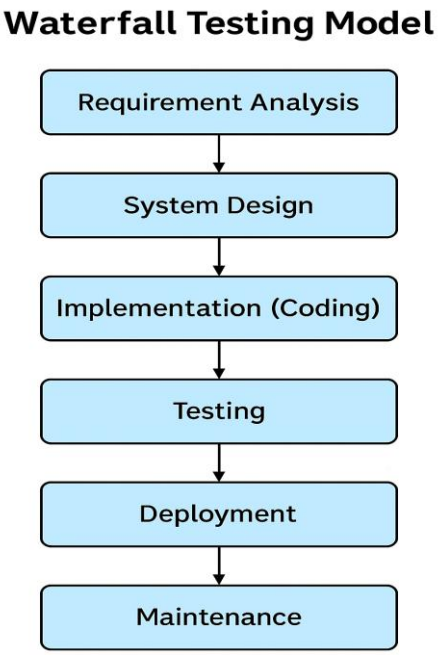
Description:

The Waterfall Model is a linear and sequential software development process where each phase must be completed before moving to the next. Testing is done after development is complete.

Phases:

1. Requirement Analysis
2. System Design
3. Implementation (Coding)
4. Testing
5. Deployment
6. Maintenance

Flowchart of waterfall testing model



Example Application:

An offline student result management system.

Sample Testing Activities:

- Test cases prepared after the coding phase.
- Test scenarios for login, result generation, and report printing.
- Test data entered manually into the system after development.

Pros:

- Simple and easy to manage.
- Clearly defined stages and deliverables.

Cons:

- No room for change once the project starts.
- Late testing phase may delay bug discovery.

Model 2: V-Model Testing (Verification and Validation Model)

Description:

The V-Model is an extension of the Waterfall Model that emphasizes testing at every development stage. Each development activity has a corresponding testing activity.

V-Model Stages:

Development Phase	Corresponding Test Phase
Requirement Analysis	Acceptance Testing
System Design	System Testing
Architecture Design	Integration Testing
Module Design	Unit Testing
Coding	—

Example Application:

An online flight booking system.

Sample Testing Activities:

- Test plans and test cases prepared in parallel with design.
- Verification of each module before integration.
- Early bug detection and correction.

Pros:

- Early testing saves time and cost.
- Better requirement traceability.
- Bugs found early in the development lifecycle.

Cons:

- More documentation needed.
- Rigid structure, not suitable for dynamic requirements.

Comparison Table:

Criteria	Waterfall Model	V-Model
Testing Time	After coding	Parallel with development
Flexibility	Low	Moderate
Bug Detection	Late	Early
Cost of Bug Fixing	High	Low
Documentation	Moderate	High
Project Size Suitability	Small to Medium Projects	Medium to Large Projects

---

Key Deliverables:

- Test Plan Document for both models.
- Sample Test Cases for login and form submission.
- Bug Report with screenshot evidence (in Excel).
- Flowcharts for both models.
- Summary report comparing results and effort.

Conclusion

This project demonstrates how the Waterfall Model and V-Model differ in their approach to testing. The Waterfall Model is suitable for small, fixed-scope projects, whereas the V-Model is better suited for complex, larger systems where early error detection is crucial. Understanding these models helps QA professionals choose the best approach based on project needs.