CSE470: Software Engineering

Waterfall Model

What is a Waterfall Model?

A sequential methodology for software project management.

There are times when the requirements for a problem are well understood—when work flows from **communication** through **deployment** in a r**easonably linear fashion**.

It is also called the **Classic Life Cycle**

When does it occur?

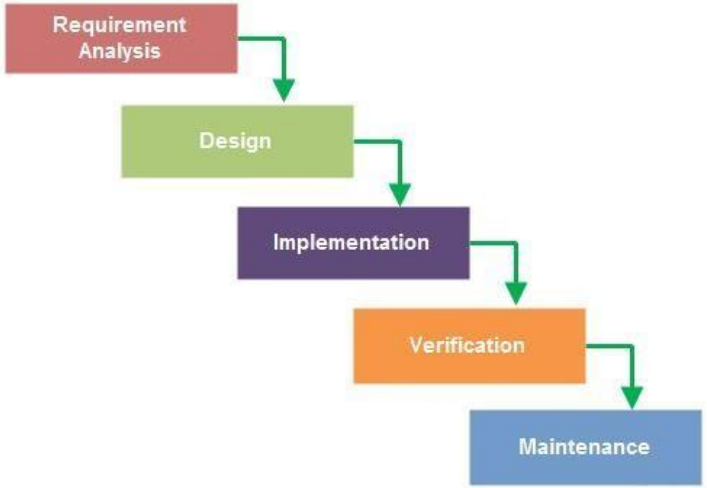
This situation is sometimes encountered when **well-defined adaptations** or enhancements to an existing system must be made

Example: An adaptation to accounting software that has been mandated because of changes to government regulations

It may also occur in a limited number of new development efforts, but only when **requirements are well defined and reasonably stable**.

Steps of the Waterfall Model

1. Requirement Collection & Analysis
2. Design
3. Coding/Implementation
4. Testing/Verification
5. Deployment
6. Maintenance



Requirement Analysis

In this phase, all the requirements are gathered from the client. Basically collecting information about what the client wants their software to do. Then our job is to:

1. Address the problem
2. Identify feasible and infeasible requirements
3. Identify how the software helps our client attain their requirements

Design

In this step, the requirements gathered in the previous step is converted into different format (Diagrams) that can be coded in a programming language.

It usually includes a high level and detailed design and architecture of the proposed software solution.

Development/Coding

After the design is fixed:

1. The design is converted into blocks
2. Those blocks are converted into code
3. Step by step converting the design to a workable software

Testing

Testing phase starts after the development phase is completed.

1. The finished product is tested based on the user requirements
2. In case of any problem, it is fixed in code (Unit Testing)
3. It is usually done in several phases

Deployment

- The software is deployed in the production environment
- Test to check if the system is performing as expected

Maintenance

- If anything goes wrong in the deployment process, it is fixed here
- Keeping the software working over time

Advantages of Waterfall Model

- Easy to Understand and Implement
- Each Step is properly defined
- Steps are performed one at a time
- Clear goals of each step

Disadvantages of Waterfall Model

- No feedback path
- Difficult to incorporate requirement changes
- No step overlapping
- Not flexible
- Limited Stakeholder Involvement
- Late error detection
- Working product after the last step of the model

When to choose Waterfall Model?

- When Requirements are well known
- Little to no changes expected
- Small to Medium Projects
- Minimum Client Involvement needed
- Resources are available and trained

The Waterfall approach involves less user interaction in the product development process. The product can only be shown to end user when it is ready.

Case Study

Case 1: One of your uncle requested you to develop an accounting calculator for his local shop.

Case 2: Your start-up company wants to develop an accounting calculator for super shops.

Question: Determine if you can implement both cases in Waterfall Model

Case 1

One of your uncle requested you to develop an accounting calculator for his local shop.

Key Points to note:

- Local shop (Small scale)
- Already existing business (Fixed Requirement)

Verdict: Waterfall can be Implemented

Case 2

Your company wants to develop an accounting calculator for start-up super shops.

Key Points to note:

- Super shop (Big scale)
- Start up (Requirements are not fixed)

Verdict: Cannot implement Waterfall Model

Reference

1. Roger S Pressman, Roger Pressman,"Software Engineering: A Practitioner's Approach", McGraw-Hill, 7th edition, 2010.
2. https://www.geeksforgeeks.org/waterfall-model/