Software Development Life Cycle
SDLC

Agenda

1. Why we need SDLC?
2. Planning
3. Requirement Analysis
4. Design
5. Implementation (Coding)
6. Testing
7. Deployment and Maintenance
8. SDLC models
9. Waterfall
10. V-Model

How Everything starts?

1. A customer has an idea and money

2. The customer contacts software companies and receives different offers

3. The customer chooses one of the software companies

4. The customer meets software company representative (Product owner) and signs a contract

Why we need **SDLC**?

Convert an idea to working software

Release the software to the clients and maintain it

Effective SDLC = Effective company

Planning

✓ The customer and the product owner or project manager have a couple of meetings

✓ In these meetings can participate domain experts, sales people, senior members

✓ They define the application

Planning

✓ Raw splitting of the application on different components

✓ Discussing the business idea and the need of the customer

✓ Risks associated with the project are discussed

Requirement Analysis

✓ The whole team meets Product Owner and Customer

✓ More detailed discussion on each product component. Splitting components on smaller chunks with detailed explanation

✓ Defining actual requirements

✓ Project manager/ Product owner defines the requirements in the Project Management System

✓ The written requirements are approved by the customer

Design

✓ Software Requirement Specification document is a reference for the product design

✓ Product Architects work on a product architecture

✓ Usually more than one design for product architecture is created

✓ Risk assessment based on budget, time constraints, modularity, frameworks, programming languages, DB, servers, browser support, mobile etc.

✓All is documented in Design Document Specification (DDS)

SKILLO

Implementation/ coding

- ✓ The actual development starts
- ✓ Code is generated based on the DDS document
- ✓ Coding standards should be followed
- ✓ Work in pair
- ✓ Implementation of UI design
- ✓ QA engineers start building actual test cases
- ✓ QA engineers contribute to usability
- ✓ Finding cheap bugs
- ✓ Code reviews and Unit tests

Testing

✓Coding of the app has finished

✓ App is deployed on a testing environment

✓ Build version is verified

✓ Finding bugs and track them in PMS, BTS

✓ Retesting fixed bugs and new functionality

✓ Regression testing

✓ Acceptance testing by the customer

Deployment and Maintenance

✓ Released to the market/customers

✓ Deployment could be on phases – Alpha, Beta, Production

✓ Based on the feedback during the different phases corrective actions are taken

✓ Releases can have a downtime

✓ Releases can be major, minor

✓ In case of production bugs Patches/Hotfixes or Rollbacks are applied

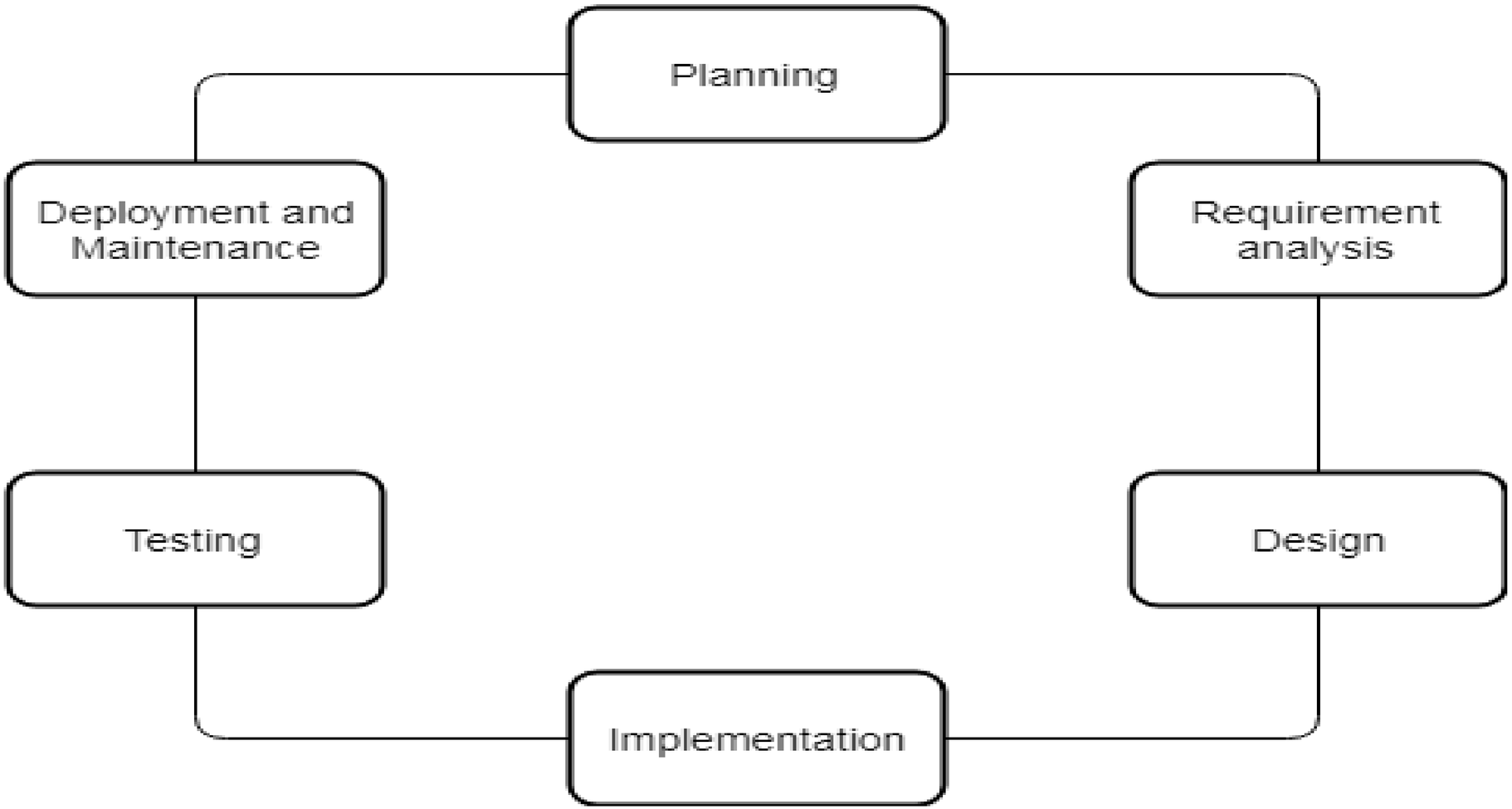
Deployment and Maintenance

✓ Code branches

✓ Our product can have multiple versions on production

✓ We should have separate branches in our source code for each production version

SDLC schema



SKILLO

How Projects Really Work (version 1.5)

Create your own cartoon at www.projectcartoon.com

How the customer explained it

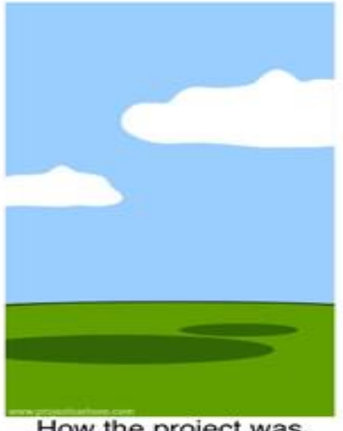
How the project leader understood it

How the analyst designed it

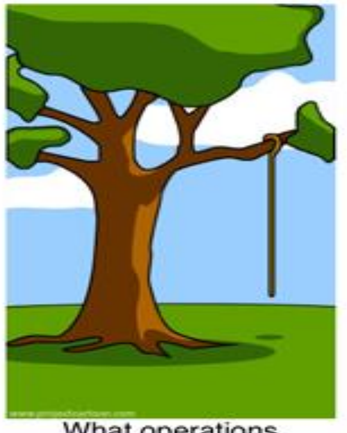
How the programmer wrote it

What the beta testers received

How the business consultant described it

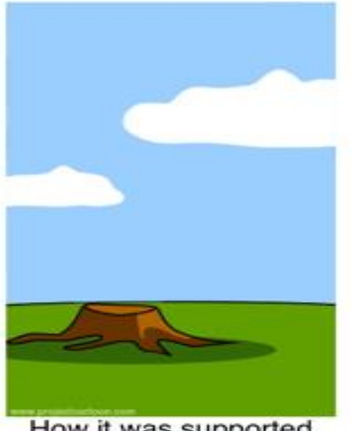


How the project was documented



What operations installed

How the customer was billed



How it was supported

What marketing advertised

What the customer really needed

SKILLO

SDLC models

1. Waterfall model

2. V model

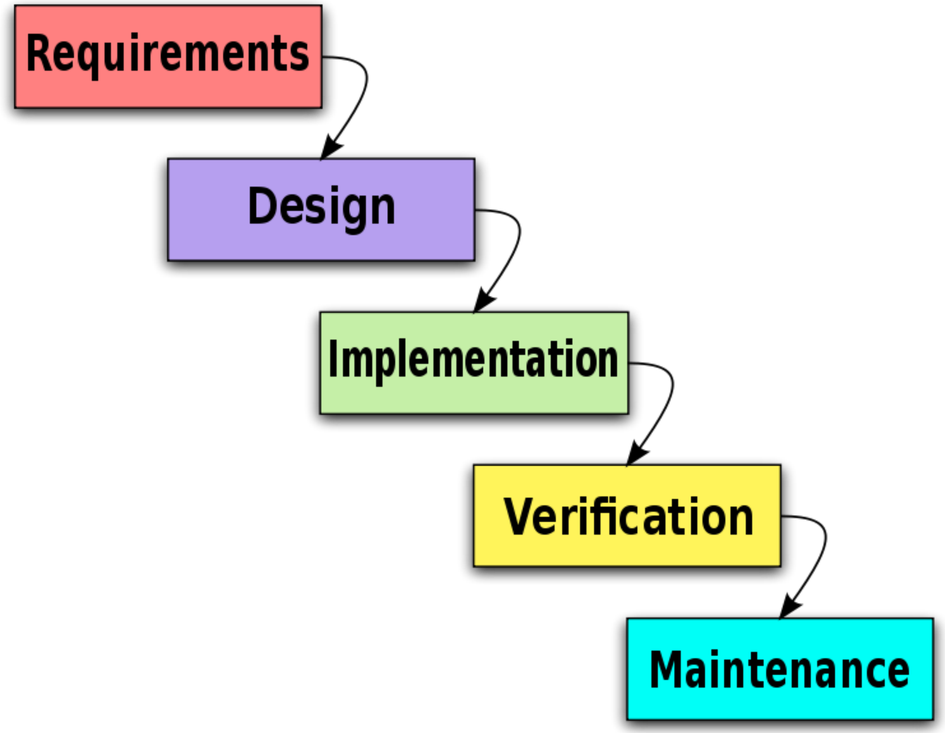
3. RAD model

4. Agile model

5. Iterative model

SKILLO

Waterfall


Requirements
Design
Implementation
Verification
Maintenance

Waterfall

✓ Each phase starts once the previous is completed

✓ Old SDLC model – cannot meet current market needs

✓ Cost of fixing issues is high

✓ Increased time to market

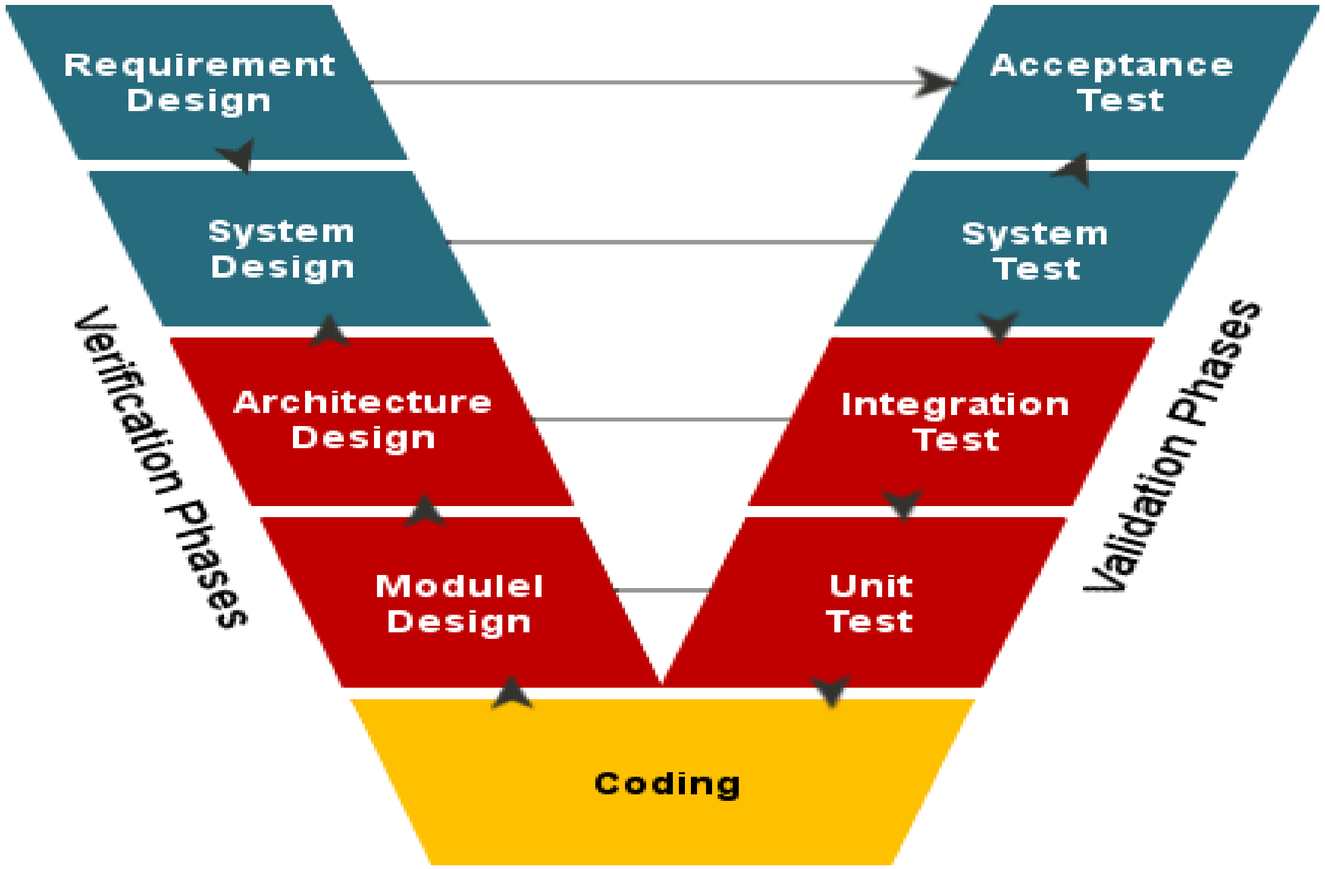
✓ Simple to understand

✓ Planning is hard and estimations are not correct

✓ Late feedback from the client

✓ High risk the project to fail before go live

V-model



SKILLO

V-model

“V” stands for **Verification** and **Validation**

Association between each phase of testing and development

Testing is Validation phase

Development is Verification phase

Verification vs Validation

Verification - Are we building the product *right*?

Validation - Are we building the *right* product?

Time for **Questions**

SKILLO