An Assignment of

Diploma in Software Testing and Automation

Module 1 - Fundamental

Submitted to

Chinmayee Shukla Ma'am

Tops Technologies Ahmedabad – S G Road

By

Piyush Dabhi

May 2024

1. What is SDLC?

Software Development Life Cycle (SDLC) is a methodology that defines the entire procedure of Software Development step by step approach to provide quality of software within the time and budget.

SDLC is a structure imposed on the development of a software product that defines the process for planning, implementation, testing, deployment, and on-going maintenance and support.

SDLC is a methodology or step-by-step approach to produce software with high quality, lowest cost in the shortest possible time by defining the phases like Planning, Analysis & Design, Coding & Implementation, and Testing & Maintenance.

2. What is Software Testing?

- Testing:
  Testing is a process which is used to identify the correctness, completeness & quality of the developed software. It is also can be stated as the process of Validating and Verifying the software program or application.

  Completeness + Correctness + Quality = Testing

- **Manual Testing:**
  It is a software testing process in which test cases are executed manually without using any automated tool. In manual software testing, a tester tests the software by following a set of predefined test cases.

- **Automation Testing:**

  Automation Testing is a software testing technique that performs using special automated testing software tools like Selenium and etc….

3. What is agile methodology?

Agile model is a combination of iterative and incremental process models with focus on customer satisfaction by rapid delivery of working software product. Agile Methods

break the product into small incremental builds. It is a methodology that helps developers test their code continuously and rapidly.

4. What is SRS?

Software Requirement Specification (SRS) is a complete specification and description of requirements of the software that need to be fulfilled for the successful development of the software system. There are three types of requirements, Functional requirement, Non-Functional Requirement and Customer Requirement

5. What is OOPS?

Object-Oriented Programming is a methodology or paradigm to design a program using classes and objects. It is used to structure the software program into simple reusable code.

6. Write Basic concept of OOPS.

Object-Oriented Programming is a methodology or paradigm to design a program using classes and objects. It is used to structure the software program into simple reusable code. Here are some basic concepts of OOPS:

1. **Object:**

   An object is the basic unit of OOP which is accessed by its properties called data member & member function. It creates the memory for the class.

2. **Class:**

   Class is logical representation of data. Class is a blueprint or a template to describe the properties and behaviour of the objects.

3. **Encapsulation:**

   Encapsulation refers to the bundling of data and methods that operate on the data within a single unit or class. wrapping up of data (data members) & methods (member functions) into single unit;

4. **Inheritance:**

Inheritance is a mechanism where a new class (subclass) can inherit properties and behaviours from an existing class (superclass).

5. **Polymorphism:**

Polymorphism is the ability to present the same interface for differing underlying forms (data types). With polymorphism, each of these classes will have different underlying data. Precisely, Poly means 'many' and morphism means 'forms'.

6. **Abstraction:**

Abstraction is the representation of the essential features of an object. Also called data hiding.

7. What is Object?

Object is a run-time entity. It is an instance of the class. An object can represent a person, place or any other item. An object can operate on both data members and member functions

8. What is Class?

Class is a user-defined data type which defines its properties and its functions. Class is the only logical representation of the data. For example, Human being is a class. The body parts of a human being are its properties, and the actions performed by the body parts are known as functions.

9. What is Encapsulation?

Encapsulation is the process of combining data and functions into a single unit called class. In Encapsulation, the data is not accessed directly; it is accessed through the functions present inside the class.
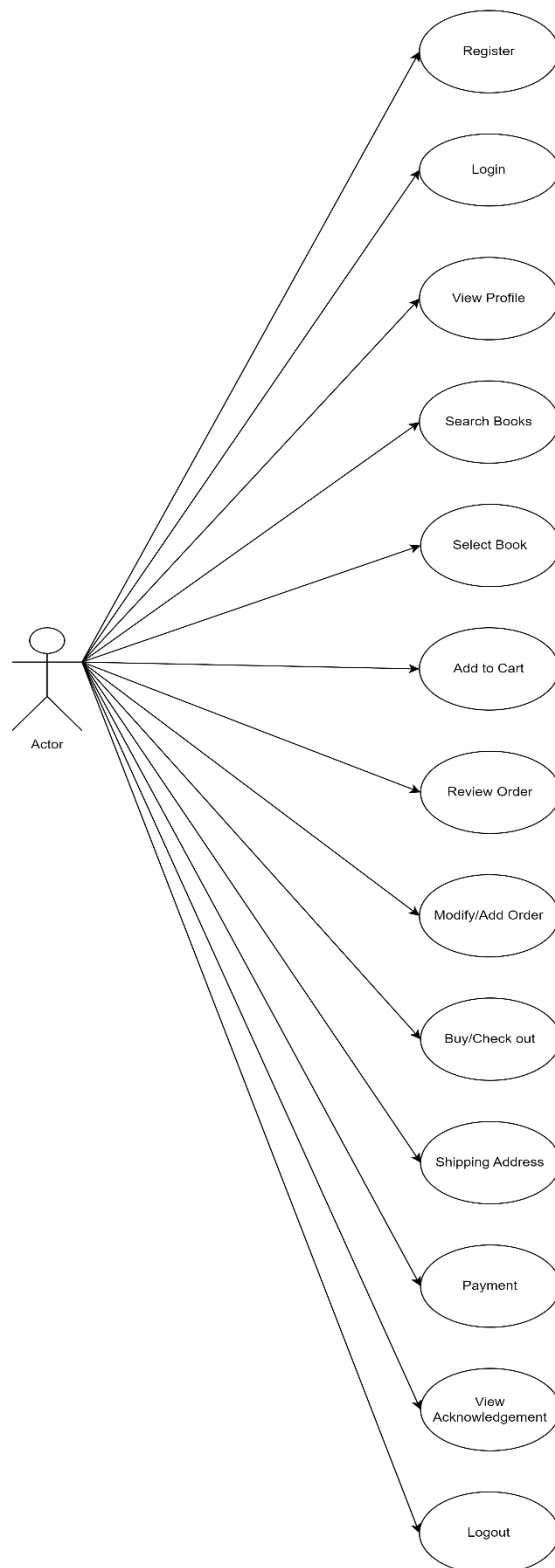
10. What is Inheritance?

Inheritance is a process in which one object acquires all the properties and behaviours of its parent object automatically. In such a way, you can reuse, extend or modify the attributes and behaviours which are defined in other classes. There are 5 types of Inheritance which is as bellow,

1. Single Inheritance

2) Multilevel Inheritance

3) Multiple Inheritance

4) Hierarchical Inheritance
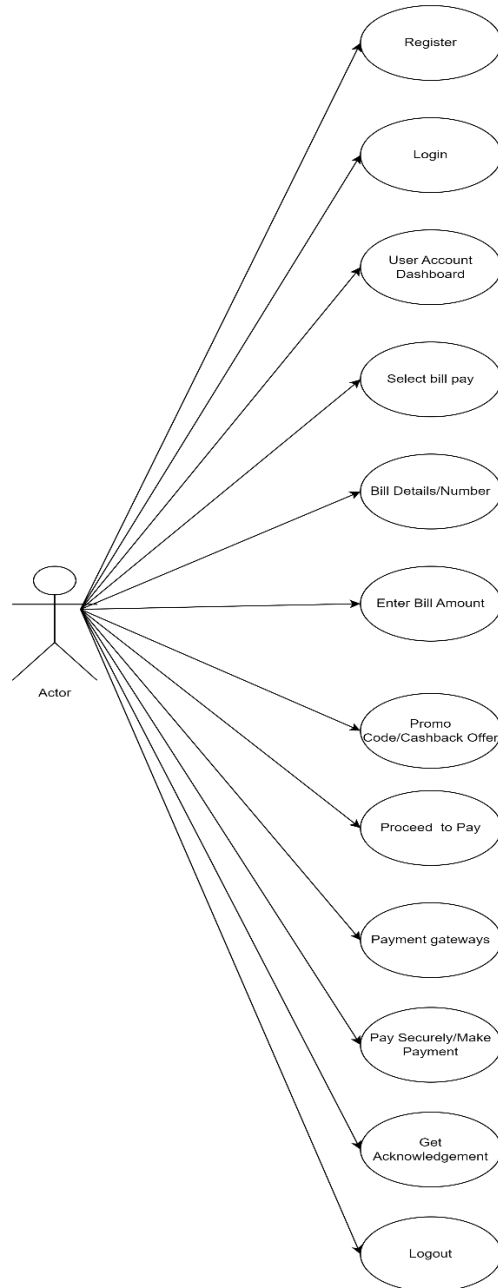
5) Hybrid Inheritance

11. What is Polymorphism?

Polymorphism is the ability to present the same interface for differing underlying forms (data types). With polymorphism, each of these classes will have different underlying data. Precisely, Poly means 'many' and morphism means 'forms'. There are two types of Polymorphism Compile time Polymorphism and Run Time Polymorphism.

Use Case Diagram for Online Book Shopping

Paytm Online Bill Payment System

Register

Login

User Account Dashboard

Select bill pay

Bill Details/Number

Enter Bill Amount

Promo Code/Cashback Offer

Proceed to Pay

Payment gateways

Pay Securely/Make Payment

Get Acknowledgement

Logout

Actor

By Piyush Dabhi

14. Write SDLC Phases with Basic Introduction.

SDLC is a methodology or step-by-step approach to produce software with high quality, lowest cost in the shortest possible time by defining the phases like Planning, Analysis & Design, Coding & Implementation, and Testing & Maintenance.

It contains 6 Stages of Software Development Life Cycle

1. **Planning/Requirements/Gathering Data:**

It is performed by the senior member of the team with inputs from the customer. There are three types of problem arise

   a. Lack of clarity

   b. Requirement Confusion – Functional and Non-Functional

   c. Requirement Amalgamation – Several different requirements may be expressed togethers

2. **Analysis/Defining:**

Define the product requirement and get them approved from the customer. This phase represents 'How' and 'What' phase.

3. **Design:**

Based on the requirements specify in the SRS (Software Requirement Specification). Usually more than one design approach for the product architecture.

4. **Implementation/Building/Developing:**

Developer use a specific programming code which follows design given by the designing team. Hardware and Software requirements.

5. **Testing:**

This stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested until the product reaches the quality standards.
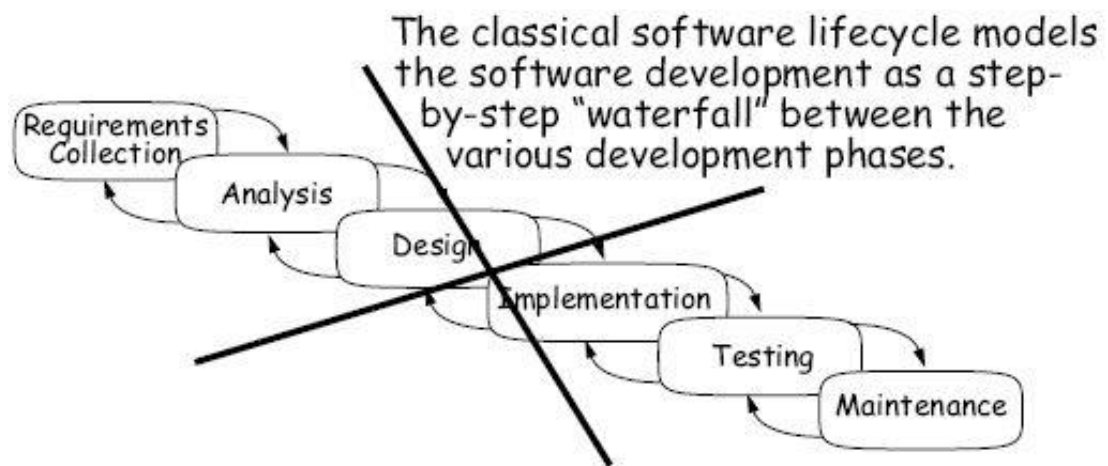
6. **Maintenance/Deployment:**

Once the product is tested and ready to be deploy it released in the appropriate market. There are three types of maintenances like corrective, adaptive and perfective maintenance.

15. Explain Phases of the Waterfall model

The Waterfall model is the earliest SDLC approach that was used for software development. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin.

It is used when product definition is stable and static, requirements are very well documented, clear and fixed. There are no ambiguous requirements and ample resources with required expertise.

The classical software lifecycle models the software development as a step-by-step "waterfall" between the various development phases.

The sequential phases in Waterfall model are –

1. **Requirement Gathering:**

   All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.

2. **Analysis/Defining:**

   Define the product requirement and get them approved from the customer

3. **Design:**

   Based on the requirements specify in the SRS (Software Requirement Specification). Usually more than one design approach for the product architecture.

4. **Implementation/Building/Developing:**

Developer use a specific programming code given by the designing team. Hardware and Software requirements.
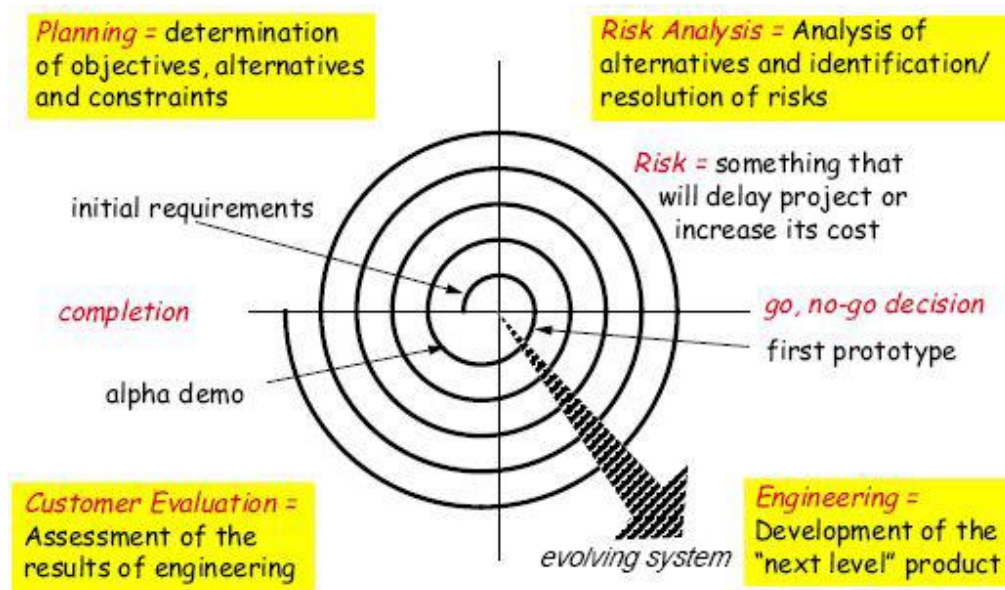
5. **Testing:**
   This stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested until the product reaches the quality standards.

6. **Maintenance/Deployment:**
   Once the product is tested and ready to be deploy it released in the appropriate market. There are three types of maintenances like corrective, adaptive and perfective maintenance.

16. Write phases of spiral model

This model is a combination of the waterfall model and the iterative model. It is very widely used in the software industry. It involves minimum risk for the customer as well as the development firms."

It defines phases like,

- **Planning**: The first phase of the Spiral Model is the planning phase, where the scope of the project is determined and a plan is created for the next iteration of the spiral.

- **Risk Analysis:** In the risk analysis phase, the risks associated with the project are identified and evaluated.

- **Engineering**: In the engineering phase, the software is developed based on the requirements gathered in the previous iteration.

- **Evaluation**: In the evaluation phase, the software is evaluated to determine if it meets the customer's requirements and if it is of high quality.

17. Write agile manifesto principles

1. Our highest priority is to satisfy the customer through the early and continuous delivery of valuable software.

2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

4. Business people and developers must work together daily throughout the project.

5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

7. Working software is the primary measure of progress.

8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

9. Continuous attention to technical excellence and good design enhances agility.

10. Simplicity–the art of maximizing the amount of work not done–is essential.

11. The best architectures, requirements, and designs emerge from self-organizing teams.

12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly.

18. Explain working methodology of agile model and also write pros and cons.

Agile model is a combination of iterative and incremental process models with focus on customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds
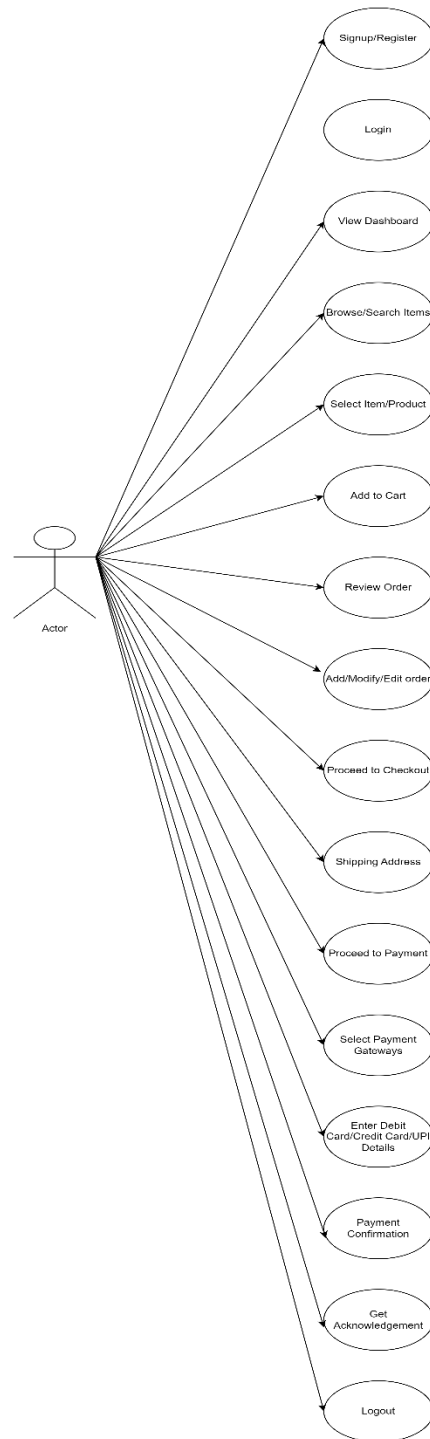
Pros	Cons
• Very realistic approach	• More risk of sustainability, maintainability and extensibility
• Rapid Delivery	• Depends heavily on customer interactions.
• Functionality can be developed rapidly	• Minimum documentation generated.
• Gives flexibility to developers	• Not useful for small projects.
• Resource requirements are minimum.	• Very high individual dependency.

Online Shopping Product Using COD

By Piyush Dabhi

Online Shopping Product Using Payment Gateway

By Piyush Dabhi