

Manual Assignment 1

1. What is SDLC

SDLC stands for Software Development Life Cycle

SDLC is a step-by-step approach to produce a high quality software with 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?

Software testing is a process which is used to assess the correctness, completeness and quality of the developed software

Software Testing is a method to check the functionality of the actual software and whether it matches expected requirements and to ensure that software product is defect free

There are two types of software testing:

- Manual software testing
- Automatic software testing

3. What is agile methodology?

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

Methodology:

Agile Methods break the product into small incremental builds, these builds are provided in iterations
Each iteration typically lasts from about one to three weeks, every iteration involves cross functional teams working simultaneously on various areas like planning, requirements analysis, design, coding, unit testing, and acceptance testing

At the end of the iteration a working product is displayed to the customer and important stakeholders
Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In agile the tasks are divided to time boxes (small time frames) to deliver specific features for a release

Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer

Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability

4. What is SRS

SRS stands for Software Requirements Specification
The SRS is a document that describes what the software will do and how it will be expected to perform

SRS is a comprehensive description of the intended purpose and environment for software under development

5. What is oops

Object Oriented Programming is viewed as a collection of objects. It is used to structure the software program into simple reusable code

It is referred as Functional testing or Black Box Testing

6. Write Basic Concepts of oops

Basic concepts of OOPS are:

- Object
- Class
- Encapsulation
- Inheritance
- Abstraction
- Polymorphism

7. What is 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

8. What is Class

Class is a collection of a data member (variables) and member function with its behavior

Class is a blueprint or a template to describe the properties and behavior of the objects

9. What is Encapsulation

Encapsulation is wrapping up of data and functions into a single unit. It hide/include/private access of data member & member function

It also restricts direct access to some of an object's components, which is done through access modifiers e.g., private, protected, public

10. What is Inheritance

One Super/Base class inherits the properties of another Sub/Derived class

Types of Inheritance:

- Single Inheritance
- Multiple Inheritance
- Multilevel Inheritance
- Hierarchical Inheritance
- Hybrid Inheritance

11. What is Polymorphism

Polymorphism enables objects to be treated as instances of their parent class, allowing methods to do different things based on the object it is acting upon

OR

An ability to take one name having many different form
This can be achieved through methods:

- Operator Overloading (within the same class):
(Compile time Polymorphism)

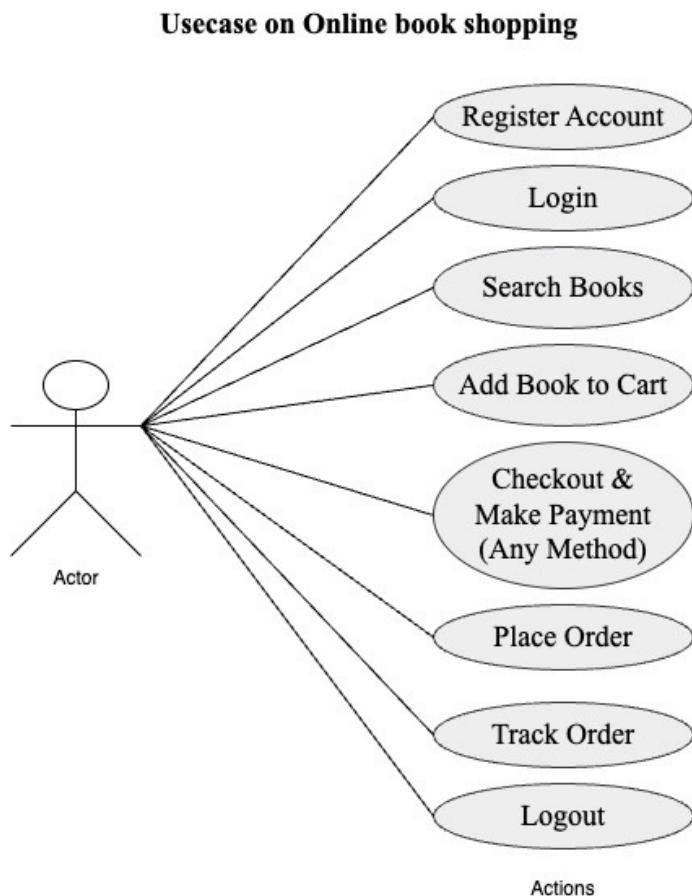
Method name should be same in single class but its behavior (Arguments & Data type) is different

- Operator Overriding (in subclasses): (Run time Polymorphism)

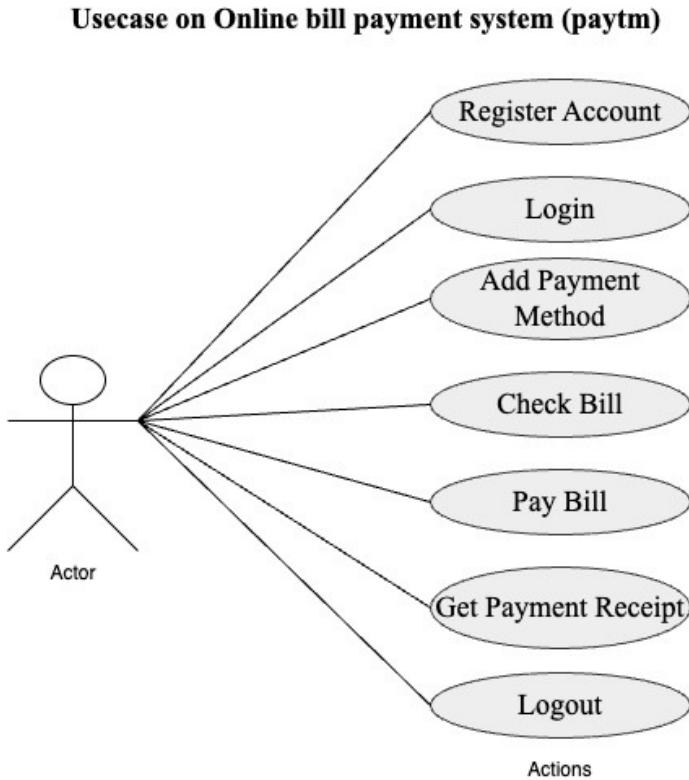
Method should be same in super class and sub class but its behavior is different

Method overriding (in subclasses) and method overloading (within the same class)

12. Draw Usecase on Online book shopping



13. Draw Usecase on Online bill payment system (paytm)



14. Write SDLC phases with basic introduction

SDLC phases are shown below:

- Requirements Collection/Gathering (Planning)

It is a very beginning phase where gathering of business requirements done from client or stakeholders

- Analysis

In this phase the analysis of the information gathered from the client is done taking into consideration of budget, timeline and quality of the product or software

- Design

Design is about building the detailed framework OR
Outline the architecture and design specifications based
on the requirements

- Implementation (Coding)

Write the actual code for the software outside of the
detailed framework

Performed by front end developers and back end
developers outside the paperwork

- Testing

Evaluate the software to ensure it meets the specified
requirements and is free of defects

Testing process is performed by testers manually or
automation process

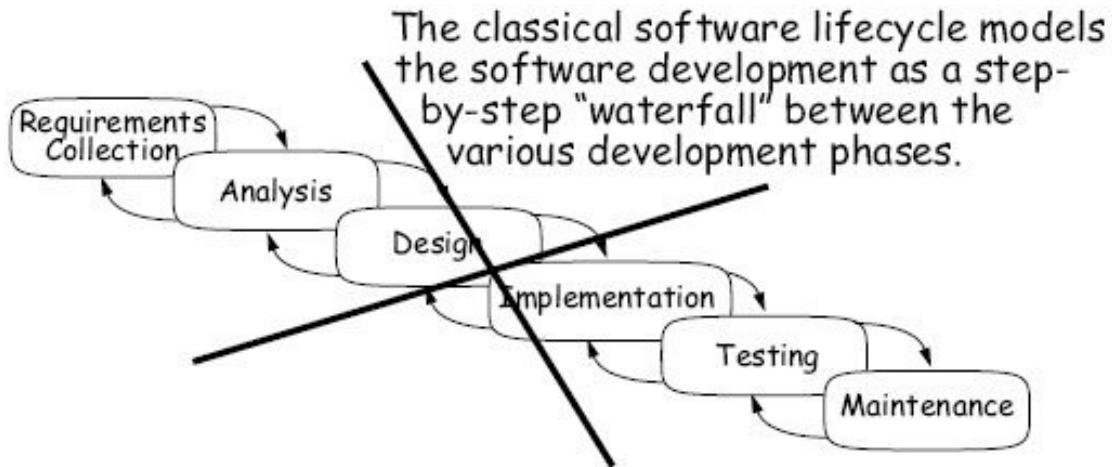
- Deployment & Maintenance

Release the software to users is Deployment phase

And at last, address any issues that arise after
deployment and make necessary updates

To increase sales after service, maintenance process is
the important phase of the SDLC

15. Explain Phases of the waterfall model



The waterfall model is a classical software lifecycle that models the software development as a step-by-step “waterfall” between the various development phases

- Requirements Collection

It is a very beginning phase where gathering of business requirements done from client or stakeholders

- Analysis

In this phase the analysis of the information gathered from the client is done taking into consideration of budget, timeline and quality of the product or software

- Design

Design is about building the detailed framework OR Outline the architecture and design specifications based on the requirements

- Implementation

Write the actual code for the software outside of the detailed framework

Performed by front end developers and back end developers outside the paperworks

- Testing

Evaluate the software to ensure it meets the specified requirements and is free of defects

Testing process is performed by testers manually or automation process

- Maintenance

Maintenance is the process of changing a system after it has been deployed

Release the software to users is Deployment phase

And at last, address any issues that arise after deployment and make necessary updates

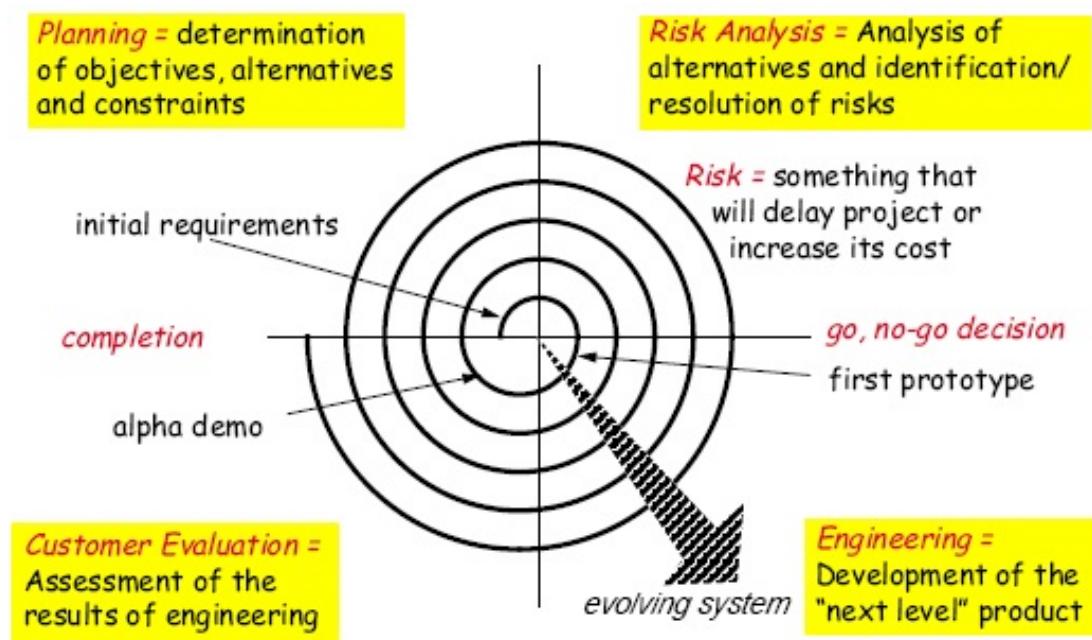
To increase sales after service, maintenance process is the important phase of the SDLC

- Corrective Maintenance: Identifying and repairing defects
- Adaptive Maintenance: Adapting the existing solution to the new platforms
- Perfective Maintenance: Implementing the new requirements. In a spiral lifecycle, everything after the delivery and deployment of the first prototype can be considered Maintenance

16. Write phases of spiral model

Spiral Model is very widely used in the software industry as it is in sync with the natural development process of any product learning with maturity also involves minimum risk for the customer as well as the development firms

Phases of spiral model are shown below:



1. Planning

Gathering of the information and planning and determination of objectives, alternatives and constraints according to client's requirements

2. Risk analysis

This phase makes spiral method different from previous models

Here analysis of alternatives and identification, resolution of risk taken under consideration

3. Engineering

In this phase the coding and development of the software is done

Prototype is made for further testing and evaluation

4. Customer evaluation

Assessment of the results of the engineered/developed prototype is done for customer perspective

17. Write agile manifesto principles

There are 4 Values of Agile Manifesto

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

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

Methodology:

Agile Methods break the product into small incremental builds, these builds are provided in iterations

Each iteration typically lasts from about one to three weeks, every iteration involves cross functional teams working simultaneously on various areas like planning,

requirements analysis, design, coding, unit testing, and acceptance testing

At the end of the iteration a working product is displayed to the customer and important stakeholders

Pros:

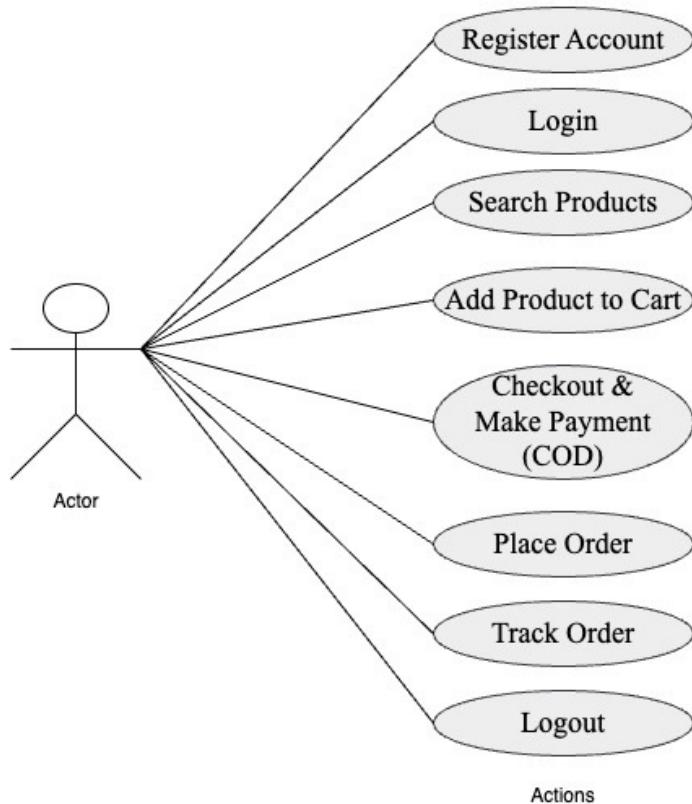
- Very realistic approach
- Rapid delivery
- Functionality can be developed rapidly
- Resource requirements are minimum
- Little or no planning required
- Promotes teamwork and cross training
- Suitable for fixed or changing requirements
- Gives flexibility to developers

Cons:

- More risk of sustainability, maintainability and extensibility
- Depends heavily on customer interactions
- Very high individual dependency
- Minimum documentation generated
- Not useful for small projects
- Not suitable for handling complex dependencies

19. Draw Usecase on Online shopping product using COD

UseCase on Online shopping product using COD



20. Draw Usecase on Online shopping product using payment gateway

Use case on Online shopping product using payment gateway

