Module-1 (Fundamental)

1) What is SDLC?

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

2) 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.

3) What is SRS?

A software requirements specification (SRS) is a complete description of the behaviour of the system
To be developed.

4) What is oops?

OOPS identifying objects and assigning responsibilities to these objects.

5) Write basic concepts of oops?

a. Object
b. Class
c. Encapsulation
d. Inheritance
e. Polymorphism
    i. Overriding
    ii. Overloading
f. Abstraction

6) What is object?

Everything in the word is an object.
Examples of objects are,

- A flower, a tree, an animal
- A student, a professor
- A desk, a chair, a classroom, a building
- A university, a city, a country
- The world, the universe
- A subject such as CS, IS, Math, History

7) What is class?

Class is define a blueprint for an object.

8) What is encapsulation?

It is a process of wrapping up of data into a single unit.

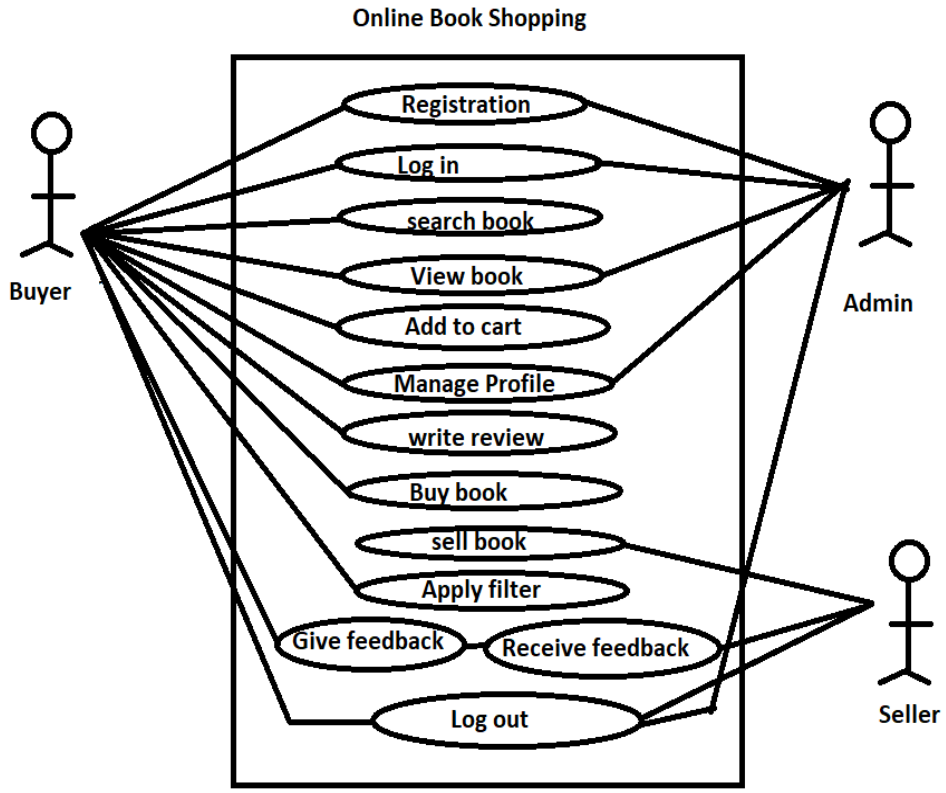
9) What is inheritance?

It is a process of hiding irrelevant data from the user.

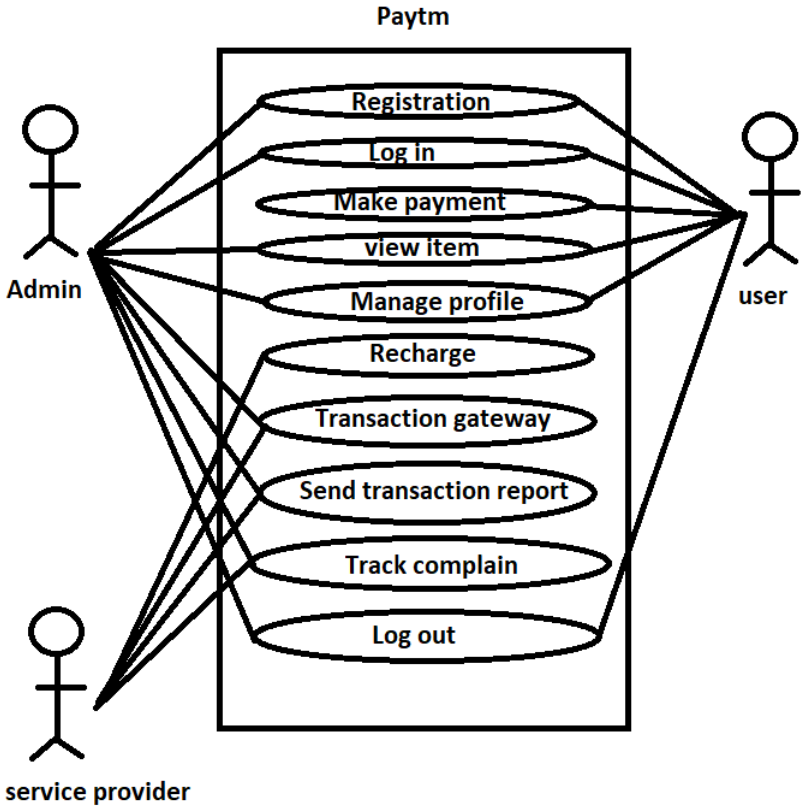
10) What is polymorphism?

Polymorphism means “having many forms”.

11) Draw Use case on online book shopping.



12) Draw Use case on online bill payment system (paytm)



13) Write SDLC phases with basic introduction.

Requirements Collection/Gathering: Establish Customer Needs

Analysis: Specify the requirements- “What”

Design: Specify a Solution – “Why”

Implementation: Construct a Solution in Software

Testing: Validate the software product.

Maintenance: Repair defects and adapt the solution to the new requirements

14) Explain Phases of the waterfall model

Requirement Gathering

- Features
- Usage scenarios
- Although requirements may be documented in written form, they may be incomplete,unambiguous, or even incorrect.

- Requirements will Change!
- User and business needs change during the project
- Validation is needed throughout the software lifecycle, not only when the “final system” isdelivered.
- Understand Functional and Non-Functional.
- Plan for change.

Analysis Phase

- This analysis represents the “**what”** phase.
- This phase defines the problem that the customer is trying to solve.
- Ideally, this document states in a clear and precise fashion what is to be built.
- The design may include the usage of existing components.
- The requirement documentaries to capture the requirements from the customer's perspectiveby defining goals.

Design

- Design Architecture Document.
- Implementation Plan
- Performance Analysis
- The Design team can now expand upon the information established in the requirementdocument.
- Test Plan

Implementation Phase

- In the implementation phase, the team builds the components.
- Critical Error Removal
- The implementation phase deals with issues of quality, performance, baselines, libraries, and debugging.
- Implementation - code

Testing Phase

- Simply stated, quality is very important. Many companies have not learned that quality isimportant and deliver more claimed functionality but at a lower quality level.
- It is much easier to explain to a customer why there is a missing feature than to explain to acustomer why the product lacks quality.
- A customer satisfied with the quality of a product will remain. loyal and wait for newfunctionality in the next version.
- Regression Testing
- Internal Testing
- Unit Testing
- Application Testing
- Stress Testing

Maintenance Phase

- The maintenance phase is the phase which comesafter deployment of the software into the field.
- Configuration and version management.
- Updating all analysis, design and user documentation.
- Repeatable, automated tests enable evolution and refactoring.

15) Write phases of spiral model

- Planning
- Risk analysis
- Engineering= Implementation/Testing
- Customer evaluation.

16) Write agile manifesto principles

Individuals and interactions - in agile development, self-organization and motivation areimportant, as are interactions like co-location and pair programming.

Working software - Demo working software is considered the best means of communication
With the customer to understand their requirement, instead of just depending on documentation.

Customer collaboration - As the requirements cannot be gathered completely in the beginning of the project due to various factors, continuous customer interaction is very important to getproper product requirements.

Responding to change - agile development is focused on quick responses to change andcontinuous development.

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

Working Methodology

- Agile SDLC model is a combination of iterative and incremental process models withfocus on process adaptability and customer satisfaction by rapid delivery of working software product
- Each iteration typically lasts from about one to three weeks.
- Agile Methods break the product into small incremental builds.
- At the end of the iteration a working product is displayed to the customer and importantstakeholders.

Advantages

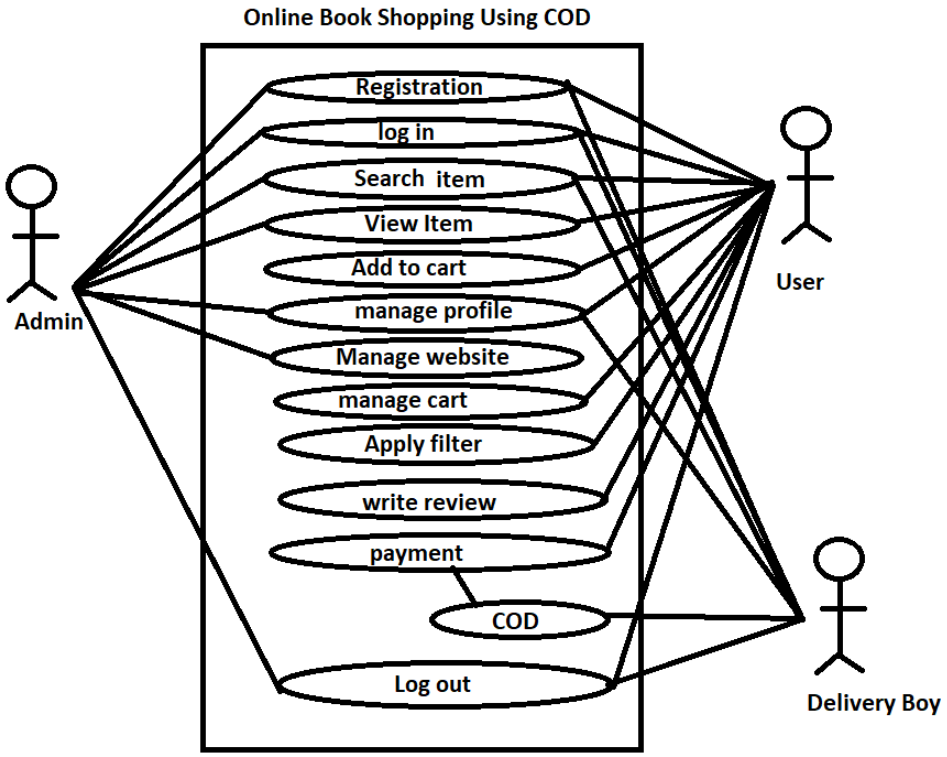
- Is a very realistic approach to software development.
- Resource requirements are minimum.
- Suitable for fixed or changing requirements.
- Good model for environments that change steadily.
- Little or no planning required.
- Easy to manage
- Gives flexibility to developers

Disadvantages

- Depends heavily on customer interaction, so if customer is not clear, team can be driven inthe wrong direction.

- Not suitable for handling complex dependencies.
- Strict delivery management dictates the scope, functionality to be delivered, and adjustmentsto meet the deadlines.
- There is very high individual dependency, since there is minimum documentation generated.
- Transfer of technology to new team members may be quite challenging due to lack of documentation.

18) Draw use case on Online shopping product using COD.



19) Draw use case on Online shopping product using payment gateway.