1. What is SDLC?

➔ SDLC stands for Software Development Life Cycle.

➔ It is a process that is used to design, analysis, planning,implementation and testing an application or software.

2. What is software testing?

➔ Testing is a process that is used to identify error, bug, defect, missing requirements and also verify completeness,correctness and quality of developed software.

3. What is agile methodology?

➔ The Agile SDLC model is a combination of iterative and incremental process models.

➔ Agile Model breaks a product into small parts or pieces that can be incrementally developed.

4. What is SRS?

➔ SRS is known as a Software Requirement Specification.

➔ SRS is a complete specification and description of requirements of software that needs to be fulfilled for successful development of a software system.
➔ These requirements can be functional as well as non-functional depending upon the type of requirement.

5. What is oops?

➔ OOP stands for Object-Oriented Programming.

➔ OOP is a methodology to design a program using classes and objects.

➔ Object-oriented programming is a computer programming model that organises software design around data, or objects, rather than functions and logic.

6. Write Basic Concepts of oops.

➔ OOP simplifies software development and maintenance by providing some concepts:

1) Object
2) Class
3) Inheritance
4) Polymorphism
    a) Overriding

b) Overloading

5) Abstraction
6) Encapsulation

7. What is an object?

➔ An object is a collection of data members and associated member functions.

➔ An object may represent a person, place or a table of data.

➔ Each object is identified by a unique name. Each object must be a member of a particular class.

➔ Example: Car, bus, bike are the objects of class vehicles.

8. What is class?

➔ A class is a collection of objects that have identical properties, common behaviour and shared relationship.

➔ A class is a blueprint that defines the variables and the methods common to all objects of a certain kind.

9. What is encapsulation?

➔ The wrapping of data and functions into a single unit or class is called encapsulation.

➔ Encapsulation enables data hiding and information hiding.

10. What is inheritance?

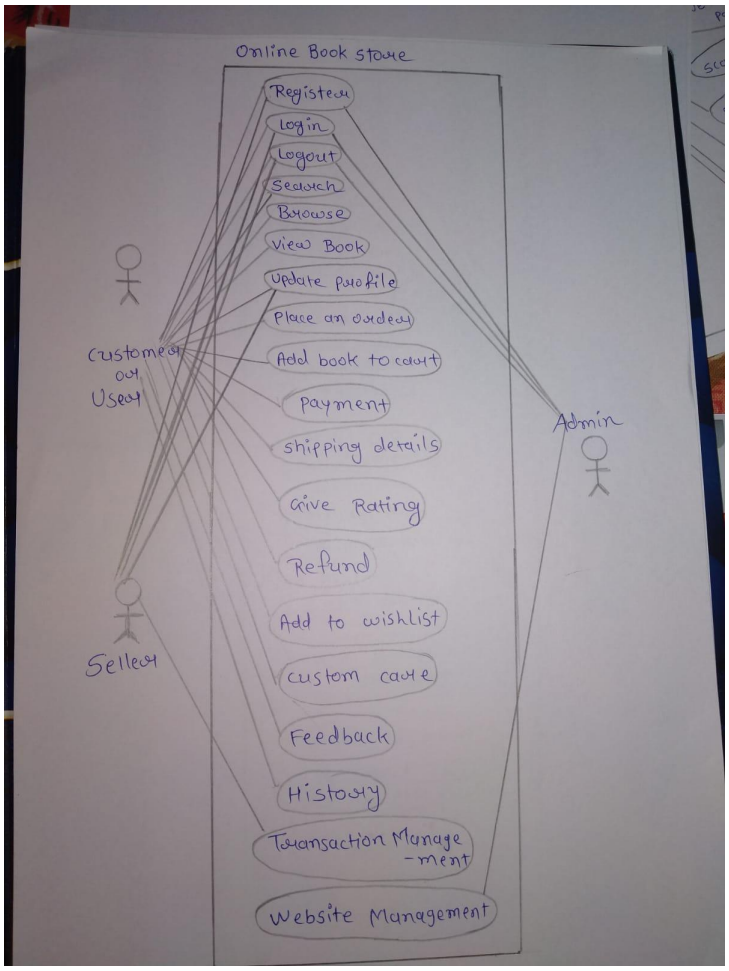
➔ Inheritance is defined as a mechanism where the sub or child class inherits the properties and characteristics of the super class or other derived classes.

11. What is polymorphism?

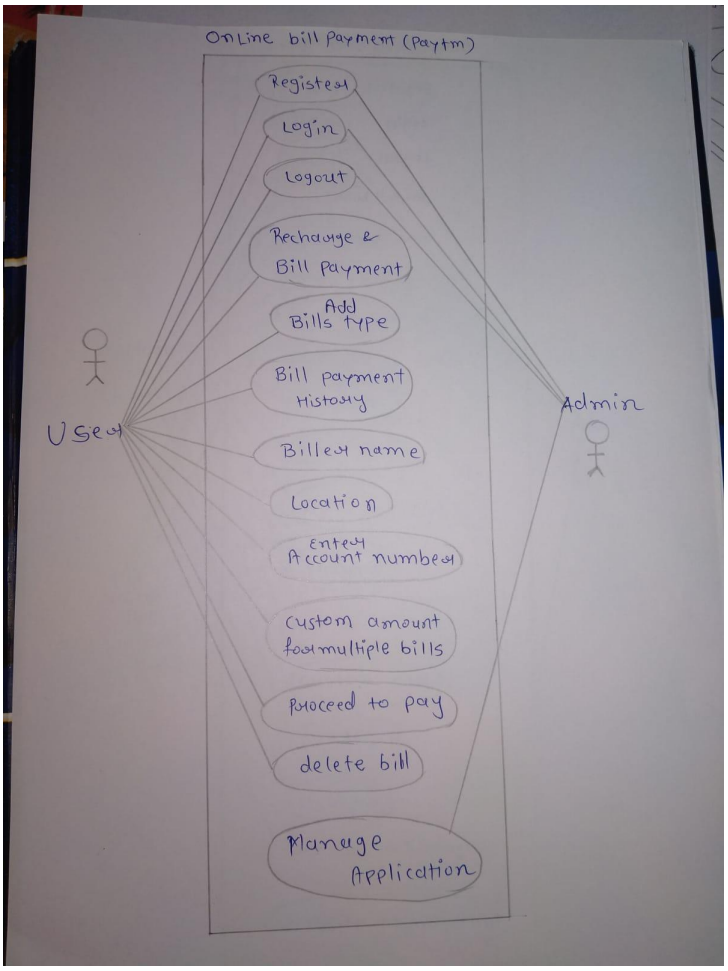
➔ The word “poly” means many and “morphs” means forms, So it means many forms.

➔ Polymorphism refers to the ability of a variable, function, or object to take on multiple forms.

12. Draw Usecase on Online book shopping.



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



14. Write SDLC phases with basic introduction.

➔ There are 6 phases in the SDLC model as given below :

A) Requirements gathering :

- During this phase, all the relevant information is collected from the client to develop a product as per their expectation.
- Any ambiguities must be resolved in this phase only.

B) Analysis :

- The analysis phase includes gathering all the specific details required for a new system.

- This includes all the specifications for software, hardware, and network requirements for the system they plan to build.

C) Design :

- In this phase, the requirement gathered in the SRS document is used as an input and software architecture that is used for implementing system development is derived.
- There are two kinds of design documents developed in this phase :
    1) High-Level Design (HLD)
    2) Low-Level Design (LLD)

D) Implementation :

- In this phase, developers start to build the entire system by writing code using the chosen programming language.
- In the coding phase, tasks are divided into units or modules and assigned to the various developers.
- It is the longest phase of the Software Development Life Cycle process.

E) Testing :

- In this phase the testing team starts testing the functionality of the entire system.
- This is done to verify that the entire application works according to the customer requirement.

F) Maintenance :

- The main focus of this phase is to ensure that needs continue to be met and that the system continues to perform as per the specification mentioned in the first phase.
- There are three types of maintenance :
    1) Corrective maintenance
    2) Adaptive maintenance
    3) Perfective maintenance

15. Explain Phases of the waterfall model.

There are 6 phases in the Waterfall model as given below:

(1) Requirements collection :

- Specifically, we need to know and understand what we have to design, what we have to develop, its processes, what will be its functionality, etc.
- It provides input material to the product being made; thus, the upcoming product is studied, finalised, and marked.
- It also gives us the extension to decide the hardware or software requirements of the product which will be designed, developed, and captured in all phases.

(2) Analysis :

- The system specifications are analysed to generate product models and business logic that will guide production. This is also when financial and technical resources are audited for feasibility.

(3) Design :

- The design phase serves to develop a concrete solution concept based on the previously determined requirements, tasks, and strategies.
- In this phase, software developers develop the software architecture and a detailed construction plan for the software, concentrating on specific components such as interfaces, frameworks, or libraries.
- The result of the design phase comprises a draft document with a software construction plan and test plans for individual components.

(4) Implementation :

- The development team works on coding the project.

- They take the design documents and ensure that their solution follows the design finalised by the architect.
- Since the application is a banking application and security was a high priority in the application requirements, they implement several security checks, audit logging features in the application.
- They also perform several other activities like a senior developer reviewing the other developers code for any issues. Some developers perform static analysis of the code.

(5) Testing :

- The testing team tests the complete application and identifies any defects in the application.
- These defects are fixed by the developers and the testing team tests the fixes to ensure that the defect is fixed.

(6) Maintenance :

- During the maintenance phase, the team ensures that the application is running smoothly on the servers without any downtime.
- Issues that are reported after going live are fixed by the team and tested by the testing team.

16. Write phases of spiral model.

It has four stages or phases :

(1) Determine objectives and find alternate solutions :

This phase includes requirement gathering and analysis. Based on the requirements, objectives are defined and different alternate solutions are proposed.

(2) Risk Analysis and resolving :

In this phase, all the proposed solutions are analysed and any potential risk is identified, analysed, and resolved.

(3) Develop and test :

This phase includes the actual implementation of the different features. All the implemented features are then verified with thorough testing.

(4) Review and planning of the next phase :

In this phase, the software is evaluated by the customer. It also includes risk identification and monitoring like cost overrun or schedule slippage and after that planning of the next phase is started.

17. Write agile manifesto principles.

The twelve principles of agile development include:

1) **Customer satisfaction through early and continuous software delivery-**
   Customers are happier when they receive working software at regular intervals, rather than waiting extended periods of time between releases.
2) **Accommodate changing requirements throughout the development process-**
   The ability to avoid delays when a requirement or feature request changes.
3) **Frequent delivery of working software -**
   Scrum accommodates this principle since the team operates in software sprints or iterations that ensure regular delivery of working software.
4) **Collaboration between the business stakeholders and developers throughout the project -**
   Better decisions are made when the business and technical team are aligned.
5) **Support, trust, and motivate the people involved -**
   Motivated teams are more likely to deliver their best work than unhappy teams.
6) **Enable face-to-face interactions -**
   Communication is more successful when development teams are co-located.

7) **Working software is the primary measure of progress -**
   Delivering functional software to the customer is the ultimate factor that measures progress.
8) **Agile processes to support a consistent development pace -**
   Teams establish a repeatable and maintainable speed at which they can deliver working software, and they repeat it with each release.
9) **Attention to technical detail and design enhances agility -**
   The right skills and good design ensures the team can maintain the pace, constantly improve the product, and sustain change.
10) **Simplicity -**
   Develop just enough to get the job done for right now.
11) **Self-organising teams encourage great architectures, requirements, and designs -**
   Skilled and motivated team members who have decision-making power, take ownership, communicate regularly with other team members, and share ideas that deliver quality products.
12) **Regular reflections on how to become more effective -**

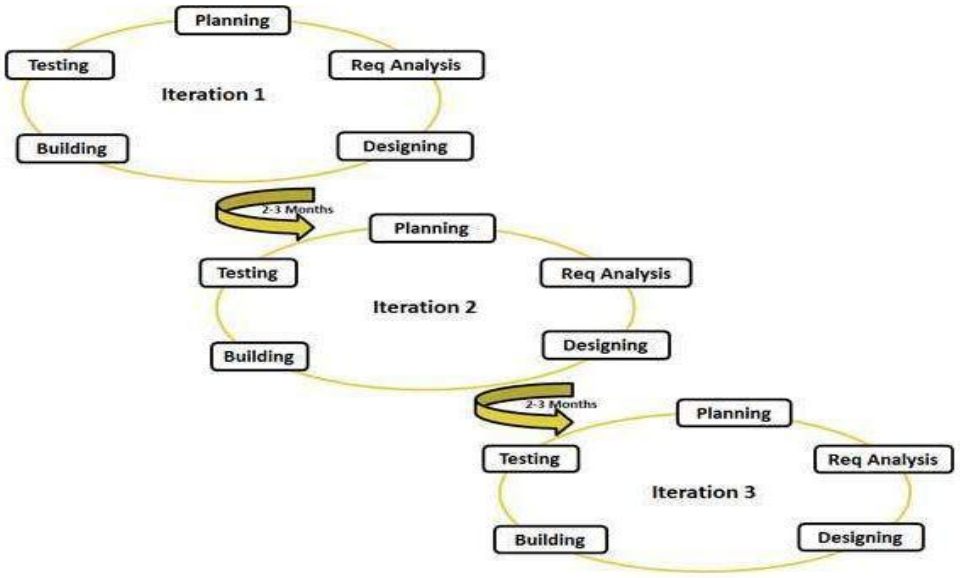
   Self-improvement, process improvement, advancing skills, and techniques help team members work more efficiently.

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

- The meaning of Agile is swift or versatile."Agile process model" refers to a software development approach based on iterative development.
- Agile methods break tasks into smaller iterations, or parts do not directly involve long term planning.
- The project scope and requirements are laid down at the beginning of the development process.
- Plans regarding the number of iterations, the duration and the scope of each iteration are clearly defined in advance.
- Each iteration is considered as a short time "frame" in the Agile process model, which typically lasts from one to four weeks.

- The division of the entire project into smaller parts helps to minimise the project risk and to reduce the overall project delivery time requirements.
- Each iteration involves a team working through a full software development life cycle including planning, requirements analysis, design, coding, and testing before a working product is demonstrated to the client.

Here is a graphical illustration of the Agile Model:



Phases of Agile Model:

1. Requirements gathering
2. Design the requirements
3. Construction/ iteration
4. Testing
5. Deployment
6. Feedback

Advantages of Agile Methodology :

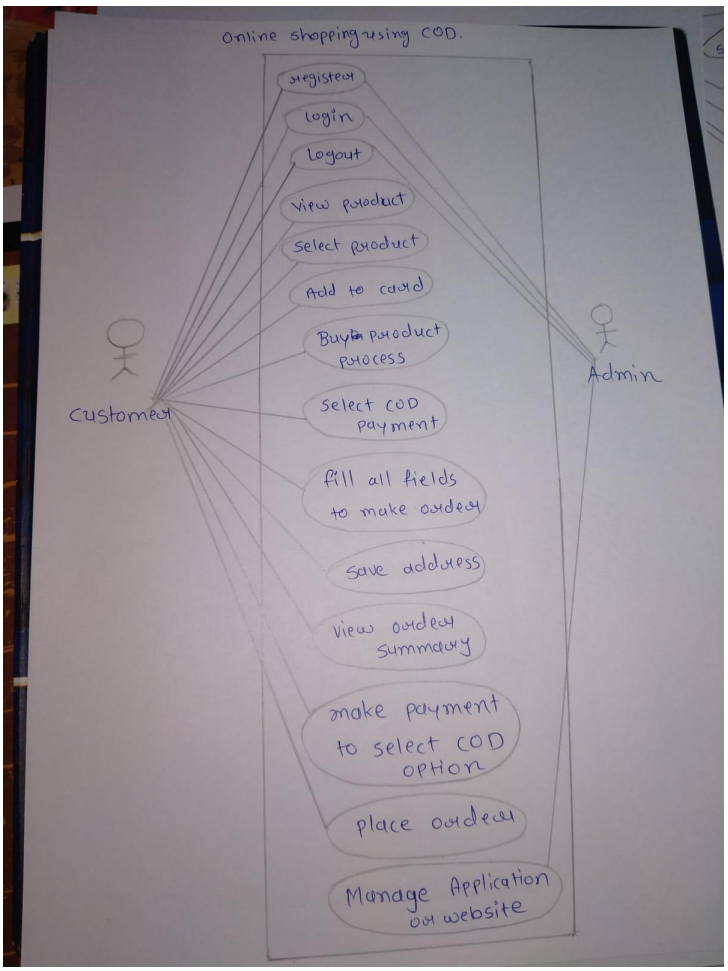
1. In Agile methodology the delivery of software is unremitting.

2. The customers are satisfied because after every Sprint working feature of the software is delivered to them.
3. Customers can have a look of the working feature which fulfilled their expectations.
4. If the customer has any feedback or any change in the feature then it can be accommodated in the current release of the product.
5. In Agile methodology the daily interactions are required between the business people and the developers.
6. In this methodology attention is paid to the good design of the product.
7. Changes in the requirements are accepted even in the later stages of the development.
8. An Agile/Scrum approach can improve organisational synergy by breaking down organisational barriers and developing a spirit of trust and partnership around organisational goals.

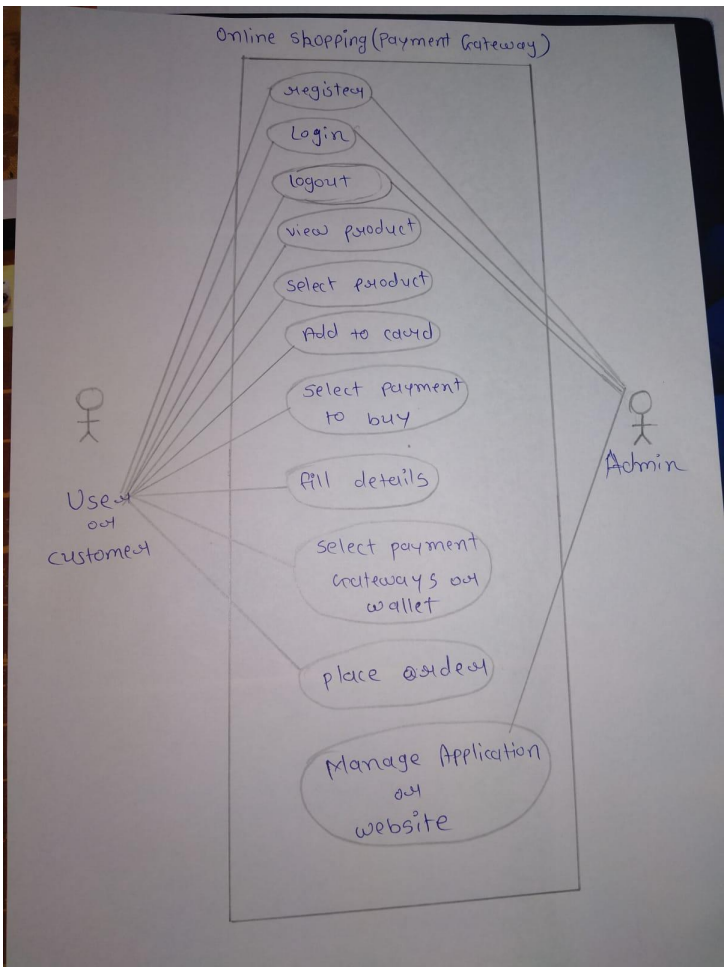
Disadvantages of the Agile Methodology :

1. In Agile methodology the documentation is less.
2. Sometimes in Agile methodology the requirement is not very clear hence it's difficult to predict the expected result.
3. In few of the projects at the starting of the software development life cycle it's difficult to estimate the actual effort required.
4. Because of the ever-evolving features, there is always a risk of the ever-lasting project.
5. For complex projects, the resource requirement and effort are difficult to estimate.

19. Draw use cases on Online shopping products using COD.



20. Draw use cases on Online shopping products using payment gateway.


Online shopping (Payment Gateway)
register
Login
logout
view product
select product
Add to cart
select payment to buy
fill details
select payment Gateways or wallet
place order
Manage Application or website
User or customer
Admin