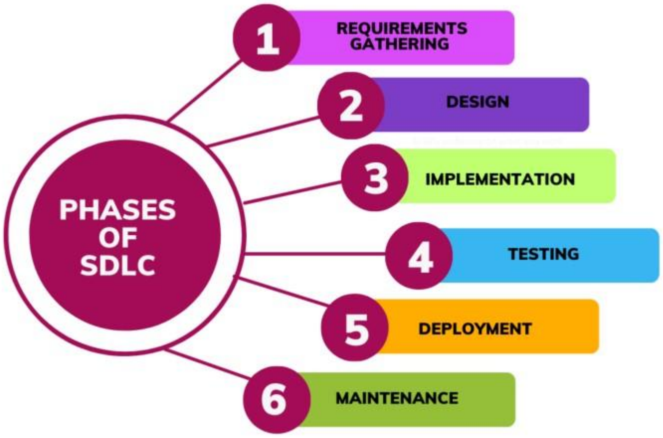
Q.1 What is SDLC?

Software Development Life Cycle(SDLC)is a structured process that is used to design develop,& test good quality software. software development life cycle, is a methodology that defines the entire procedure of software development step-by-step.

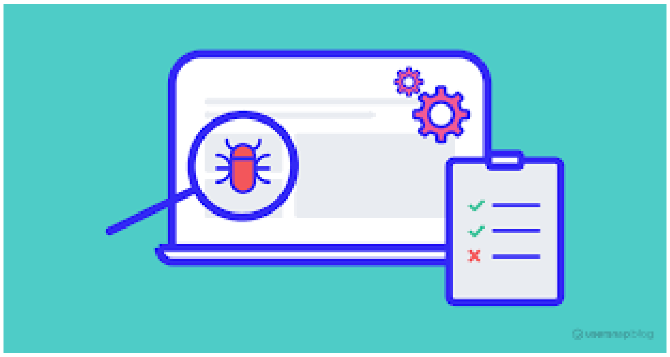


Q.2 What is Software Testing?

Software testing is a process of validating software functionality, performance, and reliability.

Software Testing is a process of software analysis and defect detecting.

Action of defect detecting are directed to defining as many as possible defect, which should be fixed.



Q.3 What is Agile methodology?

Agile is a methodology for modeling and documenting software systems based on best practices. It is a collection of values and principles that can be applied on an software development project.

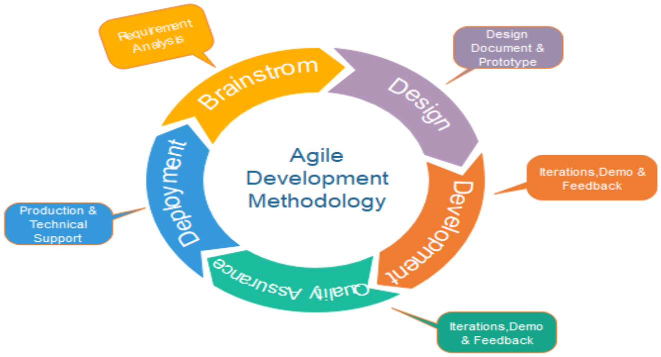

Requirement Analysis
Brainstrom
Design
Design Document & Prototype
Agile Development Methodology
Iterations,Demo & Feedback
Development
Quality Assurance
Iterations,Demo & Feedback
Deployment
Production & Technical Support

Fig. Agile Model

Q.4 What is SRS?

A software requirements specification (SRS)is a document that describes what the software will do and how it will be expected to perfume.

The SRS is a specification for a specific software product,program,or set of application that perform particular functions in a specific environment.

First, the SRS written by the client of a system.second,the SRS could be written by a developer of the system.

The two methods create entirely various situations and establish different purposes for the document altogether. The first case,SRS,is used to define the needs and expectation of the users. The second case,SRS,is the written for various purpose and serves as a contract document between customer and developer.

SRS is the a complete description of the behavior of the system to be developed. In include a set of use cases that describe all of the interactions that the users will have with the software.

- **Types of Requirements**

  1. **Customer Requirements**
  2. **Function Requirements**
  3. **Non-function Requirements**

1.Cutomer Requirements

- The customers are those that perform the eight primary function of systems engineering, with special emphasis on the operator as the key customer.
- Operation Distribution or deployment where will the system be system?
- Mission profile or scenario:-How will the system accomplish its mission objective?
- Performance and related parameters what are the critical system parameters to accomplish mission?
- Utilization environments: How are the various system components to be used?
- Effectiveness requirements:- How effective or efficient must the system be in performing its mission?
- Operational life cycle:- Hoe long will the system be in use by user?
- Environment: - What environments will the system be expected to operate in an effective manner?

2. Functional Requirements

- Functional Requirements are very important system requirements in the system design process.
- The system shall support the ability to receive e-mails.

    - To send e-mails.
    - To create new folders.
    - To filter e-mails in different folders.
    - To attach different kind of attachments.
    - To create and maintain address book.
    - They system shall support the ability to create unlimited user accounts with different e-mail address.

3. Non-Functional Requirements

- Non –functional requirements are requirements that specify criteria that can be used to judge the operation of a system rather than specific behaviors.

    EXAMPLE:-

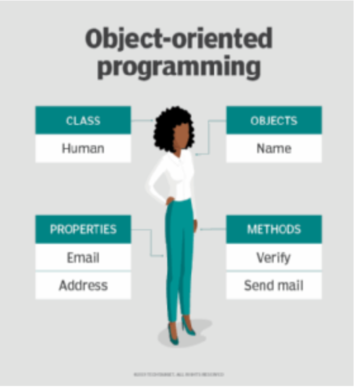
    - System must be built for a total installed cost of $1,050,000.00.
    - System must run on windows server 2003.
    - System must be secured against "Trojan attacks".

- It can be divided in to following category.
    - Usability
    - Reliability
    - Performance
    - Security

Q.5 what is OOPs?

The word **object-oriented** is the combination of two words i.e. **object** and **oriented.** The dictionary meaning of the object is an article or entity that exists in the real world. The meaning of oriented is interested in a particular kind of thing or entity.

It is a programming pattern that rounds around an object or entity are called **object-oriented programming.**



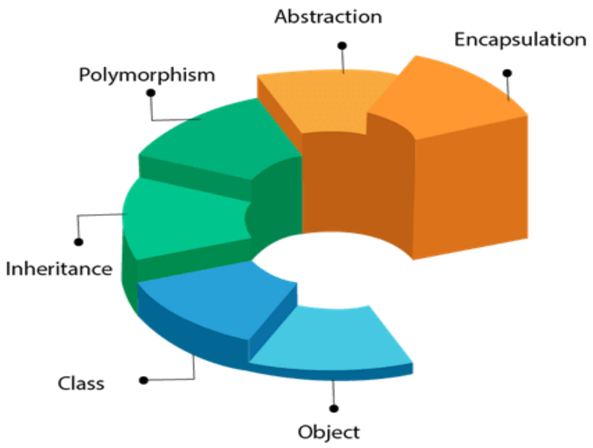
Object-oriented programming is a methodology or paradigm to design a program using classes and objects.

Q.6 Write basic concept OOPs?

Object means a real-world entity such as a pen, chair, table, computer, watch, etc. **Object-Oriented Programming** is a methodology or paradigm to design a program using classes and objects. It simplifies software development and maintenance by providing some concepts:

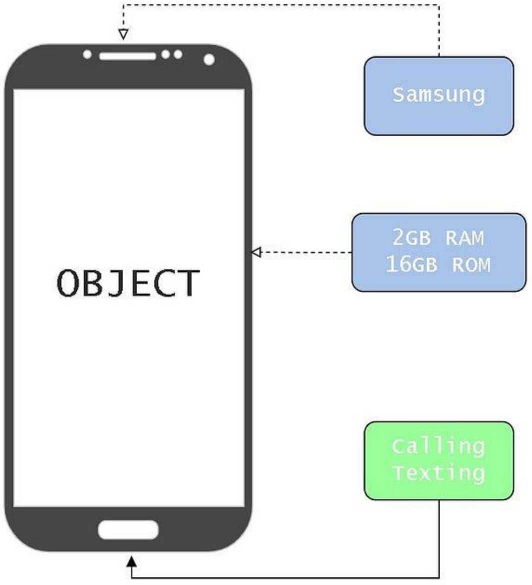
- Object
- Class
- Inheritance
- Polymorphism
- Abstraction
- Encapsulation

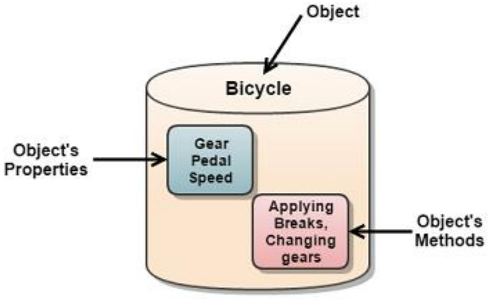
OOPs (Object-Oriented Programming System)



Q.7 What is Object?

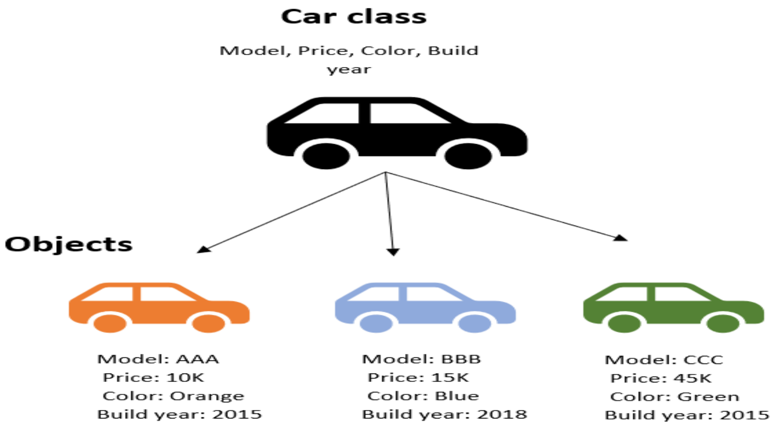
- An object is a real-world entity that has attributes, behavior, and properties.
- It is referred to as an instance of the class.
- It contains member functions, variables that we have defined in the class.
- It occupies space in the memory.
- Different objects have different states or attributes, and behaviors.





Q.8 What is class?

- Collection of objects is called class.
- It is a logical entity.
- A class can also be defined as a blueprint from which you can create an individual object. Class doesn't consume any space.
- It just describes the properties of the class and how it will work. It is the object of the class which actually runs in a program to perform an action (get its functional details) using its fields and properties.



Q.9 What is inheritance?

- Inheritance is a fundamental concept of Object-Oriented Programming (OOP) that enables a new class to inherit the properties and methods of an existing class.
- Making a class from an existing class periling the attribute of some other class.

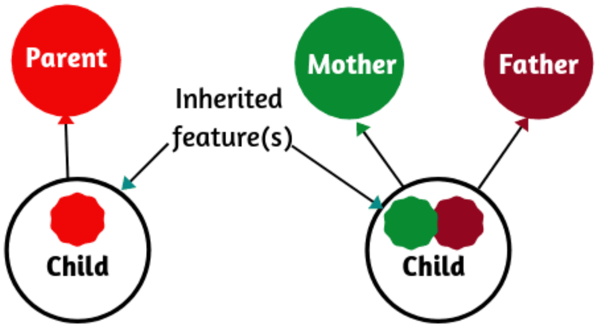
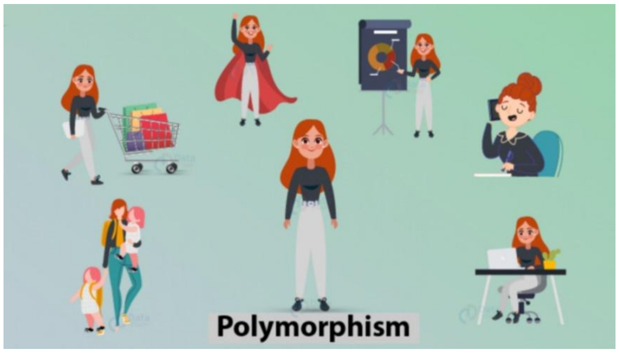


Fig: Inheritance of features in the real world

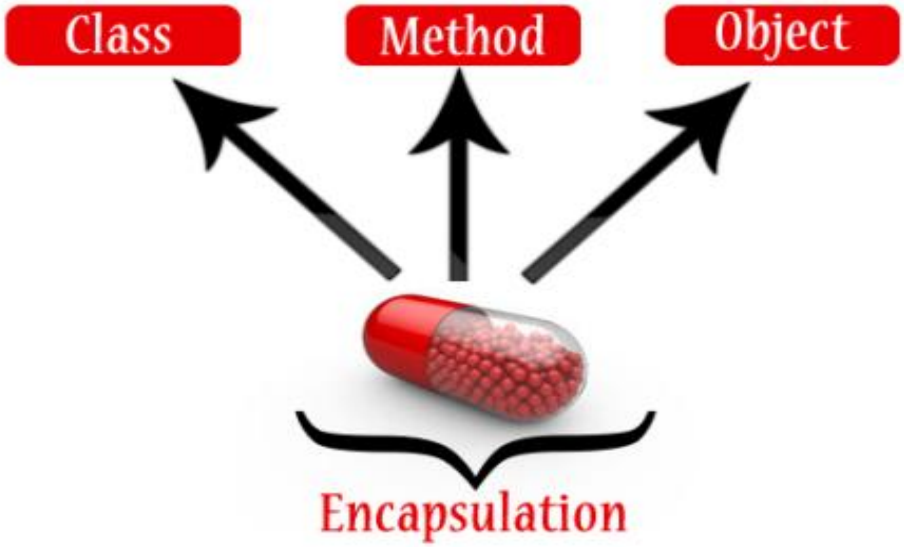
Q.10 What is polymorphism?

- **Polymorphism** is the ability of any data to be processed in more than one form. The word itself indicates the meaning as **poly** means **many** and **morphism** means **types**.
- Polymorphism is one of the most important concepts of object-oriented programming languages.
- Polymorphism means one name multiple form.



Q.11 What is Encapsulation?

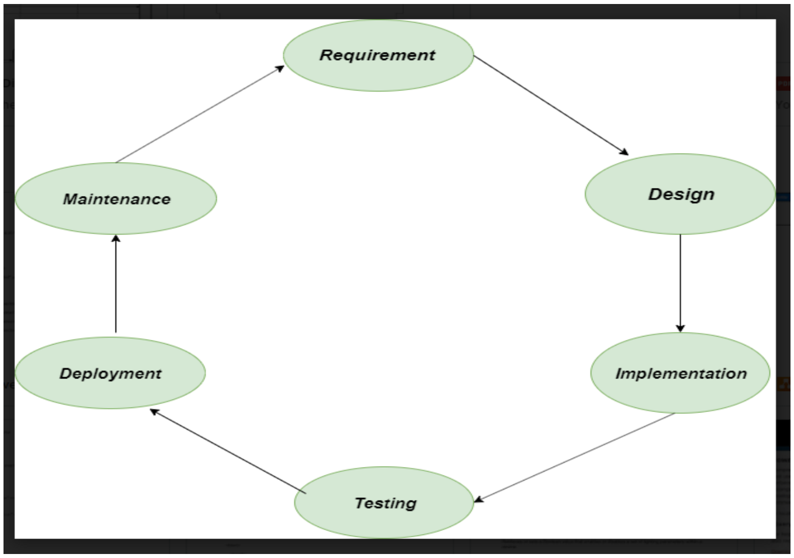
Encapsulation is a process of wrapping code and data together into a single unit, for example, a capsule which is mixed of several medicines.



Q.12 Write SDLC phaces with basic introduction?

The SDLC the following phases:

1. Requirements
2. Design
3. Implementation
4. Testing
5. Deployment
6. Maintenance



1. **<u>Requirements gathering and analysis:</u>**

    - This phase involves gathering information about the software requirements from stakeholders, such as customers, end-users, and business analysts.
    - Establish customer needs(what is the problem?)

2. **<u>Design:</u>**

    - In this phase, the software design is created, which includes the overall architecture of the software, data structures, and interfaces.

3. **Implementation or coding**:

    - The design is then implemented in code, usually in several iterations, and this phase is also called as Development.

    Things you need to know about this phase:

    - This is the longest phase in SDLC model.
    - This phase consists of Front end + Middleware + Back-end.
    - **In front-end:** Development of coding is done even SEO settings are done.
    - **In Middleware:** They connect both the front end and back end.
    - **In the back-end:** A database is created.

4. **Testing**:

    - The software is thoroughly tested to ensure that it meets the requirements and works correctly.

5. **Deployment:**

    - After successful testing, The software is deployed to a production environment and made available to end-users.

6. **Maintenance:**

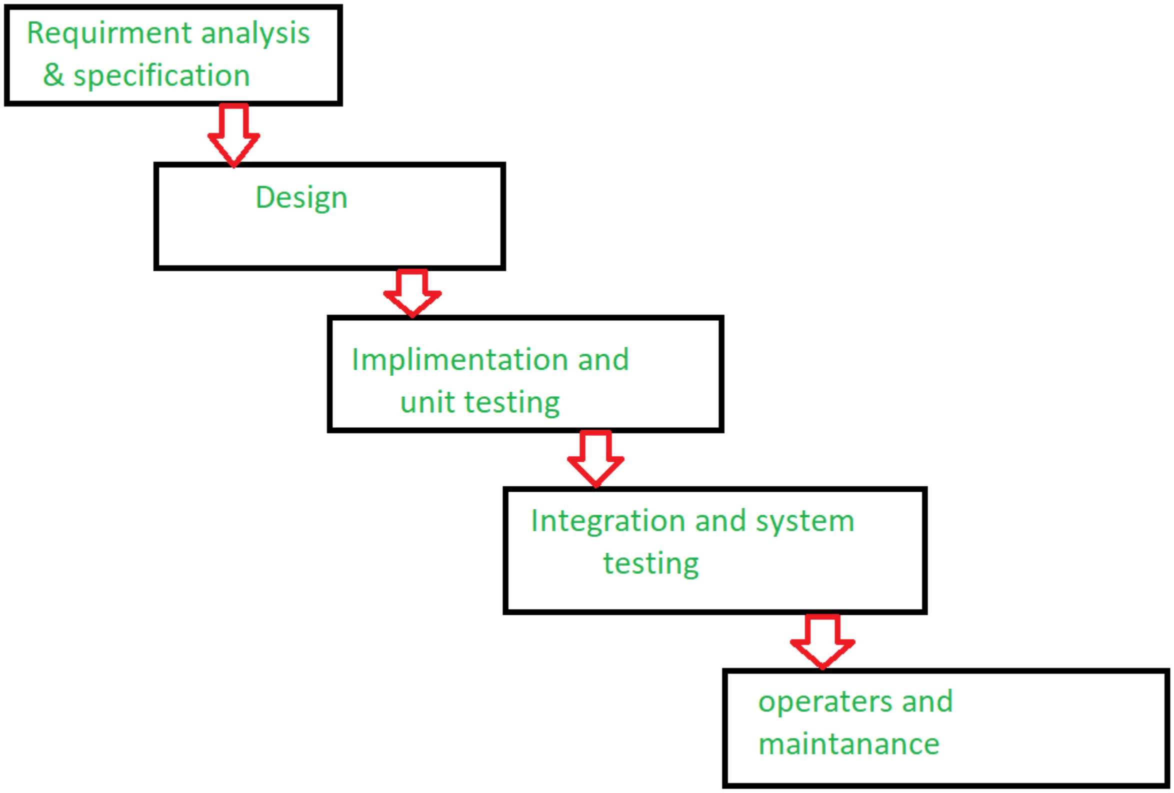
    - This phase includes ongoing support, bug fixes, and updates to the software.

Q.13 Explain phases of waterfall model.

Waterfall Model is a classical software development methodology that was first introduced by Winston W. Royce in 1970. It is a linear and sequential approach to software development that consists of several phases that must be completed in a specific order.

The Waterfall Model has six phases:-

1. **Requirements** Gathering and Analysis
2. Design Phase
3. Implementation and Unit Testing
4. Integration and System Testing
5. Deployment
6. Maintenance


Requirment analysis & specification
Design
Implimentation and unit testing
Integration and system testing
operaters and maintanance

1. Requirements Gathering and Analysis:

- The first phase involves gathering requirements from stakeholders and analyzing them to understand the scope and objectives of the project.

2. Design Phase:

- Once the requirements are understood, the design phase begins. This involves creating a detailed design document that outlines the software architecture, user interface, and system components.

3. Implementation and Unit Testing:

- The implementation phase involves coding the software based on the design specifications. This phase also includes unit testing to ensure that each component of the software is working as expected.

4. Integration and System Testing:

- In the testing phase, the software is tested as a whole to ensure that it meets the requirements and is free from defects.

5. Deployment:

- Once the software has been tested and approved, it is deployed to the production environment.

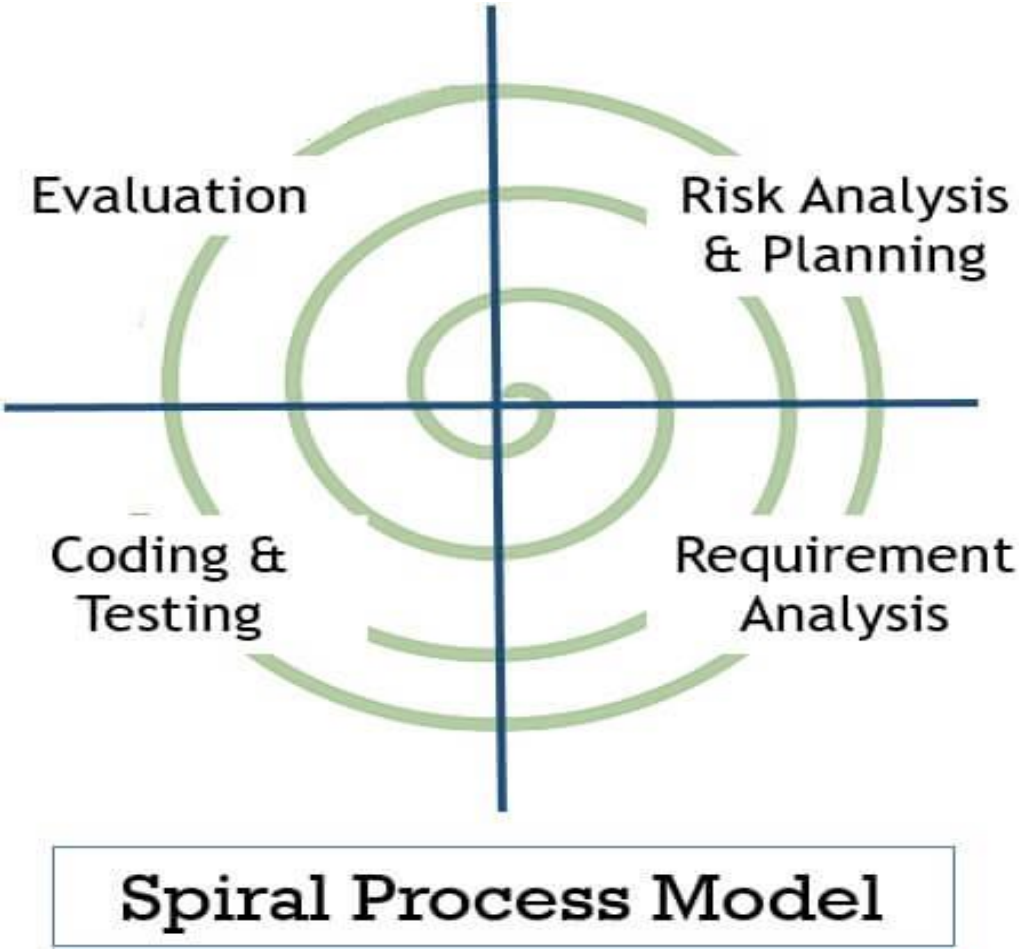
- The final phase of the Waterfall Model is maintenance, which involves fixing any issues that arise after the software has been deployed and ensuring that it continues to meet the requirements over time

Q.14 Write phases of spiral model.

In this model, we create the application module by module and handed over to the customer so that they can start using the application at a very early stage. And we prepare this model only when the module is dependent on each other. In this model, we develop the application in the stages because sometimes the client gives the requirements in between the process.

The different phases of the spiral model are as follows:

1. Requirement analysis
2. Design
3. Coding
4. Testing and risk analysis



1. **Requirement analysis :**

    - In this, the following spirals will include the documentation of system requirements, unit requirements, and the subsystem needs. In this stage, we can easily understand the system requirements because the business analyst and the client have constant communication. And once the cycle is completed, the application will be deployed in the market.

2. **Design :**

    - The second stage of the spiral model is designed, where we will plan the logical design, architectural design, flow charts, decision tree, and so on.

3. **Coding:-**

    - After the compilation of the design stage, we will move to our next step, which is the coding stage. In this, we will develop the product based on the client's requirement and getting the client's feedback as well. This stage refers to the construction of the real application in every cycle.
    - And those spirals had an excellent clarity of the requirements, and the design details of an application are known as the build with having version numbers. After that, these builds are transferred to the client for their responses.

4. **Testing and Risk Analysis :-**

    - Once the development is completed successfully, we will test the build at the end of the first cycle and also analyze the risk of the software on the different aspects such as managing risks, detecting, and observing the technical feasibility. And after that, the client will test the application and give feedback.

Q.15 Write agile manifesto principles.

The Twelve Principle of Agile Manifesto

1. **Customer Satisfaction:**

   - Manifesto provides high priority to satisfy the costumer's requirements. This is done through early and continuous delivery of valuable software.

2. **Welcome Change:**

   - Making changes during software development is common and inevitable. Every changing requirement should be welcome, even in the late development phase. Agile process works to increase the customers' competitive advantage.

3. **Deliver the Working Software:**

   - Deliver the working software frequently, ranging from a few weeks to a few months with considering the shortest time period.

4. **Collaboration:**

   - Business people (Scrum Master and Project Owner) and developers must work together during the entire life of a project development phase.

5. **Motivation:**

   - Projects should be building around motivated team members. Provide such environment that supports individual team members and trust them. It makes them feel responsible for getting the job done thoroughly.

6. **Face-to-face Conversation:**

   - Face-to-face conversation between Scrum Master and development team and between the Scrum Master and customers for the most efficient and effective method of conveying information to and within a development team.

7. **Measure the Progress as per the Working Software:**

   - The working software is the key and primary measure of the progress.

8. **Maintain Constant Pace:**

   - The aim of agile development is sustainable development. All the businesses and users should be able to maintain a constant pace with the project.

9. **Monitoring:**

   - Pay regular attention to technical excellence and good design to maximize agility.

10. **Simplicity:**

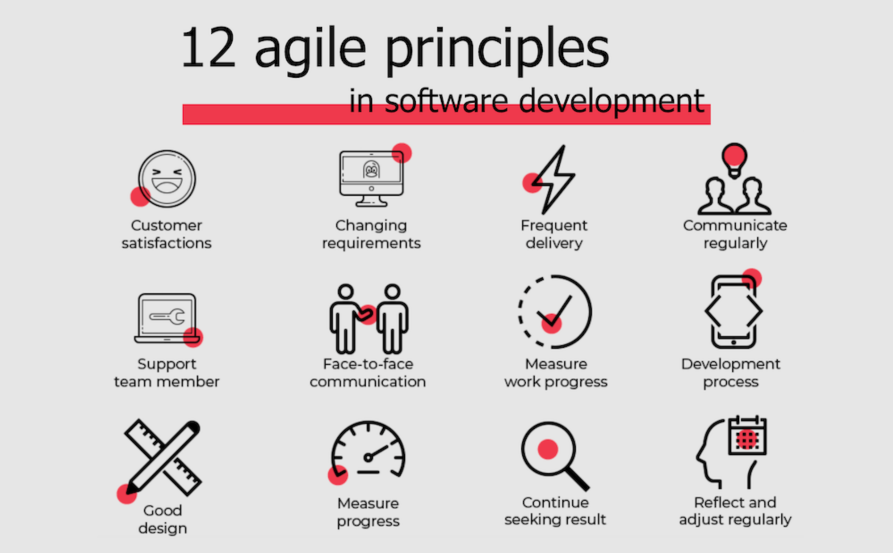
    - Keep things simple and use simple terms to measure the work that is not completed.

11. **Self-organized Teams:**

    - The Agile team should be self-organized. They should not be depending heavily on other teams because the best architectures, requirements, and designs emerge from self-organized teams.

12. **Review the Work Regularly:**

    - The work should be reviewed at regular intervals, so that the team can reflect on how to become more productive and adjust its behavior accordingly.



Q.16 Explain working methodology of agile model and also write pros and cons.

An agile methodology is an iterative approach to software development. Each iteration of agile methodology takes a short time interval of 1 to 4 weeks. The agile development process is aligned to deliver the changing business requirement. It distributes the software with faster and fewer changes.

The single-phase software development takes 6 to 18 months. In single-phase development, all the requirement gathering and risks management factors are predicted initially.

The agile software development process frequently takes the feedback of workable product. The workable product is delivered within 1 to 4 weeks of iteration.

If any enhancement is needed after product feedback in the project then its done and its re-released.

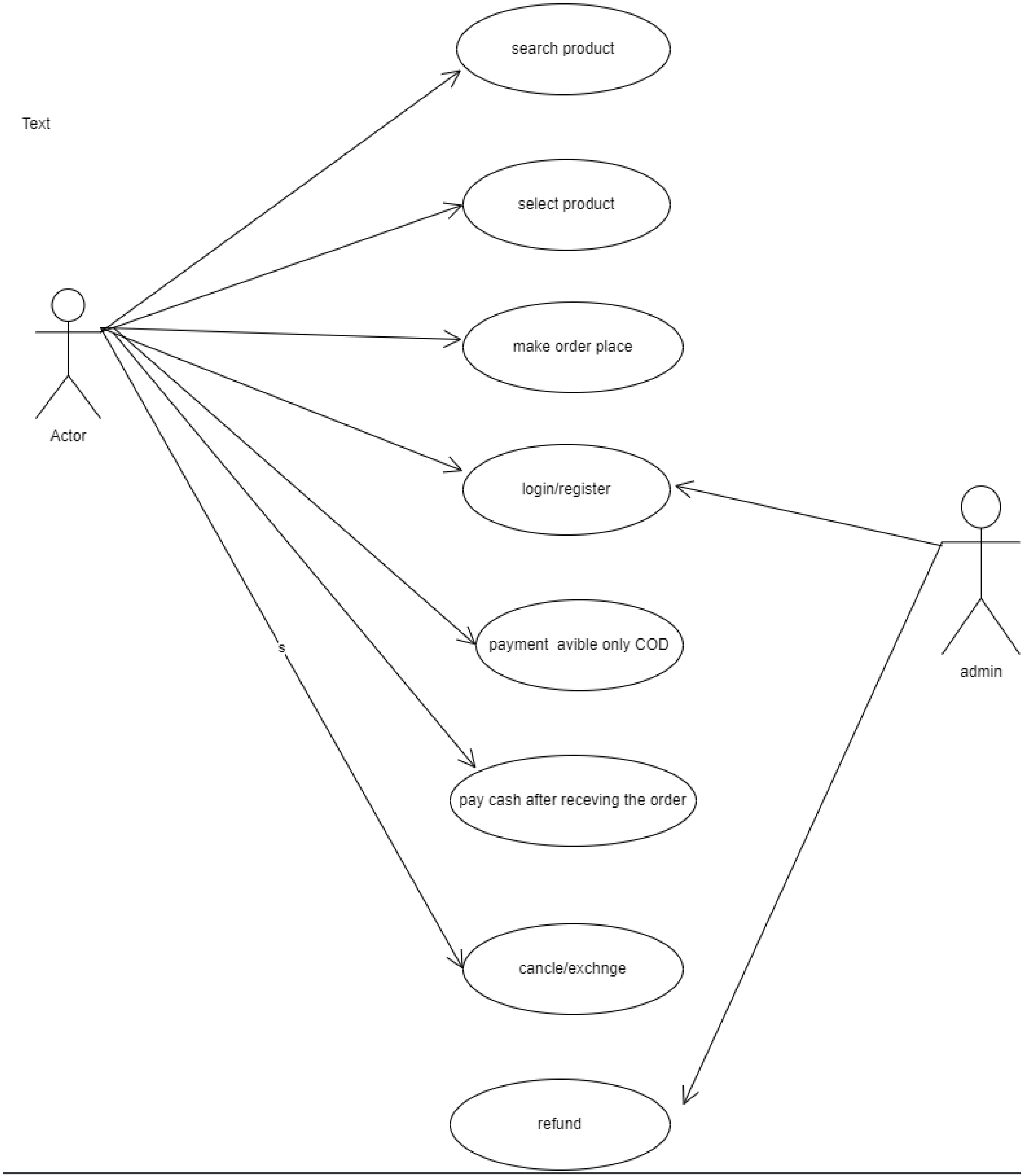
Advantages of Agile Methodology

1. Customer satisfaction is rapid, continuous development and delivery of useful software.
2. Customer, Developer, and Product Owner interact regularly to emphasize rather than processes and tools.
3. Product is developed fast and frequently delivered (weeks rather than months.)
4. A face-to-face conversation is the best form of communication.
5. It continuously gave attention to technical excellence and good design.
6. Daily and close cooperation between business people and developers.
7. Regular adaptation to changing circumstances.
8. Even late changes in requirements are welcomed.

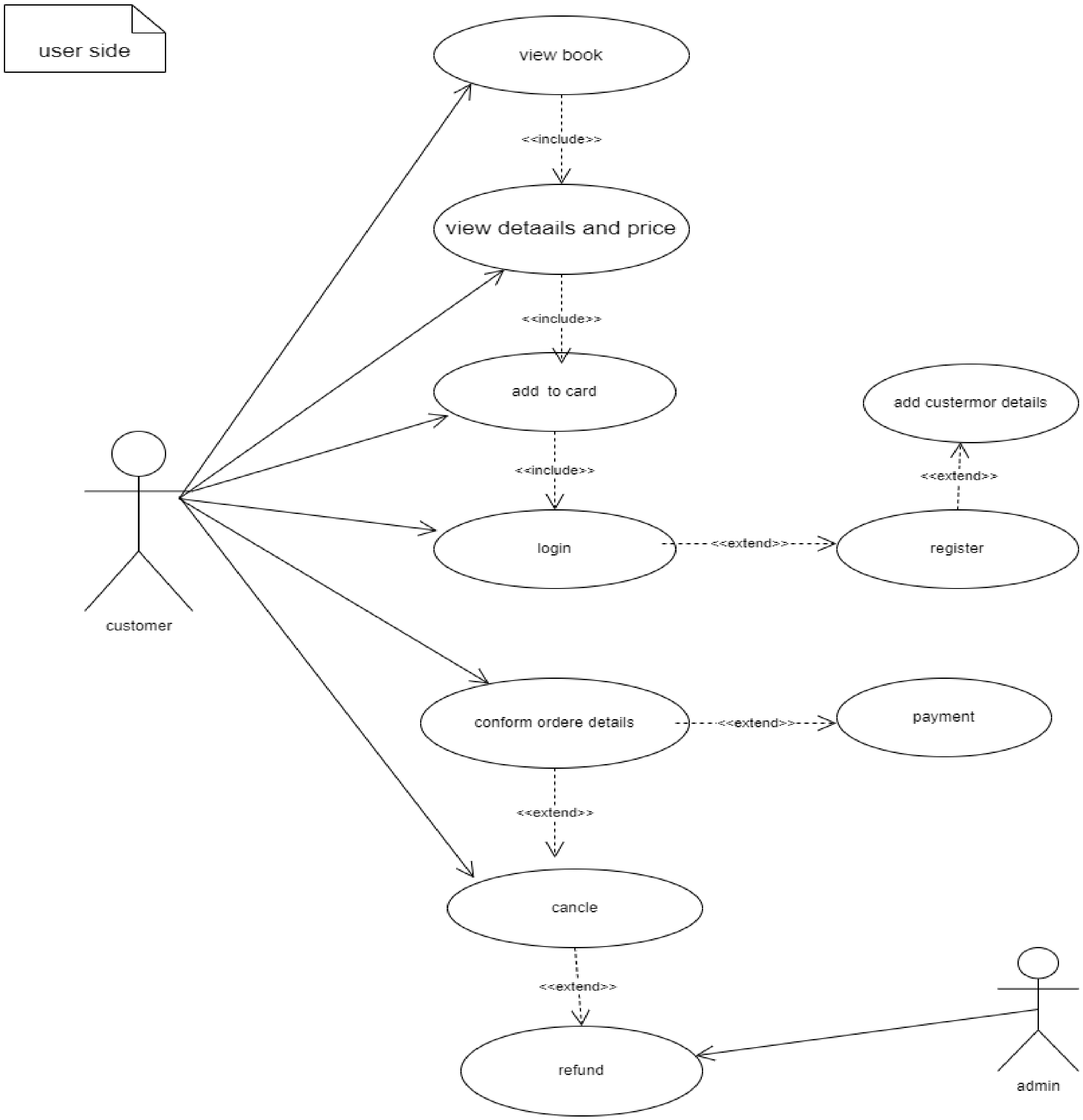
Disadvantages of Agile methodology

1. It is not useful for small development projects.
2. There is a lack of intensity on necessary designing and documentation.
3. It requires an expert project member to take crucial decisions in the meeting.
4. Cost of Agile development methodology is slightly more as compared to other development methodology.
5. The project can quickly go out off track if the project manager is not clear about requirements and what outcome he/she wants.

Q.17 Draw use case on Online shopping product using COD.

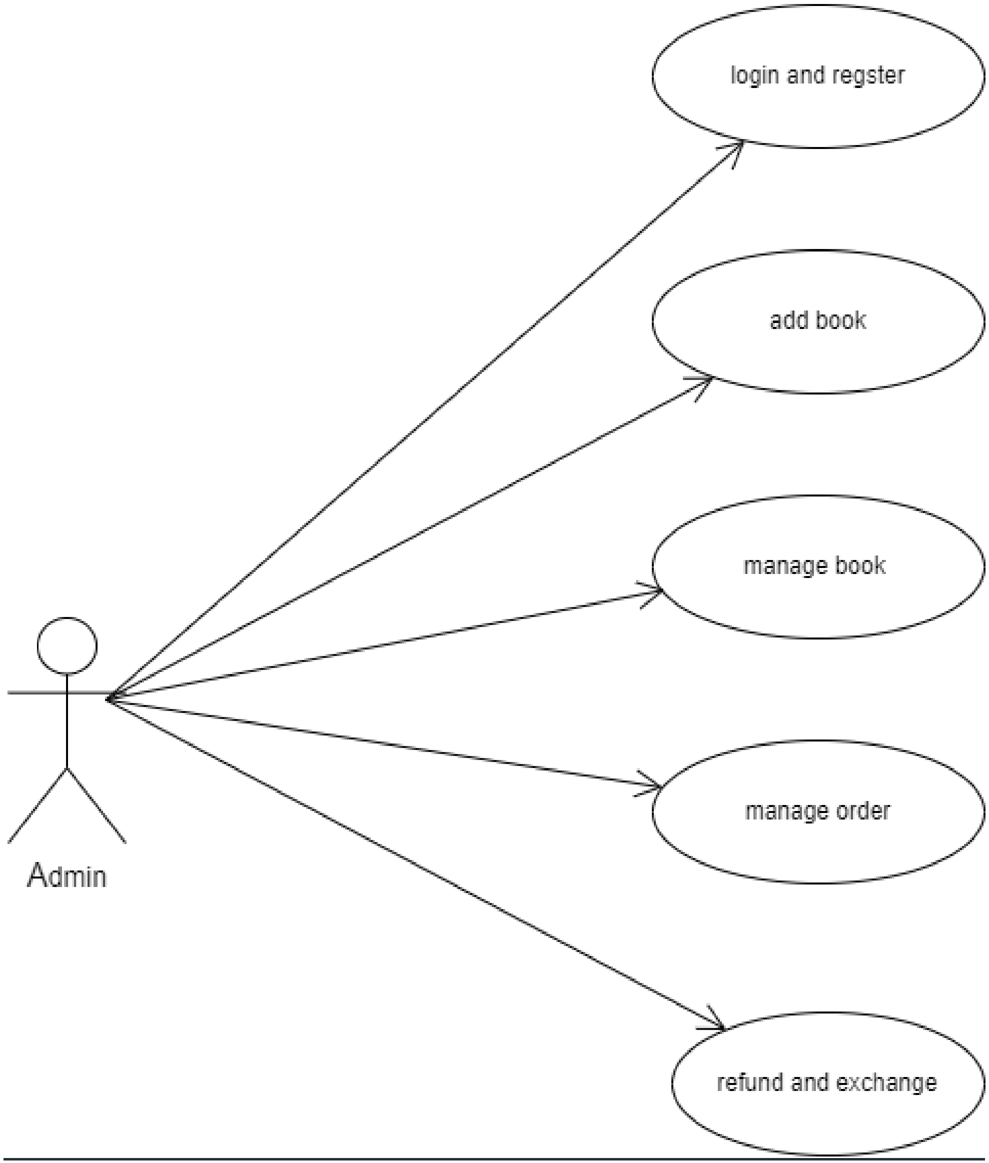


Q.18 Draw Usecase on Online book shopping.

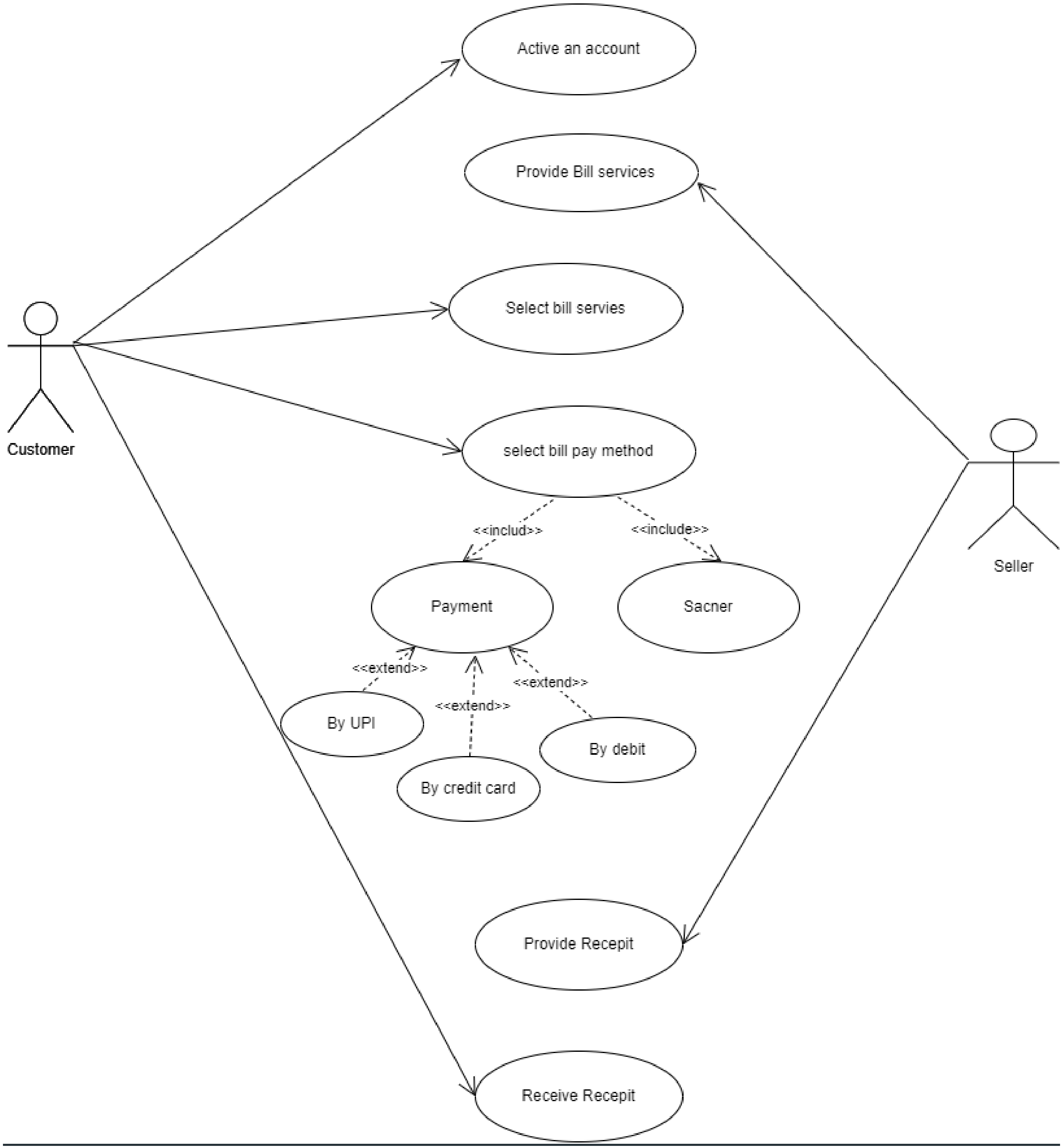


-

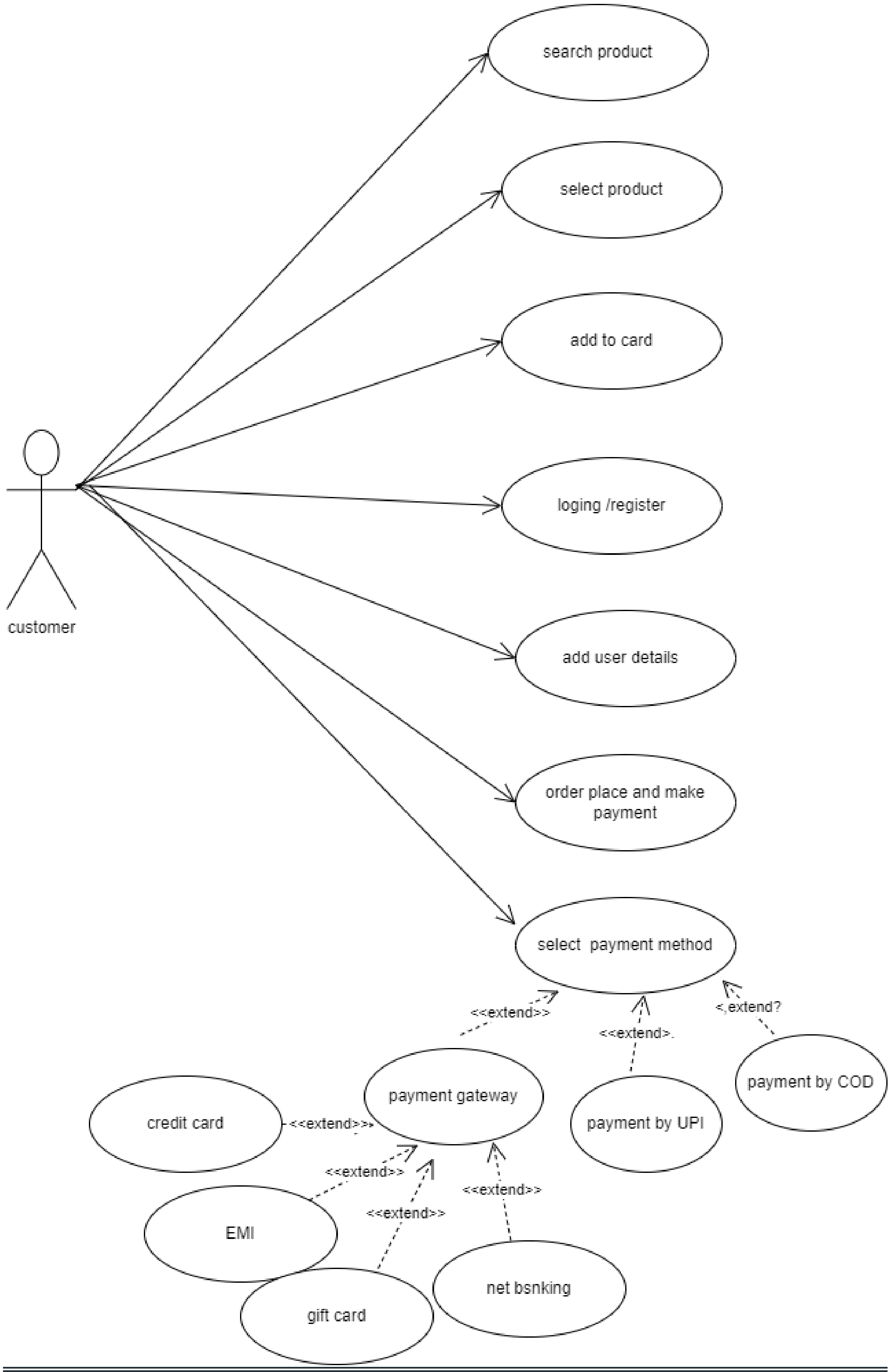
Admin


login and regster
add book
manage book
manage order
refund and exchange
Admin

Q.19 Draw Usecase on online bill payment system (paytm).



Q.20 Draw usecase on Online shopping product using payment gateway


search product
select product
add to card
customer
loging /register
add user details
order place and make payment
select payment method
<<extend>>
<<extend>.
<,extend?
payment gateway
payment by UPI
payment by COD
credit card
<<extend>>>
<<extend>>
<<extend>>
<<extend>>
EMI
gift card
net bsnking