Module – 1 (Fundamental)

1. What is SDLC ?

* The full form of SDLC is Software Development Life Cycle .

1. What is software testing ?

* Software testing is a process of identifying the correctness , completeness and quality of developed software .

1. 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.
* This method breaks the product into small incremental builds . This builds are parted in iterations . Each iteration lasts about 1-3 weeks.
* Every iteration involves the teams working simultaneously on various areas like planning , requirements analysis , designing , coding , testing etc. . At each iteration end , a working software is shown to the customer.

1. What is SRS ?

* SRS stands for Software Requirement Specification is the description of the behaviour of the system to be developed.
* It includes use cases that specifies all the interactions that the user will have with the software.
* The requirements can be specified into three parts :
  + Customer Requirements
    - Defines the basic needs based on customer specification.
  + Functional Requirements
    - Defines the requirements that are important system requirements in system design process.
  + Non-Functional Requirements
    - Defines how the system should behave or operate.

1. What is OOPS ?

* OOPS stands for Object Oriented Programming System.
* It focuses on organizing code into reusable , self-contained units(objects) that contain data and the methods (functions) that operate on that data.

1. Write basic concepts of OOPS.

* The basic concepts of OOPS are :
  + Class
  + Object
  + Encapsulation
  + Inheritance
    - Single-level inheritance
    - Multi-level inheritance
    - Hierarchical inheritance
    - Multiple inheritance
    - Hybrid inheritance
  + Polymorphism
    - Overriding
    - Overloading
  + Abstraction

1. What is object ?

* An object is an identifiable thing with some characteristics and behaviour.

1. What is class ?

* A class is the user-defined data type that represents the group of objects.
* Class class\_name {

//object

}

1. What is encapsulation ?

* It is defined as the wrapping up of data and information under a single unit.

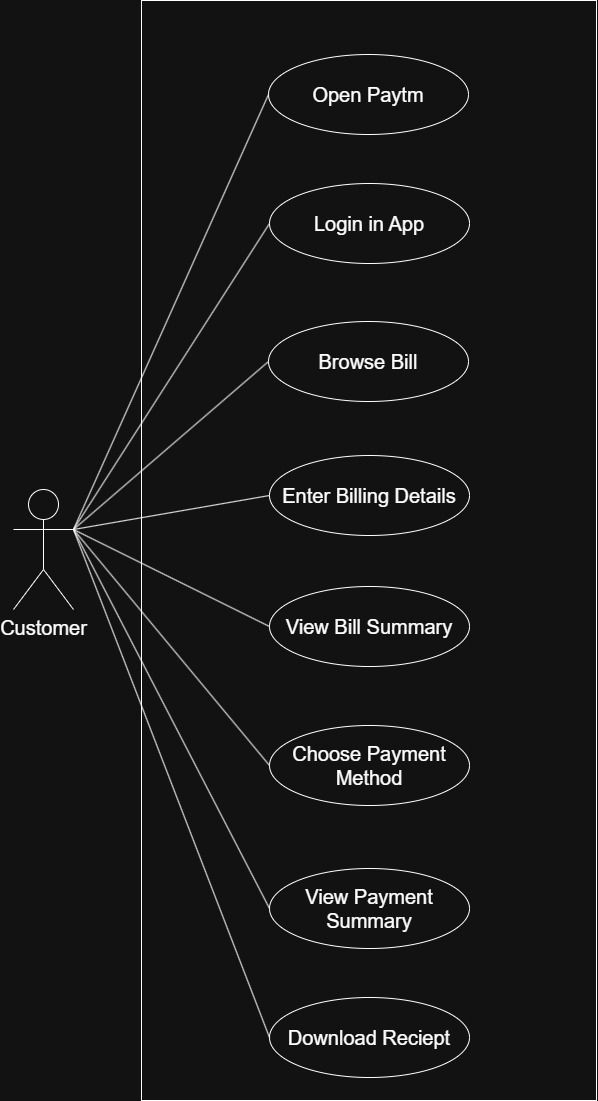
1. What is inheritance ?

* The deriving of properties in a class from another class is known as inheritance.

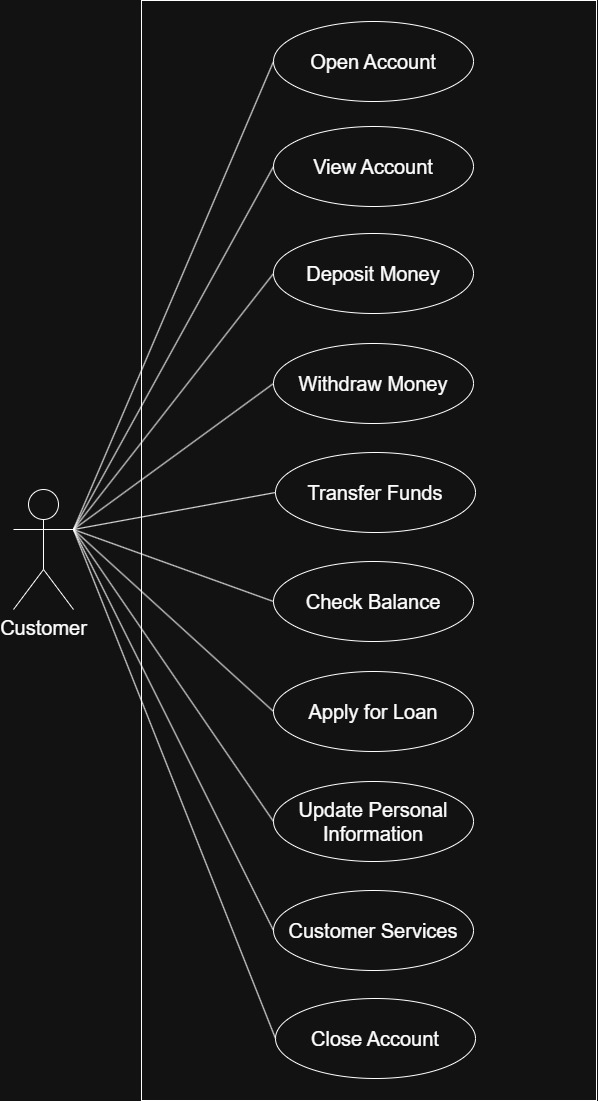
1. What is polymorphism ?

* The word polymorphism means having many forms.
* It is the ability of a method to behave differently in different scenarios.

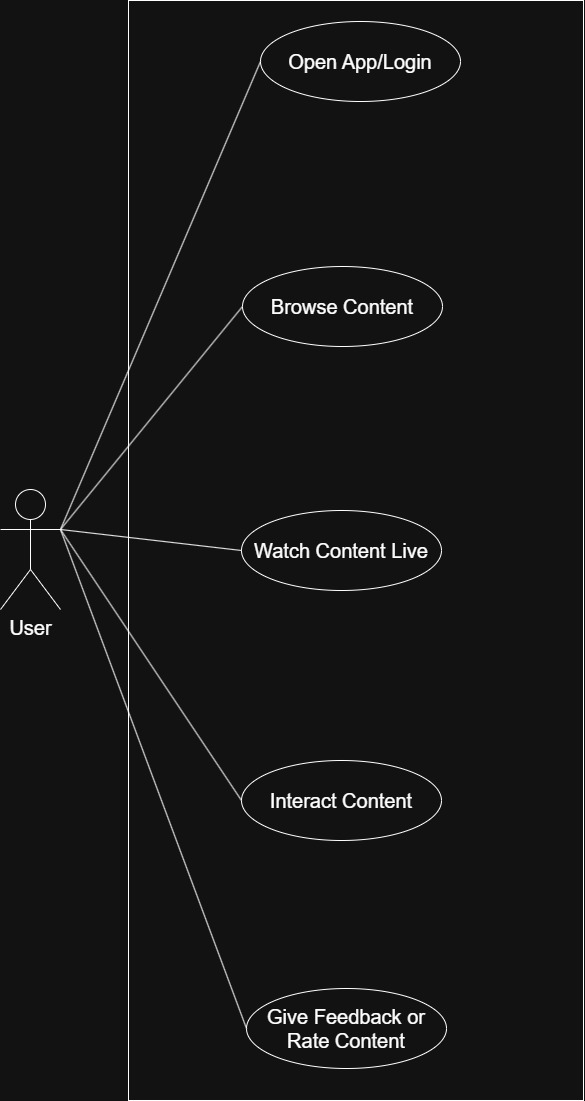
1. Draw Use-case on online bill payment system (Paytm).



1. Draw Use-case on banking system for customers.



1. Draw Use-case on Broadcasting System.



1. Write SDLC phases with basic introduction.

* The Software Development Life Cycle phases are as follows :
  + Requirement Gathering/Collection
    - In this phase , the requirements in the document in the form of diagrams and tables.
  + Analysis
    - In this phase , the documented requirements are implemented into architecture.
  + Design
    - In this phase , the design of the system begins on the basis of architecture document provided.
  + Implementation
    - In this phase , the team builds the components of the process.
  + Testing
    - In this phase , the process created is been tested thoroughly.
    - This phase is responsible for establishing high quality across all phases.
  + Maintenance
    - In this phase , in which the process is enhanced and optimized as well as fixing defects/errors.

1. Explain phases of the waterfall model.

* Requirements Collection
  + Requirements are gathered in form of a document.
* Analysis
  + Analysing the requirements as specified from document.
* Design
  + Designing the proper architecture based on analysed document.
* Implementation
  + Implementing the architecture.
* Testing
  + Testing the process.
* Maintenance
  + Optimizing and enhancing the process as well as solving errors.

1. Write phases of spiral model.

* There are four main phases :
  + Planning
    - Determination of objectives and plans as per requirement.
  + Risk Analysis
    - Analysis of risks and finding its alternative solutions.
  + Engineering
    - Development of the process/system.
  + Customer Evaluation
    - Evaluation of the software/process by the customer/client.

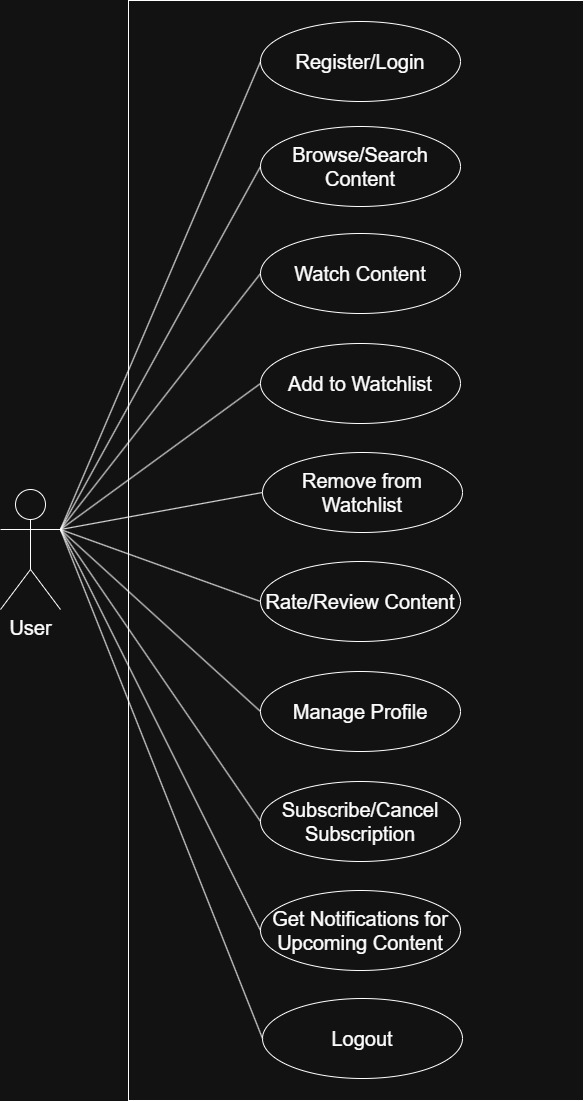
1. Write agile manifesto principles.

* The principles of agile are :
  + Customer satisfaction :- Prioritize early delivery of software.
  + Embrace change :- Welcome changing requirements.
  + Collaboration :- Business and developers should work together for good output of project.
  + Simplicity :- Prioritize simplicity and avoid unnecessary complexity.

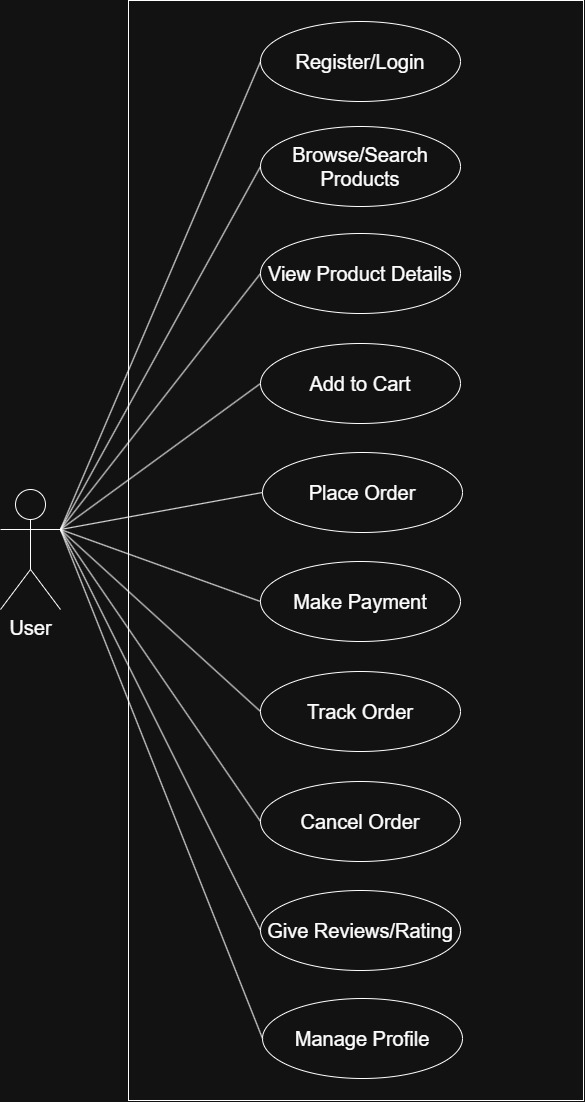
1. Explain working methodology of agile model and also write pros and cons.

* 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
  + Is a very realistic approach to software development Promotes teamwork and cross training.
  + Functionality can be developed rapidly and demonstrated. Resource requirements are minimum.
  + Suitable for fixed or changing requirements Delivers early partial working solutions.
  + Good model for environments that change steadily. Minimal rules, documentation easily employed.
  + Enables concurrent development and delivery within an overall planned context.
  + Little or no planning required easy to manage
  + Gives flexibility to developers
* Cons
  + Not suitable for handling complex dependencies.
  + More risk of sustainability, maintainability and extensibility.
  + An overall plan, an agile leader and agile PM practice is a must without which it will not work.
  + Strict delivery management dictates the scope, functionality to be delivered, and adjustments to meet the deadlines.
  + Depends heavily on customer interaction, so if customer is not clear, team can be driven in the wrong direction.
  + 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.

1. Draw Use-case on OTT platform.



1. Draw Use-case on E-commerce application.



1. Draw Use-case on Online shopping product using payment gateway.

