**SOFTWARE TESTING ASSIGNMENT**

**Module-1 (Fundamental)**

1. **What is SDLC?**

SDLC stands for Software Development Life Cycle. It gives a brief description of how a software is developed in different stages. It has 6 stages:

* + Requirements gathering
  + Analysis
  + Design
  + Implementation
  + Testing
  + Maintenance

1. **What is Software Testing?**

Software Testing is process that identifies correctness, completeness and quality of a developed software. It is also known as a process of evaluating a system or its components with the intent to find if specified requirement are satisfied or not.

It can also be stated as the process of validating and verifying that a software program or application or product:

* Meets the business and technical requirements that guided its design and development
* Works as expected
* Can be implemented with the same characteristic

1. **What is Agile Methodology?**

Agile Methodology or Agile Model is a type of SDLC Model.

Agile is a combination of Iterative and incremental process models. It focuses on process adaptability and customer satisfaction by rapid delivery of working software product.

Agile Method breaks the product into small incremental builds. These builds are provided in iterations.

Each iteration can take from one week up to three weeks to complete.

Each iteration involves cross functional teams working simultaneously on various areas like planning, Requirements Analysis, design, coding, unit testing and acceptance testing.

After every iteration a working product is displayed to the customer and important stakeholders.

1. **What is SRS?**

SRS stands for Software Requirement Specification.

It is a complete description of the behavior of the system to be developed. It includes a set of use cases that describe all of the interactions that the user will have with the software.

There are three types of SRS:

* Customer requirements
* Functional requirements
* Non-Functional requirements

1. **What is OOPS?**

OOP stands for Object Oriented Programming, that means it includes any language that easily supports object orientation.

1. **Write basic concepts of OOPS**

The basic concepts of OOPs are listed below:

* Object
* Class
* Encapsulation
* Abstraction
* Inheritance
* Polymorphism

1. **What is Object?**

Object is the basic unit of OOP. An object is anything to which both data and function that operate on data are bundled as a unit and has its own state and behavior

1. **What is Class?**

**A class is a collection of multiple objects. All the objects are defined in certain class(es).**

1. **What is encapsulation?**

Encapsulation means hiding data or binding data within class. It protects vital information from spreading.

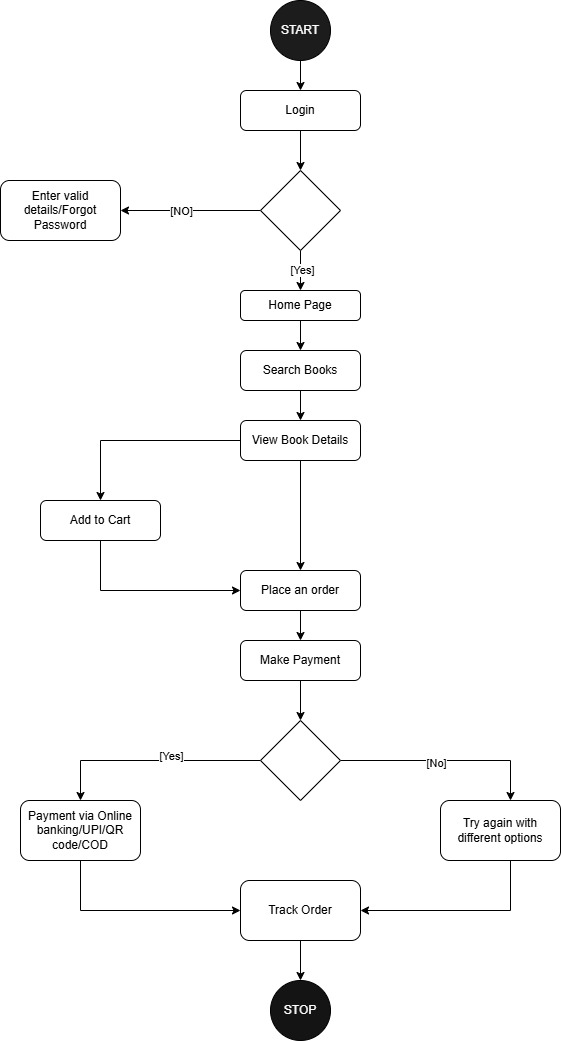
1. **What is inheritance?**

Inheritance allows a child class to access all the properties of its parent class.

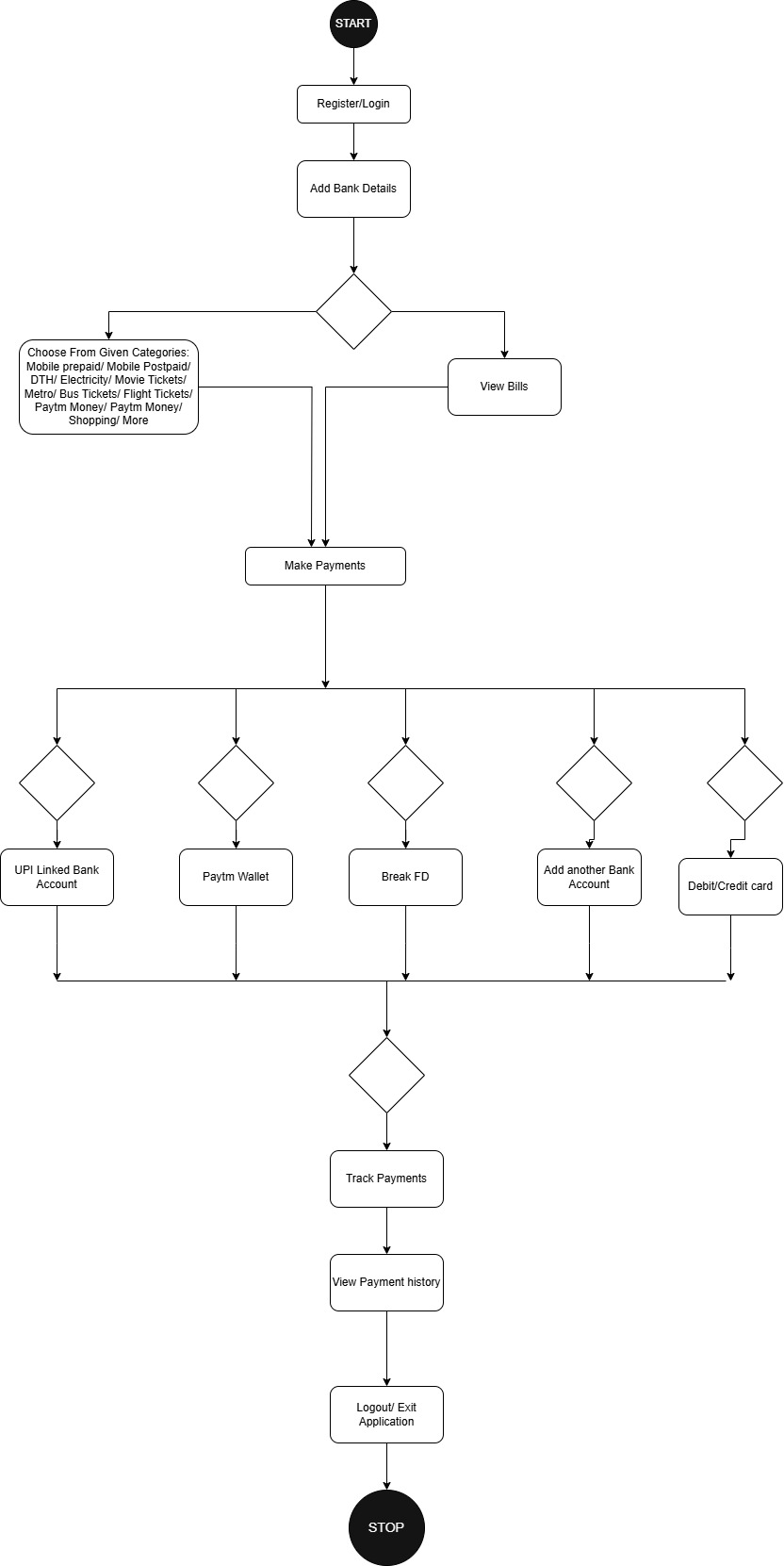
1. **What is polymorphism?**

Polymorphism allows an object to respond to the same message in different ways and also the ability to change forms.

1. **Draw use case on Online Book shopping**

****

1. **Draw use case on Online bill payment system (Paytm)**

****

1. **Write SDLC phases with basic introduction.**

SDLC has 6 phases. They are as follows:

* Requirement Gathering phase: In this phase, all the client’s requirements are specified and documented in written form. Requirements can change frequently. They can also be incomplete or complete, ambiguous or unambiguous, relevant or irrelevant. There are 2 types of requirements gathering:
  + - Functional requirements: They describe system services or functions.
    - Non-Functional requirements: They are constraints on the system or the development process.
* Analysis Phase: In this phase, all the requirements of the system are defined, independent of the accomplishment of these requirements. It has two phases: “WHAT” phase and “HOW” phase.
  + - WHAT phase: The clear and precise statements about what is to be built in documented form is called what phase.
    - HOW phase: This phase starts with the requirement document delivered by the requirement phase and maps the requirements into architecture. The deliverable design document is architecture defines the components, their interfaces and behaviors.
* Design: The designing team can prepare a model with the analyzed document. It also helps to convert the typical scenarios into a test plan.
* Implementation: In this phase, the model is used as a reference to create the desired software.
* Testing: Testing helps to validate the solution against the requirements. A few tests that might be done are: Regression Testing, Internal Testing, Unit Testing, etc.
* Maintenance: Software maintenance is the process of enhancing and optimizing deployed software, as well as fixing defects. It comes after deployment of the software into the field. The developing team will have some mechanism to document and track defects and deficiencies, configure the version management, reengineering all analysis, design and user documentation, repeated automated tests enable evolution and refactoring

1. **Explain phases of Waterfall model.**

Waterfall model will be used for short term projects where requirements are fixed. It’s unrealistic for reasons like:

> requirements must be frozen

> requirements are validated too late

It has following phases:

* Requirement gathering: Basically, all the requirements regarding the software are gathered from the customer.
* Analysis: The gathered requirements are then analyzed. The goal of the analysis part is to remove incompleteness and inconsistencies.
* Design: It converts the requirements acquired into a format that can be coded in a programming language. It includes high-level and detailed design as well as the overall software architecture
* Implementation: In this phase, the software design is translated into source code using any suitable programming language. Also, each designed module is coded.
* Testing: In this phase, the tester or the testing team puts the software on test. It is tested as a whole to ensure that it meets the requirements of the customer and is free from defects.
* Maintenance: It is the most important phase of a software life cycle. There are basically three types of maintenance:

• Corrective Maintenance: It is carried out to correct errors that were not discovered during the product development phase.

• Perfective Maintenance: It is carried out to enhance the functionalities of the system based on the customer’s request.

• Adaptive Maintenance: It is usually required for porting the software to work in a new environment such as working on a new computer platform or with a new operating system.

1. **Explain phases of spiral model.**

Spiral Model is very widely used in the software industry as it involves minimum risk for the customer as well as the development firms.

It follows a continuous cycle of few steps in order to deliver the best product that it can. The steps are as follows:

* Planning: In this phase, determination of objectives, alternatives and constraints is completed.
* Risk Analysis: This process involves pure analysis of alternatives and constraints for identification and resolution of risks.
* Engineering: The developers will be provided the model that will help in the development of the “next level” of the product.
* Customer Evaluation: During this process, the customer is showed the developed product and the assessment of the results of engineering is done.

And it keeps on repeating till the desired product with best features are made.

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

Agile Model is a combination of iterative and incremental process models. It focuses on process adaptability and customer satisfaction by rapid delivery of working software product.

Agile Method breaks the product into small incremental builds. These builds are provided in iterations.

Each iteration can take from one week up to three weeks to complete.

Each iteration involves cross functional teams working simultaneously on various areas like planning, Requirements Analysis, design, coding, unit testing and acceptance testing.

After every iteration a working product is displayed to the customer and important stakeholders.



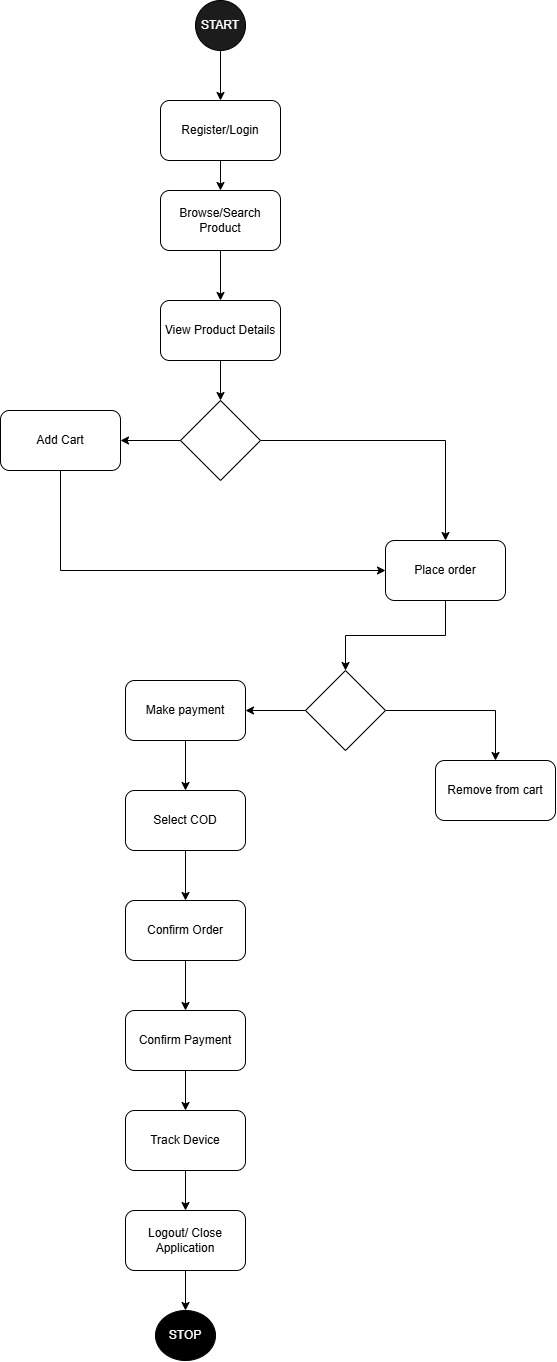
**Pros:-**

* It has a very realistic approach towards Software Development.
* It improves the functionality development and demonstration.
* It is quite suitable for fixed or hanging requirements.
* It is easy to manage as little or planning is required which gives flexibility to developers.

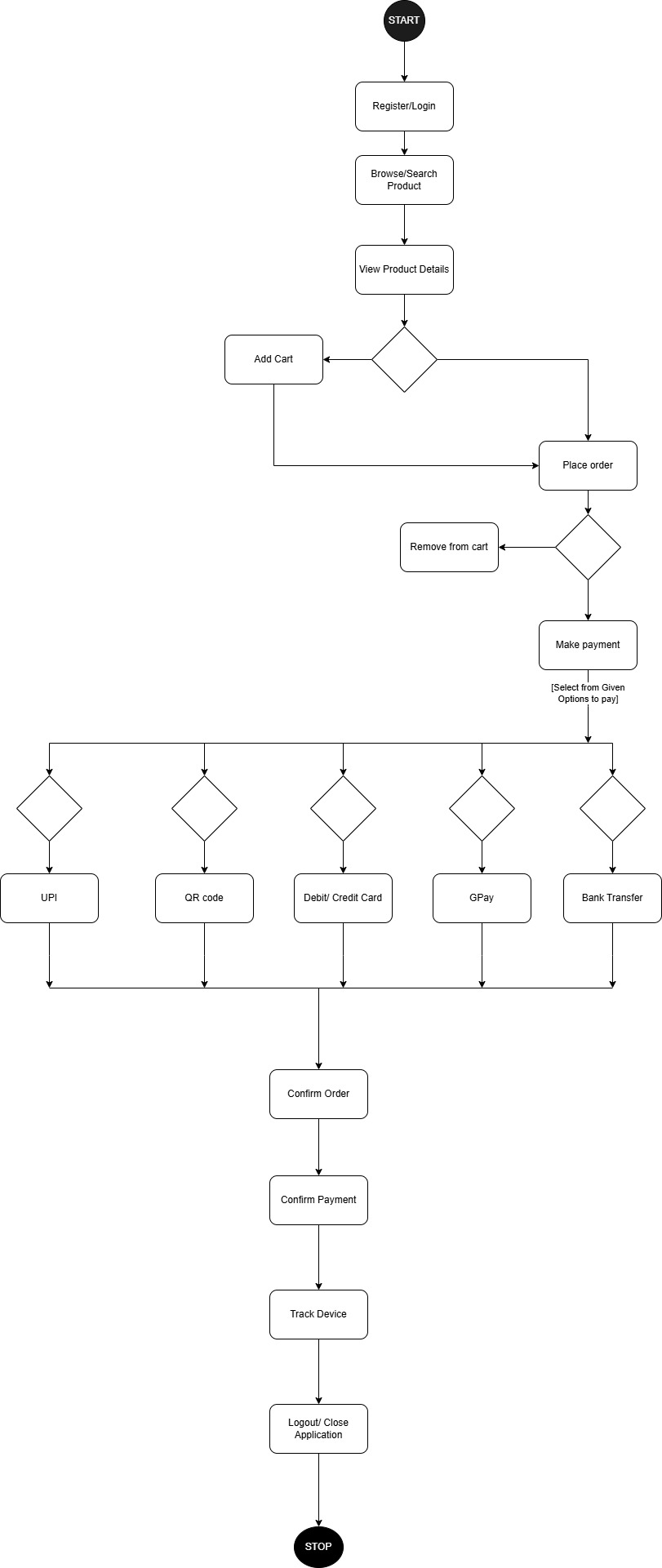
**Cons:-**

* It is not suitable for handling complex dependencies
* It also has risk of sustainability, maintainability and extensibility.
* Transfer of technology to new team members can be challenging due to lack of documentation.
* It heavily depends on the customer’s interaction and if he isn’t clear then the team can be driven in wrong direction.

1. **Draw use case on Online shopping product using COD.**

****

1. **Draw use case on Online shopping product using payment gateway.**

****