

ASSIGNMENT - 1

Module -1

1. What is SDLC

- SDLC stands for Software Development Life Cycle
- SDLC structured imposed on the development of software product
- It defines the process of planning implementation, testing Documentation, deployment and maintenance.
- A software development lifecycle is essentially a series of steps or phases that provide a model for the development and lifecycle management of an application of software.
- The software development lifecycle is the cost-effective and time-efficient process that development terms use design and high quality software.

2.What is Software Testing?

- Software Testing is Processes used to identify the correctness, completeness, quality of developed computer software.
- Software testing is the process of finding errors in the developed product.
- It also checks the real outcomes can match expected results, defect, missing requirement or gaps.

- Testing can be defined as a process of analysing software items to detect the difference between existing and required (that is defect/error/bugs).

3. What is agile methodology?

- Agile SDLC model is combination of iterative and incremental process models with focus process adaptability and customer satisfaction by rapid delivery of working software product.
- Agile methods breaks the product into small incremental builds.
- These builds are provided in iterations.
- Each iteration typically lasts from about 1 to 3 weeks.
- Every iteration involves cross functional teams working on various areas like planning, requirement analysis, design, coding, unit testing and acceptance testing.
- At the end of the iterations, a working product is displayed to the customer and important stakeholders.

4. What is SRS

- SRS stands for software requirement specification
- It includes a set of use cases that describe all of the interactions that users will have with the software.
- Use cases are also known as functional requirements which impose constraints on design or implementation (such as performance requirements, quality standards or design)

- These standards describes possible structures and qualities of software requirement specifications.
- Requirements are categorized in several ways
 - Customer Requirement
 - Functional Requirement
 - Non-Functional Requirement

5. What is OOPS

- OOPS stands for Object Oriented Programming Language
- OOPS is a programming technique in which program are
 - C++
 - PHP
 - Java
- An object based programming language is one which easily supports object-orientation.
- Object oriented programming style of programming that focus on using objects to design and build application.

6. Write Basis concepts of oops

1. Object : Any entity which has own state

Eg: Pen, Paper

2. Class : Class is a collection of object

Eg: human body

3. Abstraction : Hiding internal details and showing functionalities

Eg: login Page

4. Encapsulation : wrapping up of data or binding of data

Eg : Capsual

5. **Inheritance** : When one object acquire all the properties and behaviour of parent class

Eg : Father-son

6. **Polymorphism** : The ability to change from one form to another known as polymorphism

7. What is Object

- An object represents an individual , identifiable item , unit or entity , either real or abstract with a well-defined role in the problem domain.
- That is both data and function that operates

8. What is class

- Class is a collection of data members and member functions.
- A class represents and abstraction of the object and abstracts the properties and behaviour of that object.
- An object is a particular instance of class which has actual existence and there can be many objects for a class

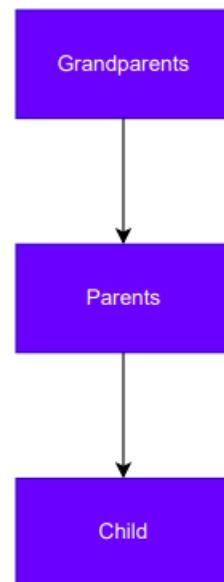
9. What is Encapsulation

- Encapsulation is the practice of including in an object everything it needs from other objects.
- The internal state is usually not accessible by other objects.

- Encapsulation is placing the data and the function that work on that data in the same place.
- Encapsulation enables data hiding, hiding irrelevant information from the users of the class and exposing only the relevant details required by the users.

10.What is Inheritance

- Inheritance means that one class inherits the characteristics of another class. This is a relationship.
- Inheritance describes the relationship between two classes.
- A class can get some of its characteristics from a parent class and then add unique features of its own.

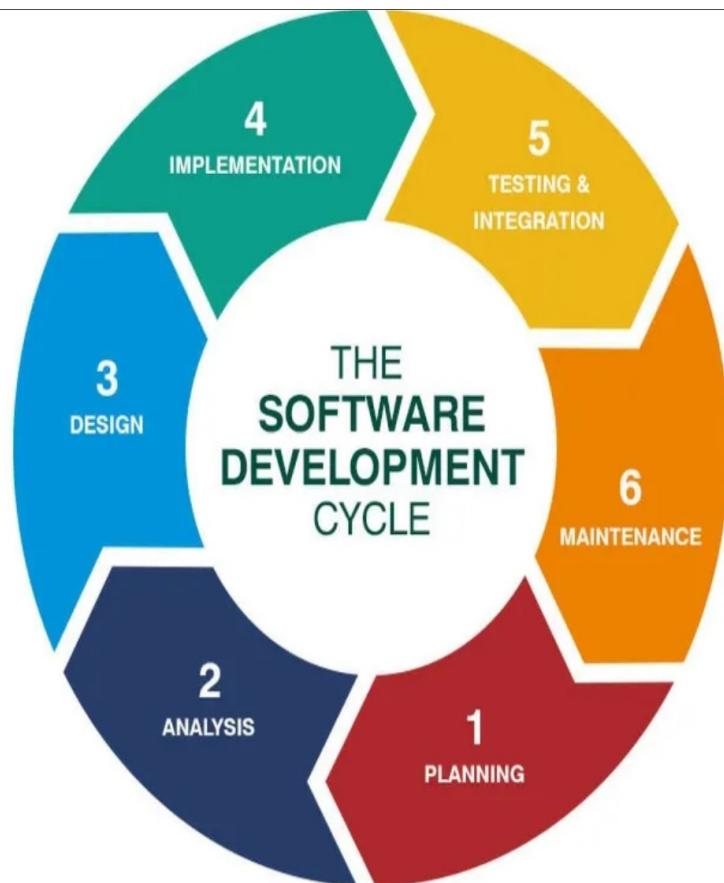


11.What is Polymorphism

- Polymorphism means having many forms.
- It allows different objects to the same message in different ways. The response specific to the type of the object.
- The ability to use an operator or function in different ways in other words giving different meaning or function to the operators or function is called polymorphism.
- The ability to change form is known as polymorphism.
- There are two types of polymorphism.
 - 1.Compile time polymorphism(Overloading)
 - 2.Runtime time polymorphism(Overriding)

12.What is SDLC phases with basic Introduction

- There are six phases of SDLC



1) Requirement Gathering

- Requirement Analysis is a process used to determine the needs and expectations of a new product.
- It involves frequent communication with stakeholders and end- users of the product to define expectations , resolve , conflict and document all key requirement.

2) Analysis

- The analysis phase defines the requirement of the system, independent of these requirement will be accomplished
- This phase defines the problem that the customer is trying to solve.

3) Design

- The design phase of sdlc is a critical step in developing the conceptual of software project.
- The design phase is stage of software developers of the product depending on the project, these can include screen designs, databases, sketches, system interfaces and prototypes.
- Client use the these details to make final product design choices.

4) Implementation

- This phase is initiated after the system has been tested and accepted by the user.
- The implementation phase deals with issues of quality, performance, baselines and debugging.
- The end deliverable is the product itself. There are already many established techniques associated with implementation.

5) Testing

- The testing phase is a different team after the implementation is completed.
- The developer builds the software then it is deployed in the testing environment.
- Then the testing team tests the functionality of the entire system, the testing is done to ensure that the entire application works according to the customer requirements.

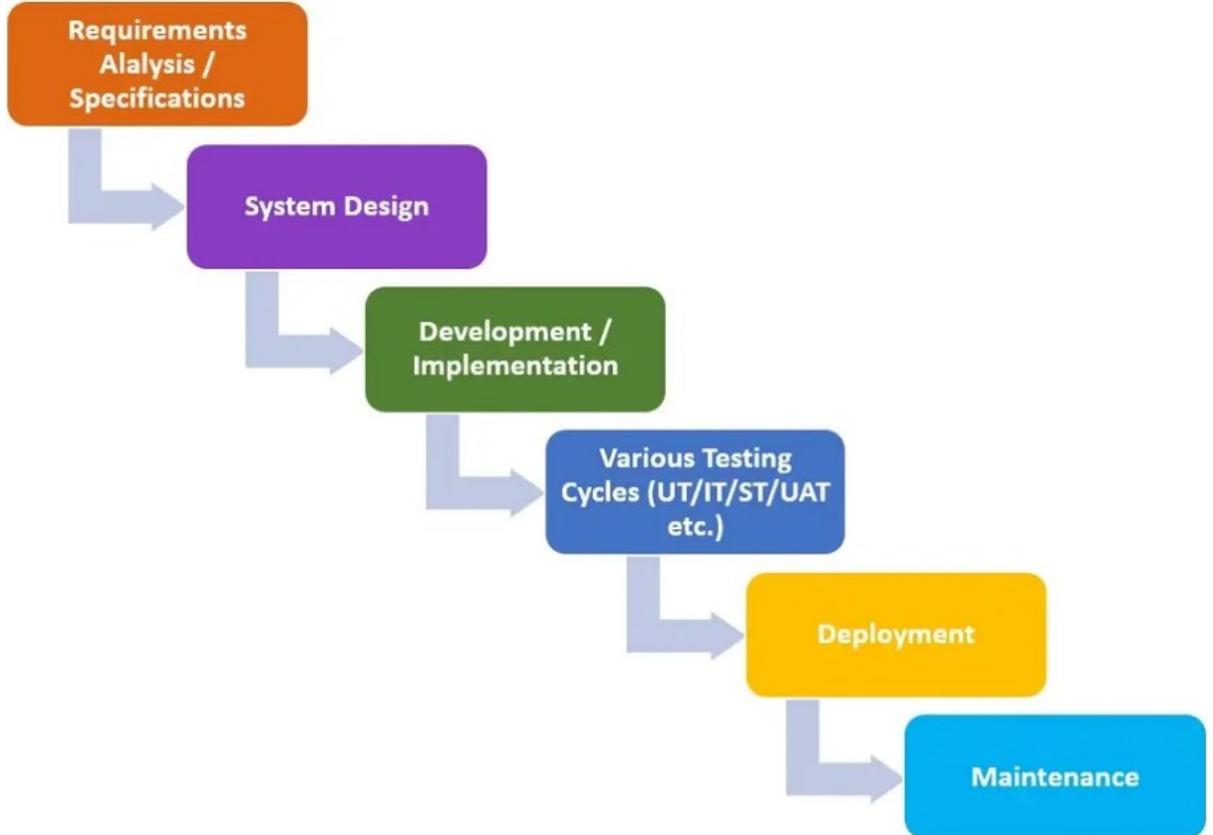
6) Maintenance

- Software maintenance is one of the activities in software engineering and is the process of enhancing and optimizing deployed software as well as fixing defects.
- Software maintenance is also one of the phases in the system development lifecycle, it applies to software development.
- The maintenance phase is the phase which comes after deployment of the software into the field.

- There are three types in maintenance
 - 1. Correctiveness maintenance
 - 2. Adaptive maintenance
 - 3. Perfective maintenance

13. Explain phases of waterfall model

- The classical software Lifecycle models the software development as a step-by-step “waterfall” between the various development phases.
- The waterfall is unrealistic for many reasons especially.



Requirement Analysis

- The collect complete needs and then analyse and define needs that must be met by the program to be built
- In this phase all requirement of the project are analysed and document I am specification document and a feasibility analysis is done to check if these requirement are valid.

System Design

- The goal of this phase is a convert the requirement acquired in the SRS 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.
- A software design document is used to document all of this effort (SDD)

Implementation

- The implementation phases is when programmers assimilates the requiarmant and specification from the previous phase and produce actual code

Testing

- Testing is a type pf software testing in which the different testing levels are performed one after the other.
- Testing is waterfall development is sequential and through, consisting of unit testing, integration testing.

Deployment

- Once the function and non-functional testing is done; the product is deployed in the customer enviroment or released into the market.

Maintenance

- Maintenance is the most important phase of a software life cycle.

- The effort spent on maintenance is 60% of the total effort spent to develop a full software.

14. Write Phases of Spiral Model

- The spiral model is one of most important software development life cycle models,which provides support for risk handling.
- The exact number of loop of the spiral is known and can vary from project to project.
- Each loop of the spiral is called a phase of the software development process.s
- The spiral model is a software development life cycle (SDLC) model that provides a systematic and iterative approach to software development.
- The spiral model is a risk-driven model,meaning that the focus is on managing risk through multiple iterations of the software development process.



- 1. Planning** : The First phase of the spiral model is the planning phase, where the scope of the project is determined and a plan is created for the next iteration of the spiral.
- 2. Risk Analysis** : In Risk Analysis phase, the risks associated with the project are identified and evaluated.
- 3. Evaluation** : In the Evaluation phase, the software is evaluated to determine if it meets the customers requirement and if it is of high quality.
- 4. Engineering** : In the engineering phase, the software is developed based on the requirement gathered in the previous iteration.

15. Explain Working Methodology of Agile model and also write prons and cons.

- 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.
- 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.

Prons-

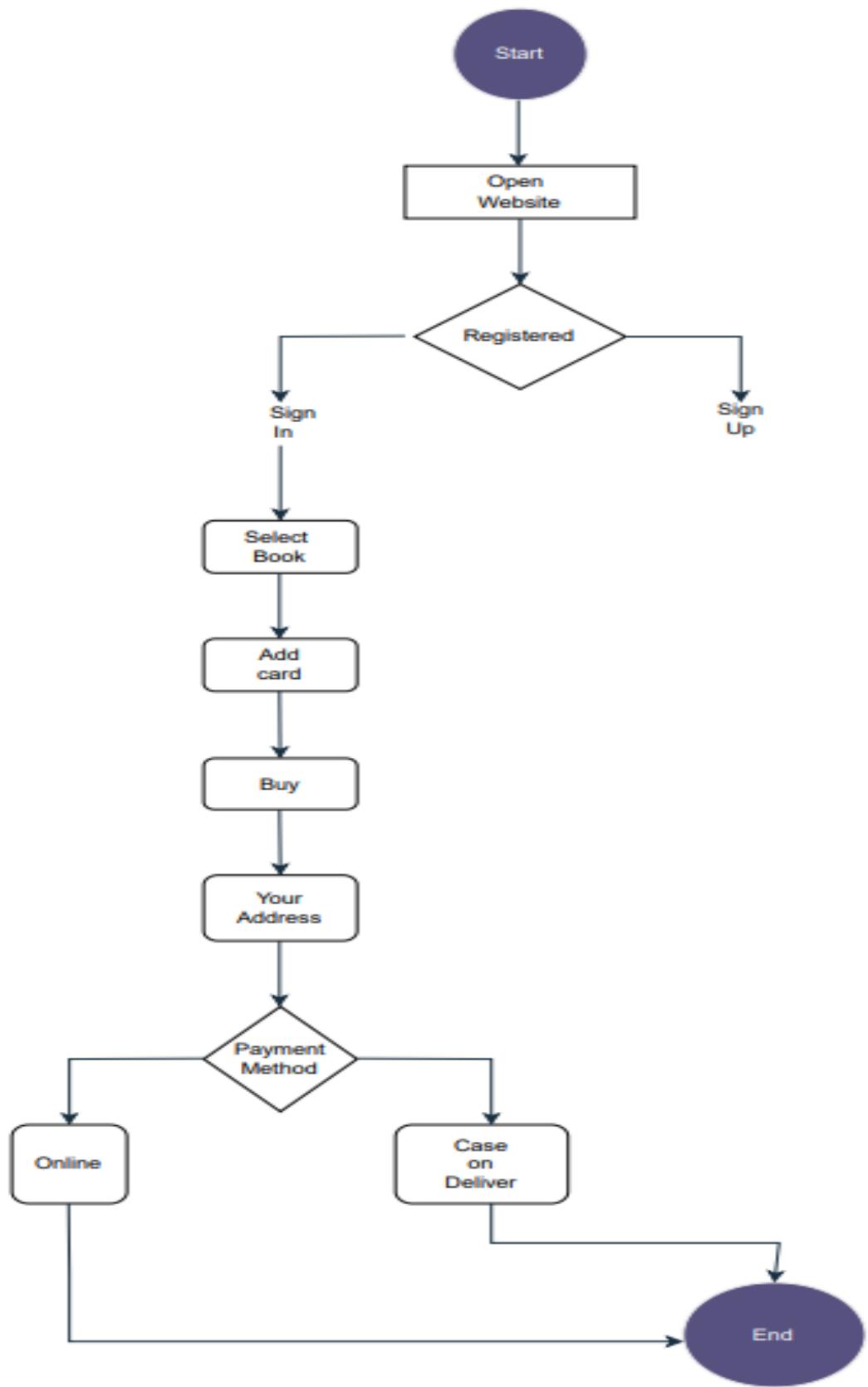
- Is a very realistic approach to software development.
- Promotes teamwork and cross training.
- Resource requirements are minimum.
- Suitable for fixed or changing requirements
- Delivers early partial working solutions.
- Little or no planning required.

- Easy to manage.
- Gives flexibility to developers.

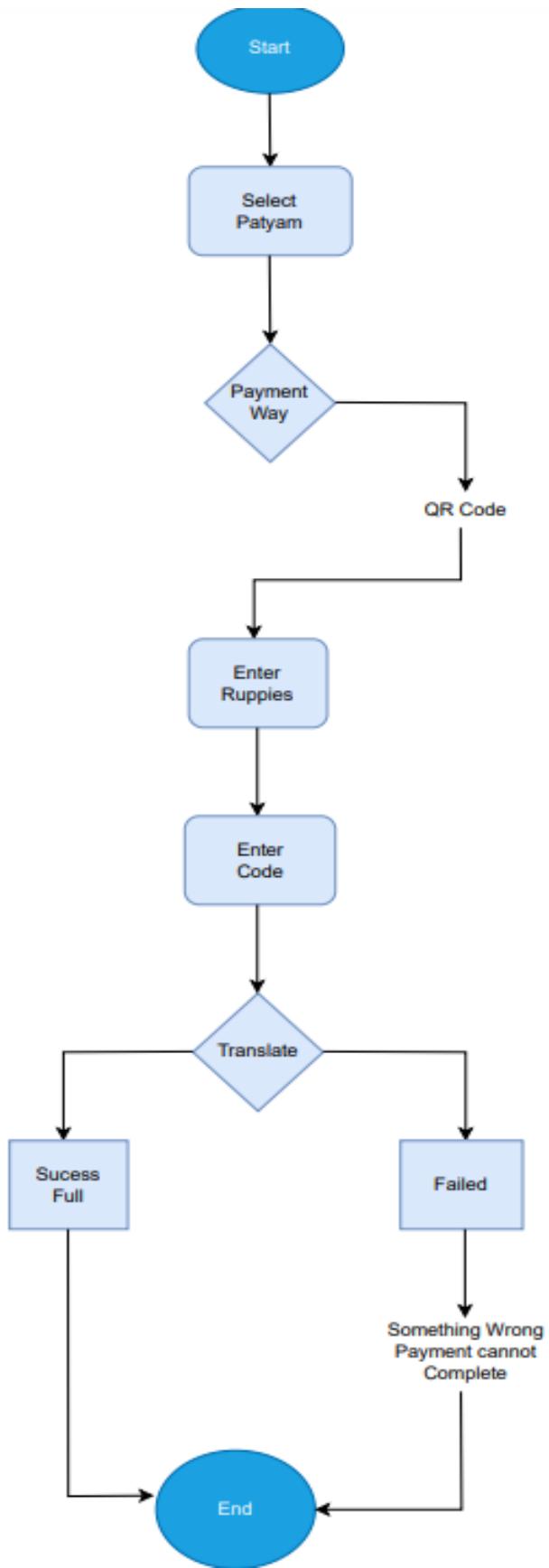
Cons-

- 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.

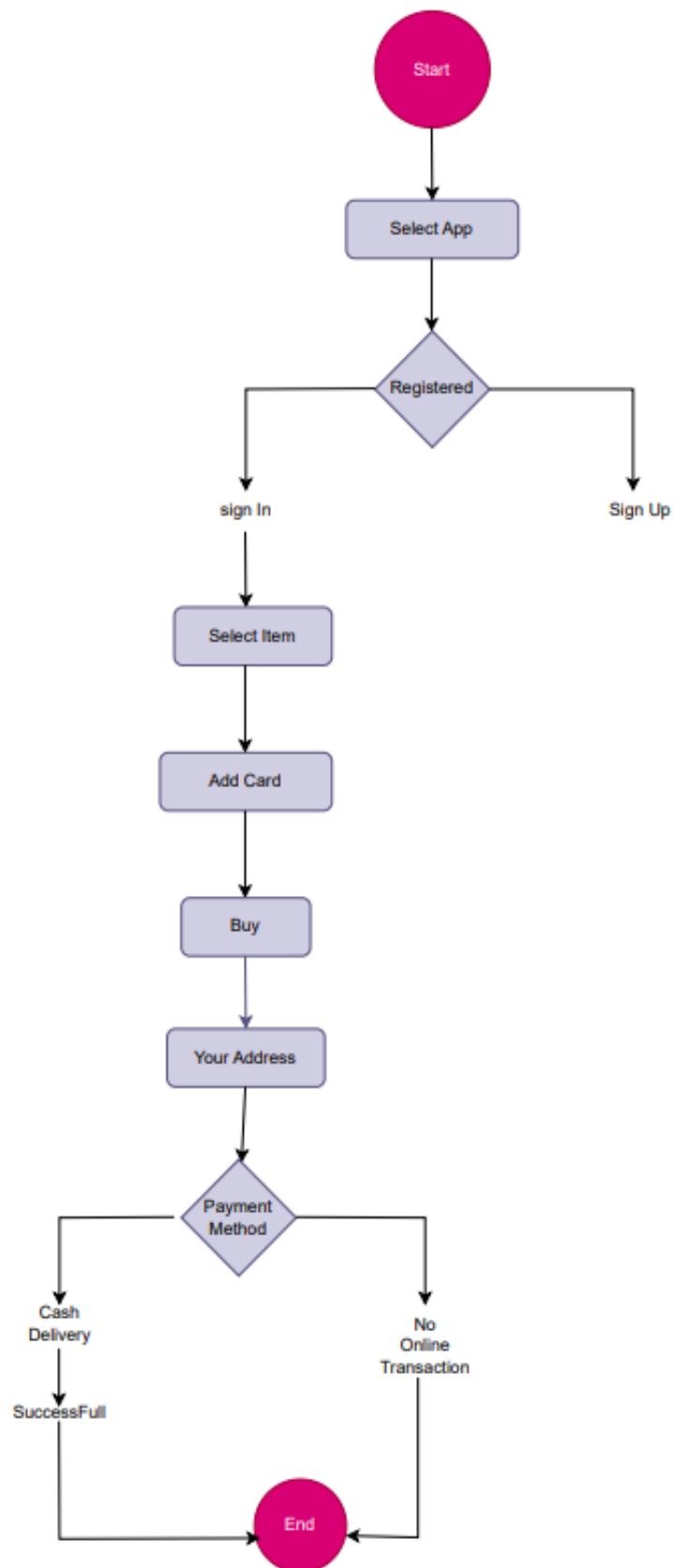
16. Draw Usecase on Online book Shopping.



17. Draw Usecase On Online bill payment system.



18. Draw Usecase On Online Shopping product using COD.



19. Draw Usecase On Online Shopping using payment gateway.

