Diploma in Software Testing

AN ASSIGNMENT ON

Course Tital: Software Testing Manual
Module -1 (fundamental)

Submitted To:
Rakesh Kharva
Technical Trainer
Tops Technologies Pvt.Ltd.

Submitted By:
Kacchadiya Dhara Mitesh

Date of submission : 29 March, 2023

Modul-1 (Fundamentals)

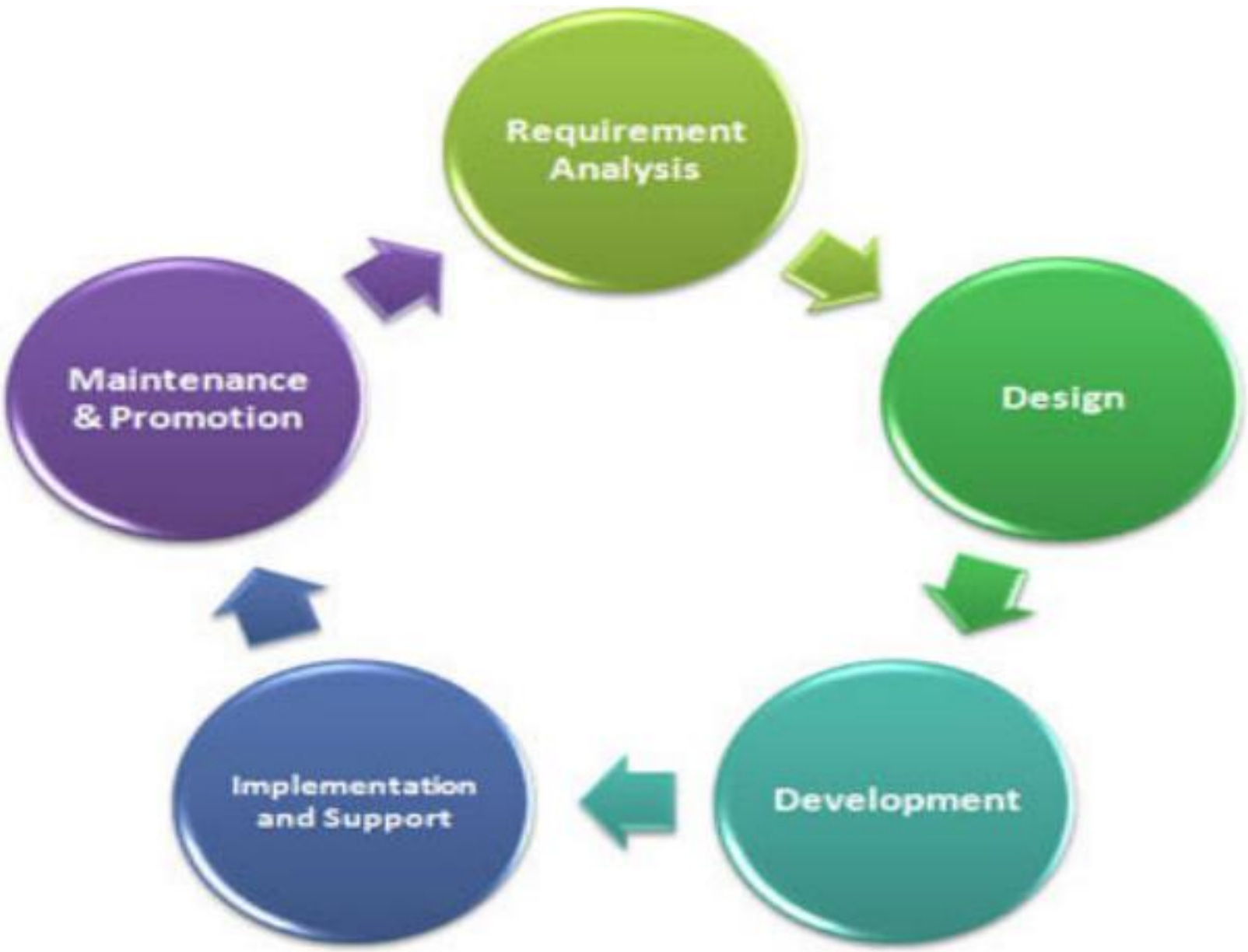
1. What is SDLC?
2. What is Software Testing?
3. What is Agile Methodology?
4. What is SRS?
5. What is OOPS?
6. What is basic concept OOPs
7. What is Object
8. What is Class
9. What is encapsulation
10. What is Inheritance
11. What is Polymorphism
12. Draw Usecase on online book shopping
13. Draw Usecase on online bill payment system (paytm)
14. Write SDLC phases with basic introduction
15. Explain phases of Water fall model
16. Write phases of spiral model
17. Write Agile manifesto principles
18. Explain working methodology of agile model and also write pros and cons.
19. Draw use-case on online shopping product using COD.
20. Draw use-case on online shopping product using-payment gateway.

What is SDLC?

SDLC full form of Software Development Life Cycle.

Dafination : A Software Development Life Cycle is essentially a series of steps, or phases, that provide a model for the development and lifecycl management of an application or piece of software.

Software Development Life Cycle



Project Methodology

Requirement Gathering :

Types of Requirements:

- **Functional Requirements:** describe system services or

Functions Compute sales tax on a purchase. Update the database on the server

- **Non-Functional Requirements: are constraints on the system or**

the development proces

Analysis Phase :

The analysis phase defines the requirements of the system, independent of how these requirements will be accomplished.

This analysis represents the “what” phase.

The requirement documentaries to capture the requirements from the customer's perspective by defining goals.

Design Phase :

- Design Architecture Document
- Implementation Plan
- Critical Priority Analysis
- Performance Analysis
- Test Plan

Implementation Phase :

- In the implementation phase, the team builds the components either from scratch or by composition.
- For example, a component may be narrowly designed for this particular
- system, or the component may be made more general to satisfy a reusability guidelines
- .Implementation - Code
- Critical Error Removal

❖ Testing Phase :

❖ Simply stated, quality is very important. Many companies have not learned that quality is important and deliver more claimed functionality but at a lower quality level
❖ .Regression Testing
❖ Internal Testing
❖ Unit Testing
❖ Application Testing

❖ Maintenance Phase :

❖ Software maintenance is one of the activities in software engineering, and is the process of enhancing and optimizing deployed software (software release), as well as fixing defects.

❖ Software maintenance is also one of the phases in the System Development Life Cycle (SDLC), as it applies to software development. The maintenance phase is the phase which comes after deployment of the software into the field.

What is Agile Methodology?

Dafination:

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 model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project. requirements. In agile the tasks are divided to time boxes (small time frames) to deliver specific features for a release.

Write Agile manifesto Principles:

The **Agile Manifesto** is comprised of four foundational values and 12 supporting principles which lead the **Agile approach** to software development.

The four core values of Agile software development as stated by the Agile Manifesto are:

- individuals and interactions over processes and tools;
- working software over comprehensive documentation;
- customer collaboration over contract negotiation; and
- responding to change over following a plan.

Working Methodology of Agile Model:

The Agile methodology is **a way to manage a project by breaking it up into several phases**. It involves constant collaboration with stakeholders and continuous improvement at every stage. Once the work begins, teams cycle through a process of planning, executing, and evaluating.

Advantages:

- ❖ Is very realistic approach to the software development.
- ❖ Prompts teamwork and cross training
- ❖ Functionality can be developed rapidly and demonstrated.
- ❖ Recourse requirements are minimum.
- ❖ Suitable for fixed and changing requirements.
- ❖ Easy to manage.
- ❖ Gives flexibility to developers.

Disadvantages:

- ❖ Not suitable for handling complex dependencies
- ❖ More risk of sustainability maintainability and extensibility
- ❖ Depends heavily on customer interaction, so if customer is not clear, team can be driven wrong direction.

What is SRS?

✧ **SRS full-form of Software Requirements Specification**

A software requirements specification (SRS) 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 users will have with the software.

Types of SRS

✧ Customer Requirements
✧ Functional Requirements
✧ No n-functional Requirements

What is OOPS

✧ **OOPS means Object Orientated Programming**

❖ Object-oriented programming (OOP) is a computer programming model that organizes software design around data, or objects, rather than functions and logic. An object can be defined as a data field that has unique attributes and behavior.

❖ Identifying objects and assigning responsibilities to these objects.

❖ Objects communicate to other objects by sending messages.

❖ Messages are received by the methods of an object

❖ An object is like a black box.

❖ The internal details are hidden.

Basic concept of OOPS :

❖ Programming is like writing.

❖ If you can write a demonstration, you can make a programming.

❖ So is also easy.

❖ But, actually, programming is not so easy, because a real good program is not easily programmed. It needs the programmers' lots of wisdom, lots of knowledge about programming and lots of experience.

What is object?

❖ An "object" is anything to which a concept applies.

❖ This is the basic unit of object oriented programming(OOP).

❖ That is both data and function that operate on data are bundled as a unit called as object.

The two part of an object:

Object = Data + Methods

or

to say the same differently

An object has the responsibility to know and the responsibility to do.

What is Class? :

- ❖ When you define a class, you define a blueprint for an object.
- ❖ A class represents an abstraction of the object and abstracts the properties and behavior of that object.
- ❖ An object is a particular instance of a class which has actual existence and there can be many objects (or instances) for a class.

Making Classes: Creating, extending or reusing abstract data types.

Making Objects interact: Creating objects from abstract data types and defining their relationships.

What is Encapsulation:

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

It is wrapping up of data into a single unit.

Example: Capsule

- Class and Object.

Class: It contains Data members and member functions with some behavioral change.

Object: It is instance of a class.

E.g.: paper cup

What is inheritance:

Inheritance means that one class inherits the characteristics of another class. This is also called a “is a” relationship.

It is acquiring the property of base class/parent class into the deliver class/child class.

Types of Inheritance : There are five types

1) Single- level Inheritance
2) Multi-level Inheritance
3) Multiple- level Inheritance
4) Hierarchical- level Inheritance
5) Hybrid -Level Inheritance

What is Polymorphism:

Poly - > Many
Morphism -> Forms
It is combination of many forms
There are 2 types of Polymorphism

1)**Compile** - Time Polymorphism
e.g. Method Overloading

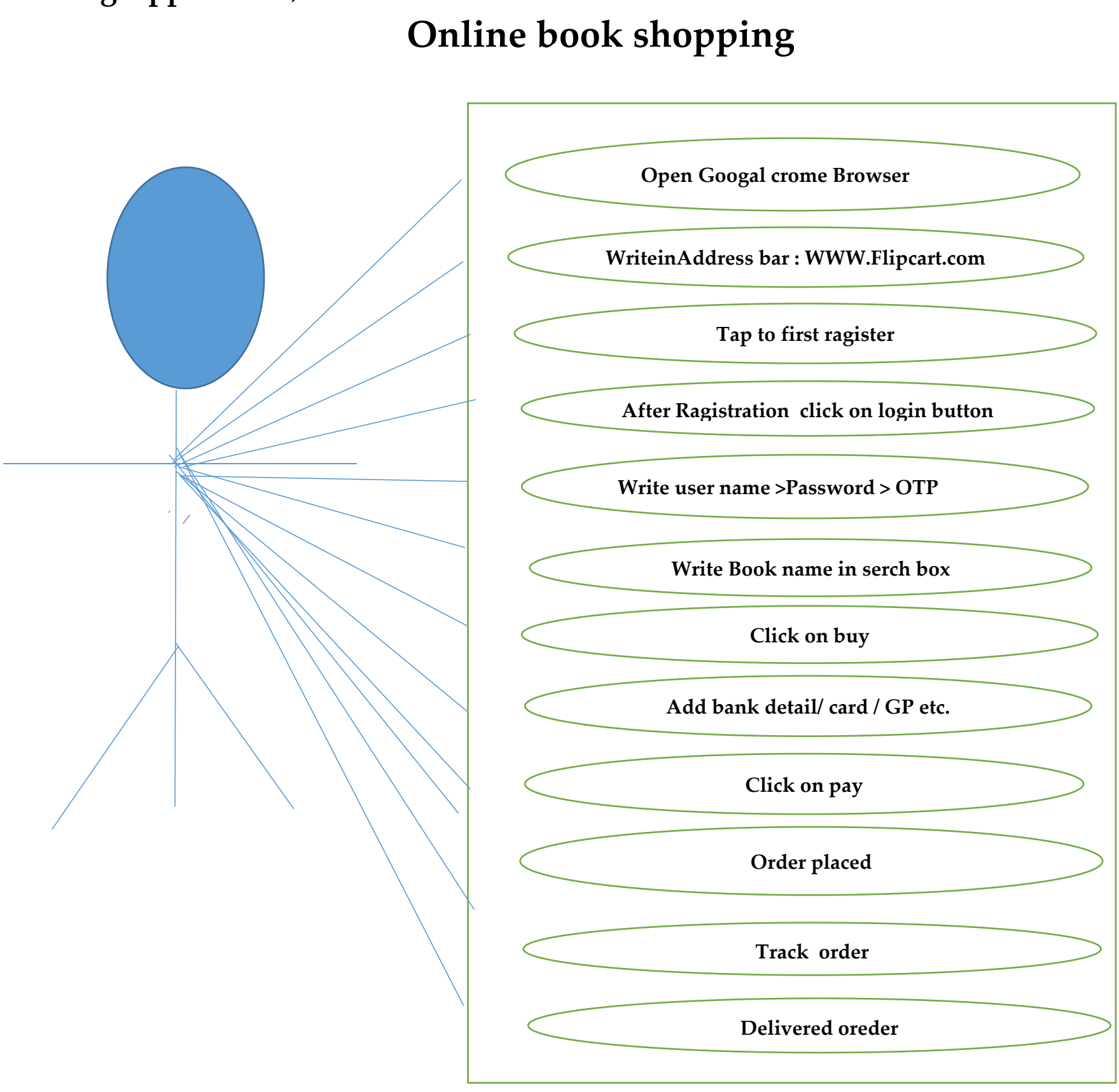
1) **Run -** Time polymorphism
e.g. Method overriding

Draw Usecase on online book shopping :

Usecase :

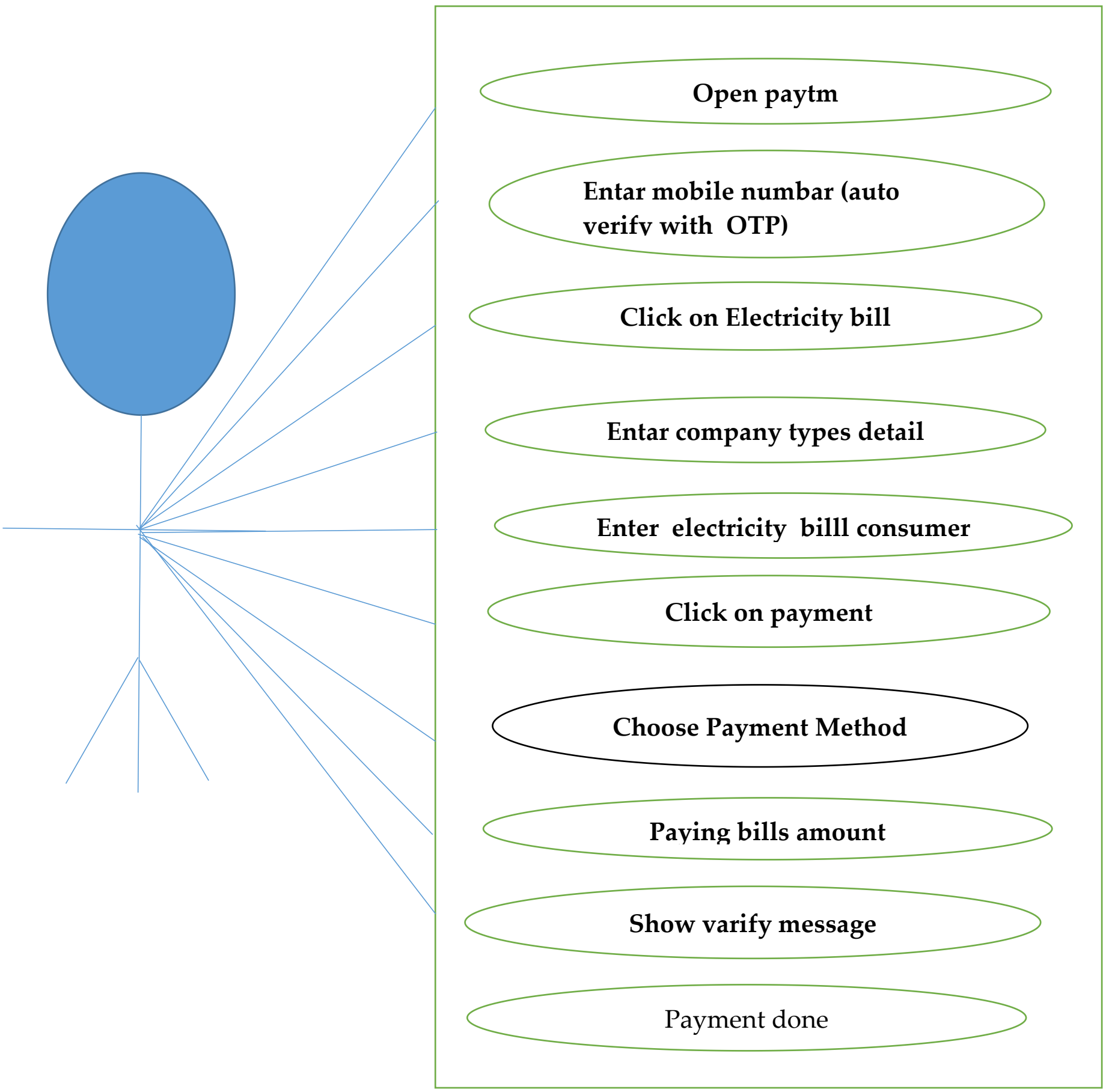
A use-case is the specification of a sequence of actions, including variants, that a system (or other entity) can perform,interacting with actors of the system.

Note :in this use case I m using website for buy a BOOK on flip-cart Com.(Not using Application) for new or first time user



Draw Usecase on online bill payment system (paytm)

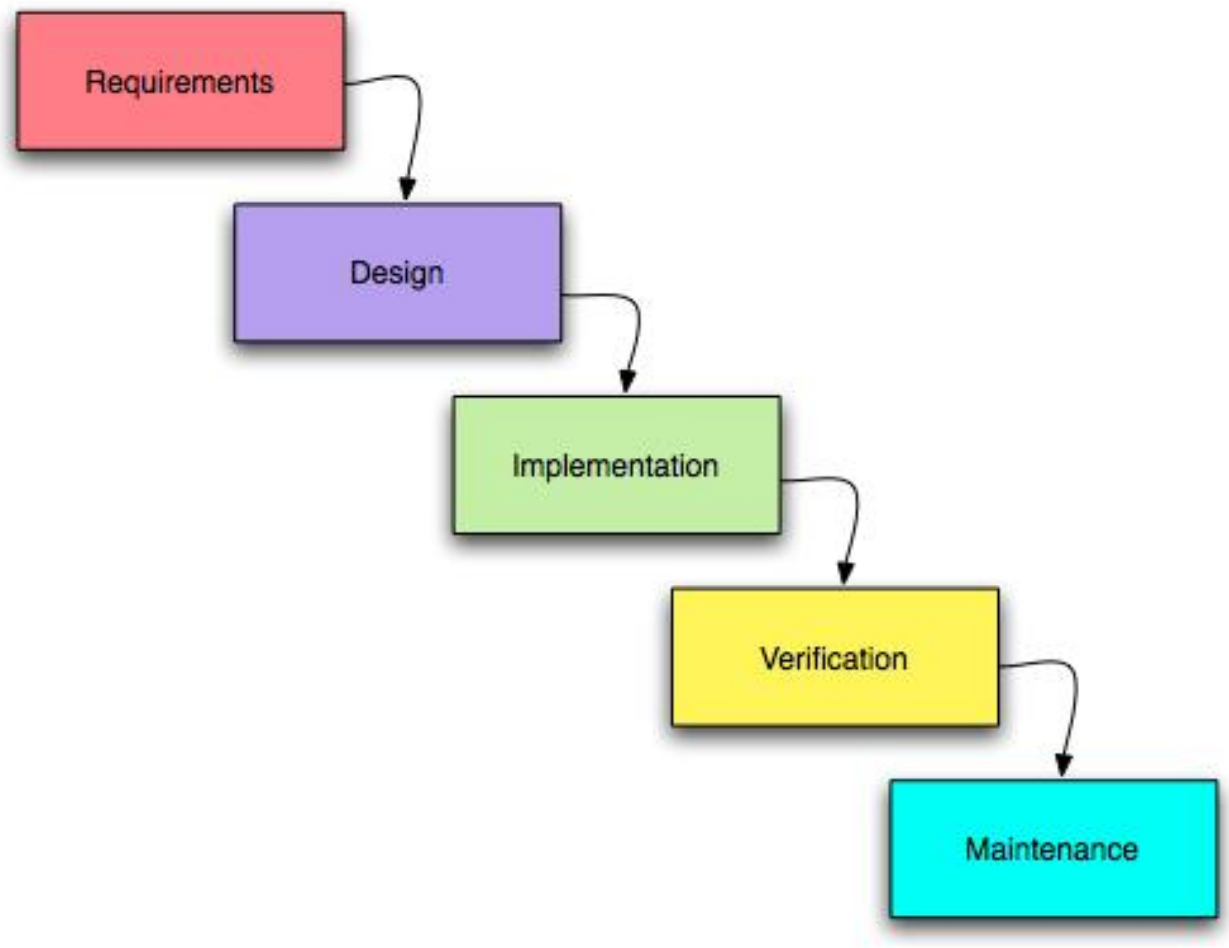
Note: Electricity bill using paytm application



Explain Phase of Water fall model:

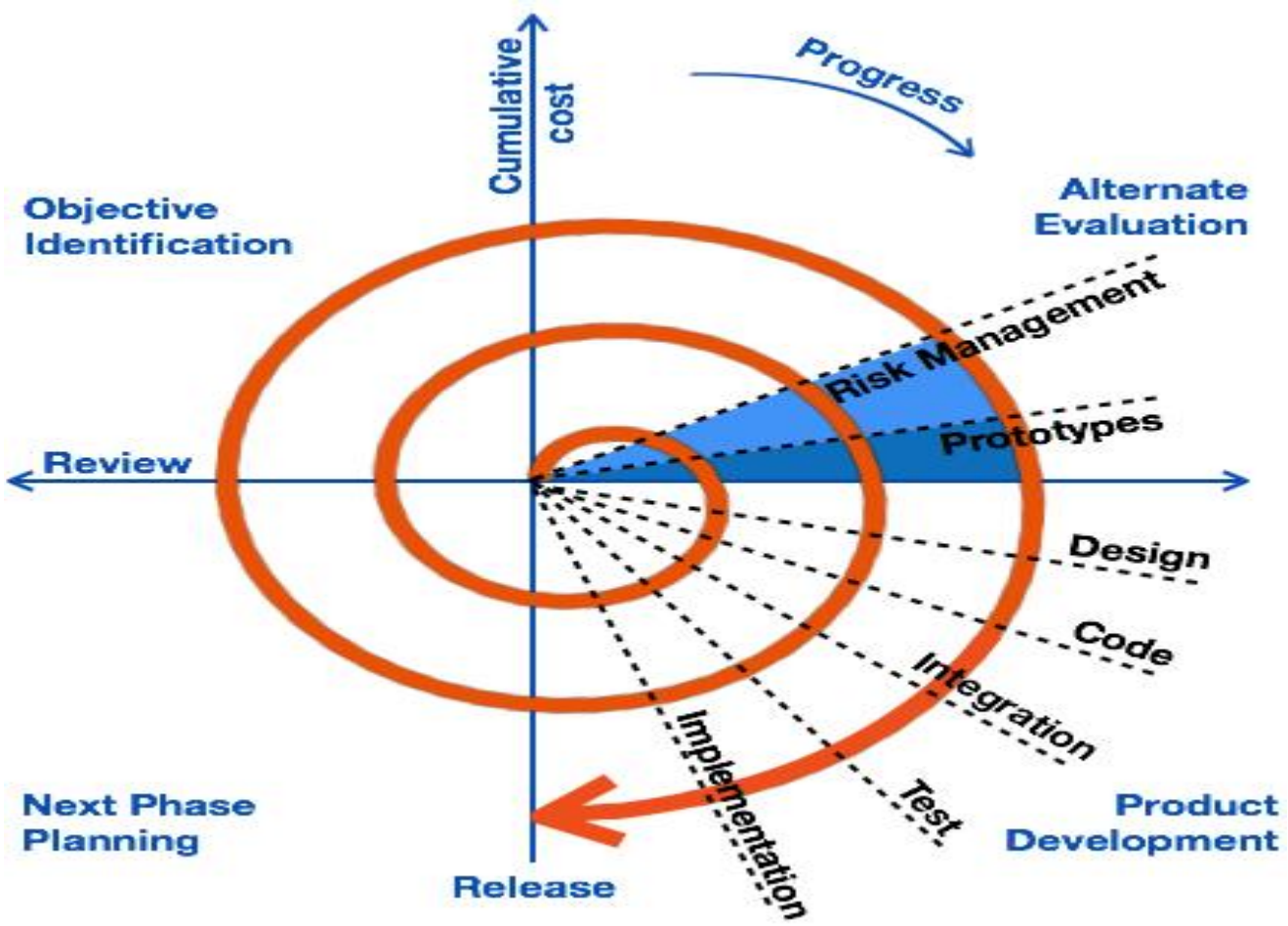
Dafination :

The classical software life-cycle models the software development as a step by step " Water fall "between the various development phases.

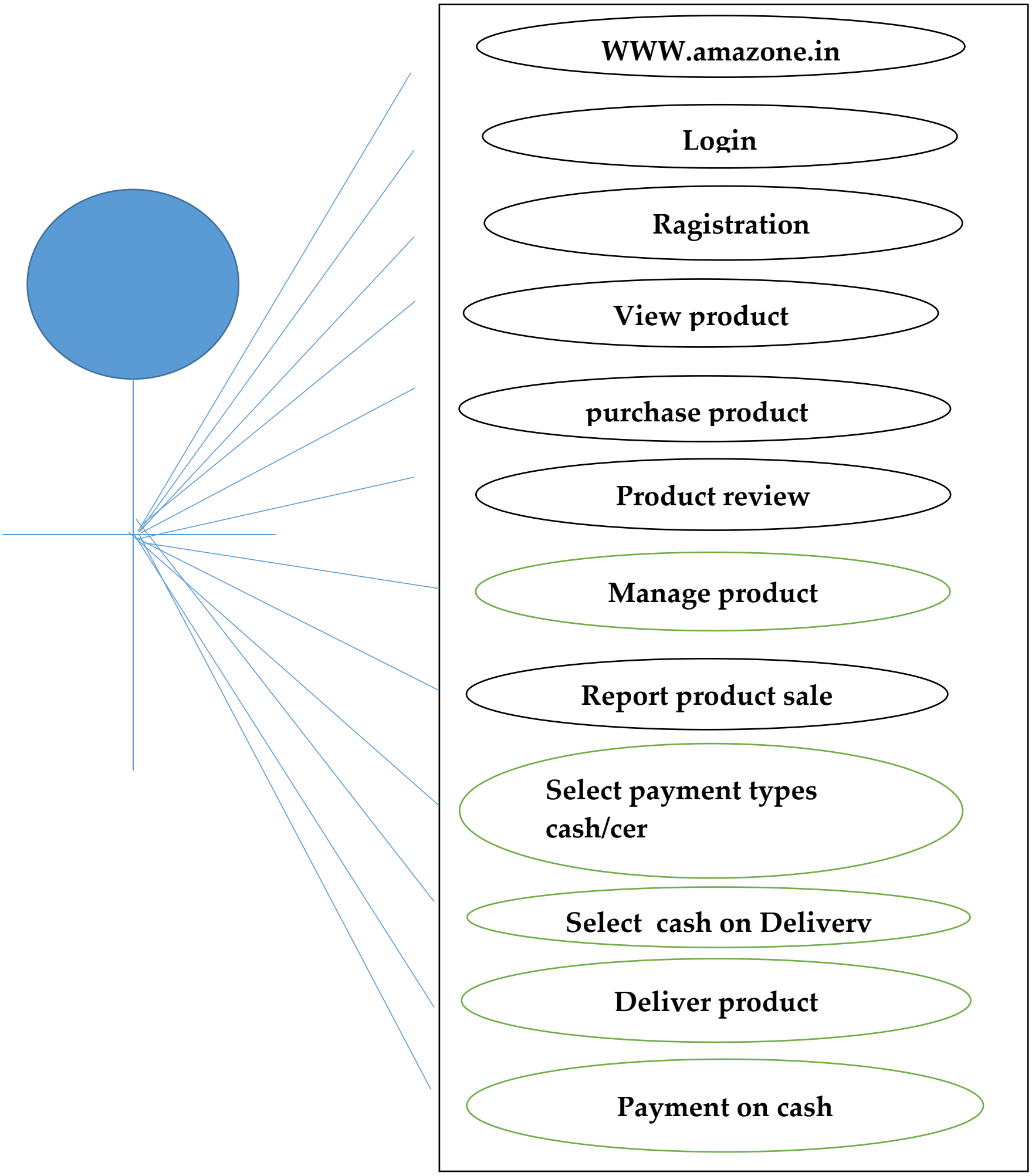


Write Phase of Bohem''s Spiral Model:

Spiral Model is very widely used in the software industry as it is in sync with the natural development process of any product i.e. learning with maturity and also involves minimum risk for the customer as well as the development firms. Following are the typical uses of Spiral model.



Draw Usecase on Online shopping product using COD:


WWW.amazone.in
Login
Ragistration
View product
purchase product
Product review
Manage product
Report product sale
Select payment types
cash/cer
Select cash on Deliverv
Deliver product
Payment on cash

Draw Usecase on online shopping product using payment gateway