Assignment - Module-1

1] What is SDLC?

- ✓ SDLC is an essential approach or step by step approach used to define software requirements, analysis, design, Implementation/ coding, testing and deployments/Maintenance for the development of the software product.
- ✓ It is a kind of series of steps which provides the various phases to produce software with high quality, lowest cost in the shortest possible time. It is a methodology to ensure the code quality at each phase of the cycle.
- ✓ SDLC methodology focuses on the following phases of software development.
  - Requirements collection/ Gathering/ Planning
  - Analysis
  - Design
  - Implementation/ Coding
  - Testing
  - Maintenance/Deployment

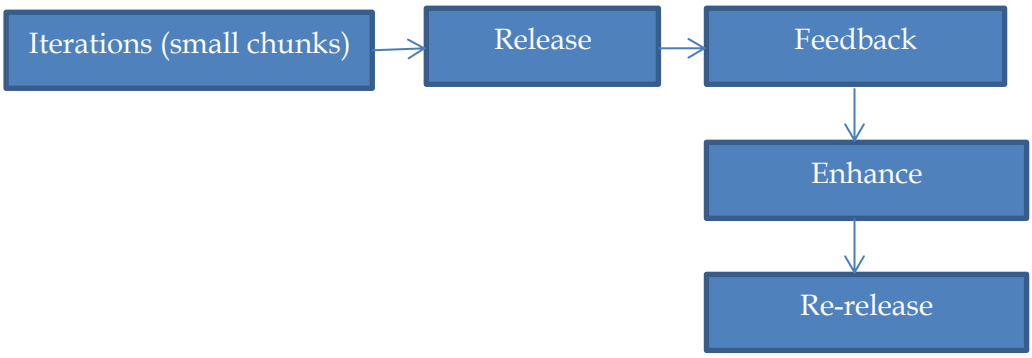
2] What is Software Testing?

- ✓ Software testing is a process used to identify the correctness, completeness and quality of developed computer software.
- ✓ Software testing is the process to evaluate a system or its components to find whether the specified requirements are satisfied or not.
- ✓ The purpose of software testing is to evaluate end to end system specifications. There should not be any gap or difference between the specified requirements and the actual requirements.

3] What is agile methodology?

Agile methodology is an advance SDLC model which emphasis on "Move Quickly" fundamental. It is a kind of model which is the advance version of Incremental model with more customer satisfaction and rapid delivery concept.

Basically agile methodology divides the product into small chunks (Iteration). Then simultaneously planning, analysis, designing, coding as well as acceptance testing can be performed and according to that release the product, according to feedback again enhance the product with adaptability and finally re-release it.



4] What is SRS?

SRS (Software Requirement Specification) is a document that consist a complete description of the behavior of the system to be developed. SRS is prepared at the stage of analysis of the system to be developed. SRS includes the requirements in several types.

- Customer requirements
- Functional requirements
- Non-functional requirements.

5] What is oops?

Object Oriented Programming system is viewed as a collection of objects. It is a kind of programming model that carry the concept of classes and objects. It is simply used to structure a software program into simple and reusable code.

6] Write Basic Concepts of oops

1. **Class :** Class is an collection of data members (variables) & member functions (procedure, methods) with its behaviours.

2. **Objects :** The basic unit of OOP is an "Object" which is always accessed by its properties called data members and member functions. Object can be anything to which concept of programming applies. Objects take up space in the memory; Objects can communicate with one another by sending messages.

3. **Encapsulation :**
   The wrapping up of data and function into a single unit (called class) is known as encapsulation. It is basically a data hiding which includes in private access of data member and member function for the class. An object is the device that supports encapsulation.

4. **Inheritance :**
   Inheritance is the process by which objects of one class acquired the properties of objects of another classes. Using a concept called inheritance; new classes can be built from the old ones.
   Inheritance provides the idea of reusability.
   Mainly 5 types of Inheritance :

   1. Single Inheritance
   2. Multilevel Inheritance
   3. Hierarchical Inheritance
   4. Multiple Inheritance
   5. Hybrid Inheritance

5. **Polymorphism:**
   Polymorphism means an Ability to take one name having many different forms. It allows different objects to respond to the same message in different ways. The ability to change form is known as polymorphism. There are 2 types of polymorphism:

   1. ***Compile time (Method Overloading)***
      The method name should be same in single class but its behaviours (arguments, datatypes) are different.
   2. ***Run time (Method overriding)***
      The whole signature of the method should be same in super class as well as in sub class but its behaviour (body part) are different).

6. Abstraction

Abstraction refers to the act of representing essential features without including the background details or explanation. Only essential part should be displayed and rest of the part will be hiding. The classes use the concept of data abstraction, they are known as an Abstract Data Types (ADT).

7] What is Object ?

Object is called an instance of a class which is used to create memory for that class. With object we can access the whole properties (member function & data members) of an class.

Syntax: classname object name= new classname();

8] What is Class?

Class is an collection of data members (variables) & member functions (procedure, methods) with its behaviours.

Syntax:

```
class classname
{
        Variable (data members);
        Method (member function);
}
```

9] What is encapsulation?

The wrapping up of data and function into a single unit (called class) is known as encapsulation. It is basically a data hiding which includes in private access of data member and member function for the class. An object is the device that supports encapsulation.

10] What is Inheritance?

Inheritance is the process by which objects of one class acquired the properties of objects of another classes. Using a concept called inheritance; new classes can be built from the old ones.
Inheritance provides the idea of reusability.
Mainly 5 types of Inheritance:

6. Single Inheritance
7. Multilevel Inheritance
8. Hierarchical Inheritance
9. Multiple Inheritance
10. Hybrid Inheritance

11] What is polymorphism?

Polymorphism means an Ability to take one name having many different forms. It allows different objects to respond to the same message in different ways. There are 2 types of polymorphism:

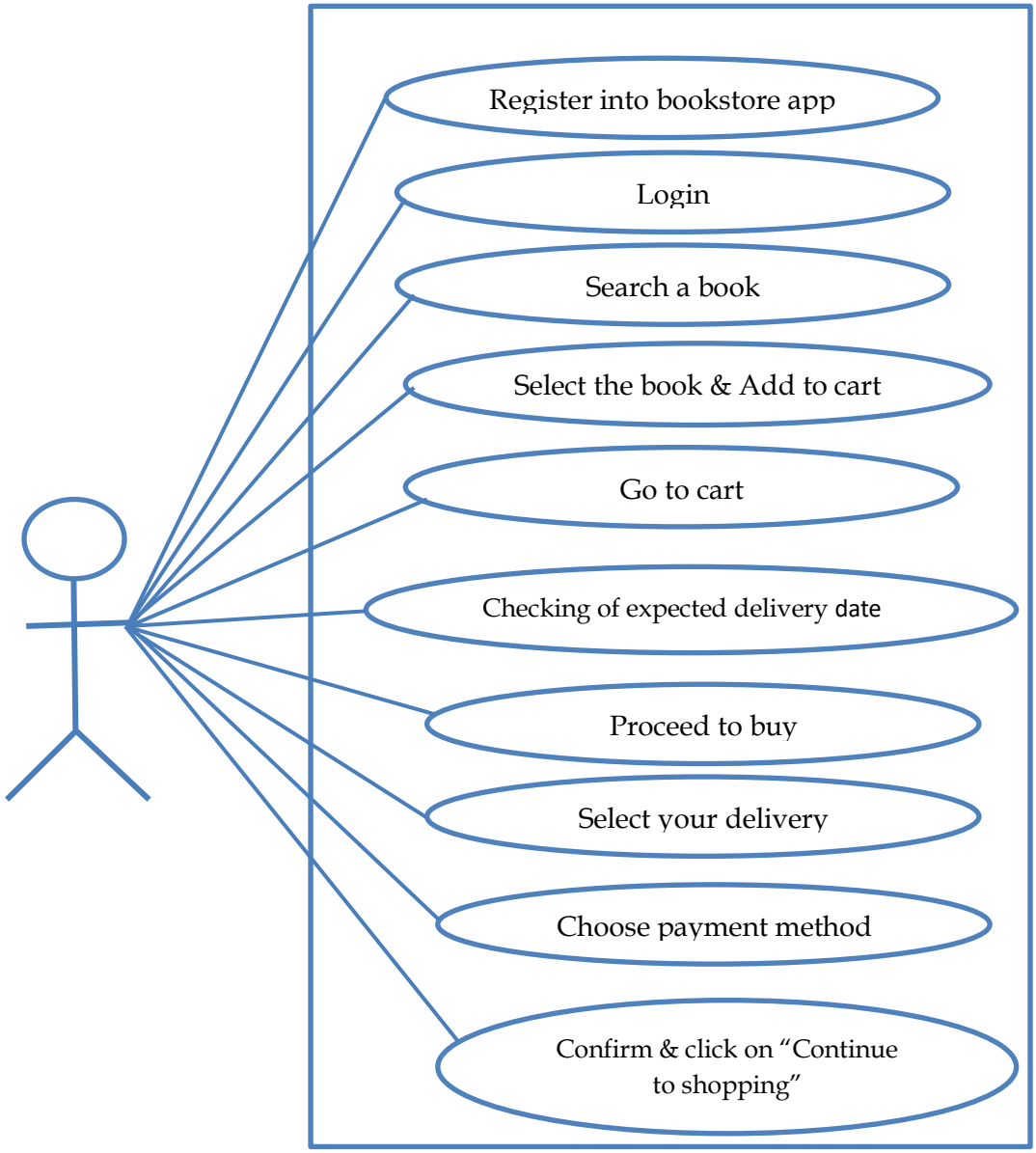
Compile time (Method Overloading)

The method name should be same in single class but its behaviours (arguments, datatypes) are different.

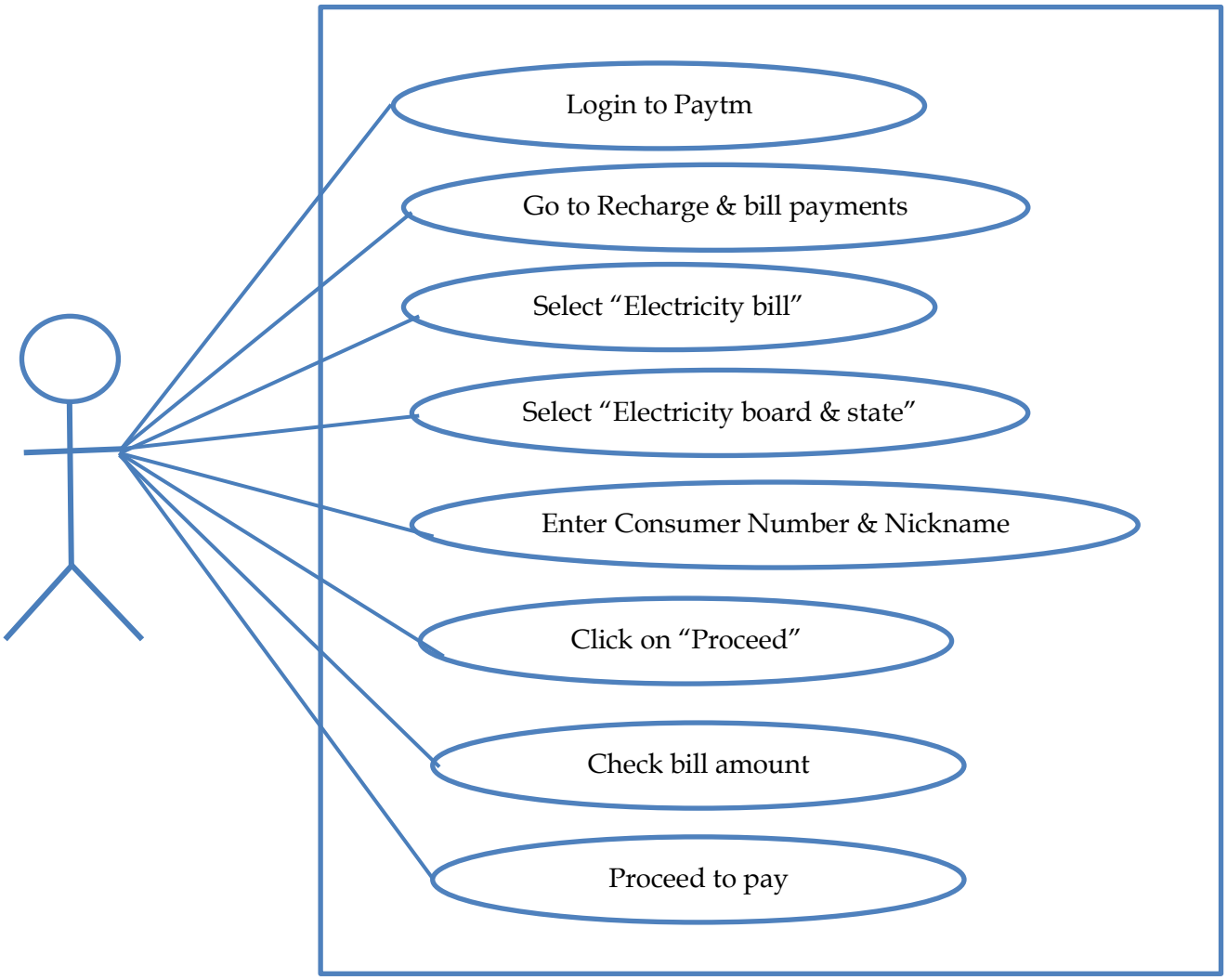
Run time (Method overriding)

The whole signature of the method should be same in super class as well as in sub class but its behaviour (body part) are different).

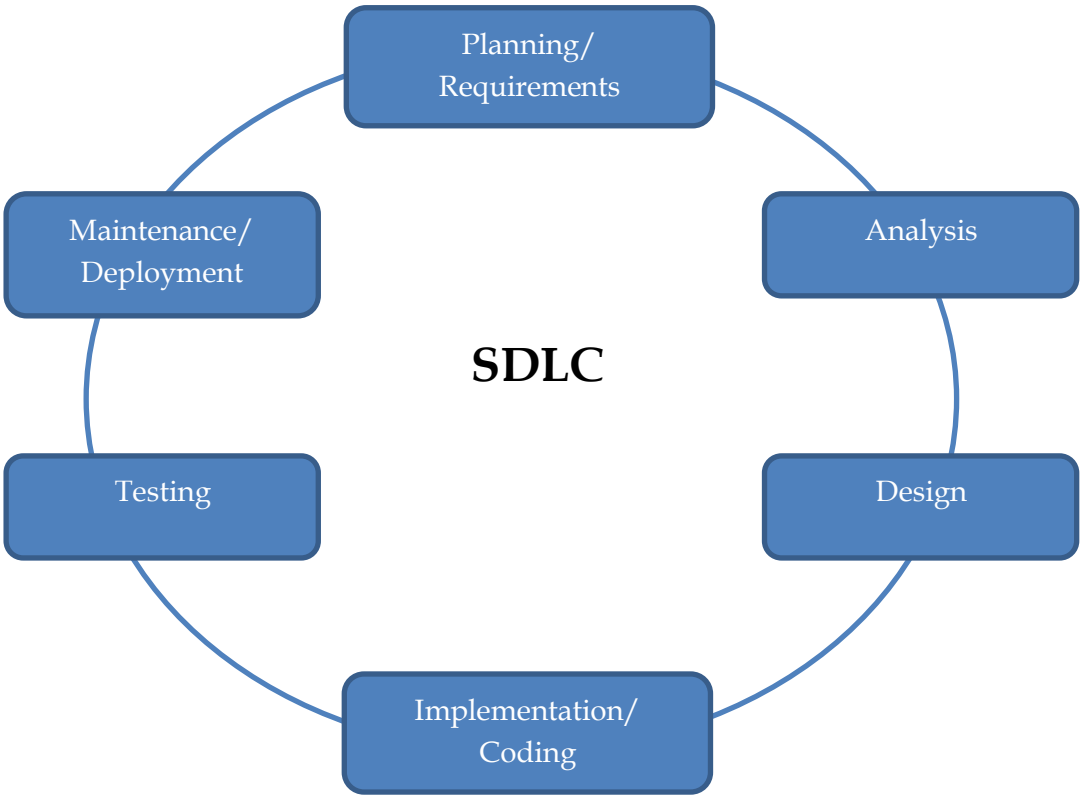
12] Draw Usecase on Online book shopping.



13] Draw Usecase on online bill payment system (paytm) .



14] Write SDLC phases with basic introduction.



- ***Planning/ Requirement gathering***

This is an initial phase for performing requirement analysis by the developers. While gathering all the requirements, the problems may arise like lack of clarity, requirements confusion, requirements amalgamation. In this stage customer requirements, functional requirements as well as non-functional requirements should be plan out.

- ***Analysis phase***

The analysis phase defines how the requirements will be accomplished. In this phase the requirements are documented, is called SRS (Software Requirement Specification) which shows the complete description of the behavior of the system to be developed.

- ***Design Phase***

The designing phase models how the system will work. It includes architecture, user interface, platform, programming, communication, security etc. Basically this phase is used to check the system is feasible for the customer technically, practically and financially.

- ***Implementation***

Implementation translates the designing part into coding part. The developer will start building the entire system by writing code using programming language.

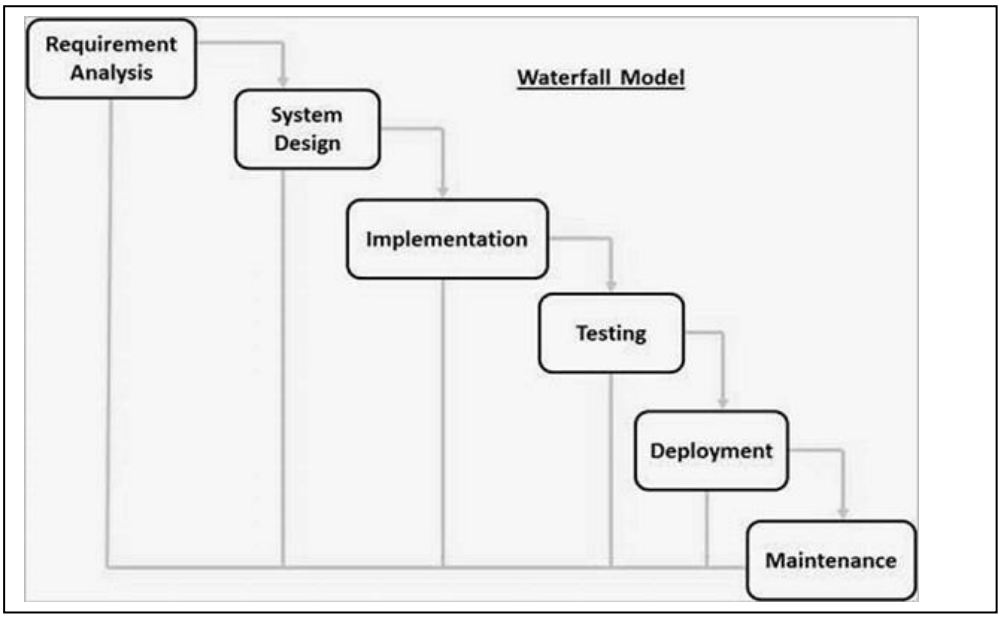
- ***Testing***

After implementation stage, a developer deploys the software into testing environment. The testing team tests the functionalities of the entire system and ensures that the customer requirements are satisfied accordingly or not.

- ***Maintenance/ Deployment***

Once the testing is done, the product is ready for deployment. It is the SDLC phase where the developed software is taken care of. Here software is updated timely depends on technology. Corrective maintenance, adaptive maintenance, and productive maintenance can be performed by changing environments.

15] Explain Phases of the waterfall model.



Requirement Analysis: All the possible requirements of the system to be developed is captured and documented here.

System Design: According to the requirement gathering in first stage, here designing models are prepared.

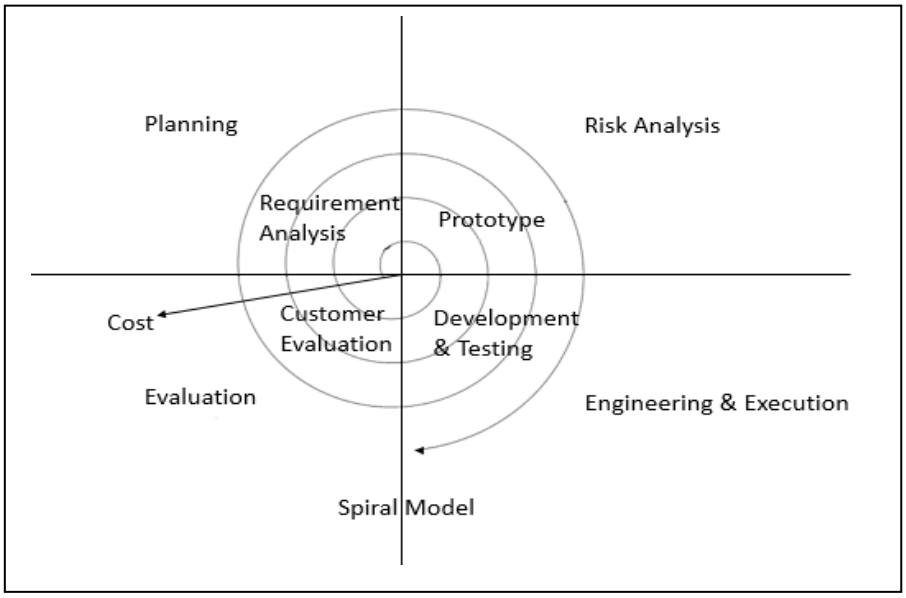
Implementation: According to designs, program will be implemented here or coded here. It can be implemented with unit or in integrated form.

Testing : The most essential phase where checking of end to end specifications is done. Functional or non-functional testing can be done here.

Deployment : Once the functional and non-functional testing is done; the product is deployed in the customer by releasing into market.

Maintenance : Enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

16] Explain Phases of the Spiral model.



Planning Phase: Requirements are gathered during the planning phase.

Risk Analysis: In this phase, a process is undertaken to identify risk and alternate solutions. If any risk is found during the risk analysis then alternate solutions are implemented.

Engineering Phase: In this phase software is **developed**, along with testing at the end of the phase. Hence in this phase the development and testing is done.

E**valuation phase:** This phase allows the customer to evaluate the output of the project to date before the project continues to the next spiral.

17] Write agile manifesto principles.

Agile methodology is a combination of iterative and incremental model with focus of more adaptability and customer satisfaction with rapid delivery of the product. It has the following key principles:

❖ Individual & Interactions over processes and tools.
❖ Working software over comprehensive document
❖ Customer collaboration over contract base negotiation.
❖ Responding to change over the given plan.

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

Agile methodology is a combination of iterative and incremental model with focus of more adaptability and customer satisfaction with rapid delivery of the product. Basically agile model breaks the product into small iterations. Then for each iteration, planning, analysis, designing, unit & integration testing, acceptances testing all the processes are performed simultaneously. And at the end, the product will be shown to customer or important stake holders.

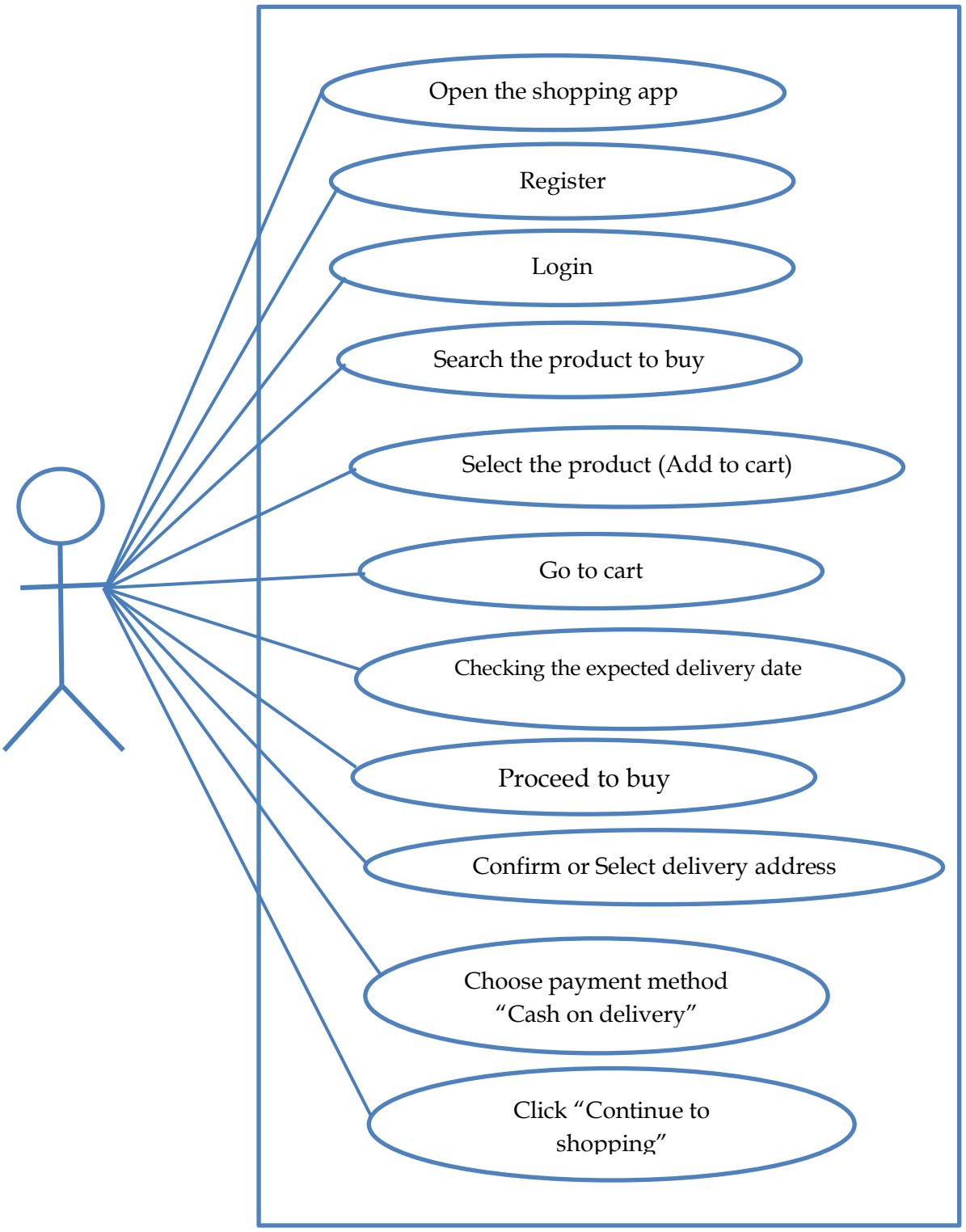
Pros:

- sometimes can be given It is a realistic approach for the software development.
- This methodology is widely and very useful for the frequently delivery of product.
- Resources requirements are minimum.
- Suitable for both static and changing requirements.
- Easy to plan and manage. So it will gives flexibility to the developers.
- For time saving, this model is really good choice.
- Face to face communication with the client gives more acceptance level of the product to be developed.

Cons:

- More risk on maintainability and extensibility.
- Less documentation is generated.
- Depends heavily on customer interaction, a wrong direction.
- Sometimes in Agile methodology the requirement is not very clear hence it's difficult to predict the expected result.
- Difficult to measure about overall resource requirements and efforts.

19] Draw usecase on Online shopping product using COD.



20] Draw use case on Online shopping product using payment gateway.