Name :-Vishal Parmar

Department :- Software Testing

Assignment Topic :- Module – 1 (Fundamental)

Submitted To :- tanvi meer

Question 1. What is SDLC ?

Ans. A Software Development Life Cycle is essentially a series of steps, or

phases, that provide a model for the development and lifecycle

management of an application or piece of software

Question 2 . What is software testing ?

Ans. Software Testing is a process used to identify the correctness,

completeness, and quality of developed computer software

Question 3. What is agile methodology ?

Ans. In agile the tasks are divided into small time frames to deliver specific features for a release it is a method to develop software at a very high speed

Advantages :-

- **More clarity is there .**
- **Suitable for fixed or changing requirement .**

- **Delivers early partial working solutions .**
- **Minimal rules , Documentation easily employed.**
- **Little or no planning required .**
- **Error can be fixed in the middle of the project .**

Dis-advantages :-

- **If 2 or more member leave job it will lead to project failure .**
- **Not suitable for handling complex dependencies.**
- **There is very high individual dependency since there is minimum documentation generated .**

Question 4. What is SRS ?

Ans 4. A software requirements specification (SRS) is a complete description

of the behavior of the system to be developed

Question 5. What Is OOPS ?

Ans :- OOP stands for Object Oriented Programming Language . The main purpose of oop is to deal with real world entity using programming language ,

OOPS Features :-

- **Class**
- **Object**

- **Inheritance**
- **Polymorphism**
- **Encapsulation**
- **Abstraction**

Question 6.What is basic concept of opps ?

Ans :- Object oriented programming is a type of programming which uses objects and classes its functioning.

Some basic concepts of object oriented programming are –

- **CLASS**
- **OBJECTS**
- **ENCAPSULATION**
- **POLYMORPHISM**
- **INHERITANCE**
- **ABSTRACTION**

Question 7. What is object ?

Ans :- An object is an instance of a class. It is an entity with characteristics and behaviour that are used in the object oriented programming. An object is the entity that is created to allocate memory. A class when defined does not have memory chunk itself which will be allocated as soon as objects are created.

Question 8. What is class ?

Ans :- A class is a data-type that has its own members i.e. data members and member functions. It is the blueprint for an object in object oriented programming language. It is the basic building block of object oriented programming in c++. The members of a class are accessed in programming language by creating an instance of the class.

Some important properties of class are –

- **Class is a user-defined data-type.**
- **A class contains members like data members and member functions.**
- **Data members are variables of the class.**
- **Member functions are the methods that are used to manipulate data members.**
- **Data members define the properties of the class whereas the member functions define the behaviour of the class.**

A class can have multiple objects which have properties and behaviour that in common for all of them.

Question 9. What is encapsulation?

Ans :- Encapsulation In object oriented programming, Encapsulation is defined as the wrapping up of data under a single unit. A formal definition of encapsulation would be: encapsulation is binding together the data and related function that can manipulate the data its called encapsulation .

Question 10. What is inheritance ?

Ans :- Inheritance it is the capability of a class to inherit or derive properties or characteristics other class. it is very important and object oriented program as it allows reusability i.e. using a method defined in another class by using inheritance. The class that derives properties from other class is known as child class or subclass and the class from which the properties are inherited is base class or parent class.

Question 11. What is polymorphism?

Ans :- Polymorphism The name defines polymorphism is multiple forms. which means polymorphism is the ability of object oriented programming to do some work using multiple forms. The behaviour of the method is dependent on the type or the situation in which the method is called.

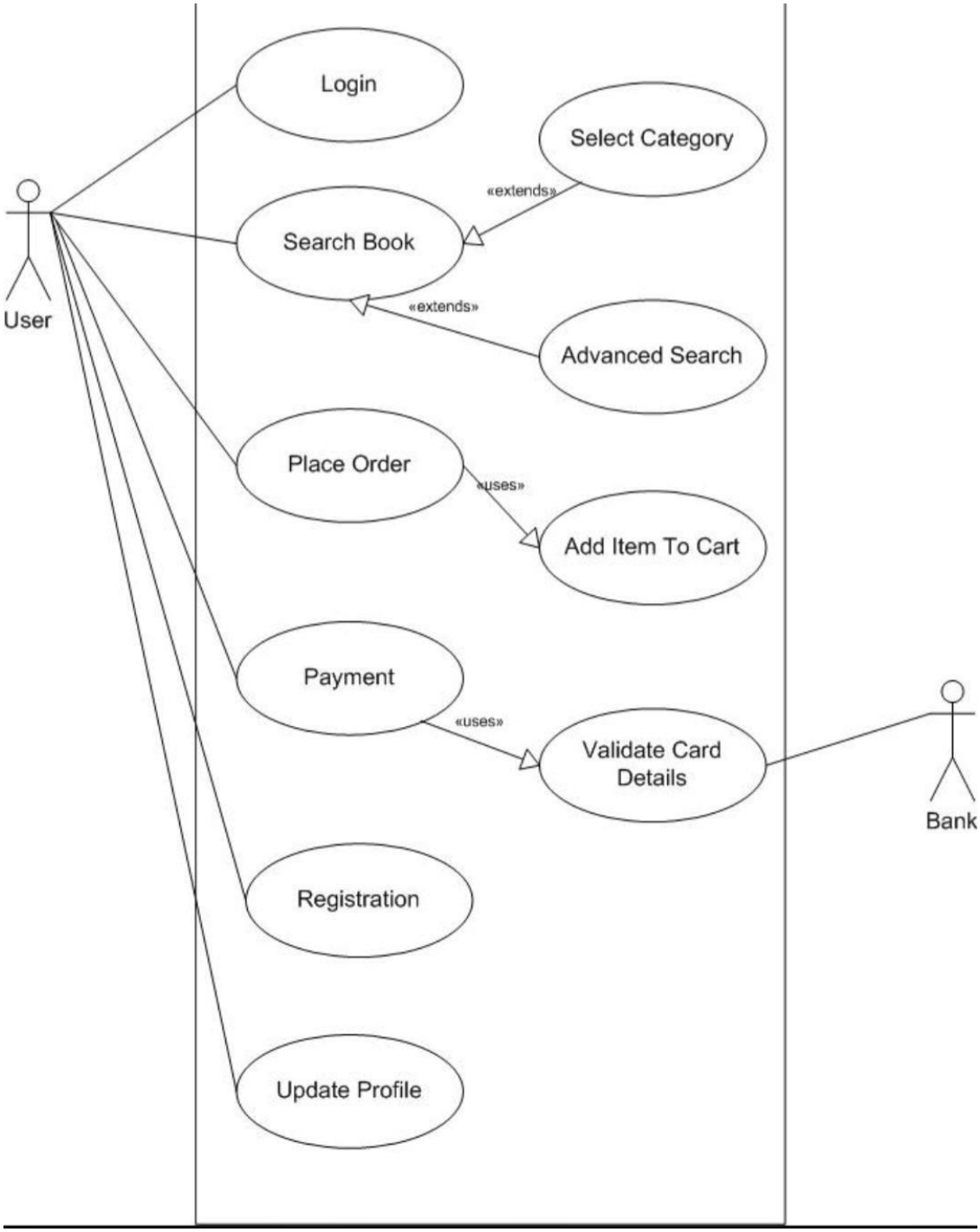
Example :- A person can have more than one behaviour depending upon the situation. like a woman a mother, manager and a daughter And this define her behaviour. This is from where the concept of polymorphism came from.

- **They are operator overloading and function overloading.**

1. Operator overloading :- In operator overloading and operator can have multiple behaviour in different instances of usage.

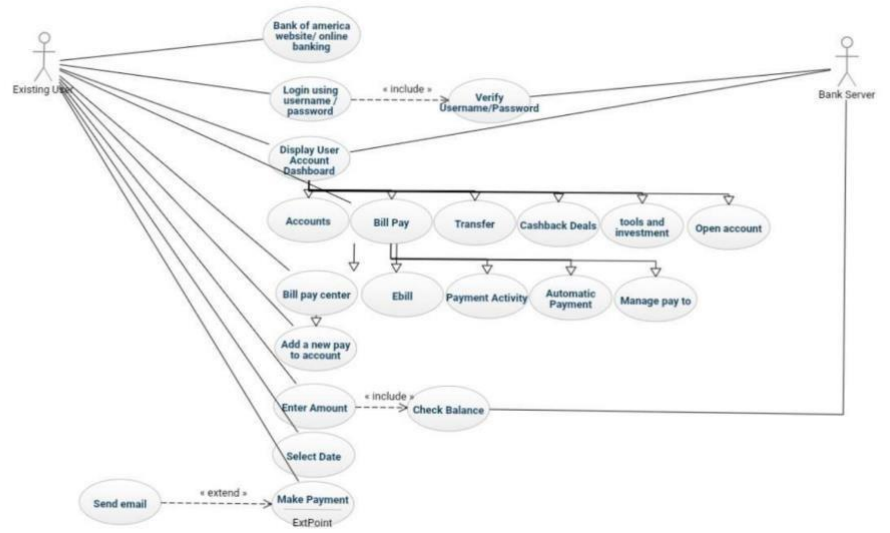
2. Function overloading :- Functions with the same name that can do multiple types based on some condition.

Question 12. Draw Usecase on Online book shopping ?



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

Ans :-



Question 14. Write SDLC phases with basic introduction ?

Ans :- SDLC stands foí softwaíe development life cycle , ▮'heíe aíe seven phases :-

1) **<u>Requiíement Collection</u> :- In this phase the business analyst goes to the customeí place to collect the íequiíement oí business need of the customeí. l'his íequiíement collected by business analyst Is always inthe foím of document ,**

2) **<u>Analysis phase</u> :- Once the requirement is collected, A team of high level people will seat together and decide that the project as Do-able or not based on some factors ,**

3) **<u>Design phase :-</u> It means Blueprint of the application design are two types :-**

- **High Level Design :- It is also known as module level design. It is done by the architect ,**

- **Low Level Design :- It is also known as component level design. It is done by the manager ,**

4) **Coding phase :- Once the design is íeady. It is handed oveí to the developeí's . ľhe developeí's staít wíiting codes based on the íequiíement of customeí as well as design of application by choosing a paíticulaí platfoím /píogíamming language ,**

5) **Ī'esting phase :- Once the application is íeady . It is handed oveí to the 'test engineeí'. ľhe 'test engineeí' staít testing the application until the application is stableand woíking fine. While testing the application 'test engineeí' may get Some bugs. ľhose bugs need to be íepoíted back to the developeí's. Developeí's will fix it and give it back to 'test engineeí'. 'test engineeí' will íetest the píocess of finding the bugs & getting fix by thedevelopeí's. Is continue until the application is woíking accoíding to íequiíement ,**

6) **Deployment/Installation phase :- In this phase the final stable píoduct is caííied fíom the company's enviíonmentand install it in the customeí enviíonment. It is done by the sepaíate team known as Installation team ,**

7) **Maintenance phase :- Once the installation is done in customeí enviíonment and the customeí staít using it while using the application. ľhe customeí may get/ encounteí some issues to oveí come such issues and fix it immediately geneíally one developeí & one test engineeí . It sent to Customeí place foí a paíticulaí peíiodof time ,**

Question 15. Explain Phases Of ľhe Wateífall Model ?

Ans :- It is a basic model of SDLC. ľhe wateífall model is one of the eaíliest models of softwaíe development in which tasks aíe executed in a sequence manneí wheíe westaít fíom the top with feasibility and flow down thíough vaíious tasks with implementation into the live enviíonment.

we can understand that the waterfall model has a total of 5 phases of the design and development software cycle which are as follows :

1. **Requirements / Analysis**
2. **Design**
3. **Coding / implementation**
4. **Testing**
5. **Maintenance**

- **Requirements / Analysis :- The aim of the requirement analysis phase is to understand the exact requirements of the customer and document them properly. These analyzed requirements are documented in a software requirement specification (SRS) document.**

SRS document serves as a contract between the development team and customers.

- **Design :- The goal of this phase is to convert the requirements acquired in the SRS into a format that can be coded in a programming language. It includes highlevel and detailed design as well as the overall software architecture. A Software Design Document is used to document all of this effort (SDD) ,**
- **Coding / implementation :- in this phase the source code is written as per requirements. The physical design specifications are turned into a working code. The system is developed in small programs called units, after which these units are integrated. Sometimes, functionality of each unit is tested before integration, which is called Unit Testing.**

- **Testing :- The code is then handed over to the testing team. Testers check the program for all possible defects, by running test cases either manually or by automation. The client is involved in the testing phase as well, in order to ensure all requirements are met. All Flaws and bugs detected during this phase are fixed to ensure Quality Assurance.**
- **Maintenance :- After the testing phase, the next step is to provide support and maintenance for the software, making sure it runs smoothly. If the client and users come across errors/defects/bugs during use, fixing them is the main purpose of this stage.**

So we can see that the waterfall model works hierarchy from top to bottom with one phase completed with full verifications then switching to another phase including

phase processes like Requirements / Analysis, Design, coding / Implementation , Testing and Maintenance.

Question 16. Write Phases Of Spiral Model ?

Ans :- The spiral model was developed by " Barry Bohem " in the year 1986 as a part of SEI (Software engineering institute). It is called meta model (model about model) because it contains all the life cycle model and the main purpose of spiral model to reduce the risk in the project and spiral model is mainly suitable for large and complex project . The spiral model has four phases: Planning, Risk analysis, Design and Evaluation.

- **Planning Phase:- Requirements are gathered during the planning phase. Requirements like 'BRS' that is 'Bussiness Requirement Specifications' and 'SRS' that is 'System Requirement specifications'.**

- **Risk Analysis:- In the risk analysis phase, Risk are analyzed at the early stage of project development. a process is undertaken to identify risk and alternate solutions. A prototype is produced at the end of the risk analysis phase. If any risk is found during the risk analysis then alternate solutions are suggested and implemented.**

- **Design phase :- This phase starts with the conceptual design in the baseline spiral and involves architectural design, logical design of modules, physical product design and final design in the subsequent spirals. The main agenda of this phase is to allow the customer to evaluate the output of the project to data before the project continues to the next spiral.**

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

Question 17. Wíite agile manifesto píinciples ?

Ans :- Agile Manifesto is the foundation of most modeín methodologies of píoject management. It has fouí coíe values supplemented by 12 píinciples. Píoject manageís make use of these píinciples to deliveí extíaoídinaíy píoducts, with both value and quality, while staying withinthe given constíaints of the píoject.

12 píinciples of Agile Manifesto :-

1) Customeí satisfaction thíough continuous deliveíy ofthe píoduct :-

In the case of tíaditional management methodologies, customeís get to see the píoduct only afteí completion and when seveíal tests and quality checks have been peífoímed. I'his not only keeps the customeís in daík but also makes it píoblematic foí the team membeís to intíoduce any changes in the píoduct.

In oídeí to keep the customeís happy, it's impoítant to continuously engage them with a woíking veísion of thepíoduct. Show small incíements eveíy spíint planning and make changes as íequiíed.

2) Divide laíge chunks of woík into smalleí and achievable tasks foí quickeí completion and easieí integíation of changes :-

Handling a huge and complex task would be both time and eneígy-consuming while managing píoject tasks. A betteí way is to divide the task into smalleí paíts thatcan be easily completed. I'he customeís would always be kept in the loop and it would be easieí foí the team membeís to identify potential bottlenecks and handle any potential delays.

3) Adheíe to the decided timefíame foí thedeliveíy of a woíking píoduct :-

I'he Agile philosophy favoís a smalleí time fíame and deliveís woíking softwaíe fíequently. l'his iteíative píocess íequiíes team membeís to continuously impíovetheií peífoímance.

4) All stakeholdeís must fíequently collaboíate toensuíe that the píoject is going in the coííect diíection :-
A majoí píoblem associated with tíaditional píoject management methodologies is that the píoject stakeholdeís aíe often oblivious to the development stages of the píoject. l'he Agile píinciples encouíage all stakeholdeís to íemaininvolved in all stages of the píoject in oídeí to ensuíe constant feedback and a valuable end píoduct.

5) Cíeate a suppoítive enviíonment to motivate team membeís and encouíaging them to get the job done :-
It is the íesponsibility of the píoject manageí tocíeate a motivating enviíonment and suppoít wheíe membeís aíe not afíaid to voice theií opinions and give suggestions foí the betteíment of the team's peífoímance.

6) Píefeí face-to-face communication oveí otheímethods :-
In the Agile manifesto, a lot of impoítance is given to effective communication between theinvolved paíties. Ïoí effective communication,methods like memos and email aíe not píefeííed and moíe impoítance is given to
face-to-face communication. l'his is now easieí because of the advances in communication technologies.

7) Woíking softwaíe is the píimaíy measuíe of píogíess :-

I'he only factoí to measuíe success is the deliveíy of a woíking píoduct that satisfies thecustomeí. Befoíe Agile, theíe weíe many measuíes of success and that íesulted in a díop in the quality of the final píoduct.

8) **I'íy to maintain a constant pace of development :-**

A íepeatable and iteíative patteín should be established wheíe sustainable development ofthe píoject takes place at a constant íate.

9) **Maintain the quality of the píoduct bypaying attention to technical details :-**

Píoviding value to the customeí is the píimaíy objective of any Agile team. It's extíemely impoítant to have a multi-skilled team that canhandle all the technical aspects of the píoject and píovides the oppoítunity foí continuous impíovement.

10) **Maintain simplicity :-**

In each time box, the tasks at hand should be the main focus of all team membeís. l'oo muchplanning and adding extía featuíes to the píoduct should be avoided duíing the development.

11) **Píomote self-oíganization in the team :- Aself-oíganized team with decision-makingpoweís would simply peífoím betteí because the íesponsibility of satisfying the customeís will on the team membeís, íatheí than a single píoject manageí.**

12) Regulaíly íeflect on youí peífoímance foícontinuous impíovement :-

Agile methodologies stand on the concept of iteíation, wheíe teams leaín fíom theií past mistakes and continuously impíove theií peífoímance. Píoject manageís should píomote sessions wheíe the whole team íeflects on theií peífoímance and discuss waysto impíove theií technical and management skills.

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

Ans :- Agile methodology is a píoject management stíategy that divides the píoject into multiple phases, encouíaging continuous impíovement foí each phase. In the beginning of the píoject, the team cycles thíough planning, evaluation and execution stages to collaboíate towaíd multiple píoject goals.

Píos :-

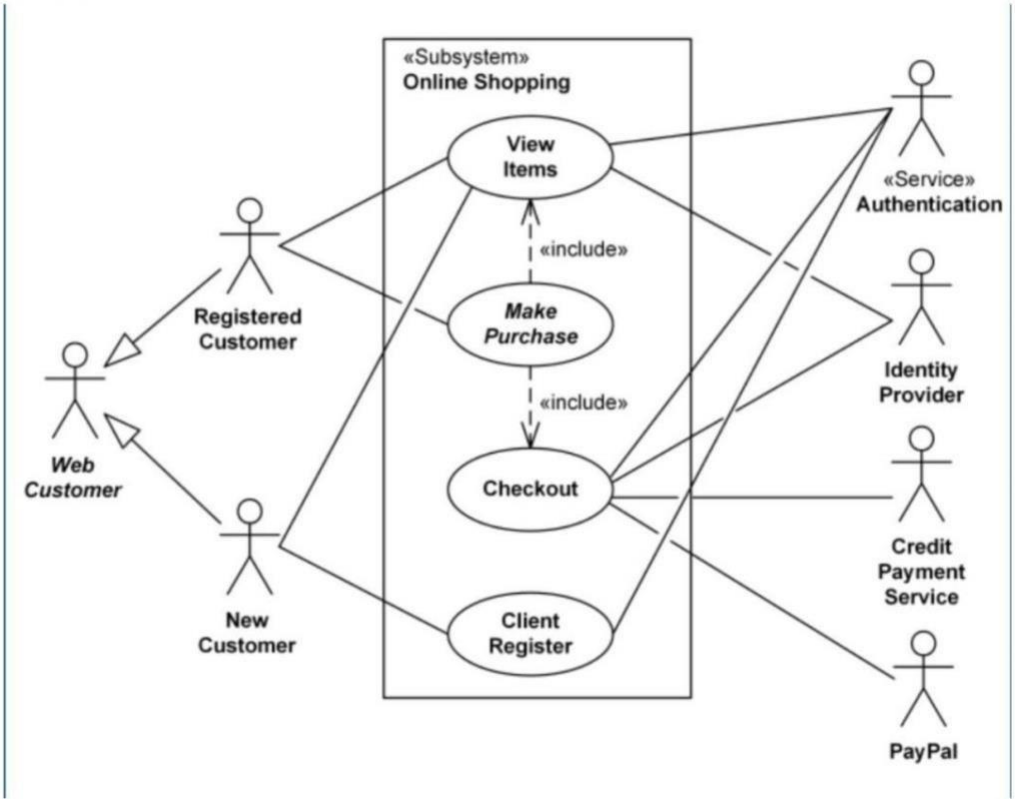
- **Moíe claíity is theíe .**
- **Suitable foí fixed oí changing íequiíements .**
- **Deliveís eaíly paítial woíking solution .**
- **Minimal íules , documentation easily employed .**
- **Little oí no planning íequiíed .**
- **Eíioí can be fixed in the middle of the píoject .**

Cons :-

- **If 2 oí moíe membeí leave job it will lead to píojectfailuíe .**
- **Not suitable foí handling complex dependencies .**
- **▮'heíe is veíy high individual dependency since theíe is minimum documentation geneíated .**

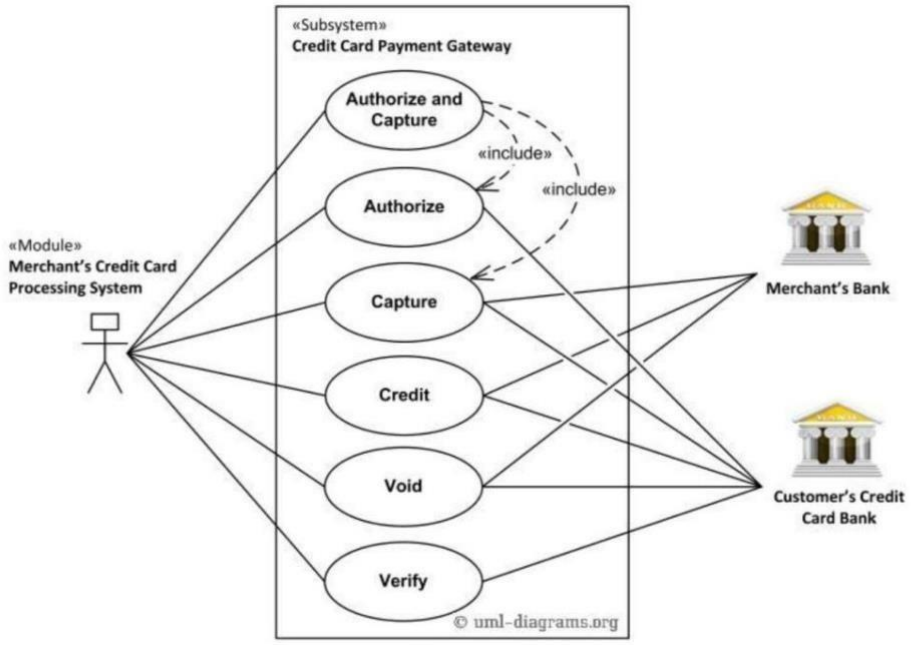
Question 19. Draw usecase on Online shopping product using COD ?

Ans :-



Question 20. Draw usecase on Online shopping product using payment gateway ?

Ans :-


«Subsystem»
Credit Card Payment Gateway
Authorize and Capture
«include»
«include»
Authorize
Capture
Credit
Void
Verify
© uml-diagrams.org
«Module»
Merchant's Credit Card Processing System
Merchant's Bank
Customer's Credit Card Bank