Name :- Pankesh *Prajapat*

Department :- Software Testing

Assignment Topic :- Module – 1 (Fundamental)

Submitted To :-

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.

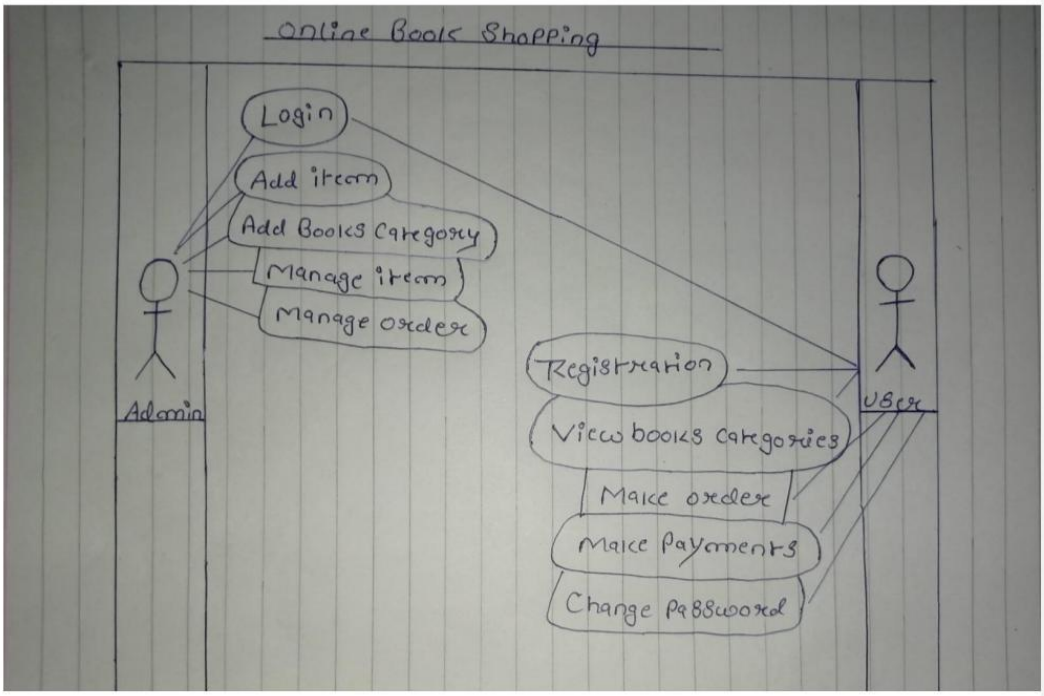
Example :- A person can have more than one behaviour depending upon the situation. like a woman a mother, manager and a daughter And this define her behaviour. This is from where the concept of polymorphism came from.

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

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

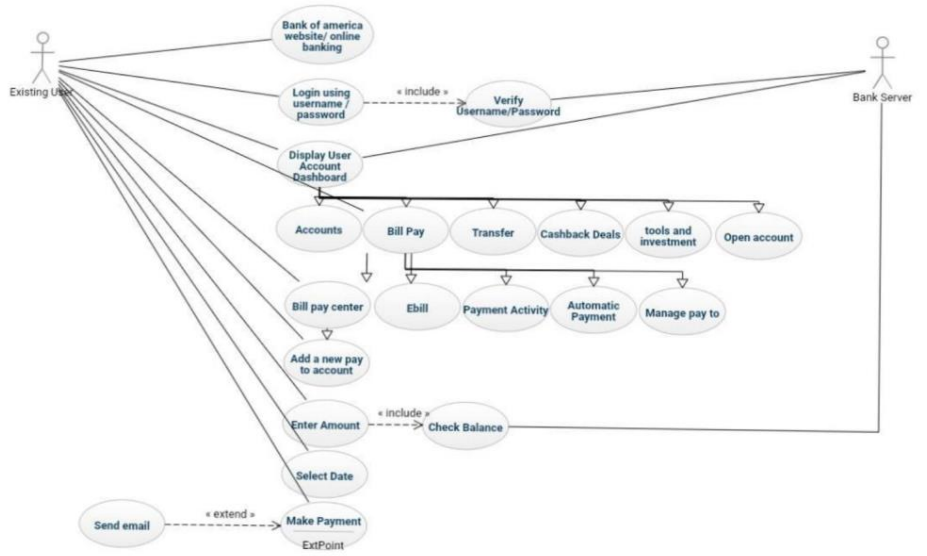
2. **Function overloading :- Functions with the same name that can do multiple types based on some condition.**

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 for software development life cycle ,
There are seven phases :-

1) **<u>Requirement Collection</u> :- In this phase the business analyst goes to the customer place to collect the requirement or business need of the customer. This requirement collected by business analyst Is always in the form 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) **<u>Coding phase</u> :- Once the design is ready. It is handed over to the developer's . The developer's start writing codes based on the requirement of customer as well as design of application by choosing a particular platform / programming language ,**

5) **<u>Testing phase</u> :- Once the application is ready . It is handed over to the 'test engineer'. The 'test engineer' start testing the application until the application is stable and working fine. While testing the application 'test engineer' may get Some bugs. Those bugs need to be reported back to the developer's. Developer's will fix it and give it back to 'test engineer'. 'test engineer' will re-test the process of finding the bugs & getting fix by the developer's. Is continue until the application is working according to requirement ,**

6) **<u>Deployment/Installation phase</u> :- In this phase the final stable product is carried from the company's environment and install it in the customer environment. It is done by the separate team known as Installation team ,**

7) **<u>Maintenance phase</u> :- Once the installation is done in customer environment and the customer start using it while using the application. The customer may get/ encounter some issues to over come such issues and fix it immediately generally one developer & one test engineer . It sent to Customer place for a particular period of time ,**

Question 15. Explain Phases Of The Waterfall Model ?

Ans :- It is a basic model of SDLC. The waterfall model is one of the earliest models of software development in which tasks are executed in a sequence manner where we start from the top with feasibility and flow down through various tasks with implementation into the live environment.

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 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) ,**
- **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. Write agile manifesto principles ?

Ans :- Agile Manifesto is the foundation of most modern methodologies of project management. It has four core values supplemented by 12 principles. Project managers make use of these principles to deliver extraordinary

products, with both value and quality, while staying within the given constraints of the project.

12 principles of Agile Manifesto :-

1) Customer satisfaction through continuous delivery of the product :-

In the case of traditional management methodologies, customers get to see the product only after completion and when several tests and quality checks have been performed. This not only keeps the customers in dark but also makes it problematic for the team members to introduce any changes in the product.

In order to keep the customers happy, it's important to continuously engage them with a working version of the product. Show small increments every sprint planning and make changes as required.

2) Divide large chunks of work into smaller and achievable tasks for quicker completion and easier integration of changes :-

Handling a huge and complex task would be both time and energy-consuming while managing project tasks. A better way is to divide the task into smaller parts that can be easily completed. The customers would always be kept in the loop and it would be easier for the team members to identify potential bottlenecks and handle any potential delays.

3) Adhere to the decided timeframe for the delivery of a working product :-

The Agile philosophy favors a smaller time frame and delivers working software frequently. This iterative

process requires team members to continuously improve their performance.

4) **All stakeholders must frequently collaborate to ensure that the project is going in the correct direction :-**
   A major problem associated with traditional project management methodologies is that the project stakeholders are often oblivious to the development stages of the project. The Agile principles encourage all stakeholders to remain involved in all stages of the project in order to ensure constant feedback and a valuable end product.

5) **Create a supportive environment to motivate team members and encouraging them to get the job done :-**
   It is the responsibility of the project manager to create a motivating environment and support where members are not afraid to voice their opinions and give suggestions for the betterment of the team's performance.

6) **Prefer face-to-face communication over other methods :-**
   In the Agile manifesto, a lot of importance is given to effective communication between the involved parties. For effective communication, methods like memos and email are not preferred and more importance is given to face-to-face communication. This is now easier because of the advances in communication technologies.

7) Working software is the primary measure of progress :-

The only factor to measure success is the delivery of a working product that satisfies the customer. Before Agile, there were many measures of success and that resulted in a drop in the quality of the final product.

8) Try to maintain a constant pace of development :-

A repeatable and iterative pattern should be established where sustainable development of the project takes place at a constant rate.

9) Maintain the quality of the product by paying attention to technical details :-

Providing value to the customer is the primary objective of any Agile team. It's extremely important to have a multi-skilled team that can handle all the technical aspects of the project and provides the opportunity for continuous improvement.

10) Maintain simplicity :-

In each time box, the tasks at hand should be the main focus of all team members. Too much planning and adding extra features to the product should be avoided during the development.

11) Promote self-organization in the team :-

A self-organized team with decision-making powers would simply perform better because the responsibility of satisfying the customers

will on the team members, rather than a single project manager.

12) Regularly reflect on your performance for continuous improvement :-
Agile methodologies stand on the concept of iteration, where teams learn from their past mistakes and continuously improve their performance. Project managers should promote sessions where the whole team reflects on their performance and discuss ways to improve their technical and management skills.

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

Ans :- Agile methodology is a project management strategy that divides the project into multiple phases, encouraging continuous improvement for each phase. In the beginning of the project, the team cycles through planning, evaluation and execution stages to collaborate toward multiple project goals.

Pros :-

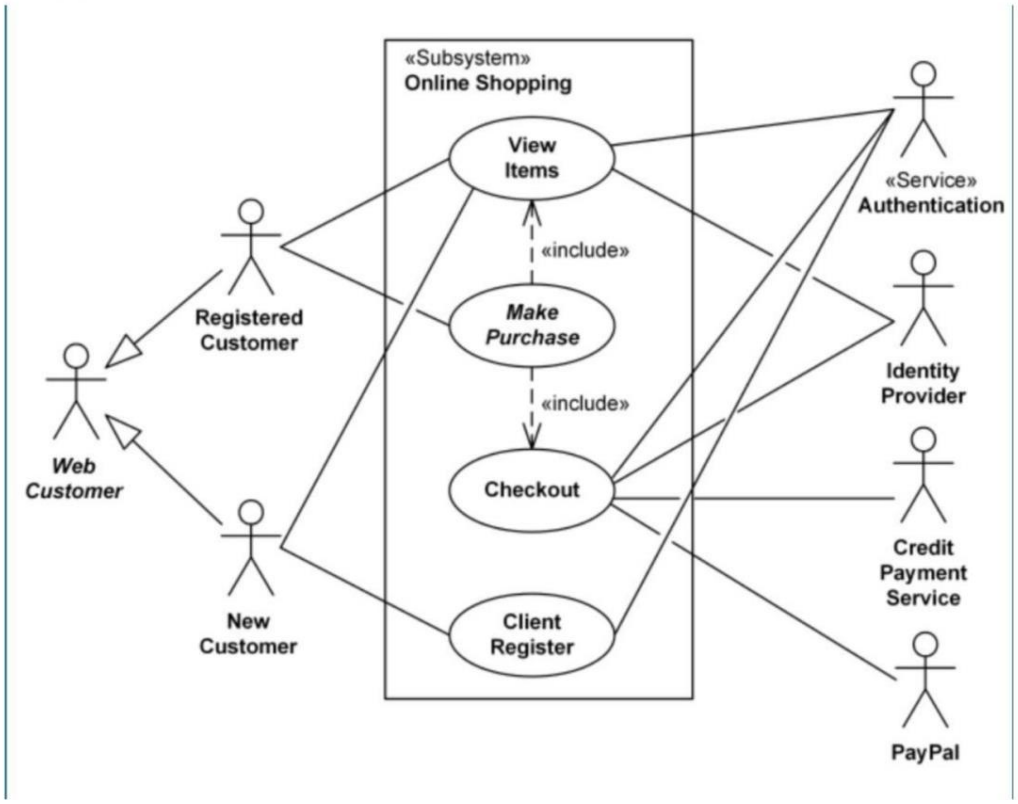
- **More clarity is there .**
- **Suitable for fixed or changing requirements .**
- **Delivers early partial working solution .**
- **Minimal rules , documentation easily employed .**
- **Little or no planning required .**
- **Error can be fixed in the middle of the project .**

Cons :-

- **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 .**

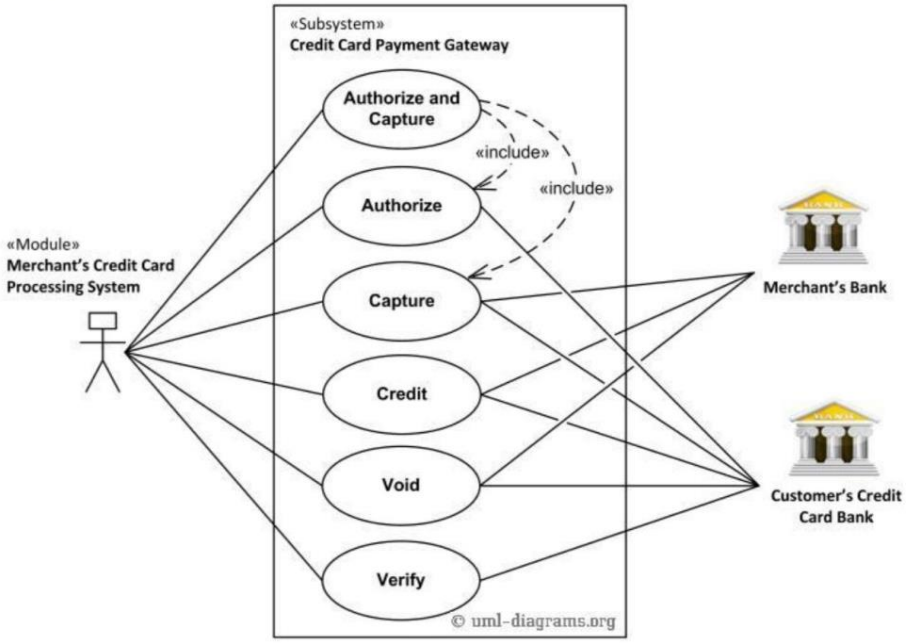
Question 19. Draw usecase on Online shopping product using COD ?

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
«Module»
Merchant's Credit Card Processing System
Merchant's Bank
Customer's Credit Card Bank
© uml-diagrams.org