**Software testing assignment**

MODULE -1 (FUNDAMENTAL)

**1.What is SDLC?**

Software development cycle is a process used by the software industry to design, develop and test high quality softwares. The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times.

The lifecycle defines a methodology for improving the quality of the software and all the development process. It is an sequential development.

**2.What is software testing?**

Software testing is the act of examining the artifacts and the behaviour of the software under test by validation and verification. Software testing can also provide an objective, independent view of software to allow the business to appreciate and understand the risks of software implementation. Software testing can determine the correctness of software under the assumption of some specific hypothesis but testing cannot identify all the failures within the software. The primary purpose of testing is to detect software failures so that defects may be discovered or corrected.

**3.What is agile methodology?**

In agile methodology the tasks are divided to time boxes means small time frames to delivery specific features for a release. Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features by required by the customer. A project management framework that breaks project down into several dynamic phases, commonly called as sprints.

**4.What is SRS?**

The full form of SRS is Software Requirements Specification which describes clear description of all requirements. SRS is provide an overview of the project and its goals. Define the scope of the project and explain what the software is intended to do. SRS is developed based the agreement between customers and contractors. Ideally, this document describes what is to be built. The requirement documentaries to capture the requirements from the customers perspective by defining goals. The SRS will contain information about all the software components that make up the product.

**5.What is oops?**

The full form of OOPS is object-oriented programming is a programming paradigm based on the concept of object. That is a set of data contained in fields, the code, indicating procedures instead of usual logic-based system. Different parts of it perform actions on real world items, creating actual interactions between people and machines. The strategy is advantageous for collaborative development when projects are divided into groups due to the organization of object-oriented software. Code reuse, scalability, and efficiency are other advantage of OOP.

**6.Write basic concepts of oops?**

There are six concepts of object-oriented programming.

* Object
* Class
* Encapsulation
* Inheritance
* Polymorphism
* Abstraction

**7.What is object?**

Object means the real-world entity (things, people, place, process). Objects are instances of a class created with specifically defined data. When class is defined initially, the description is the only object that is defined. Object means the data and functionality, behaviour and property and method.

**8.What is class?**

Class is the Blueprint of an object. Object oriented programming is a programming paradigm based on concepts of objects. The object may contain data as a form of instance variables and behaviours in form of methods.

First, we need to understand two concepts’ objects and class.

For example,

Fruit is a class and fruit types grapes; apple are the objects.

**9.What is encapsulation?**

Encapsulation is basically information hiding. It describes the idea of bundling data and methods that work on that data within one unit. This principle states that all important information is contained inside an object and only select information is exposed. The implementation and state of each object are privately held inside a defined class. Other objects do not have access to this class or the authority to make changes. They are only able to call a list of public functions or methods. This characteristic of data hiding provides greater program security and avoids unintended data corruption.

**10.What is inheritance?**

Inheritance means sub-class inherits from the super-class. Classes can reuse code from other classes. Relationships and subclasses between objects can be assigned, enabling developers to reuse common logic while still maintaining unique hierarchy. This property of OOP forces a more through data analysis, reduces development time and ensures a higher level of accuracy.

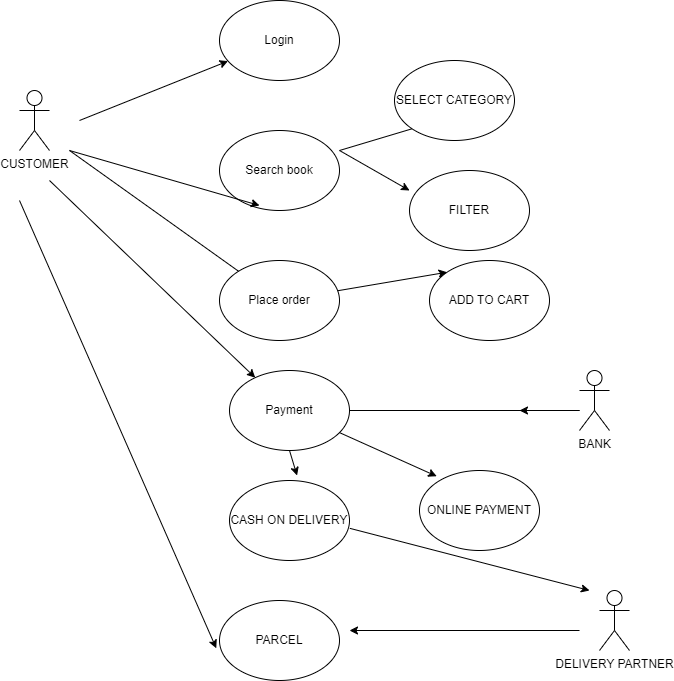
In animal class, there are methods and attributes that are common to all animals by using inheritance concept other child classes can use those attributes and methods in the parent class.

There are some inheritance types: Single inheritance, multiple inheritance, hybrid inheritance, multilevel inheritance, hierarchical inheritance.

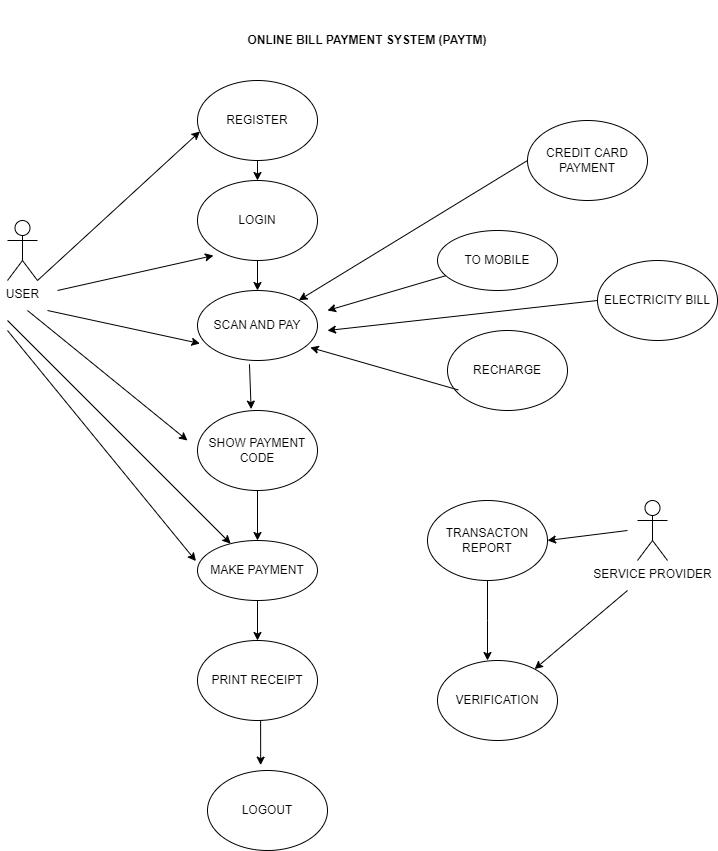
**11.What is polymorphism?**

Objects are designed to share behaviours and they can take on more than one form. Polymorphism allows different types of objects to pass though the same interface.

**12.Draw use case on online book shopping?**



**13.Draw use case on online bill payment system?**

****

**14. Write SDLC phases with basic introduction?**

There are 7 phases of software development cycle.

1.Planning

2.Analysis

3.Design

4.Build

5.Testing

6.Diploy

7.Maintain

1.PLANNING

In the planning the requirement is most important factor. The team holds discussion with various stakeholders from problem and domain, tries to bring out much information as possible on their requirement. The software development team works to carry on the project.

The requirement is collected from various ways;

* Studying the existing or absolute system and software
* Conducting the interviews of users and developers’
* Referring the database
* Collecting answers from the question makers

Requirements definitions usually consist of natural language supplemented by diagrams and table.

There are two types of requirements;

1. Functional requirement – Describe system service and functions.
2. Nonfunctional requirement – Describe constraints on the system or the development process.

To properly decide what to make, what not to make, and what to make first, you can use a feature prioritization framework that takes into account the value of the software, the cost, the time it takes to build and other factors.

2.ANALYSIS

It defines the requirement of the system independents of how these requirements will be accomplished. It is carried out by senor develop testers of the team with information from the client of the pre sloes market studies and domain specialist of the industry. The deliverable result at the end of this phase is a requirement document (SRS) which describes clear description of all requirements. Ideally, this document describes what is to be built. The requirement documentaries to capture the requirements from the customers perspective by defining goals.

3.DESIGN

A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external modules. The internal design of all the modules of the proposed architecture should be clearly.

\*High level design:

Brief description and name of each module, an outline about the functionality of every module, interface relationship and dependencies between modules, database table identified along with their key elements, complete architecture diagrams along with technology details.

\*Low level design:

Functional logic of the modules, database tables which include type and size, complete detail of the interface, addresses all types of dependency issues, listing of error messages, complete input and output for every module.

4.BUILD

This is the longest phase of software development cycle.

In this stage of SDLC the actual development starts and the product is built. Developers start to build the entire system by writing code using the chosen programming language such as C, C++, PHP, Java and pascal. Tasks are divided into units or modules and assigned to the various developers. Developers need to follow certain predefined coding guidelines and also use a programming tools like compiler, interpreters, debugger to generate and implement the code.

5.TESTING

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all stages of SDLC. This stage refers to the testing only stage of the product where product defects are reported, tracked, fixed, and retested, until the product reaches the quality standards defined in the SRS software requirement specification.

6.DEPLOYMENT

Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometimes product deployment happens in stages as per the business strategy of that organization. The product may first be released in a limited segment and tested in the real business environment. It is sort of User Acceptance Testing (UAT). Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment.

7.MAINTENANCE

After the product is released in the market, its maintenance is done for the existing customer base. Maintenance is the process of changing a system after it has been deployed. In the maintenance stage bug fixing, upgrade, enhancement these activities are occur.

**15.Explain the phases of waterfall model?**

The waterfall model illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete. In this waterfall model, the phase do not overlap. The whole process of software development is divided into separate phases. In the waterfall model, typically the outcome of one phase acts as the input for the next phase sequentially.

The sequential phases in waterfall model are-

1. Requirement/Analysis - All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
2. Design- The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
3. Coding- with inputs from the system design, the system is first developed in small programs called units, which is integrated in next phase. Each unit is developed and tested for its functionality, which is referred to as unit testing.
4. Testing- All the units developed in the coding phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
5. Deployment- Once the functional and nonfunctional testing is done; the product id deployed in the customer environment or releases into the market.
6. Maintenance- There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

**16.Write phases of spiral model?**

The spiral model has four phases:

1. Determine objectives- This phase includes requirement gathering and analysis. Based on the requirements, objectives are defined and different alternate solutions are proposed.
2. Identify and resolve risks- All the proposed solutions are analysed and any potential risk is identified analysed, and resolved.
3. Develop and test – This phase includes the actual implementation of the different features. All the implemented features are then verified with through testing.
4. Review and planning of the next phase- In this phase, the software is evaluated by the customer. It also includes risk identification and monitoring like cost overrun and schedule slippage and after that planning of the next phase is started.

**17.Write agile manifesto principles?**

* Customer satisfaction
* Support team member
* Good design
* Measure progress
* Face to face communication
* Changing requirements
* Frequent delivery
* Measure work progress
* Continue seeking results
* Communicate regularly
* Development progress
* Reflect and adjust regularly

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

Focus on process adaptability and customer satisfaction by rapid delivery of working software product.

Break the product into small incremental build.

These builds are provided in iterations.

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.

Pros-

Functionality can be developed rapidly and demonstrated.

Suitable for fixed or changing requirements.

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.

Cons-

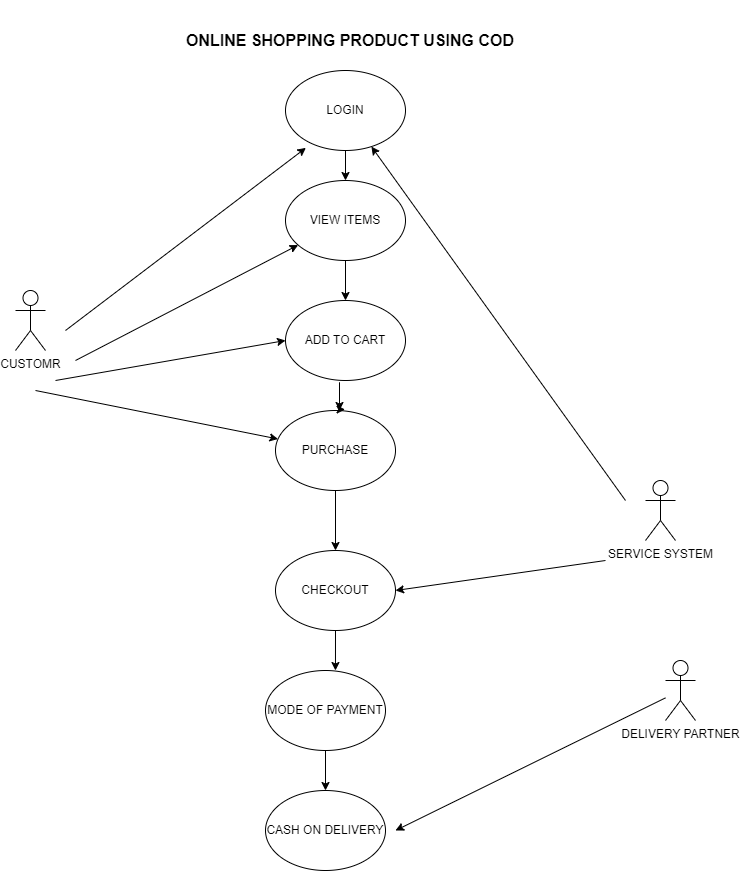
Not suitable for handling complex dependencies.

An overall plan, agile leader and agile PM practice is a must without which it will not work.

Depends heavily on customer interaction so if customer is not clear, team can be driven int the wrong direction.

There is very high individual dependency, since there is minimum documentation generated.

**19.Draw use case on online shopping product using COD?**



**20.Draw use case on online shopping product using payment gateway?**

