**Software Testing Assignment-1**

Module–1(Fundamental)

1. What is SDLC

SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

1. What is software testing

Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that software product is Defect free.

1. What is agile methodology

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.

1. What is SRS

A software requirements specification (SRS) is a document that describes what the software will do and how it will be expected to perform. It also describes the functionality the product needs to fulfill all stakeholders (business, users) needs.

1. What is oops

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.

1. Write Basic Concepts of oops

Object oriented programming is a type of programming which uses objects and classes its functioning. The object oriented programming is based on real world entities like inheritance, polymorphism, data hiding, etc. It aims at binding together data and function work on these data sets into a single entity to restrict their usage.

Some basic concepts of object oriented programming are −

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

1. What is object

An entity that has state and behavior is known as an object e.g., chair, bike, marker, pen, table, car, etc. It can be physical or logical (tangible and intangible). The example of an intangible object is the banking system.

* An object is *a real-world entity*.
* An object is *a runtime entity*.
* The object is *an entity which has state and behavior*.
* The object is *an instance of a class*.

1. What is class

A class is a group of objects which have common properties. It is a template or blueprint from which objects are created. It is a logical entity. It can't be physical.

A class in Java can contain:

1. Fields
2. Methods
3. Constructors
4. Blocks
5. Nested class and interface
6. 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.

Encapsulation in Java is the process of wrapping up of data (properties) and behavior (methods) of an object into a single unit; and the unit here is a Class (or interface).

1. What is inheritance

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

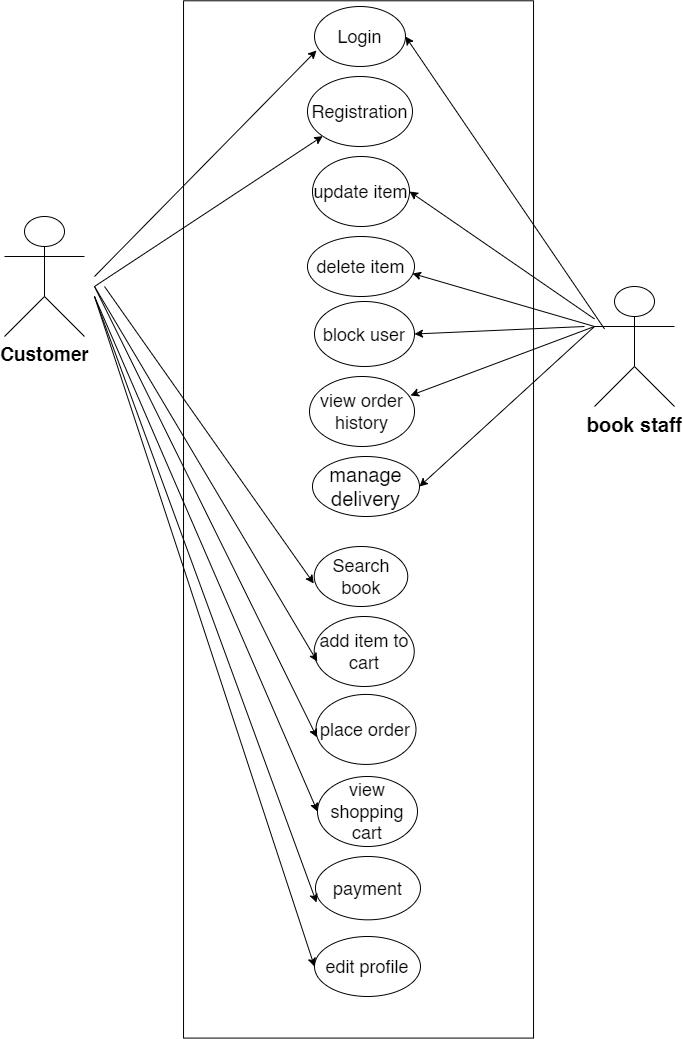
In general, Java supports single-parent, multiple-children inheritance and multilevel inheritance (Grandparent-> Parent -> Child) for classes and interfaces. Java supports multiple inheritances (multiple parents, single child) only through interfaces.

1. What is polymorphism

Poly refers to many. That is a single function or an operator functioning in many ways different upon the usage is called polymorphism. E.g. the message displayDetails() of the Person class should give different results when send to a Student object (e.g. the enrolment number).

* The ability to change form is known as polymorphism.There is two types of polymorphism in Java Compile time polymorphism(Overloading) Runtime polymorphism(Overriding)

1. Draw Usecase on Online book shopping



1. Write SDLC phases with basic introduction

1.Requirement Gathering

Requirements definitions usually consist of natural

language, supplemented by (e.g., UML) diagrams

and tables.

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 process.

* Non-functional requirements may be more critical than

functional requirements.

* If these are not met, the system is useless!

2. Analysis Phase

* The analysis phase defines the requirements of the system,

independent of how these requirements will be accomplished.

* This phase starts with the requirement document delivered by the

requirement phase and maps the requirements into architecture.

* The architecture defines the components, their interfaces and

behaviors.

* The deliverable design document is the architecture.

This phase represents the “how” phase.

3.Design Phase

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

4. Implementation Phase

* In the implementation phase, the team builds the components either from scratch or by composition Given the architecture document from the design phase and the requirement document from the analysis phase, the team should build exactly what has been requested, though there is still room for innovation and flexibility.

5. Testing Phase

6. Maintenance Phase

1. Explain Phases of the waterfall mode

The sequential phases in Waterfall model are −

* **Requirement Gathering and analysis** − All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
* **System 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.
* **Implementation** − With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
* **Integration and Testing** − All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
* **Deployment of system** − Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
* **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.

**15. Write phases of spiral model**

1. **Determine objectives and find alternate solutions –** This phase includes requirement gathering and analysis. Based on the requirements, objectives are defined and different alternate solutions are proposed.
2. **Risk Analysis and resolving –**In this quadrant, all the proposed solutions are analyzed and any potential risk is identified, analyzed, and resolved.
3. **Develop and test:** This phase includes the actual implementation of the different features. All the implemented features are then verified with thorough 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 or schedule slippage and after that planning of the next phase is started.
5. Write agile manifesto principles

* The Agile Manifesto is a document that identifies four key values and 12 principles that its authors believe software developers should use to guide their work. Formally called the Manifesto for Agile Software Development, it was produced by 17 developers during an outing on Feb. 11-13, 2001, at The Lodge at Snowbird ski resort in Utah.
* The developers called themselves the Agile Alliance. They were seeking an alternative to the existing [software development](https://www.techtarget.com/whatis/definition/software-development) processes that they saw as complicated, unresponsive and too focused on documentation requirements.

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

* Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customersatisfaction by rapid delivery of working software product.
* Agile Methods break the product into small incremental builds.
* These builds are provided in iterations.
* Each iteration typically lasts from about one to three weeks.
* 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.

Advantages of Agile SDLC

* Project is divided into short and transparent iterations.
* It has a flexible change process.
* It minimizes the risk of software development.
* Quick release of the first product version.
* The correctness of functional requirement is implemented into the development process.
* Customer can see the result and understand whether he/she is satisfied with it or not.

Disadvantages of Agile SDLC

* The development team should be highly professional and client-oriented.
* New requirement may be a conflict with the existing architecture.
* With further correction and change, there may be chances that the project will cross the expected time.
* There may be difficult to estimate the final coast of the project due to constant iteration.
* A defined requirement is absent.