**Software Testing Assignment**

**Module – 1 (Fundamentals)**

1. **What is SDLC?**

* SDLC is a structure imposed on the development of a software product that defines the process for planning, implementation, testing, documentation, deployment, and ongoing maintenance and support.
* 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.

1. **What is software testing?**

* Software Testing is a process used to identify the correctness, completeness, and quality of developed computer software.

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 is a complete description of the behavior of the system to be developed.

1. **What is OOPS?**

* Object Oriented Programming is a computer programming model that organizes software design around data or objects rather than function and logic. An object can be defined as a data field that has unique attributes and behavior.

1. **Write Basic Concepts of OOPS?**

* There are 6 Concepts OOPS.

1. Class
2. Object
3. Encapsulation
4. Inheritance
5. Polymorphism
6. Abstraction
7. **What is Object?**

* Objects are an instance of a class created with specifically defined data.

1. **What is Class?**

* Class is a collection of data member (variable) and member function (process, method) with its behavior.

1. **What is Encapsulation?**

* Encapsulation is data hiding.
* Encapsulation is wrapping up of data into single unit.

1. **What is Inheritance?**

* Inheritance is properties of parent class extends into child class.

1. **. What is Polymorphism?**

* Polymorphism is ability to take one name having many forms or multiple forms.

1. **. Draw Use case on online book shopping?**

1. **. Draw Use case on online bill payment system (Paytm).**
2. **. Write SDLC phases with basic introduction.**

* SDLC has 6 phases.

1. **Requirement Gathering.**

* All the relevant information is collected from the customer to develop a product as per their expectation.

1. **Analysis.**

* The analysis phase defines the requirements of the system, independent of how these requirements will be accomplished.

1. **Design.**

* Design Architecture Document.
* The Design team can now expand upon the information established in the requirement document.

1. **Implementation.**

* In the implementation phase, the team builds the components either from scratch or by composition.
* In the implementation stage, the team develop the customer product.

1. **Testing.**

* In this phase, the developed software is tested by tester and any defects found are assigned to developers to get them fixed.

1. **Maintenance**

* Software maintenance is one of the activates in software engineering, and is the process of enhancing and optimizing deployed software (software release), as well as fixing defects.

1. **. Explain Phases of the Waterfall model.**

* **Phases of the Waterfall model:**
* **Requirement Gathering:** During this phase, detailed requirements of the software system to be developed are gathered from client.
* **Analysis:** The analysis phase defines the requirements from client and plan how these requirements will be accomplished.
* **Design:** Plan the programming language, database and other high level technical details of the project.
* **Implementation:** Coding the software.
* **Testing:** In this phase, test the software to verify that it is built as per the specifications given by the client.
* **Maintenance:** Once client system is ready to use, you may later require change the code as per customer request.

1. **. Write phases of Spiral model.**

* **Spiral Model phases :**
* **Planning:** It includes estimating the cost, schedule and resources for the iteration. It also involves understanding the system requirements for continuous communication between the system analyst and the customer.
* **Risk Analysis:** Identification of potential risk is done while risk mitigation strategy is planned and finalized.
* **Engineering:** It includes testing, coding and deploying software at the customer site.
* **Evaluation:** Evaluation of software by the customer. Also, includes identifying and monitoring risk such as schedule slippage and cost overrun.

1. **. Write agile manifesto principles.**

* **Agile manifesto principles:**
* **Individuals and Interactions:** In agile development, self-organization and motivation are important, as are interaction like co-location and pair programming.
* **Working Software:** Demo working software is considered the best means of communication.
* **Customer Collaboration:** As the requirements cannot be gathered completely in the beginning of the project due to various factors, continuous customer interaction is very important to get proper product requirements.
* **Responding to change:** Agile development is focused on quick responses to change and continuous development.

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

* **Working methodology of agile model:**
* 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.
* Agile method break the product into small incremental builds, 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:**
* It is a very realistic approach to software development.
* Promotes teamwork and cross training.
* Functionality can be developed rapidly and demonstrated.
* Recourses requirements are minimum.
* Suitable for fixed or changing requirements.
* Delivers early partial working solutions.
* Good model for environments that change steadily.
* Minimal rules, documentation easily employed.
* Little or no planning required, easy to manage.
* Gives flexibility to developers.
* **Cons:**
* Not suitable for handling complex dependencies.
* More risk of sustainability, maintainability and extensibility.
* An overall plan, an 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 in the wrong direction.
* Transfer of technology to new team members may be quite challenging due to lack of documentation.

1. **. Draw Use case on Online shopping product using COD.**
2. **. Draw Use case on Online shopping product using payment gateway.**