Software Testing Assignment

Module–1(Fundamental)

1. What is SDLC?

* Software development life cycle, is a process for planning, creating, testing, and deploying an information system.
* It is a structured process that enables the production of high-quality, low-cost software, in the shortest possible production time.
* The goal of the SDLC is to produce superior software that meets and exceeds all customer expectations and demands.

2. What is software testing?

* Software Testing is a process used to identify the completeness, correctness and quality of developed computer software.
* Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that software product is Defect free.
* It involves execution of software/system components using manual or automated tools to evaluate one or more properties of interest.
* The purpose of software testing is to identify errors, gaps or missing requirements in contrast to actual requirements.

1. What is agile methodology?

4 What is SRS?

* A software requirements specification (SRS) is a complete description of the behavior of the system to be developed.
* It is a detailed description of a software system to be developed with its functional and non-functional requirements.
* The SRS is developed based on the agreement between customer and contractors. It may include the use cases of how user is going to interact with software system.

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.

6 Write Basic Concepts of oops

* Object
* Class
* Encapsulation
* Inheritance
* Polymorphism
* Overriding
* Overloading
* Abstraction

7 What is object

* Object is an instance /example / part of a class

1. What is class

* Class is a blueprint for an object.

1. What is encapsulation

* In Java encapsulation is a process to wrapping up of data into single unit.

1. What is inheritance

* Inheritance is the process of adapting behaviour or property of parent class “here two class are in parent -child relation”.

11 What is polymorphism

* ‘Poly’ means ‘many’ ‘morphs’ means ‘forms’

12 Draw Usecase on Online book shopping

13 Draw Usecase on online bill payment system (paytm)

14 Write SDLC phases with basic introduction

* 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.

**The Phases of the Software Development Life Cycle (SDLC)**

|  |  |
| --- | --- |
| 1)Requirement | Establish Customer Needs |
| 2)Project Planning. | Model And Specify the requirements-  “What” |
| 3) Design. | Model And Specify a Solution – “Why” |
| 4)Coding & Implementation | Construct a Solution In Software |
| 5)Testing. | Validate the solution against the  requirements |
| 6)Maintanence | Repair defects and adapt the solution to  the new requirements |

15 Explain Phases of the waterfall model ?

* Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.

**SDLC Waterfall Model**

**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.

16 Write phases of spiral model

* It has four stages or phases: The planning of objectives, risk analysis, engineering or development, and finally review. A project passes through all these stages repeatedly and the phases are known as a Spiral in the 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.

17 Write agile manifesto principles

The following 12 Principles are based on the [**Agile Manifesto**](https://www.agilealliance.org/agile101/the-agile-manifesto/).

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

2. Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.

3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

4. Business people and developers must work together daily throughout the project.

5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

7. Working software is the primary measure of progress.

8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

9. Continuous attention to technical excellence and good design enhances agility.

10. Simplicity–the art of maximizing the amount of work not done–is essential.

11. The best architectures, requirements, and designs emerge from self-organizing teams.

12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

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

* Agile methodology works by breaking projects down into little bits of user functionality, prioritizing them, and then continuously delivering them in 2-4 week cycles called iterations or sprints. Teams operate in short cycles aimed at continuous improvement to develop only what the users want.

Following are the PROS-

* It is a very realistic approach to software development
* Promotes teamwork and cross training.
* Functionality can be developed rapidly and demonstrated.
* Resource 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.
* Enables concurrent development and delivery within an

Overall

* Planned context.
* Little or no planning required
* Easy to manage
* Gives flexibility to developers

Following are the 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.
* Strict delivery management dictates the scope, functionality to be delivered, and adjustments to meet the deadlines.
* Depends heavily on customer interaction, so if customer is not clear, team can be driven in the wrong direction.
* There is very high individual dependency, since there is minimum documentation generated.
* Transfer of technology to new team members may be quite challenging due to lack of documentation.

19 Draw usecase on Online shopping product using COD.

20 Draw usecase on Online shopping product using payment gateway