Software testing Assignment

Module-1(Fundamental)

1. What is SDLC ?

Ans. The software development life cycle is a process used by the software industry to design ,develop and test high quality software. It include some phases like planning,

Analysis, design, build, testing. Deploy and maintain

1. What is agile methodology?

Ans. The agile methodology is a way to manage a project by breaking it up into several phases. It involves constant collaboration with stakeholders and continuous improvement at every stage. Once the work begins, teams cycle through a process of planning, executing, and evaluating.

1. What is SRS ?

Ans. A software requirements specification (SRS) is a document that describes the nature of a project, software or application . In simple words, SRS document is a manual of a project provided it is prepared before you kick-start a project/application.

1. What is oops ?

Ans. Object – oriented programming is a programming paradigm based on the concept of “objects”.

* An object can be defined as a data filed that has unique attributes and behavior.

* 1. write basic concepts of oops ?

Ans. OOPS aims to construct a program using classes and objects. oops concepts help the programmer control and access the data in a program easily.

* In addition, it improves the code readability and reusability. Some of the other programming languages that use the oops approach are python , ruby, c ++ , c # , and many more.
  1. What is object ?

Ans. Object is a real life entity.

* Oriented programming (oops) is a programming paradigm based on concepts of objects. the object may contain data as a form of instance variables and behavior in form of methods.

-

* 1. What is class ?

Ans. Classes are user-defined data types that act the blueprint for individual objects, attributes, and methods. Objects are instances of a class created with specifically defined data. Objects can correspond to real-world objects or an abstract entity.

* 1. What is encapsulation ?

Ans. Encapsulation is basically information hiding , It describes the idea of bundling data and methods that work on that data within one unit.

* In here access to data need to be controlled using access modifiers (public, private, protect etc.) and expose them to the outside world using getters and setters.

* 1. What is inheritance ?

Ans. Inheritance means, sub-class inherits form the super-class.

* In animal class ,there are methods and attributes that are common to all animals by using inheritance concepts other child classes can use those attributes and methods in the parent class.
* An example lion has a picture, locations and lion eat ( ) etc.

* 1. What is polymorphism ?

Ans. In simple words ,we can define polymorphism as the ability of a message to be displayed in more than one form . A real-life example of polymorphism is a person who at the same time can have different characteristics . like a man at the same time is a father , a husband and a employee.

* 1. What is RDBMS ?

Ans. The software used to store , manage ,query ,and retrieve data stored in a relational database is called a relational database management system ( RDBMS) . The RDBMS provides an interface between users and applications and the database , as well as administrative functions for managing data storage, access , and performance.

* 1. What is SQL ?

Ans. SQL is structured query language , which is a computer language for storing , manipulating and retrieving data stored in relational database.

* 1. Write SQL Commands ?

Ans.

* DDL = Data definition language
* DML = Data manipulation language
* DCL = Data control language - DQL = Data query language

16) Write SDLC phases with basic introduction ?

Ans.

1. Planning = A high level plan is being worked out with a business intent to take care the resources required for creating ,modifying or upgrading a service or the solution.

1. Analysis = The software development team works to carry on the project.

* The team holds discussions with various stakeholders from problem domain and tries to bring out as much information as possible on their requirements.

* 1. Design = In his third. phase ,software design document are prepared as per the requirement specification document
* This helps define overall system architecture.

* 1. Build = Now the real work begins the development or build phases marks the end if the initial section of the process . Additionally , this phase signifies the start of production. The development stage is also characterized by installation & change.

* 1. Testing = This phases involves system integration and system testing normally carried out by a quality assurance professional to determine it the proposed design meets the initial set of business goals .

* 1. Deploy = It includes a release specifically for a market facing group of people and gets it tested in a real-time environment for their acceptance. It is a sort of user acceptance testing (UAT).

* Focuses on fixing some usability bugs or enhancements crucial for the market perspective or can also give a green status for delivering it to the target customers.

* 1. Maintain = The last phase is when end users can fine- tune the system . if they wish, to boost performance , and new capabilities or met additional users requirement.

1. Explain phases of the waterfall model ?

Ans. a) analysis & requirement : The second phases Is where teams consider the function requirement of the project or solution. It’s also where system analysis takes place or analyzing the needs of the end users to ensure the way system can meet their expectations.

* + 1. system design : the third . phase ,software design document are prepared as per the requirement specification document

This helps define overall system architecture

* + 1. development : now the real work begin the development phase marks the end of the initial section of the process additionally . this phase signifies the start of production. The development phase is also characterized by installation & change .
    2. testing : this phase involves system integration ans system testing (of programs and procedures) normally carried out by a quality assurance (QA) professional to determine design meets the proposed design meets the initial set of business goals.
    3. implementation: the fifth phase is when the majority of the code for the program is written , and when the projects input into production by moving the data and components from the old system and placing in the new system via a direct customer

1. Write phases of spiral model ?

Ans. 1. Determine objective

* + 1. identify and resolve risks
    2. Development and test W
    3. plan the next iteration 19) Write agile manifesto principal ?

Ans.

* 1. Individuals and interactions – In agile development ,self – organization and motivation are important , as are interaction like co-location and pair programing

.

* 1. Working software – Demo working software is considered the best means of communication.

- With the customer to understand their requirement ,instead of just depending on documentation.

1. Customer collection – As the requirement 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.
2. Responding to change – agile development is focused on quick responses to change and continuous development .

1. What is join ?

Ans. JOIN is used to combine & get the data form different tables. INNER JOIN returns rows when there is a match in both tables. Left join returns all rows form the left table , even if there are no matches in the right table. RIGHT JOIN returns all rows from the right table , even if there are no matches in the left table.

1. Write type of joins ?

Ans.

1. INNER JOIN
2. LEFT JOIN
3. RIGHT JOIN
4. FULL JOIN

22) Explain working methodology of agile model and also write pros and cons ?

Ans. In agile the task are divided to time boxes (small time frames ) to deliver specific factures for a release .

* Iterative approach is taken and working software build is delivered after each iteration .
* Each build is incremental in teams of features ; the final build holds all the feature required by the customer.

Pros :- functionality can be developed rapidly and demonstrated

* Suitable foe 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:- not suitable for handling complex dependencies .

* An overall plan, an agile leader and agile PM practice is a must without which it will not work .
* Depend 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 quiet challenging due to lack of documentation .





