**Assignment 1**

**1. What is SDLC? ANS-**SDLC stands for software development life cycle model. It is describe the sequence of phase or step to develop any software.

**2. What is agile methodology? ANS-**Focus on process adaptability and customer satisfaction by rapid delivery of working software product. Break the product into small incremental builds. These builds are providing iterations. Every iteration involves cross functional team working simultaneously on various area like planning, requirements analysis, design, coding, unit testing and acceptance testing. At the end of iteration a working product is displayed to the customer and important stakeholders.

**3. What is SQL ANS-**SQL is structure Query language. Which is computer language for storing, manipulating and retrieving data store in relational database system.

**4. What is RDBMS ANS-**RDBMS stand for relation database management system is a database management system that is based on the relational model. It has the following major components: Table, row, record, field , column and attribute.

**5. What is SRS?** a. SRS stand for software requirement specification.b. a software requirements specification is a document that captures complete description about how the system is expected to perform.c. SRS is a document created by a system analyst after the requirements are collected from various customers.d. The SRS provides the requirements in technical language so that they can be comprehended and useful by the software development team to project planning, design, coding and testing.e. It should also provide a realistic basis for estimating product costs, Risk and schedules.f. It is usually signed off at the end of requirements engineering phase.

**6. Write SDLC phases with basic introduction SDLC:** It aim to produce high Quality system that meets or exceeds customer expectations, work effectively and efficiently in the current and planned information technology infrastructure and is inexpensive to maintain and cost effective to enhance. **1. Requirement Gathering**: Business analyst collects the requirement from the customer / client business needs and documents the requirement specification and provides the same to development team. **2. Analysis phase:**  Once the requirement gathering is done the next step is to define and document the product requirement and get them approved from the customer. This is done through software requirement specification document (SRS). **3. Design Phase:** High Level Design(HLD):It Gives the architecture of the software product to be developed and is done by architects and senior developers.Low Level Requirement: It done by the senior developer. It describes how each and every feature in the product should work and how every component should work.The outcome from this phase is High level document and Low level document.

****

**4. Development / coding Phase:** This is the phase where we start building the software and start writing the code for the product. The outcome from this phase is source code document and the developed product. **5.Testing Phase:**  When the software is ready. It is send to the test team. Once the QA makes sure that the software is error free, It goes to the next stage which is Implementation. The outcome of this phase is the Quality Product. **6.Deployment Phase**: Its include a release specifically for market facing group of people. And get it tested in real time environment for there acceptance . It is a sort of user acceptance testing (UAT). Focuses on fixing some useable bugs.  **7.Maintenance phase:** After the software clear all the SDLC Phase without any issues then it goes into the .Maintenance phase. It allow the customer to request for upgrade and get the fixes for problem internally and externally identification.

**7. 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. The following illustration is a representation of the different phases of the Waterfall Model.



The sequential phases in Waterfall model are –

1. **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.
2. **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.

**3. 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.

**4.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.

**5.Deployment of system** –

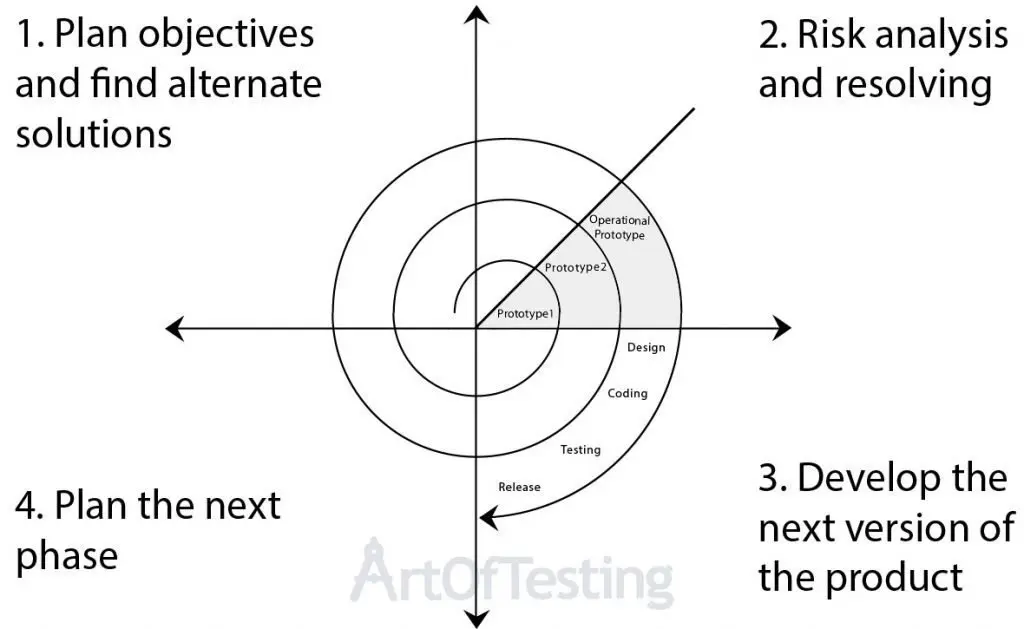
Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.

**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.

All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model, phases do not overlap.

**8. Write phases of spiral model**



The spiral model is an SDLC model that combines elements of an iterative software development model with a waterfall model. It is advisable to use this model for expensive, large and complex projects.  In its diagrammatic representation, we have a coil having many cycles or loops. The number of cycles varies for each project and is usually specified by the project manager. Each spiral cycle is a stage in the software develop The Spiral Model allows the product to be rolled out and refined in each phase of the spiral, with the ability to build prototypes in each stage. A prototype is created at the beginning of each phase as a risk management technique. The most important feature of the model is that once the project starts, it has the ability to manage unknown risks. Let’s go through the different phases of the Spiral model first and after that, we would be able to see how risk is handled in this model. **Spiral Model Phases** 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.**Developandtest:**  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.

**9. What is OOPs** 1. Identifying objects and assigning responsibilities to these objects. 2. Objects communicate to other objects by sending messages. 3. Messages are received by the methods of an object. 4. An object is like a black box. 5. The internal details are hidden. 6. Object is derived from abstract data type. 7. Object-oriented programming has a web of interacting objects, each house-keeping its own state. 8. Objects of a program interact by sending messages to each other. **10. Write Basic Concepts of oops** 1.Object 2. Class 3. Encapsulation 4 Inheritance 5. Polymorphism 6. Abstraction

**11 .What is object**

This is the basic unit of object oriented programming (OOP).

That is both data and function that operate on data are bundled as a unit called as object.

**12.What is class**

A class represents an abstraction of the object

And abstracts the properties and behaviour of that object.

**13.What is encapsulation**

Encapsulation is the practice of including in an objecte very thing it needs hidden from other objects. The internal stateis usually not accessible by other objects.

**14.What is inheritance**

**15.What is polymorphism**

Polymorphism means “having many forms”.

It allows different objects respond to the same message in different ways, the response specific to the type of the object.

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

Agile is a methodology and there are different implementations following this methodology.

Following are some of the most widely used agile implementations-

**Scrum** – Scrum is one of the most popular implementations of agile where different roles like – product owner, scrum master and team members are assigned to different participants of software development. Daily scrum meeting is organized for the updates and a build is delivered in a two to three-week cycle called a sprint.

**Extreme Programming (XP)** – Extreme programming is an agile implementation in which frequent customer feedback and changes are incorporated with a focus on quality software. The quality of the software is maintained by following the coding practices like pair programming(code reviews, unit testing, etc.) to the extreme level. Hence, the name extreme programming.

**Kanban**– Kanban is a development approach in which the tasks are organized in a Kanban board, wherein we can track the progress of the work, helping in decision making.

**Crystal Clear** – Crystal clear development like other agile methodologies focuses on frequent delivery and feedback. It is a lightweight approach based on the fact that customization of processes and practices is required to meet the project-specific requirements.

**Lean Software Development** – Lean software development methodology is based on seven lean principles – eliminate waste(like any code not adding value), amplify learning, decide late, deliver fast, empower team, build integrity and See the Whole(see the product as a whole).

**Advantages of Agile Methodology** Following are the advantages of Agile methodology-1**.**Agile is very much suited for projects where requirements and the end product is not very clear.2**.**It promotes customer satisfaction as their feedbacks and changes are embraced. 3.It reduces risk factors as early deliverables are made visible to the end-users. 4.Exhaustive planning is not required at the beginning of the development process.5**.** It is easy to manage with minimal rules and more flexibility.6.Dividing the project into incremental deliverable builds leads to more focus on the quality of the product. **Disadvantages of Agile Methodology** Following are the disadvantages of Agile methodology- 1.As it is highly customer-centric, so it can pose a problem when the customer does not have a clear understanding of the product and process.2.Lack of formal documentation and designing leads to a very high dependency on individuals for training and other tasks. 3.For complex projects, the resource requirement and effort are difficult to estimate.4. Frequent deliverables, feedback, and collaboration can be very demanding for some customers.5.Because of the ever-evolving features, there is always a risk of the ever-lasting project.

**17. Write agile manifesto principles 12 Key Principles of Agile Manifesto**

1. **Customer Satisfaction** – The first principle which is the highest priority of any project is customer satisfaction by achieving quick and continuous delivery of the software.
2. **Accommodate Changing Requirements** – Agile process easily accommodates requirements change in the development process; the changes can be introduced throughout the development period and even with the changes the software will be developed without unnecessary delays.
3. **Frequent Delivery** – The agile process encourages frequent iterations of working software in a short time (2 weeks to 2 months).
4. **Continuous Communication** – In agile, the developers and the business stakeholders are in constant communication via various meetings.
5. **Motivate Team Members** – Agile makes emphasis on motivation and encouragement of the individual resources; which ultimately helps in delivering high-quality work.
6. **Face-to-face Communication** – For efficient communication, the development teams should be co-located so that they can have face-to-face interactions.
7. **Working Software** – Measure of progress for any agile project is to deliver working software to the customer.
8. **Sustainable Development** – Agile promotes sustainable development. In each iteration, various teams involved in an agile process should maintain a constant speed at which they can deliver working software.
9. **Improve Design** – Developers and project managers should continuously focus on improving design; technical quality.
10. **Simplicity** – Agile process emphasis on keeping things simple; teams should focus only on the necessary tasks and features; and remove any complex features.
11. **Self-organizing Teams** – Agile teams need to be self-organizing and cross-functional; such teams will help to deliver good quality software by taking ownership, having good and continuous communication, making right and quick decisions, etc.
12. **Regular Assessment** – Teams should assess their working style and their past work continuously at periodic intervals to learn from their mistakes and improve their performance in the next iterations.
13. providing an efficient way to access the database.

**18. What is join?**  A join is an SQL operation performed to establish a connection between two or more database tables based on matching columns, thereby creating a relationship between the tables.

**19.Write type of join?** a. Inner join b. Right join c. Left join

**20. Write SQL Commends** 1.DDL (Data Defination lang.) 2.DML (Data Manipulation Lang.) 3. DCL (Data Control Lang.) 4. DQL (Data Query Lang.)3