**Module 1(Fundamentals)**

**Q.1. What is SDLC?**

* SDLC means Software Development Life Cycle.
* SDLC is a structure imposed on the development of a software product that defines the process of planning, implementation, testing, documentation, deployment and ongoing maintenance and support.
* SDLC is essentially a series of steps or phases , that provide a model of the development and lifecycle management of an application or piece of software.

**Q.2. What is Software testing?**

* Software Testing is a process used to identify the correctness, completeness and quality of developed computer software.
* Testing is executing a system in order to identify any gaps, errors or missing requirements in contrary to the actual desire or requirements.
* It can also be started as the process of Validating and Verifying that a software program or application or product:
* Meets the business and technical requirements that guided its design and development.
* Works as expected.
* Can be implemented with the same characteristic.

**Q.3. What is Agile Methodology?**

* Agile model is combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product.
* Agile method believes that every projects needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In this model the tasks are divided to time boxes to deliver specific features for a release.
* Agile method break the product into small incremental builds.

**Q.4. What is SRS?**

* SRS means Software Requirement Specifications.
* SRS is complete description of the behavior of the system to be developed. It includes a set of use cases that describes all the interactions that the users will have with the software.
* Use cases are also known as function requirements.
* Requirements are categorized in several ways like Functional, Non-Functional & Customer Requirements.

**Q.5. What is oops?**

* OOP means Object Oriented Programming.

**Q.6. Write Basic concepts of OOPs?**

* There are 6 basic concepts of OOPs

1) Class

2) Object

3) Inheritance

4) Polymorphism

5) Abstraction

6) Encapsulation

**Q.7. What is Object?**

* An Object is a particular instance of a class.

**Q.8. What is Class?**

* Class is a structure in which you can have member functions and member variables are there.

**Q.9. What is Encapsulation?**

* Wrapping data in to single unit.

**Q.10. What is Inheritance?**

* To access property of one class to another class.
* There are 5 types of Inheritance.

1) Single

2) Multi-level

3) Multiple

4) Hierarchical

5) Hybrid

**Q.11. What is Polymorphism?**

* Same Function name but having different Functionalities.
* There are 2 types of Polymorphism.

1) Compile Type (Overloading)

2) Run type (Overriding)

* **Abstraction :** Hiding Internal details and showing essential information to user.

**Q.12. Draw Use case on Online book Shopping?**

**Q13.Draw Use case on online bill payment system? (PAYTM)**

**Q.14. Write SDLC phases with Basic Introduction.**

* 6 Phases of SDLC :-

1) Requirement Gathering - It Establish customer need. It must be in clear and Documented form. It must be precise and easy-to-read.

2) Analysis – This phase defines the Requirements of the system, Independent of how these requirements will be accomplished. The Deliverable result at the end of this phase is a Requirement Document.

3) Design – In this phase, Design Architecture Document, Implementation Plan, Critical Priority Analysis, Performance Analysis, Test Plan. The Design Team can expand upon the information established in the requirement document. It must guide this Decision Process.

4) Implementation – In this Phase, The team builds the components either from Scratch or by composition. The Team should build exactly what has been requested, though there is still room for innovation and flexibility.

5) Testing – Here correctness, completeness and quality of a software is verified. The Testing is the separate phase which is performed by the different team after the implementation completed. If the testing is done after the-fact or continuously, testing is usually based on a regression technique split into several major focuses, namely internal, unit, application and stress.

6) Maintenance – It is the process of enhancing or optimizing deployed software as well as fixing defects. This is the phase which comes after deployment of the software in to field.

**Q.15. Explain Phases Of Waterfall Model.**

* The Classical software lifecycles Models the software development as a “step-by-step” waterfall between the various development phases.
* The Waterfall unrealistic for many reasons especially:
* Requirement must be frozen to early in the life cycle.
* Requirements are validated too late.
* Phases of Waterfall models are Requirement Collection, Analysis, Design, Implementation, Testing and Maintenance.

**Q.16. Write Phases of Spiral model.**

* 1) Planning – Determinations of objectives, alternatives and constraints.
* 2) Risk Analysis – Analysis of alternatives and identifications/resolutions of risk.
* 3) Engineering – Development of the next level product
* 4) Customer Evaluation – Assessment of the result of engineering.

**Q.17. Write Agile Manifesto Principles.**

* Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In agile the tasks are divided to time boxes (small time frames) to deliver specific features for a release.

**Q.18. Explain Working Methodology of agile model And also write Pros and Cons.**

* Agile model is 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. Each build is incremental in terms of features; the final build holds all the features required by the customer.
* Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability.
* Pros : It promotes team work & cross training , can be developed rapidly, suitable for fixed requirements, Minimum rules, documentation easily employed, little or no planning required, easy to manage and give flexibility to developers.
* Cons : More risks or sustainability, maintainability and extensibility, Not suitable for handling complex dependencies, depends heavily on customer interactions, so customer is not clear , team can be driven in the wrong direction, high individual dependency, Transfer of technology to the new team member may be quite challenging due to lack of documentation.

**Q.19. Draw Use case on online Shopping product using COD.**

**Q.20. Draw Use case on online Shopping product using Payment Gateway.**