

Q1. What is SDLC?

A1. SDLC stands for Software Development Life Cycle. It is the systematic or organized process that is used to develop a software that includes planning, preparing, implementation, testing, documentation and maintenance. It is made as per the ISO/ IEC 12207 standards.

Q2. What is agile methodology?

A2. Agile is an improvised model of Iterative & incremental method. This focusses more on the process adaptability. The work is done in smaller incremental builds. So, the final result is better.

Q3. What is SRS?

A3. SRS stands for Software Requirements Specifications. This is the complete description of the behavior of the software that has to be developed. It comprises of the set of the use cases that will show all the scenarios between the user and the software. There are 3 types of requirements that are taken care by SRS:

1. Customer requirements
2. Functional requirements
3. Non- functional requirements

Q4. What is OOPS?

A4. OOPS is Object Oriented Programming System. It is a programming system that deals with the object to design a software.

Q5. Write basic concepts of OOPS.

A5. There are 6 basic concepts of OOPS-

* Object
* Class
* Encapsulation
* Abstraction
* Inheritance
* Polymorphism

Q6. What is object?

A6. Object is the basic unit of OOPS. Object is defined as the data field that has its unique attributes (properties) and behavior. Object accesses all the properties of the class except the private properties. We can call object as the instance of a class and there can be multiple instances of a class in a program.

Object can be real or abstract but it should have defined attributes.

The first step of OOPS is to identify all the objects by the programmer and see how are they interrelated.

Example of object: pen, chair, car or a human being with properties like name, address etc.

Q7. What is class?

A7. Class is one of the basic concepts of OOPS which is a group of similar entities. It is a type of blue print for individual object, attributes and methods.

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

For e.g., we have a class called “Expensive Cars” it could have objects like BMW, VOLVO, Mercedes etc.

Its properties (data) can be price, speed.

Methods (process) that can be performed with these cars can be driving, reverse parking, applying break etc.

Q8. What is encapsulation?

A8. Encapsulation of data is wrapping of the data or code into single unit. In OOPS the variables of a class are always hidden from other classes. It can only be accessed using methods or processes of the current class.

This provides greater data security and avoids data corruption.

For e.g., in a school, a student cannot exist without a class.

Q9. What is inheritance?

A9. Inheritance is that an object acquires properties and behaviors of the parent class. It is like creating a parent-child relationship between two classes.

The main purpose of inheritance is that classes can reuse code from other classes. It reduces the development time and provides higher level of accuracy.

For e.g., vehicle is the parent class (base class/ super class) and car/ bus/ truck are the child classes (subclass or derived class).

Types of inheritance-

1. Single inheritance- in this a subclass is allowed to inherit from only one base class.
2. Multiple inheritance- in this a subclass can inherit from more than one base class.
3. Hierarchical inheritance- in this more than one child/ subclass is inherited from single base/ parent class.
4. Multilevel inheritance- in this a subclass is inherited from other subclass, which basically inherits from a parent class.
5. Hybrid inheritance- this is combination of multiple & hierarchical inheritances.

Q10. What is polymorphism?

A10. Polymorphism is a concept in which the object, variable or function takes on multiple or more than one form. It allows different objects to respond to the same message in different ways.

For e.g., in English the verb ‘run’ has different meanings depending upon with which noun it is used- laptop, foot race or business.

All have different meanings or functions of the same verb ‘run’.

There are 2 types of polymorphism-

1. Compile time (Method overloading)- when there are multiple functions with same name but different parameters, then these functions are said to be overloaded.
2. Run time (Method overriding)- method overriding occurs, when the subclass/ derived class has a definition for one of the member functions of the parent/ base class. Then the base function is said to be overridden.

Q11. What is RDBMS?

A11. RDBMS stands for Relational Database Management System. It is a database management system that is based on the relational model. It is the basis for SQL. It stores the data into row-based table structure that connects the related data elements.

Q12. What is SQL?

A12. SQL stands for structured query language. It is the standard language for storing, manipulating and retrieving data in databases. It is a domain specific language.

Q13. Write SQL commands.

A13. The commands of SQL are:

* DDL- Data Definition Language

It fetches certain records from one or more tables- SELECT

* DML- Data manipulation Language

INSERT-creates a record

UPDATE- modified record

DELETE- deletes record

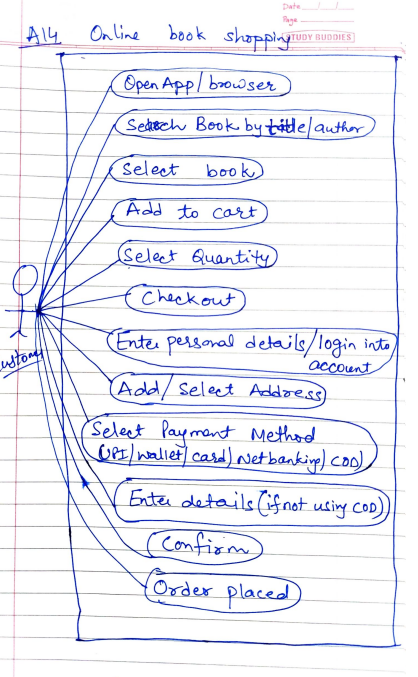
* DCL- Data Control Language

GRANT- gives a privilege to user

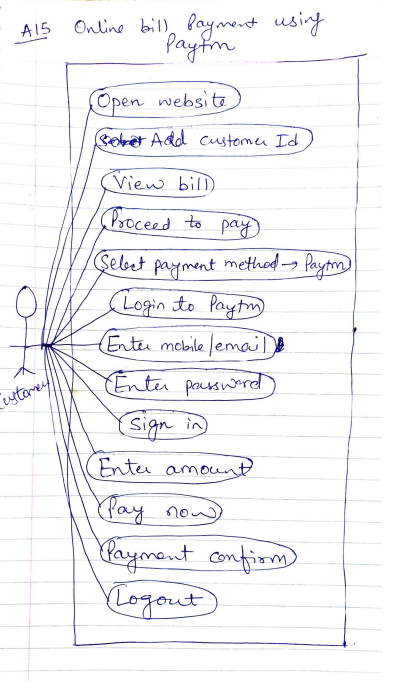
REVOKE- takes back privileges granted from user

* DQL- Data Query Language

Q14. Draw Usecase on Online book shopping.



Q15. Draw Usecase on Online bill payment system. (paytm)



Q16. Write SDLC phases with basic information.

A16. SDLC has 6 basic phases:

1. **Requirement-** this is the first phase of development. In this the needs of the customer are collected. What is the requirement, what features need to be included, time line to finish, feedbacks if required etc.

There are two types of requirements - Functional- the ones that can be worked upon or met.

Non-Functional- the requirements which are difficult to be fulfilled or could be a constraint on the system.

The common problems that arise in requirement are-

* Lack of clarity- when there is difference between the need of the customer and understanding of the information collector, that is called lack of clarity. It is usually impossible to write a document which has both clarity and easy to understand to all.
* Requirements confusion- there can be confusion to differentiate between functional & on- functional requirements. This also accounts as a problem in the requirement collection phase.
* Requirements Amalgamation- many requirements as needed by the customer should be merged to get the final need, so this amalgamation of requirements can be a problem if not understood properly.

1. **Analysis Phase**- this is the 2nd phase of SDLC. This phase has two parts:

What phase- in the what part of the analysis, the requirement document is prepared keeping in mind the need of the customer and trying to understand the goals the customer is trying to meet. The outcome of this step is the Requirement document that sums up all.

How phase- in this step the requirement architecture is made as per the Requirement document. The architecture comprises all the models, systems, interfaces, size, platform etc. that are needed to make the software for the customer.

1. **Design phase-** in this phase the design team actually designs the plan of the final software as per the requirement architecture. This design is done on the priority basis, what needs to be designed first and then what next etc. Then this design is finally ready to be implemented for final outcome.
2. **Implementation phase-** in this phase the software as per the architecture made in the design phase is made. There is room for innovation and this is flexible to be changed later, if required. The actual software that will be given to the customer back is made in this phase. They look for the performance, basics, debugging etc. in this phase as this is the final outcome.
3. **Testing phase-** Quality of any product is very important. So, in the testing phase the quality standards are checked of the product. If it works satisfactory as per the needs of the customer or not. The is an important part in SDLC. There are many types of testing available depending on the type of the product and requirement that has to be tested.
4. **Maintenance phase-** this is the final phase of the SDLC. Maintenance is an important part as it not only fixes the arising problems but also has a room for new innovations, versions that can be added to the already existing design. There are three types of maintenance-

* Corrective- it is to correct the flaws.
* Adaptive- it is to update the software or design to be adaptive to the new versions or updates.
* Perfective- it is done to add the new requirements or need to the already existing design.

Q17. Explain Phases of waterfall model.

A17. In the waterfall model all the 6 basic steps of SDLC are followed: Requirement collection, Analysis, Design, Implementation, Testing and Maintenance.

The main feature of waterfall is that is good for non-flexible, short and clearly defined projects. There is no going back to the requirement document or new features can’t be added at the later stage. These are quick to be completed, well understood and less time consuming.

The disadvantages of this model are: they are rigid in nature, not good for complex or longer projects. These have high risks as these can’t be changed at the later stage as per new requirement of the customer.

Q18. Write phases of spiral model.

Spiral model has 4 phases:

1. **PLANNING-** in the first phase, the goals and the objectives are determined. Alternatives to those goals are made and limitations or constrains are discussed.
2. **RISK ANALYSIS**- in the second phase, the alternatives are analyzed on the basis of the risk involved with them. The risks are identified and the solutions to overcome those are defined.
3. **ENGINEERING**- this is the 3rd phase of the model where the development of project or system is done. After the development, testing is also done in this phase. Basically, the product is made in this phase.
4. **CUSTOMER EVALUATION**- this is the last phase of the model. In this phase, the product is given to the customer and his feedback is taken. This is a critical phase as depending upon the feedback of the customer, further development of the product takes place.

Q19. Write agile manifesto principles.

A19. Agile manifesto has 4 principles:

* Individuals & interactions- agile prefers individuals over processes & tools. Work is divided between individuals for better progress. Individuals interact with the customer to understand their requirements and work as per that. Constant interactions are fruitful for the development of the product.
* Working software- Agile has a working software rather than just the document to work with the customer. This helps in understanding the product both for the customer and the team that is working on the development of the product.
* Customer collaboration- there is a constant participation of the customer in this model as the complete requirement is not collected in the beginning. So, every now and then the customer is interacted to get further product requirements.
* Responding to change- the last principle of agile method is that it is quick to inculcate the changes in the project. It responds quickly to the change in the requirements.

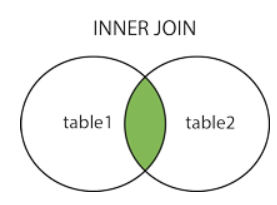
Q20. What is join?

A20.Join is a statement used to combine data or rows from two or more tables based on a common field between them. Join is used to combine records from two or more tables in the database.

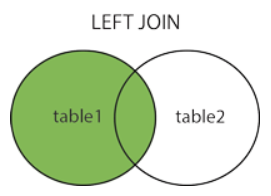
Q21. Write types of joins.

A21. There are 4 types of joins:

1. **INNER JOIN**- this join returns values, which have matching values in both the tables.

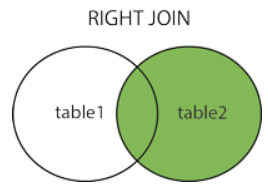


1. **LEFT JOIN-**this join returns all the rows of the table on the left side of the join and matches rows for the table on the right side of the join.

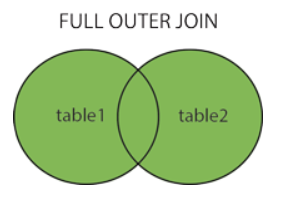


1. **RIGHT JOIN-** RIGHT JOIN is similar to LEFT JOIN. This join returns all the rows of the table on the right side of the join and matching rows for the table on the left side of the join.

Or we can say in a simple manner that it returns all records from the right table, and the matched records from the left table.



1. **FULL JOIN**- FULL JOIN creates the result-set by combining results of both LEFT JOIN and RIGHT JOIN. The result-set will contain all the rows from both tables.



Q22. Explain working methodology of agile model and also write pros and cons.

A22. Agile model works on the principle that every model is different and the method used to make the final software should be tailor made to suit the requirements of that project. The tasks in agile model are divided into smaller time frames that deliver the result simultaneously to be seen as a final project.

Iterative approach is there and a working model is made after each iteration

All the features are there in the final product.

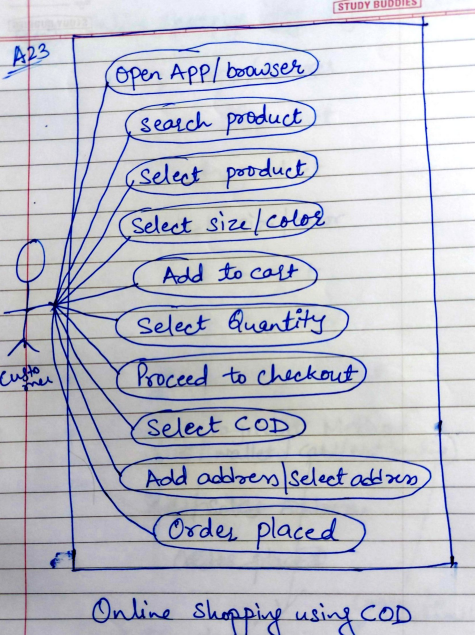
**Pros of Agile-**

* Limited or no planning is required.
* Working solutions are seen at early stage, so changes can be done easily.
* Team work is encouraged.
* It is good for both fixed & changing requirements.
* Less documentation is required.
* Easy to manage.
* As it develops simultaneously in many teams so flexibility and adaptability is there.

**Cons of Agile-**

* More risk involved.
* A large team is needed that has project manager, team leader also.
* Maintaining ongoing development is risky.
* It depends more on customer interaction which can delay the project and end result can be different.
* Hiring new team members in the ongoing project is different as less or no document is there.

Q23. Draw Usecase on Online shopping product using COD.



Q24. Draw Usecase on Online shopping product using payment gateway.

