**Que.-1 what is SDLC**

Ans.-

* SDLC Structure imposed on the development of Software product that defines the Process for Planning, Implementation, Testing, Documentation, Deployment and Ongoing Maintenance and Support. There are a number of different development models.
* SDLC Phases 1. Requirement Gathering

2. Analysis

3. Design

4. Implementation

5. Testing

6. Maintenance

**Que.-2 what is Agile** **Methodology**

Ans.-

* Agile Methodology is combination of iterative and incremental process model with focus on process adaptability and customer satisfaction by rapid delivery of working software product.
* Agile model Break the product into small incremental builds.
* This build are provided in Iteration.
* Each Iteration typically last from about one to three week.

**Que.-3 what is SRS.**

Ans.-

* A software requirements specification (SRS) is a document that describes what the software will do and how it will be expected to perform. It is also describes the functionality the product needs to fulfill all stakeholders (Business Users) needs.

SRS can summarized into Four Ds:

1. Define your product’s purpose.
2. Describe what you’re building.
3. Detail the requirements.
4. Deliver it for approval.

**Que.-4 what is OOPS.**

Ans.-

* Object- Oriented Programming is a computer programming model that organized software design around data, or objects rather than function and logic. An object can be defined as a data field that is unique attributes and behavior.

**Que.-5 write basic concept of OOPS.**

Ans.-

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

**Que-6 what is object.**

Ans.-

* An object is an instance of class.it is an entity with characteristics and behavior that are used in the object oriented programming. Object is realistic.

**Que.-7 what is class.**

Ans.-

* A class is data-type that has its own member i.e. data member and member functions. It is blueprint for an object in object oriented programming language. It is the basic building block of object oriented programming in C++.
* Class is user defined data-type.
* A class contains member like data member and member functions.
* Member function are the method that are used to manipulate data members.
* Data member define the properties of the class whereas the member function define the behavior of the class.

**Que.-8 what is encapsulation.**

Ans.-

* Encapsulation is the concept of wrapping together of data and information in single unit.
* Encapsulation is binding together the data and related function that can manipulate the data.

**Que.-9 what is inheritance.**

Ans.-

* It is capability of a class to inherit or derive properties or characteristics other class.
* It is very important and object oriented program as it allows reusability i.e. using a method defines in another class by using inheritance.

Following types of inheritance.

1. Single inheritance
2. Multiple inheritance
3. Multi-level inheritance
4. Hierarchical inheritance
5. Hybrid inheritance

**Que.-10 what is polymorphism.**

Ans.-

* Polymorphism is the ability of object oriented programming to do some works using multiple forms.
* In C++, Polymorphism achieved using two ways,

1. Operator Overloading: In operator overloading operator can have multiple behavior in different instance of usage.
2. Function overloading: Function with same name that can do multiple types based on some conditions.

**Que.-11 what is RDBMS**

Ans.-

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

**Que.-12 what is SQL?**

Ans.-

* SQL Structure is query language, which is computer language storing, manipulation and retrieving data store in relational database.
* SQL is the standard language for relation database system. All relational database management systems like MySQL, My access and Oracle, Sybase, Informix, Postures and SQL server use SQL standard database language.

**Que.-13 Write SQL Commands.**

Ans.-

1. DDL-Data Definition Language

* CREATE :- Create new table, a view table, or object in database
* ALTER :- Modified an existing database object, such as table
* DROP :- Delete an entire table, a view of a table or other object in the database

1. DML-Data Manipulation Language

* INSERT:- Create record
* UPDATE :- Modified records
* DELETE:- Delete records

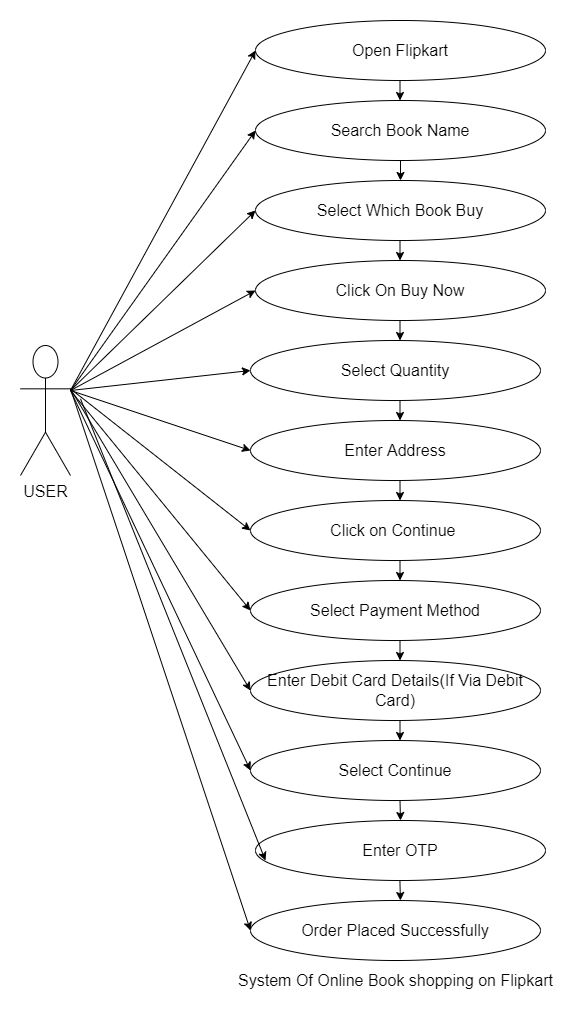
1. DCL-Data Control Language

* GRANT:- Give privilege to user
* REVOKE:- Take back privileges granted from users

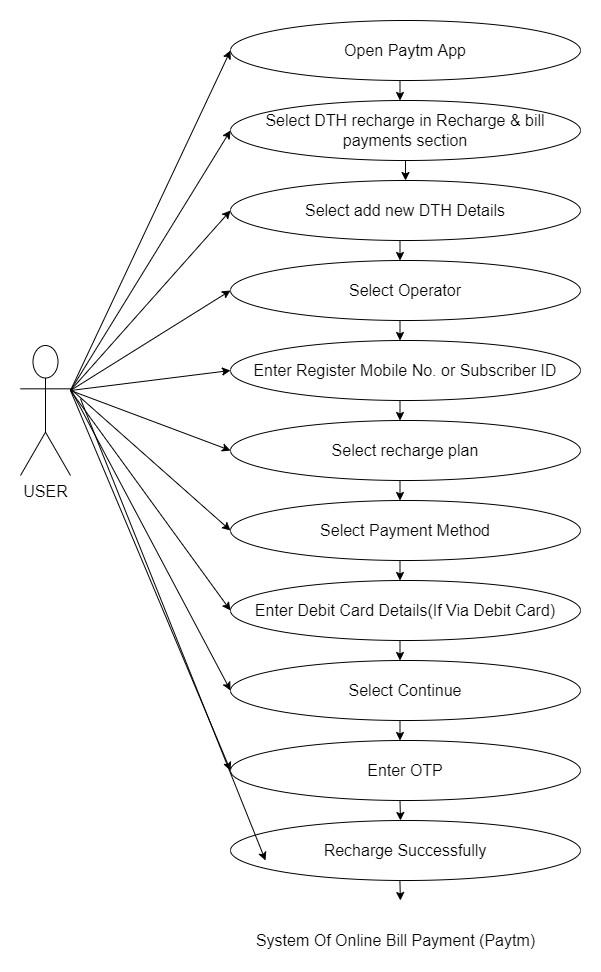
1. DQL-Data Query Language

* SELECT:- Retrieves certain record from one or more tables

**Que.-14 draw Use case on online book shopping**



**Que.-15 draw usecase on online bill payment system (paytm)**



**Que.16 write SDLC Phase with basic introduction.**

Ans.-

1. Requirement Gathering

* Establish customer needs
* Requirements definition usually consist of natural language, supplemented by diagram and tables.
* Type of requirements

1. Functional Requirements:- Describe system service or function.
2. Non-functional requirements :- are constraints on the system or development process

* Three type of problem arise

1. Lack of clarity:
2. Requirements confusion: Functional and Non-functional requirements tend to be intertwined.
3. Requirement Amalgamation: Several Different requirement to be express together.
4. Analysis Phase

* The analysis phase defines the requirements of the system, independent of how these requirements will be accomplished.
* This phase defines the problem that the customer trying to solve.
* This analysis represent the **“What”** phase.
* The deliverable result at the end of this phase is a requirement document.
* Ideally, this document states in clear and precise fashion what is to be built.

1. Design Phase

* This phase represent **“How”** Phase.
* This phase start with requirement document delivered by the requirement phase and mate the requirement to architecture.
* Details on computer programing language and environments, machine, packages, application architecture, distributed architecture layering, memory size, platforms algorithm, data structure, global type definition, interfaces and many other engineering details established.

1. Implementation Phase

* In the implementation phase, the team builds the component either by scratch or by composition.
* Given architecture document from design phase and requirements document from analysis phase, the team should build exactly what has been requested.
* The end deliverable is product itself. There are already many established techniques associated with implementation.

1. Testing Phase

* The testing phase is a separate phase which is performed by different team after implementation phase completed.
* Simply stated, quality is very important. Many companies have not learned that quality is important and deliver more claimed functionality but a lower quality level.

1. Maintenance Phase

* Maintenance is the process of changing a system after it has been deployed.
* Software maintenance is one of the activity in the software engineering and is the process of enhancing and optimizing deployed software (software release), as well as fixing defects.
* Corrective Maintenance: Identifying and repairing defects.
* Adaptive Maintenance: Adapting existing solution to the new platforms.
* Perfective Maintenance: Implementing a new requirements.

**Que.-17 write phases of waterfall model.**

Ans.-

* Requirements: The first phase involves understanding what needs to design and what is its function, purpose etc. Here, the specification of the input and output or the final product are studied and marked.
* Analysis: The analysis phase defines the requirements of the system, independent of how these requirements will be accomplished
* Design: The requirement specifications from the first phase are studied in this phase and system design helps in specifying hardware and system requirements and also helps in defining overall system architecture.
* Implementation: With input from system design, the system is first developed in small programs called units, which are integrated into the next phase.
* Testing: The analysis phase defines the requirements of the system, independent of how these requirements will be accomplished.
* Maintenance: Maintenance is the process of changing a system after it has been deployed. Software maintenance is one of the activity in the software engineering and is the process of enhancing and optimizing deployed software (software release), as well as fixing defects.

**Que.-18 write phases of spiral model.**

Ans.-

The spiral model is an SDLC model that combines elements of an iterative software development model with waterfall model. It is advisable to use this model for expensive, large and complex projects.

Phases:

1. Planning:

* Determination of objectives, alternatives and constraints.
* This phase includes requirement gathering and analysis. Based on requirements, Objective are define and different solution are proposed.

1. Risk Analysis:

* Analysis of alternative and identification/resolution of risk.
* All the proposed solution risk are analyzed and any potential risk is identified, analyzed and resolve.

1. Engineering:

* This phase includes the actual implementation of the different features. All the implemented are than verified with testing.

1. Customer Evaluation:

* 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 next phase started.

**Que.-19 write agile manifesto principles.**

Ans.-

1. Individual and Interaction: In agile development, self-organization and motivation are important, as are interaction like co-location and pair programming.
2. Working software: Demo working software is considered best means of communication with the customer to understand their requirement, instead of just depending on documentation.
3. Customer collaboration: As the requirements cannot be gathered completely in the beginning of the project due to various factors, Continues customer interaction is very important to get proper product requirements.
4. Responding to change: Agile development is focused on quick response to change and continues change.

**Que.-20 what is join?**

Ans.-

* The JOIN clause is used to combine rows from two or more tables, based on a related column between them.

**Que.21 write type of joins.**

Ans.-

1. Inner joint: Returns records that have matching values in both tables
2. Left (Outer) Join: Returns all records from the left table, and the matched records from the right table
3. Right (Outer) Join: Returns all records from the right table, and the matched records from the left table
4. Full (Outer) Join: Returns all records when there is a match in either left or right table.

**Que.-22 explain working methodology of agile model and also write pros and cons.**

Ans.-

* Agile SDLC model is a 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.
* These builds are provided in iterations.
* Each Iteration Lasts from about one to three weeks.

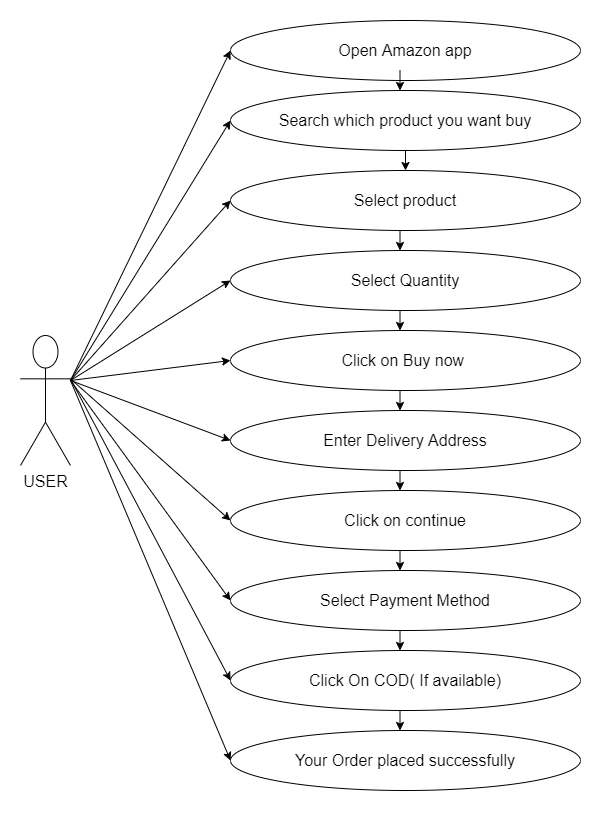
Pros:

* Easy to manage
* Give flexibility to developers
* Little or no planning required
* Is a very realistic approach to Software Developers
* Promotes team works and Cross training.
* Functionality can be rapidly and demonstrated.
* Resource required minimum.
* Suitable for fixed or changing requirements.
* Delivers early partial working solution.
* Good model for environments that change steadily.
* Minimal rules, Documentation easily employed.
* Enables concurrent development and delivery overall planned context.

Cons

* Depends on heavily on customer interaction, So if customer not clear, team can be lead in wrong direction.
* There is very high individual dependency, since there is minimum documentation generated.
* Transfer to technology to new team member may be quite challenging due to lack of
* Not suitable for handling complex dependencies.
* More risk of sustainability, maintainability and extensibility.
* An overall plan, an agile leader and agile PM practice is must without which it will not work.
* Strict delivery management dictate the scope, Functionality to be delivered and adjustment to meet the developments.

**Que.23 draw use case on online shopping product using COD.**



**Que.24 draw use case on online shopping product using payment gateway.**

