**Module–1(Fundamental)**

1. **What is SDLC**

SDLC means software development life cycle.it is structure imposed by development of software product which define process.

for instance, planning, implementation, testing, designing and maintenance.

There are 5 types of phases.

1. Analysis
2. Design
3. Implementation
4. Testing
5. Maintenance
6. **What is agile methodology?**

Agile methodology is combination of iterative model and incremental process models.

Agile methods break the product into small parts.

It is provided in iterations. As well as every iteration involves cross functional teams working simultaneously on various areas.

Like planning, analysis, design, coding, testing, and acceptance testing.

1. **What is SRS**

SRS means software requirements specification .it is a complete description of the behaviour of the system to be developed.

It includes a set of use cases that describe all of the interactions that the users will have with the software.

Moreover Use cases are also known as functional requirements.

In addition to use cases, the SRS also contains non-functional requirements.

There are 3 types of Requirements

Customer Requirements

Functional Requirements

Non-Functional Requirements

1. **What is oops**

Oops means Object Oriented Programming.

it is Identifying objects and assigning responsibilities to these objects.

The Objects communicate to other objects by sending messages.

Messages are received by the methods of an object

Object is derived from abstract data type

1. **Write Basic Concepts of oops**

There are 6 Concepts of OO

Object

Class

Encapsulation

Inheritance

Polymorphism

Abstraction

There are 2 type of polymorphism

Overriding

Overloading

1. **What is object**

object represents an individual, identifiable item, abstract.

it is both data and function that operate on data are

bundled as a unit called as object.

Object is the basic unit of object oriented programming.

1. **What is class**

Class is like blueprint for an object.

A class represents an abstraction of the object and abstracts the properties and behaviour of that object.

1. **What is encapsulation**

Wrapping of data into single unit.

Encapsulation is the practice of including in an object everything it needs hidden from other objects.

1. **What is inheritance**

Inheritance means that one class inherits the characteristics of another class. This is also called is a relationship.

Inheritance describes the relationship between two classes

There are 5 types of inheritance:

1. Single:

A //parent class

|

V

B //child class

2. Multiple:

A B

| |

-----------------

|

v

C

3. multilevel

A //grand parent

|

V

B //parent

|

V

C //child

4. hierarchical:

A

|

-----------------

| |

V V

B C

5. Hybrid:

A

|

-----------------

| |

v V

B C

| |

-----------------

|

V

D

1. **What is polymorphism**

Polymorphism means having many forms.

Poly refers too many.

That is a single function or an operator functioning in many ways different upon the usage is called polymorphism.

There is two types of polymorphism in Java.

Overloading

In this method, multiple methods having same name can appear in a class with different signature.

Overriding

In this method, same methods having same name can appear in a class with same signature.

1. **What is RDBMS**

RDBMS means Relational Database Management System.

RDBMS is the basis for SQL and for all modern database systems like that MS SQL Server, IBM DB2, Oracle, MySQL, and

Microsoft Access.

A Relational database management system is a database management system.

Most of databases are relational so tis called RDBMS.

database contains 1 or more *tables*

table contains 1 or more *records*

record contains 1 or more *fields*

fields contain the data

tables are related on common fields

**12. What is SQL**

SQL means Structured Query Language.

SQL is a language of database, it includes database creation, deletion, fetching rows and modifying rows etc.

SQL is an ANSI (American National Standards Institute)standard it is also many different versions of the SQL language.

SQL is the standard language for Relation Database System. All relational database management systems like MySQL, MS Access, and Oracle, Sybase, Informix, postures and SQL Server use SQL as standard database language.

**13. Write SQL Commands**

There are 4 types of SQL commands

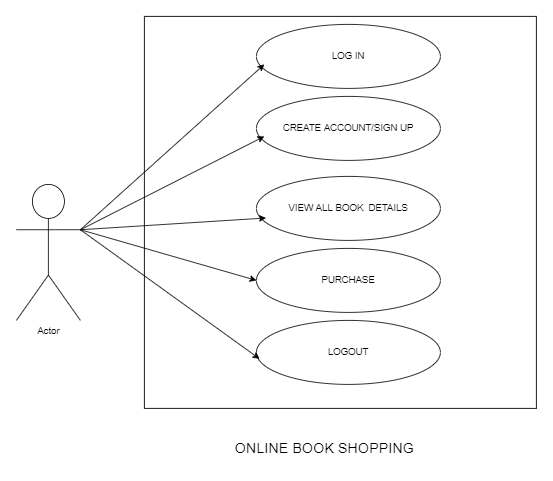
DDL – Data Definition Language

DML – Data Manipulation Language

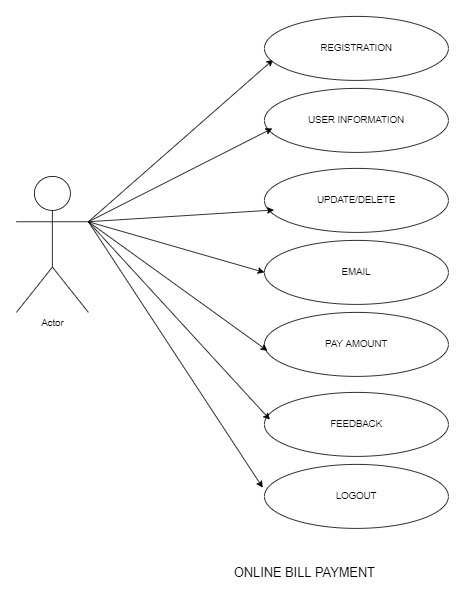
DCL – Data Control Language

DQL – Data Query Language

**14. Draw Usecase on Online book shopping**



**15.Draw Usecase on online bill payment system (paytm)**



**16. Write SDLC phases with basic introduction**

There are 5 types of SDLC phases

* Analysis :

The analysis phase defines the requirements of the system, independent of how these requirements will be accomplished.

* Design:

The Design team can now expand upon the information established in the requirement document

* Implementation:

In the implementation phase, the team builds the components either from scratch or by composition.

Critical Error Removal

The implementation phase deals with issues of quality, performance, baselines, libraries, and debugging.

* Testing:

Simply stated, quality is very important.

Many companies have not learned that quality is important and deliver more claimed functionality but at a lower quality level

* Maintenance:

Software maintenance is one of the activities in software engineering, and is the process of enhancing and optimizing deployed software as well as fixing defects.

Updating all analysis, design and user documentation

Repeatable, automated tests enable evolution and refactoring.

**17. Explain Phases of the waterfall model**

This model used in short project or website.

it is Requirements are very well documented, clear and fixed.

Product definition is stable.

it is not dynamic.

There are no ambiguous requirements.

many resources with required expertise are available to support the product.

The project is short.

* **Pros:**
* Simple and easy to understand
* Phases are processed and completed one at a time.
* Works well for smaller projects and it is very well understood.

Easy to arrange tasks.

* Process and results are well documented.

**Cons:**

* It is High amounts of risk.
* It is Not a good model for complex and object-oriented projects.
* It Cannot changing requirements.
* Poor model for long and ongoing projects.

**19. Write agile manifesto principles**

There are 4 types of manifesto principal.

**Individuals and interactions:** - in agile development, self organization and motivation are important, as are interactions like co-location and pair programming.

**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.

**Customer collaboration :**- As the requirements 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.

**Responding to change:** - agile development is focused on quick responses to change and continuous development.

**21. Write type of joins.**

There are 4 types of join.

INNER JOIN: returns rows when there is a match in both tables.

LEFT JOIN: returns all rows from the left table, even if there are no matches in the right table.

RIGHT JOIN: all rows from the right table, even if there are no matches in the left table.

FULL JOIN: rows when there is a match in one of the tables. DDL

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

**Pros:**

* It is a very realistic approach to software development
* Promotes teamwork and cross training.

Functionality can be developed rapidly.

Resource requirements are minimum.

Suitable for fixed or changing requirements

Delivers early partial working solutions

* Good model for environments that change steadily.

Minimal rules, documentation easily employed.

Enables concurrent development and delivery within an overall planned context.

Little or no planning required

Easy to manage

Gives flexibility to developers

**Cons:**

Not suitable for handling complex dependencies.

More risk of sustainability, maintainability and extensibility.

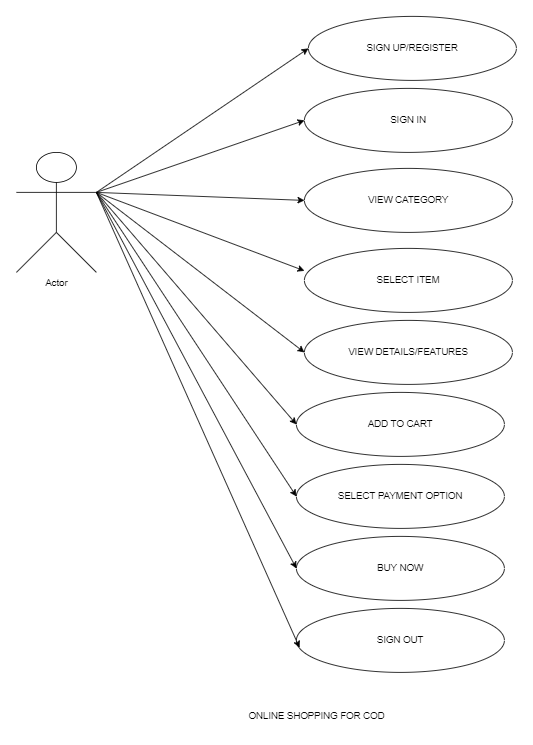
An overall plan, an agile leader and agile PM practice is a must without which it will notwork.

Strict delivery management dictates the scope, functionality to be delivered, and adjustments to meet the deadlines.

Depends 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

**23. Draw usecase on Online shopping product using COD.**



**24. Draw usecase on Online shopping product using payment gateway.**

