**Software Testing Assignment**

**Module–1(Fundamental)**

**1) What is SDLC?**

* SDLC is a structure on the development of a software to define as planning, implementation, testing, documentation, development and ongoing maintenance.

1. **What is software testing?**

* Testing is a process used to identify correctness, completeness and quality of developed computer software.

1. **What is agile methodology?**

* 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.

1. **What is SRS.**

* A software requirement specification is a complete description of the behaviour of the system to be developed.

1. **What is oops.**

* Object oriented programming system: black box, functional

1. **Write basic concept of oops**

* Object
* Class
* Encapsulation
* Inheritance
* Polymorphism
* Overriding
* Overloading
* Abstraction

1. **What is object.**

* Object is a instances of an class

1. **What is class.**

* Class in an collection of data member and member function with its behaviours

1. **What is encapsulation.**

* Encapsulation is data hiding, wrapping up of data into single until

**10) What is inheritance.**

* Properties of parent class extends into child class
* Main purpose is reusability, extendibility
* There are mainly 5 types

1. Single
2. Multilevel
3. Hierarchical
4. multiple : java does not support directly
5. hybrid : java does not support directly

**11) What is polymorphism.**

* Ability to take one name having different or many forms.
* There are mainly 2 types

1. Method overloading
2. Method overriding

**12) Write SDLC phases with basic introduction.**

* **Requirement gathering/collection** – there are 3 types of problem arise
* Lack of clarity
* Requirements confusion
* Requirements amalgamation
* **Analysis**-This phase identifies the customer problem and results in a clear requirement document that defines what need to be built.
* **Design**- This phase creates the design and implementation plan based on the requirement document including priority, performance, and test analysis.
* **Implementation**- In the implementation phase, the team builds the components either from scratch or by composition.
* **Testing phase**- Simply stated, quality is very important. Many companies have not learned that quality is important.
* **Maintenance-** Maintenance is the process of changing a system after it has been deployed
* Corrective maintenance
* Adaptive maintenance
* Perfective maintenance

**13) Explain phases of the waterfall model.**

* 6 phases of waterfall model
* Requirement gathering
* Analysis
* Design
* Implementation
* Testing
* Maintenance
* **When to use waterfall model?**
* Requirement are very well documented, clear and fixed
* Product definition is stable
* Technology is understood and is not dynamic
* There are no ambiguous requirement
* Project is short
* **Why waterfall model**
* Simple and easy to understand and use
* Easy to use due to rigidity of the model. Each phase has specific deliverable and a review process.
* Phases are processed and completed at a time.
* Clearly define stages.
* Process and results are well documented.
* **Why not waterfall model?**
* High amount of risk and uncertainty
* Not a good project for complex and object-oriented projects. Poor model for long and ongoing projects.
* It is difficult to measure progress within stages.
* Cannot accommodate c hanging requirement

**14) Write the phases of spiral model**

* Spiral mode is divided in 4 models.
* Planning
* Risk analysis
* Engineering
* Customer evaluation

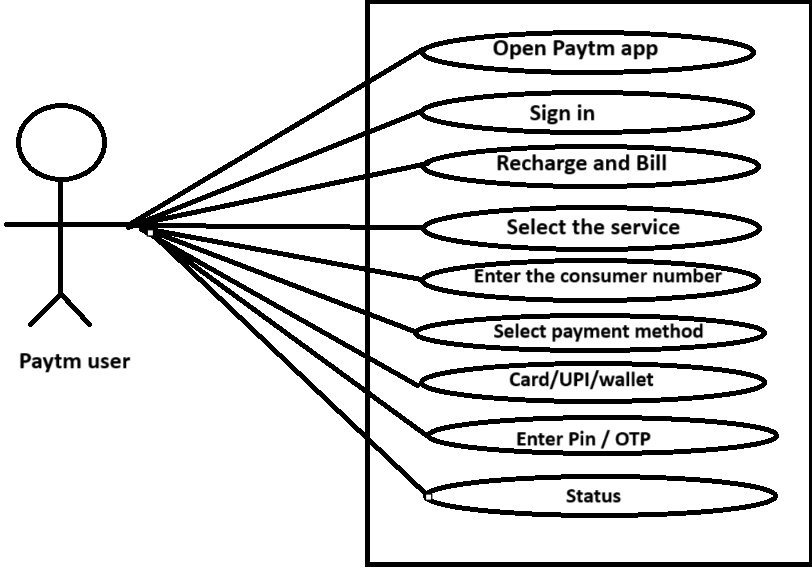
**15)Write agile manifesto principles.**

* Customer satisfaction through early and continues software delivery . welcome changing requirements anytime during development. Face to face communication is most effective. Close collaboration business and developers.

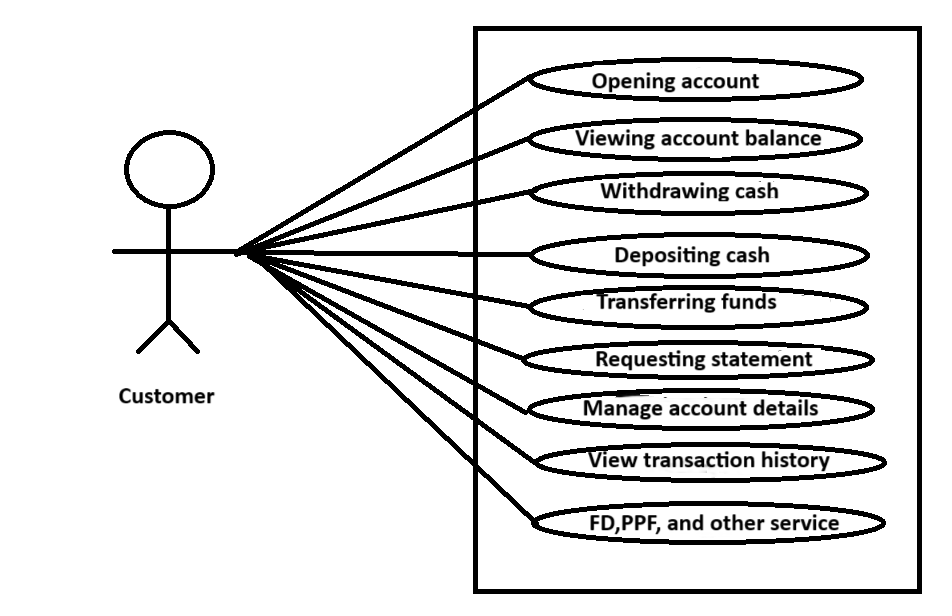
**16) Explain working methodology of agile model and also write pros and cons.**

* The project is broken into small parts called sprints. Team plan, design , development , test , deliver working software in each sprint.
* **Pros of agile**
* Resource requirements are minimum
* Suitable for fixed or changing requirements Delivers early partial working solutions.
* Flexible for developers
* Easy to manage
* **Cons of agile**
* Not good for complex dependencies
* Risky for long-term maintenance
* Strict delivery timeliness

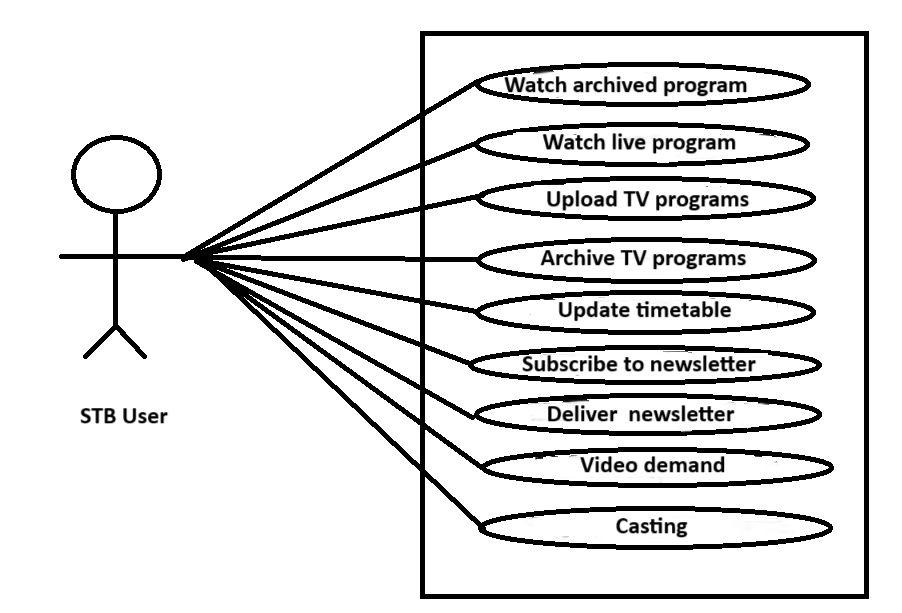
**17) Draw usecase on online bill system ( Paytm )**



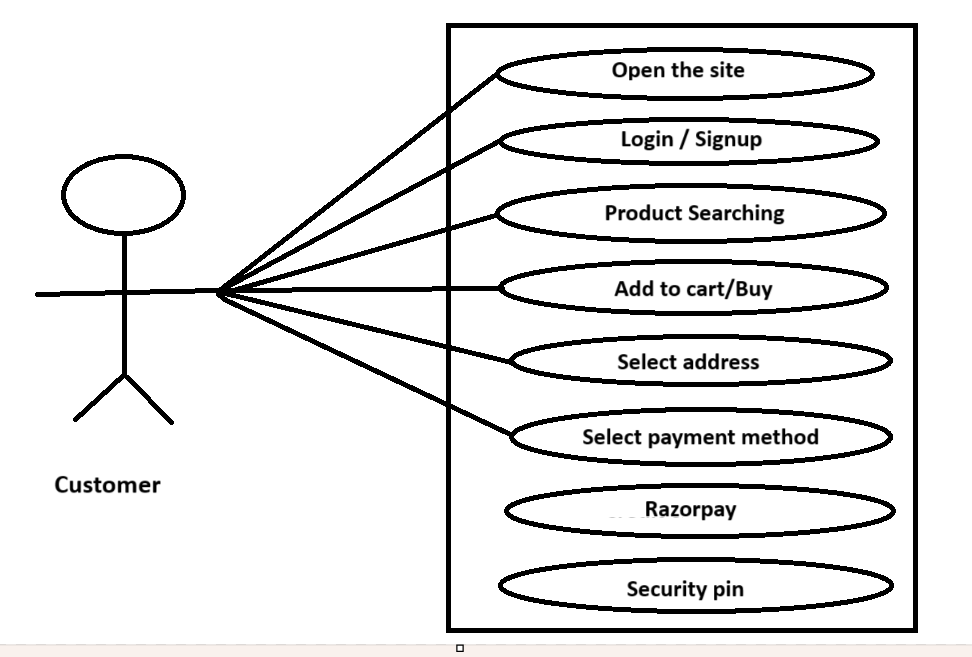
**18) Draw useccase on banking system for customers.**

****

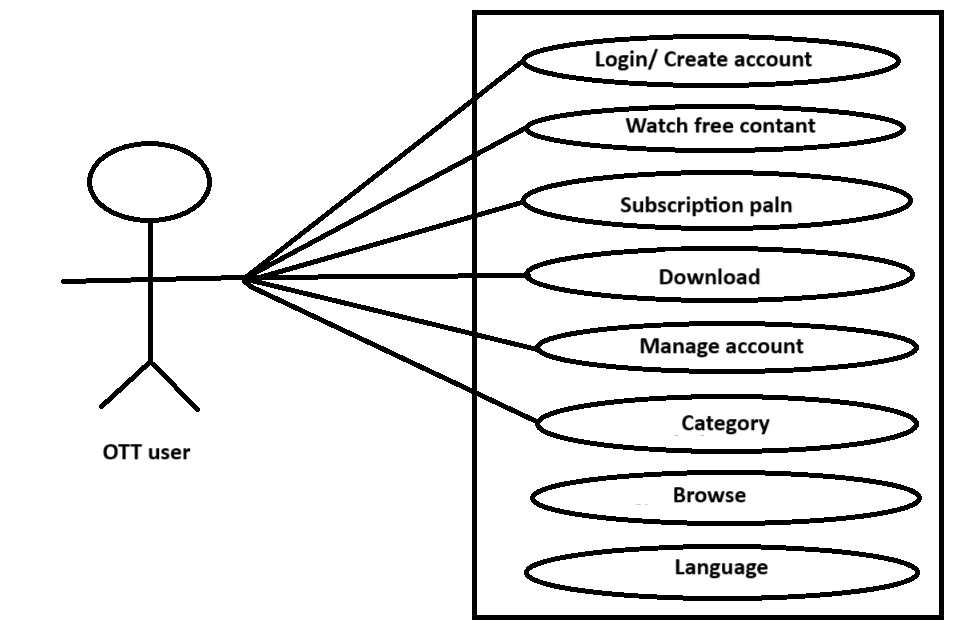
**19) Draw usecase on broadcasting system.**

****

**20) Draw usecase on Online shopping product using payment gateway.**

****

**21) Draw usecase on OTT Platform.**



**22) Draw usecase on E-commerce application.**

