**Software Testing Assignment**

**Module-1 (Fundamental)**

Q-1. What is SDLC?

* SDLC means software development life cycle.
* A SDLC is essentially a series of steps, or phases, that provide a model for the development and lifecycle management of an application or piece of software.

SDLC Phases

1. Requirements collection/Gathering
2. Analysis
3. Design
4. Implementation
5. Testing
6. Maintenance

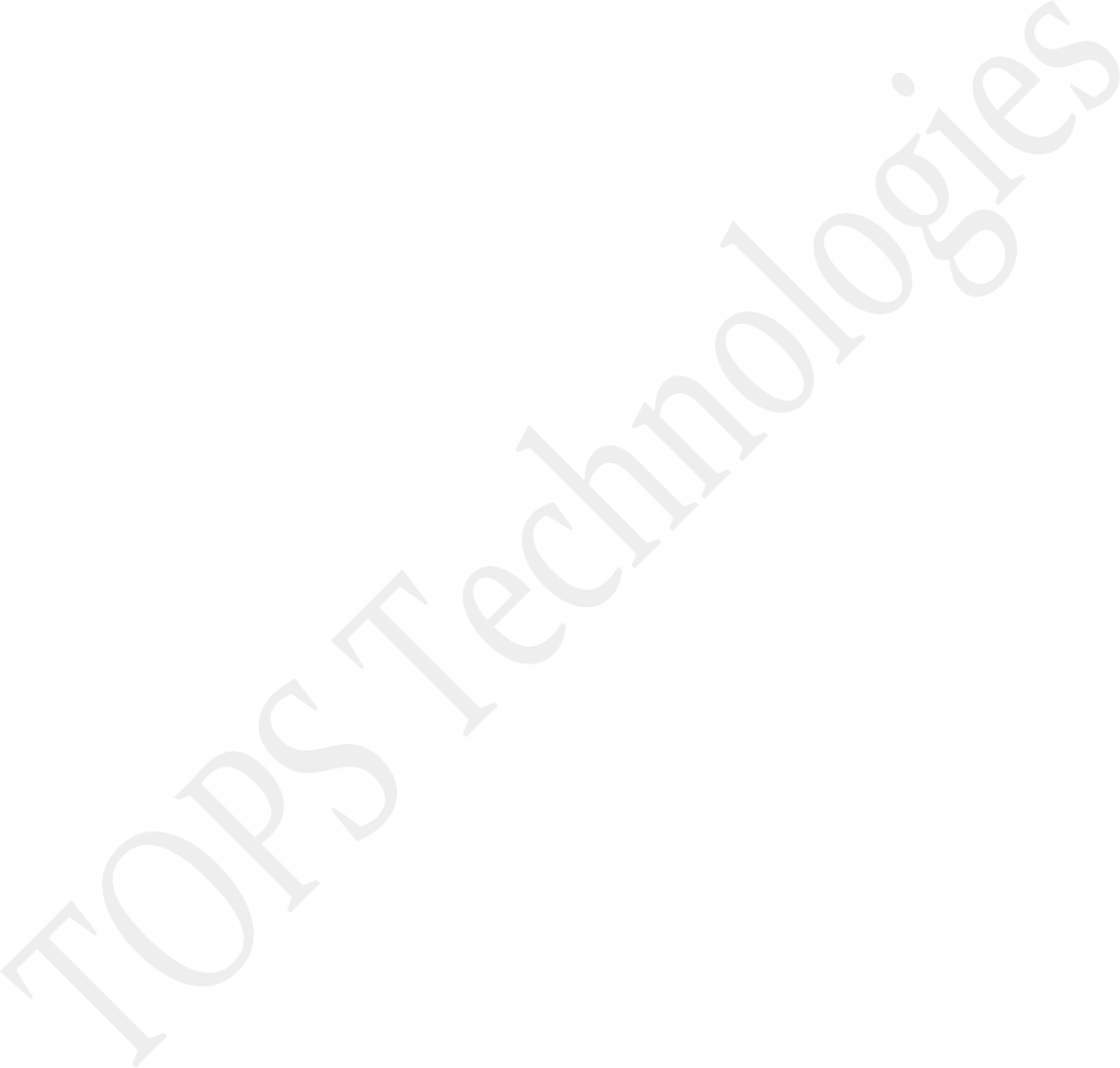
Q-2. What is software testing?

* Software testing is a process used to identify the correctness, completeness and quality of developed computer software.
* Testing is the process of evaluating system or its component with the intend to find that whether it satisfies the specified requirements or not.
* Testing can be defined as a process of analzing software item to detect the differences between existing & required condition.

Q-3. What is agile methodology?

* It is a combination iterative and incremental model.
* It divides the software into small incremental builds,this build are provided in iterations, that means the big projects are divided into small chunks(iterations).
* Each iteration last about one to three weeks.
* Each iteration involves all the team members working simultaneously on areas like planning, requirement analysis, design, coding, unit testing and acceptance testing.

Q-4. What is SRS?

* A software requirements speciﬁcation(SRS) is a complete description of the behavior 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.
* Use cases are also known as functional requirements.In addition to usecases, the SRS also contains nonfunctional (orsupplementary) requirements.
* Non-functional requirements are requirements which impose constraints on the design or implementation (such as performance requirements, quality standards, or designconstraints).

Q-5. What is oops?

* Object oriented programming is way of writing the programs in organized way.
* Object are like a black box where data are hidden.

Q-6. Write basic concepts of oops?

* Class
* Object
* Inheritance
* Polymorphism

1.over ridding

2.overloading

* Encapsulation
* Abstraction

Q-7. What is object?

* Object gives the permission to access functionality of class.

Q-8. What is class?

* Class is a collection of data member and member function.

Int a=10, b=20;

Void func(int r, int y){

A+b

}

Void func(int t, int q){

}

Q-9. What is encapsulation?

* Encapsulation is the practice of including in an object every thing it needs hidden from other objects. The internal stateis usually not accessible by other objects.
* Encapsulation is placing the data and the functions that work on that data in the same place.While working with procedural languages,it is not always clear which functions work on which variables but object-oriented programming provides you framework to place the data and the relevant functions together in the same object.

Q-10. What is inheritance?

* Making a class from an existing class.Derving the attribute of some other class.

Q-11. What is polymorphism?

* One name multiple form.
* Type: Over riding
* Same name of function with same parameter but definition will be different.
* Overloading

1.**Function overloading**:Same function name but diferent parameter.

2.**Constructor overloading**:Same constructor name but different parameter.

3.**Operator overloading**:Using the operator to add the object instead of variable operands.

Q-12. Write SDLC phases with basic introduction?

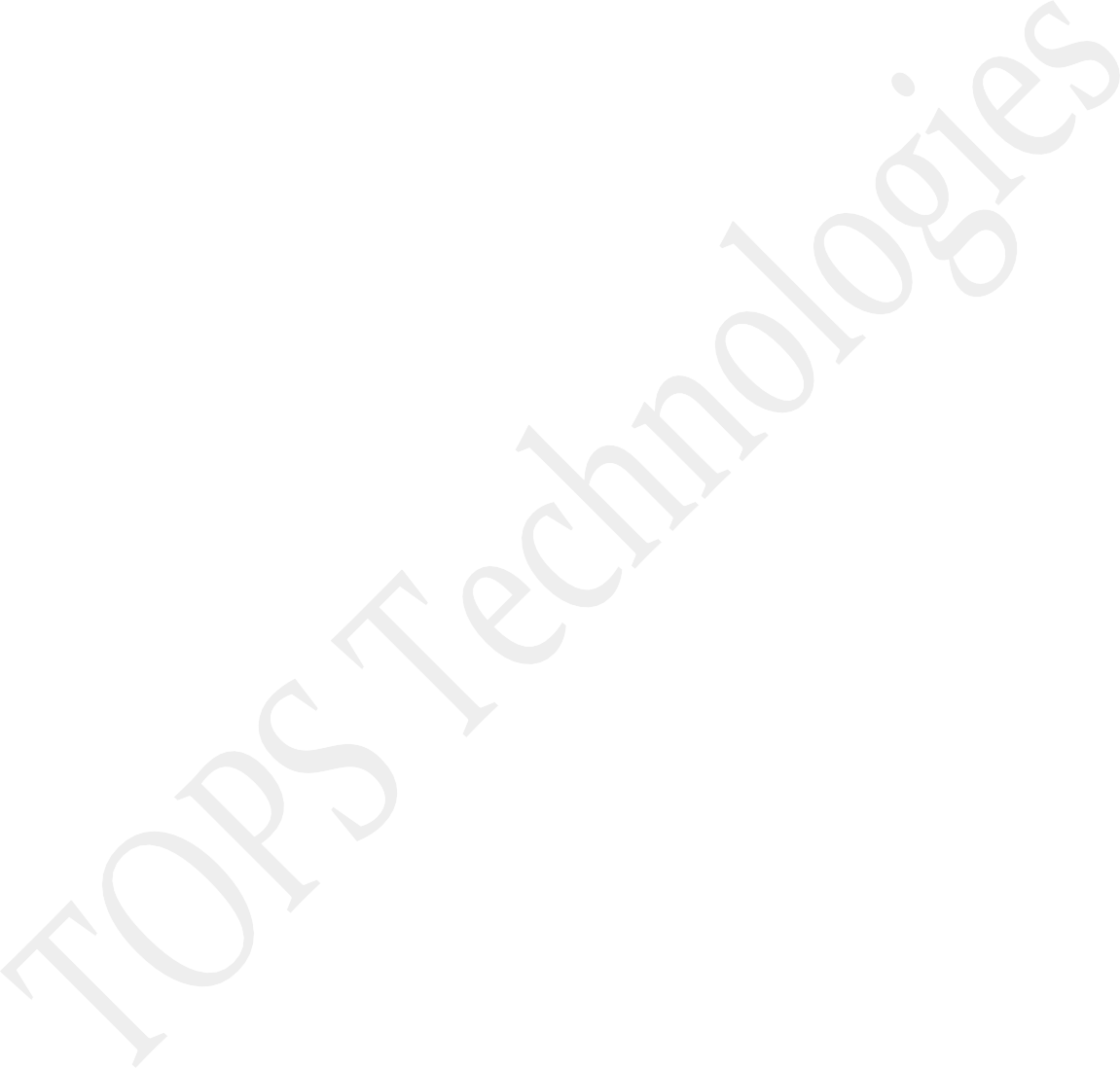
1. **Requirements collection/Gathering**

* Features
* Usagescenarios
* Although requirements may be documented in written form,they may
* Be incomplete, unambiguous, or evenincorrect.
* Requirements will Change!

###### Types of Requirements:

* **Functional Requirements**:describe system services or Functions.
* Compute sales tax on a purchase
* Update the database on the server
* **Non-Functional Requirements**:are constraints on the system or the development process.
* Non-functional requirements may be more critical than Functional requirements.
* If these are not met,the system is useless!

**2.Analysis Phase**

* The analysis phase deﬁnes the requirementsof the system, independentof how these requirements will be accomplished.
* Ideally,this document states in a clear and precise fashion what is to be built.
* This analysis represents the “what” phase.
* The requirement documentaries to capture the requirements from the customer's perspective by deﬁning goals.
* This phase starts with the requirement document delivered by the requirement phase and maps the requirements into architecture.
* The architecture deﬁnes the components,their interfaces and Behaviors.
* The deliverable design document is the architecture.
* This phase represents the “how” phase.

**3.Design Phase**

* Design architecture document
* Implementation plan
* Critical priority plan
* Performance analysis
* Test plan

**4.Implementation Phase**

* In the Implementation phase, the team builds the components either from scratch or by composition.
* Implementation – code
* Critical Error Removal
* The implementation phase deals with issues of quality, performance, Baselines, libraries, and debugging.

**5.Testing phase**

* 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.
* A customer satisfied with the quality of a product will remain loyal and wait for new functionality in the next version.
* Quality is a distinguishing attribute of a system indicating the degree of excellence.
* Regression testing
* Unit testing
* Stress testing

**6.Maintenance Phase**

* Maintenance is the process of changing a system after it has been deployed.
* **Corrective maintenance**:Identifying and repairing defect.
* **Adaptive maintenance**:Adapting the existing solution to the new platforms.
* **Perfective maintenance**:Implementing the new requirements.

Q-13. Explain Phases of the waterfall model?

* Waterfall model (classical software life cycle).
* The classical software life cycle is a the software development. as a step-by-step “ waterfall ” between the various development phases.
* This flow like water

**Requirements**

**Collection**

**Analysis**

**Design**

**Implementation**

**Testing**

**Maintenance**

* Requirement must be frozen to early in the life cycle.
* Requirement are validated too late.
* **Applications (when to use )**
* Product definition is stable.
* Requirement are very well and documented, clear and fixed.
* The project is short.
* Technology is understood and not dynamic.
* **Advantages of waterfall model**
* Quality of the product is good.
* Since requirement changes are not allowed so finding buges will less.
* Process and results are well documented.
* Phases are processed and completed one at the time.
* Clearly defined stages.
* Easy to arrange tasks.
* **Disadvantages of waterfall model**
* High amounts of risk and uncertainty.
* Not good model for complex and objects-oriented project.
* Requirement changes are not allowed.
* Total investment is more because time to take for rework on defect is time consuming to high investment.
* Testing start only after the coding.
* It is difficult progress within stages.

Q-14. Write phases of spiral model?

1. **Planning** = determination of objectives, alternatives and constraints.
2. **Risk analysis** = analysis of alternatives/resolution pf risk.
3. **Engineering** = development of the “next level” product.
4. **Customer evaluation** = assessment of the results of engineering.

* **Applications (when to use )**
* Spiral Model is very widely used in the software industry as it is in synch with the natural development process of any product.
* When costs there are a budget constraint and risk evalution is important.
* For medium to high-risk projects.
* Long-term project commitment because of potential changes to economic priorities as the requirements change with time.
* Customer is not sure of their requirements which are usually the case.
* Significant changes are expected in the product during the development cycle.
* **Advantages of spiral model**
* Changing requirement can be accommodated.
* Allow for the extensive use to prototypes.
* Users see the system early.
* **Disadvantages of spiral model**
* Management is more complex.
* End of project may not be known early.
* Not suitable for small or low risk projects and could be expensive for small projects.
* Process is complex.
* Spiral may go indefinitely.

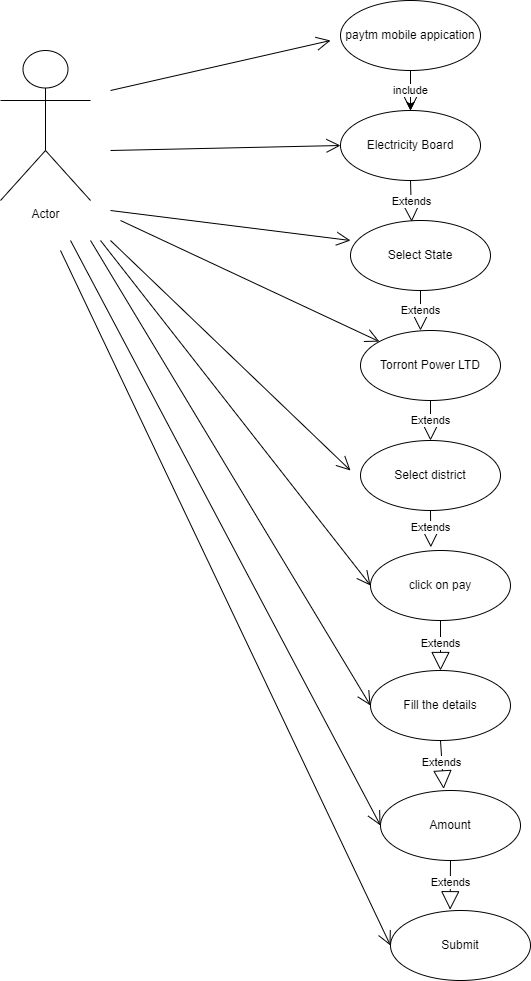
Q-15. Write agile manifesto principles?

* It is a combination iterative and incremental model.
* It divides the software into small incremental builds,this build are provided in iterations, that means the big projects are divided into small chunks(iterations).
* Each iteration last about one to three weeks.
* Each iteration involves all the team members working simultaneously on areas like planning, requirement analysis, design, coding, unit testing and acceptance testing.
* At the end of the iteration the working product is displayed to the customer or the important stake holder and it is released in the market.
* After the release we check for the feedback of the deployed software.
* If any enhancements needed in the project then it’s done and it’s re-released.

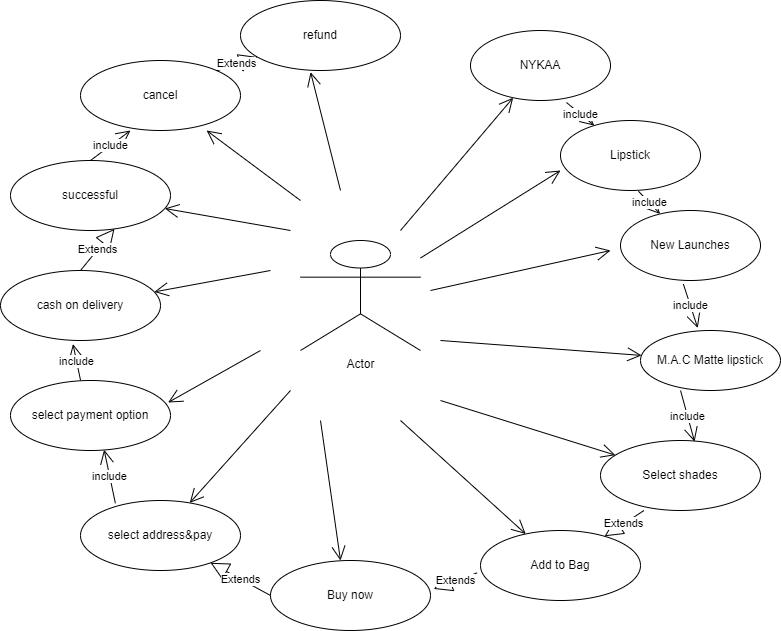
Q-16. Explain working methodology of agile model and also write pros and cons?

* It is a combination iterative and incremental model.
* It divides the software into small incremental builds,this build are provided in iterations, that means the big projects are divided into small chunks(iterations).
* Each iteration last about one to three weeks.
* Each iteration involves all the team members working simultaneously on areas like planning, requirement analysis, design, coding, unit testing and acceptance testing.
* At the end of the iteration the working product is displayed to the customer or the important stake holder and it is released in the market.
* After the release we check for the feedback of the deployed software.
* If any enhancements needed in the project then it’s done and it’s re-released.
* **Advantages of agile method**
* Frequent delivery.
* Face to face to communication with the customer.
* Less time.
* Adaptability.
* Promotes teamwork and cross training.
* Suitable for fixed and changing requirement.
* Resource requirement are minimum.
* Easy to manage.
* Gives flexibility to developer.
* **Disadvantages of agile method**
* Less documentation.
* Maintenance problems.
* Not suitable for complex dependencies.
* Transfer of technology to new member may be quite challenging due to lack of documents.

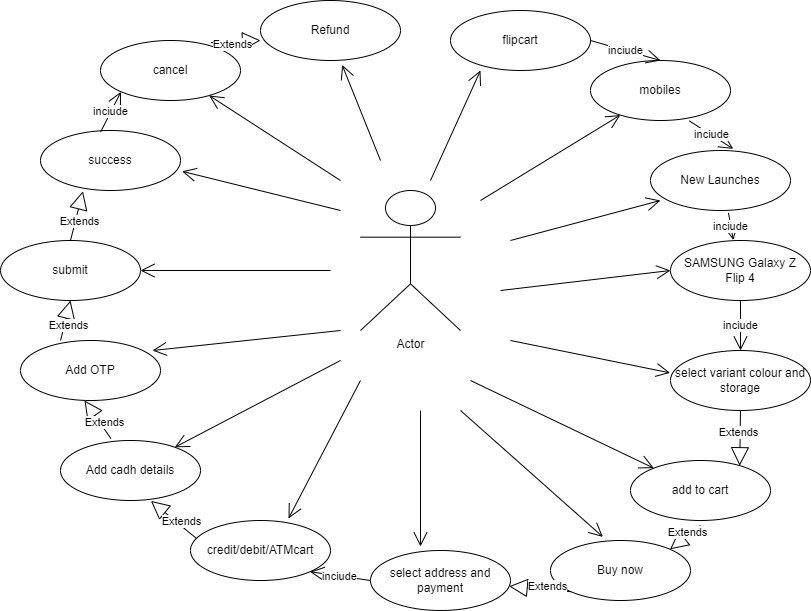
Q-17. Draw Usecase on online bill payment system (paytm)?



Q-18. Draw usecase on Online shopping product using COD?



Q-19. Draw usecase on Online shopping product using payment gateway?



Q-20. Draw Usecase on Online book shopping?

