Software Testing Assignment

Module–1(Fundamental)

- What is SDLC

ANS: - Development of software product that defines the process for Planning, implementation, testing ,documentation, deployment and ongoing maintenance and support.

-A software development life cycle is an essentially a series of steps or phase, provide that Model for the development and lifecycle management of an application or piece of software.

- What is software testing?

ANS :-Software testing is a process used to identify the correctness, completeness and quality of developed computer software

- Testing is the process of evaluating a system or its component(s) with the intent to find that whether it satisfies the specified requirements or not.

- The process consisting of all life cycle activities, both static and dynamic, concerned with planning, preparation and evaluation of software products and related work products to determine that they satisfy specified requirements, to demonstrate that they are fit for purpose and to detect defects.

- What is agile methodology?

ANS:- Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In agile the tasks are divided to time boxes (small time frames) to deliver specific features for a release.

- Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability.

Pros

Is a very realistic approach to software development Promotes teamwork and cross training.

Resource requirements are minimum.

Suitable for fixed or changing requirements Delivers early partial working solutions.

Good model for environments that change steadily.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 not work.

There is very high individual dependency, since there is minimum documentation generated.

Transfer of technology to new team members may be quite challenging due to lack of documentation.

- What is SRS?

A software requirements specification (SRS) 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. Use cases are also known as functional requirements.

Non-functional requirements are requirements which impose constraints on the design or implementation.

Types of Requirements

Requirements are categorized in several ways. The following are common categorizations of requirements that relate to technical management:

1. Customer Requirements
2. Functional Requirements
3. Non-Functional Requirements

- What is oops?

ANS :- Programming is like writing.

If you can write a demonstration, you can make a program. So, programming is also easy. But, actually, programming is not so easy, because a real good program is not easily programmed.

It needs the programmers' lots of wisdom, lots of knowledge about programming and lots of experience.

It is like writing, to be a good writer needs lots of experience and lots of knowledge about the world.

- Write Basic Concepts of oops

ANS:-

Class

Object

Encapsulation

Inheritance

Polymorphism

* Overriding

*Overloading

Abstraction

- What is object

ANS:- This is the basic unit of object oriented programming(OOP).

That is both data and function that operate on data are bundled as a unit called as Object.

- What is class?

ANS:- Blueprint for an object.

An object is a particular instance of a class which has actual existence and there can be many objects (or instances) for a class.

- What is encapsulation?

ANS:- Encapsulation is the practice of including in an object everything it needs hidden from other objects. The internal state is usually not accessible by other objects.

- What is inheritance?

ANS:- Inheritance means that one class inherits the characteristics of another class. This is also called a “is a” relationship

- What is polymorphism?

ANS:- Polymorphism means “having many forms”.

It allows different objects to respond to the same message in different ways, the response specific to the type of the object.

- Write SDLC phases with basic introduction

ANS: - 1. Requirement collection/ Gathering - Establish Customer Needs

2. Analysis - Model And Specify the requirements- “What”

3. Design - Model And Specify a Solution – “Why”

4. Implementation - Construct a Solution In Software

5. Testing - Validate the solution against the requirements

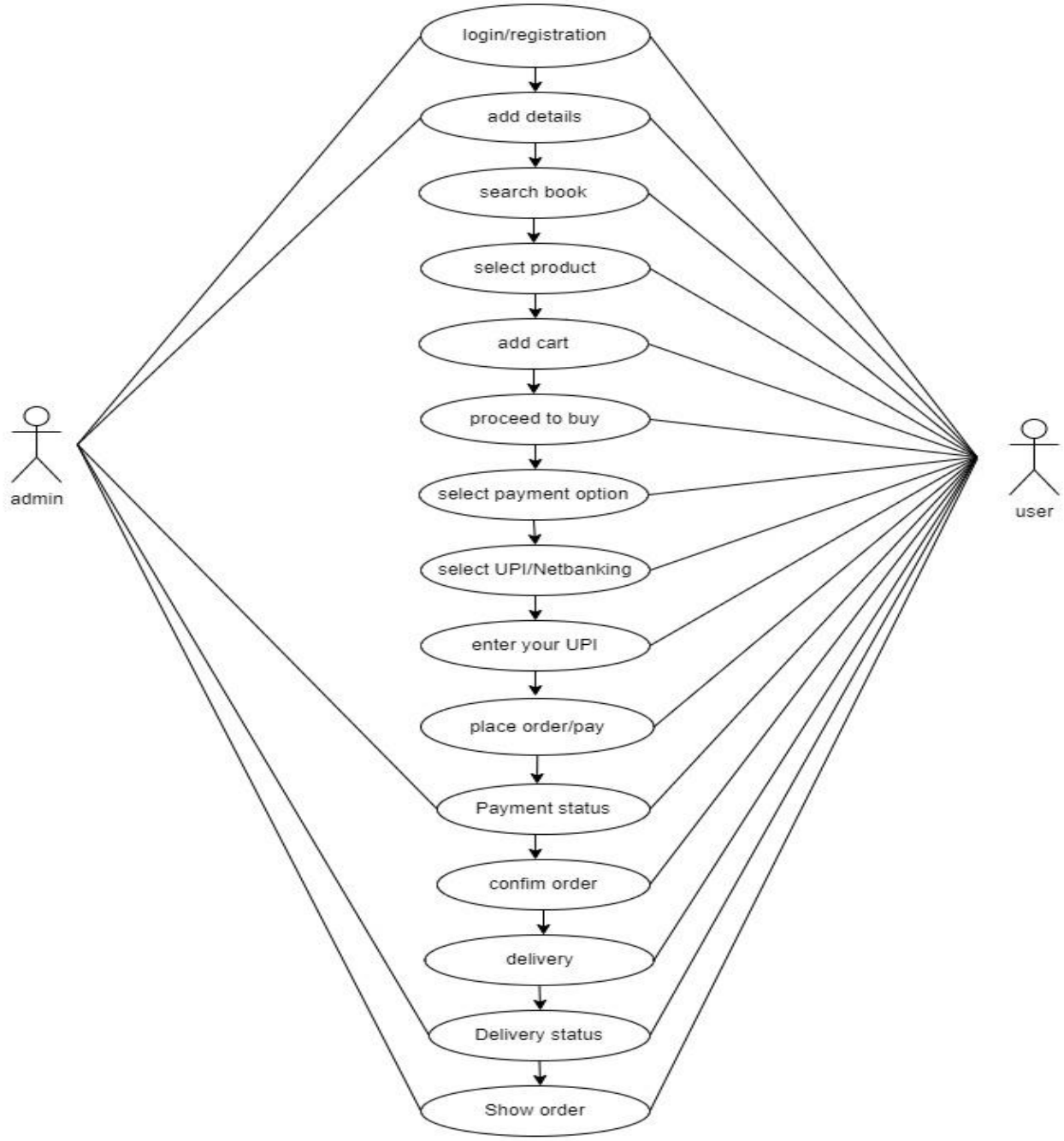
6. Maintenance - Repair defects and adapt the solution to the new requirements

- Write agile manifesto principles?

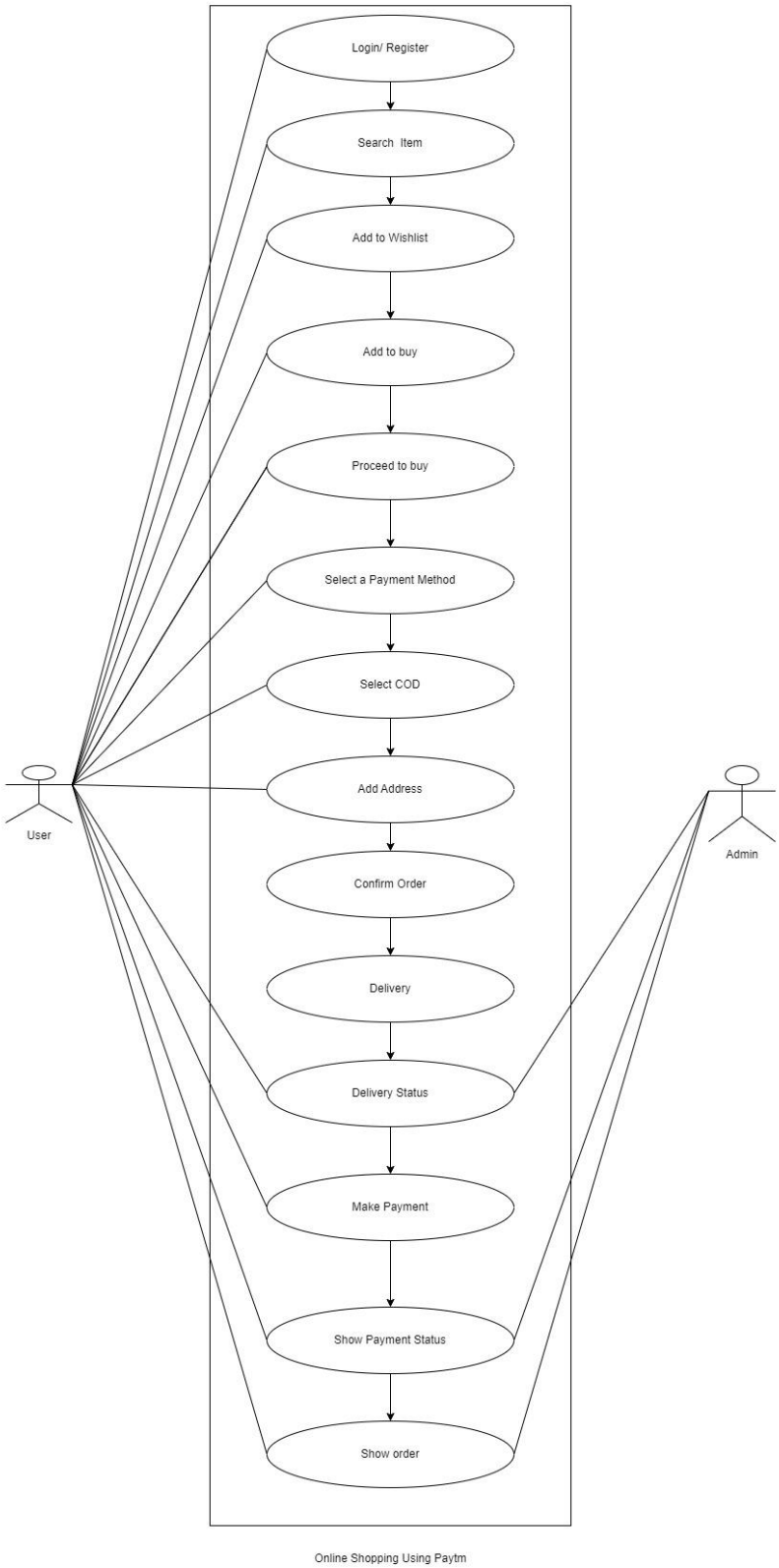
There are four Principles of Agile manifesto.

1. Individuals and interactions. In agile development self organisation and motivation are important as are interaction like co-location and pair programming.
2. Working Software. Demo working software is consider the best means of communication with the customer to understand their requirement instead of just depending on documentation.
3. Customer Collaboration As per requirement can not be gathering completely in the beginning of the project due to various factors, continues customers interaction is very important to proper product requirements
4. Responding to change Agile development is focused on quick responses to change and continuous development.

- Draw Usecase on Online book shopping



- Draw Usecase on online bill payment system (paytm)



- Explain Phases of the waterfall model?

ANS:- The waterfall is unrealistic for many reasons, especially:

- Requirements must be "frozen" to early in the life cycle
- Requirements are validated too late and The project is short.

Requirements are very well documented, clear and fixed.

Technology is understood and is not dynamic.

There are no ambiguous requirements.

Ample resources with required expertise are available to support the product.

PROS: Simple and easy to understand and use Easy to manage due to the rigidity of the model.

Each phase has specific deliverables and a review process.

Phases are processed and completed one at a time.

Works well for smaller projects where requirements are very well understood.

Process and results are well documented.

CONS:- No working software is produced until late during the life cycle.

High amounts of risk and uncertainty.

Not a good model for complex and object-oriented projects.

Poor model for long and ongoing projects.

Not suitable for the projects where requirements are at a moderate to high risk of changing. So risk and uncertainty is high with this process model.

It is difficult to measure progress within stages.

- Write phases of spiral model?

ANS:- When costs there are a budget constraint and risk evaluation 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.

Requirements are complex and need evaluation to get clarity.

Pros

Changing requirements can be accommodated.

Allows for extensive use of prototypes Requirements can be captured more accurately.

Users see the system early.

Development can be divided into smaller parts and more risky parts can be developed earlier which helps better risk management.

Cons:

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.

Large number of intermediate stages requires excessive documentation.

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

ANS:- Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements.

Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability.

Pros: Is a very realistic approach to software development

Promotes teamwork and cross training.

Functionality can be developed rapidly and demonstrated.

Resource requirements are minimum.

Suitable for fixed or changing requirements Delivers early partial working solution.

Good model for environments that change steadily.

Cons: Not suitable for handling complex dependencies.

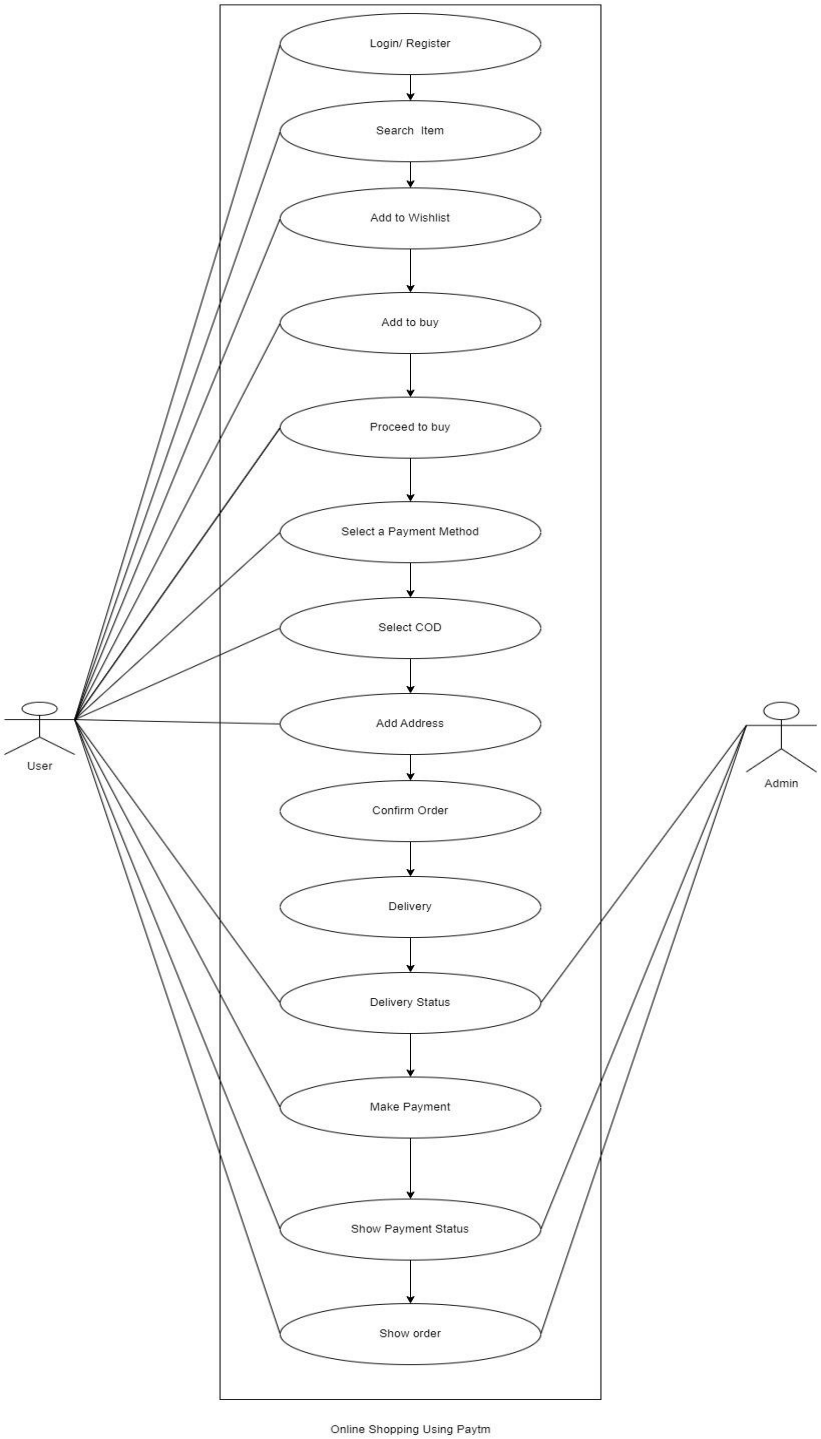
More risk of sustainability, maintainability and extensibility.

An overall plan, an agile leader and agile PM practice is a must without which it will not work.

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.

- Draw usecase on Online shopping product using COD.



• Draw usecase on Online shopping product using payment gateway.