

Software Testing Assignment

What is SDLC ?

- SDLC stands for Software Development lifecycle.
- SDLC is a process that Produces software with the highest quality and lowest cost in the Shortest time.
- SDLC consists of a Detailed plan describing how to Develop, Maintain Specific software.

What is Software Testing ?

- Software Testing is the process used to Identify the Correctness, Completeness and Quality of developed computer software.

What is Agile Methodology ?

- Agile SDLC Model is Combination of Iterative and Incremental Process Model.It is designed for faster software development.
- Agile Method breaks the product into small Incremental Builds ,This Builds are provided in iterations.
- The Goal of each Agile method is ready to change and deliver software as fast as possible.

What is SRS ?

- A *Software Requirements Specification(SRS)* is a document that describes what the software will do and how it will be expected to perform.
- It also describes the functionality the product needs to fulfill the needs of all stakeholders.

What is OOPS ?

- The Basic Concept of OOPS is to Create Object ,Re-use them throughout the program and manipulate these Objects to get results.
- OOP allows us to establish software as a collection of objects that consist of state and behaviour of object.

Write Basic Concepts of OOPS

- *Objects*
- *Class*
- *Abstraction*
- *Inheritance*
- *Polymorphism*
- *Encapsulation*

What is Object ?

- Object is Instance of Class.
- An Object in OOPS is nothing but a Self-Contained Component which consist of Method and Property to make a particular type of data .
- An Object in OOPS can include a Data structure , a Variable or a Function. It has memory Location Allocated.

What is CLASS ?

- When we define a Class , we define a blueprint for an Object.
- A class is a design or pattern from which objects are created.

What is ABSTRACTION ?

- Abstraction is the process of hiding the internal details of an application from the user.
- Abstraction is used to describe things in simple terms.

What is INHERITANCE ?

- Inheritance means one class inherits Characteristics of another Class.
- This is very useful Concept of OOPS since this feature reduce the code size.
- Code Reusability can be achieved through this concept.

What is POLYMORPHISM ?

- Polymorphism is the ability of an object to take on many forms.
- It describes the concept that you can access objects of different types through the same interface.
- Mainly there are two types
 - 1) Overloading
 - 2) Overriding

What is ENCAPSULATION ?

- Encapsulation is a way to restrict the direct access to some component of an object, so users cannot access state values for all the variables of a particular object.
- Encapsulation is the process of Wrapping data and behaviour of an object into Single unit .
- Encapsulation enables Data Hiding , Hiding Irrelevant information from the user .

Write SDLC Phases with Basic Introduction

- ☐ SDLC is a Structure Imposed On The Development Of a Software Product that defines the process of Planning, Implementation Testing Documentation Deployment, And Ongoing Maintenance And Support There Are Various SDLC Phases Describe Below.

1)Requirement Gathering

- First Phase Of SDLC model is Collect Requirement From Client.
- This Stage Gives Clear Picture of the Scope Of The Entire Project and the Anticipated Issues Opportunities, and Directives Which Triggered The Project.
- Requirement Gathering Stage need Teams to get Detailed And Precise Requirement.

Three Types Of Problem Can Arise During Requirements Gathering Phase

- Lack Of Clarity: It Is Hard To Write Documents That Are Both Precise And Easy to read.
- Requirement Confusion: Functional And Non-Functional Requirements tend To Be Intertwined.
- Requirement Amalgamation: Several Different Requirements may Be Expressed to gather.

2) Analysis Phase

- Analysis Phase Defines The Requirements of The System Independent Of How These Requirements Will Be Accomplished
- The Deliverable Result At The end Of This Phase Is Requirement Document.
- Ideally This Document States In a Clear And Precise Fashion What Is To Be built.
- The Architecture Defines The Components Their Interfaces And behaviors.
- Details On Computer Programming Languages And Environments Machines ,Packages,Application Architecture, Distributed Architecture Layering ,Memory Size Platform Algorithms Data Structures Global Type Definitions ,Interfaces And Many Other engineering Details are established.

3) Design Phase

- Design Architecture Document
- Implementation Plan
- Critical Priority Analysis
- Performance Analysis
- Test Plan
- The Design Team Can Now Expend Upon Information Established in the requirement document.
- The Requirement Document Must Guide This Decision Process.
- The Architecture Team Also Convert The Typical Scenarios Into Test Plan

4) Implementation Phase

- Implementation Start Once The Developer gets the Design Document The Software Design Is Translated Into Source Code All The Component Of The Software Are Implemented In This Phase
- The Team Should Build Exactly What Has Been Requested though There Is Still Room For Innovation And Flexibility

5) Testing Phase

- Once The Software Is Complete And It Is Deployed In The Testing Environment The Testing Team Starts Testing Functionality Of The Entire System
- This Is Done To Verify That The Entire Application Works According To The Customer Requirement
- During This Phase QA And Testing Team May Find Some Bugs/Defects Which They Communicates To Developers The Development Team Fixes The Bug And Send Back QA For Re-Test This Process Continues Until The Software Is Bug-Free Stable And Working According To The Business needs of that System.

6) Maintenance Phase

- Once The Software Testing Phase Is Over And No Bug Or Error Left In The System Then The Final Deployment Process Starts Based On The Feedback Given By the Product Manager The Final Software Is Released And Checked For Deployment Issues .
- Maintenance Is The Process Of Changing A system After It has been deployed
- Corrective Maintenance : Identifying And Repairing Defects

Explain Phases of Waterfall Model

1) Requirement Gathering Stage:

During this phase detail requirements of the software system to be developed are gather from client.

2) Design Stage:

- Plan the programming language eg JAVA,PHP, .net
- Or database like Oracle, MySQL, etc.
- Or other high level technical details of the projects.

3) Built Stage:

After design stage it is built stage, that is nothing but coding the software.

4) Test Phase:

In This Phase You Test The Software To Verify That It Is Built As Per the Specifications given by The Client

5) Deployment Phase:

Deploy The System In The Respective Environment

6) Maintenance Phase:

Once Your System Is Ready To Use You May Later Require Change The Code As Per Customer Request

Explain Phases Of Spiral Model:

1) Planning

It Includes Estimating The Cost , Schedule And Resources For The Iteration It Also Involves Understanding The System Requirements For Continuous Communication Between The System Analyst and The Customer

2) Risk Analysis:

Identification Of Potential Risk Is Done while Risk Mitigation Strategy Is Planned And Finalized

3) Engineering:

It Includes Testing Coding And Deploying Software At The Customer Site.

4) Evaluation:

Evaluation Of Software By Customer Also Includes Identifying And Monitoring Risks Such As Schedule Slippage and Cost Overrun

Write Agile Manifesto Principal

- Individual Interaction
- Working Software
- Customer Collaboration
- Responding To Change

Explain Working Methodology Of Agile Model And Also write Pros And Cons

- Agile SDLC Model Is Combination Of Iterative and Incremental Process Model Which Focus On Process Adaptability and Customer Satisfaction by Rapid Delivery Of Working Software product
- Agile Method Breaks The Product into Small Incremental Builds This Builds Are Provided in Iterations
- Each Iteration Typically Lasts From About One To Three Weeks
- At The End Of The Iteration A Working Product Is Displayed To The Customer and Important Stakeholders

Pros of Agile Model:

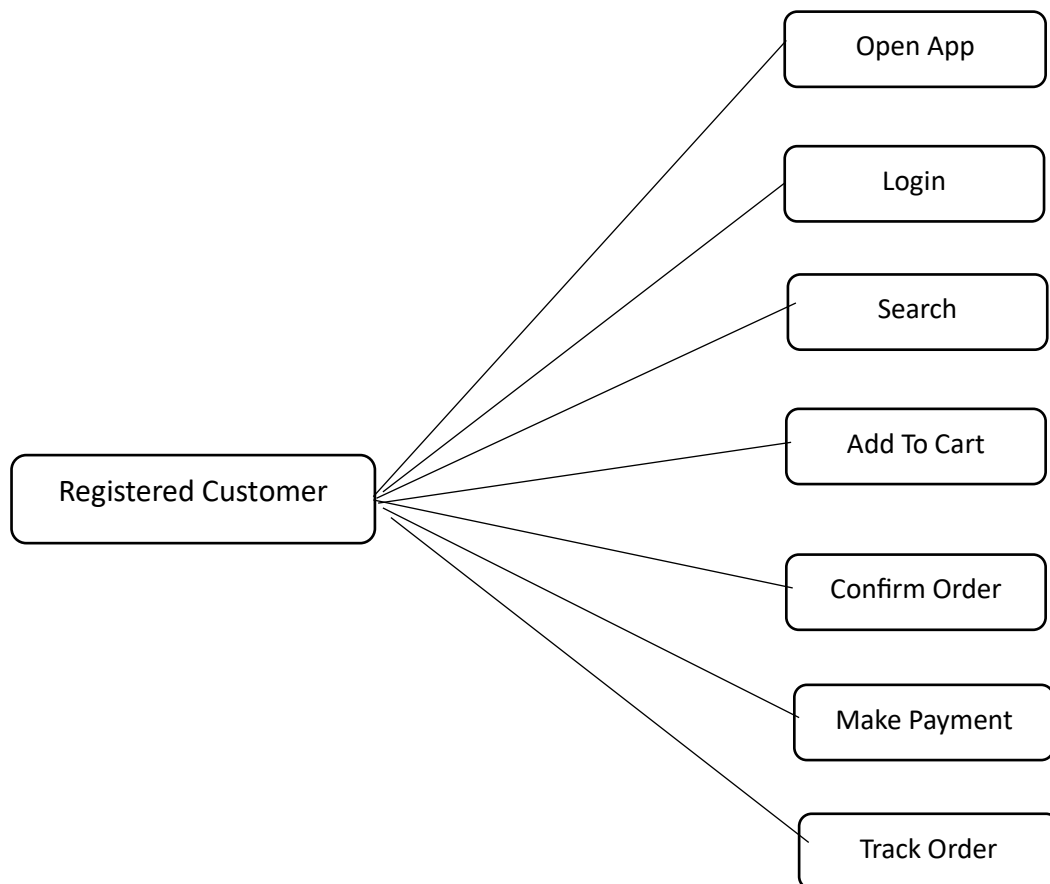
- It Is Very Realistic Approach To Software Development
- Promote Teamwork And Cross Training
- Functionality Can Be Developed Rapidly And Demonstrated
- Resource Requirements Are Minimum
- Suitable For Fixed and Changing Requirements
- Delivers Early Partial Working Solutions
- Good Model For Environments That Changes Steadily
- Minimal Rules ,Documentation Easily Employed

Cons of Agile Model:

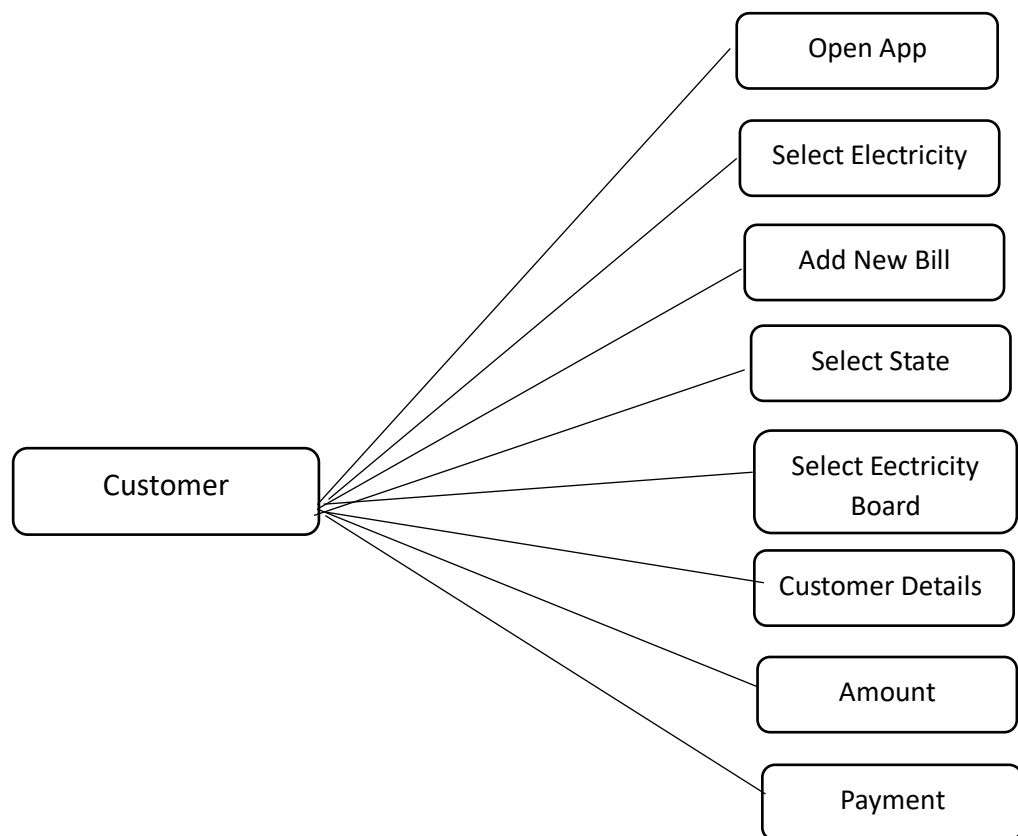
- Not Suitable For Handling Complex Dependencies
- More Risk Of Sustainability, Maintainability ,Extensibility

- 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.

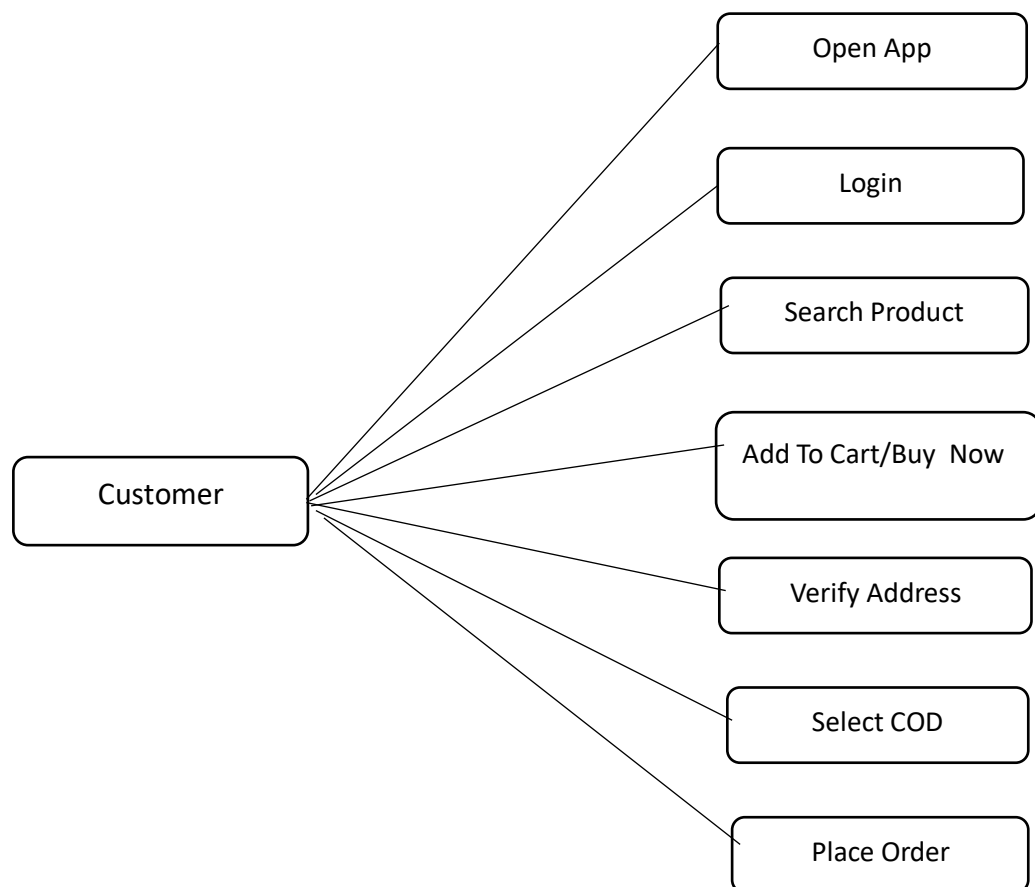
Draw use case on online book shopping



Draw use case on online Bill Payment



Draw use case on online Shopping Product Using COD.



Draw use case on online Shopping Product Using Payment gateway.

