### **Software Development Life Cycle (SDLC) - What is SDLC?**

SDLC defines the phases in the **building** of software or application.

* + - Project Planning
    - Requirement Gathering (Gathering information used to plan project, Identifying risks)
    - Design (How the application will be built)
    - Coding (developing) (Based on requirements, developers will write the application)

### **Testing**

* + - Deployment (Releasing product)
    - Maintenance (Make sure product is stable, looking at customer report on bugs and fixing it)

### **What is Software Testing?**

* + Process of executing a program or application with the intent of find software bugs using functional and automation tools
  + Process of validating/verifying a software program/application
  + Testers should **test to break** the approach, not test to pass.

### **What is the difference between STLC and SDLC?**

* + STLC is part of SDLC. It can be said that STLC is a subset of the SDLC set.
  + STLC is limited to the testing phase but SDLC is about the complete development of a software product. where the quality of software or product ensures. SDLC has vast and vital role in complete development of software or product.
  + However, STLC is a very important phase of SDLC and the final product or the software cannot be released without passing through the STLC process.
  + STLC is also a part of the post-release/ update cycle, the maintenance phase of SDLC where known defects get fixed or a new functionality is added to the software.

### **What is the requirement?**

Requirements convey the expectation of users for the software or product.

### **When does the testing start?**

* + Testing starts from testing the requirements (not after the coding phase which seems like the most likely answer.)
  + We have to make sure the requirement is correct in the first place. With the wrong requirement it is impossible to build bug-free application.

### **How to tell if the requirement is good or bad?**

* + Requirement must be (SMART)
    - **S**pecific 🡪User should be able to login. Authorized user with valid username and password should be able to login
    - **M**easurable 🡪 User should able to login very fast (in 2 second after clicking login button).
    - **A**ttainable
    - **R**ealistic
    - **T**estable 🡪 User should able to download the receipt very fast (in 2 second

### **Why do we test?**

* + To build bug free application.
  + To satisfied end user and client.
  + To build great product to generate more revenue.
  + I love testing and testing is my passion.

### **What is tester’s main responsibility?**

* + To find bug as much as possible as early as possible. Make sure most of the bug gets fixed.
  + To satisfy the end user and client by delivering bug free and user-friendly application.

### **Is 100% testing possible?**

* + We can’t test the application 100% since there are unlimited scenarios that we can’t even imagine.
  + Software testing is risk-based activity based on **priority of the functionality** we can test as much as much as possible.
  + Even though 100% testing is not possible, but I believe 100% customer satisfaction is certainly possible.

### **What is positive testing? Happy Path testing?**

* + Testing the application with valid inputs. Also called “**Happy Path**” Testing. Ex. If you log in with valid username and password it is positive testing.

### **What is negative testing? Happy Path testing?**

* + Testing the application with invalid inputs. Ex. If you log in with invalid username and password it is negative testing.

### **What is testing hierarchy?**

**Unit testing** 🡪 Developers test each module or block of code during development.

Component Testing 🡪 Component is a standalone functionality that can work by itself. Ex. Amazon Buyer Functionality, Seller Functionality, Prime Video Functionality.

**Integration Testing** 🡪 Combine all of the Functionalities. When I integrate them, can I still use all of the functions? Make sure they all still work.

**System Testing** 🡪 End-to-End testing. Test everything from beginning to end.

**Acceptance Testing** 🡪 Hire a UAT (User Acceptance Testing) Team or Business Analyst can also do Acceptance Testing. After testing has been complete you have to get another team to do acceptance testing so they can confirm the QA teams testing was successful and have the product ready for the customer.

### **What is Functional testing?**

Functional testing team can also be called manual testers it can also be done by automation team (automation functional testing). Similar to Black box testing or manual testers. Just testing the specific functionality of the application. Ex. Can user login? Can user logout? Not testing look and feel of application.

when you start to work on application, first you will do functional test as well.

### **What is non-functional testing?**

Performance testing, Security testing, Ex. Can 2000 user’s login to the application at the same time? Can user move to next page in 1 second?

### **What is unit testing? Have ever done unit testing?**

It is part of the white box testing. It’s done by the developers before they deploy the code from Development environment to QA environment.

Since it is performed by developers, I have never done unit testing yet. But I think I can learn it and do it if it is needed.

### **What is black box testing?**

Black box testing is the software testing method which is used to test the software without knowing the internal structure of code or program. In black box testing the tester only knows what the software suppose to do, they don’t care what is going on inside the box, how it works and what is the logic. They only cares if the software doing it is job.

### **What is Integration Testing?**

Integration testing is type of black box testing. Integration testing tests after integration component if the system works fine or not. Single component might work fine by itself but when integrated with multiple component it might fail. That’s why framework need to integration test after integrated something.

Integration testing is done after unit testing.

### **What is Acceptance testing? Called user acceptance testing**

The Acceptance testing will be performed after QA testing or system testing. After the system test has corrected all or most defects, the system will be delivered to the user or customer for acceptance testing. Acceptance testing is basically done by the user or customer although other stakeholders may be involved as well.

In some companies, it is done by UAT team. After UAT team performing the acceptance testing the code will go to production.

Development environment( where developers write code and perform unit testing)

QA environment (where we test the application.)

UAT environment (after the code is tested QA environment it will be deployed to the UAT environment. UAT testing team will perform testing to make sure it fits the business requirement. It is also called staging environment.

Production environment( is when the end, user can see the real application)

### **Software Testing Life Cycle (STLC) - What is STLC?**

STLC defines the phases in **testing** of software or application development. In STLC process in different activities are carried out to improve the quality of the product.

* + Requirements review
  + Test planing
  + Test designing
  + Test environment setup
  + Test execution
  + Test reporting

### **Smoke Test** 🡪

**TESTING order:** Code 🡪 Unit Testing 🡪 Integration Testing 🡪 Sanity Testing 🡪***Smoke Testing*** 🡪 Functional Testing

In our project, there are five modules like; login, view user, user detail page, new user creation and task creation

In these five modules, the developer will first perform the smoke testing by executing all the major functionality of modules like; user is able to login with valid login credentials or not, after login new user can be created or not, user that is created is viewed or not etc.

### **What is Agile?**

Agile is a framework of values and principles to manage teams and projects. Agile methodology is a practice that helps continuous iteration of development and testing in the software development process.

Scrum : Team plans for amount of work for the next sprint

Kanban : No sprint planning, stories are picked up as is, but you still have everything else

### **Why do we need Agile? Waterfall and Agile?**

Because waterfall methodologies have following disadvantage;

Requirement cannot be change or hard to change once document is signed.

In waterfall before completing the one phase you can’t move to the next phase. For example, before coding phase is completed testing cannot be started.

Customer can’t see what they are going to get until very late stage in development life cycle.

It takes longer time to go to the production. By the time product goes to the market it might be outdated already.

Agile has following advantages :

The change is welcomed. For example, after the sprint demo if client does not like something, we can take their feedback and improve the product. Requirement change is OK.

Since it is iterative development process, the development team can developed piece of functionality, get feedback and improve next iteration. So, the product will be continuously improve.

Waste is eliminated in agile with the help of scrum master. For example, if I am blocked, I don’t have to wait and waste my time. Since team members communicates with each other efficiently we can be more productive by preventing duplicated effort.

Waterfall emphasizes tools and platform, like C# .NET, but agile emphasizes people. You can have best tool but at the end people are using those tools. I believed inspired people can make amazing products even they have less money or less resources.

### **What kind of Agile methodology did you use in your previous projects?**

### **Scrum is an Agile framework, right? Name few other Agile frameworks.**

Yes, Scrum is an Agile framework. Few other Agile frameworks are –Feature Driven Development Test Driven Development, Kanban

### **What are the different roles in Scrum? Scrum roles?**

Product owner

* Responsible product backlog
* Define the features of the product
* Decide on release date and content
* Prioritized user stories.
* just features and priority every iteration, as needed
* Accept or reject work results in the **sprint review meeting.**

Scrum team or development team

* + Development Team is the portion of the team that creates the product

Scrum master – Scrum master is the leader and the coach for the scrum team who checks whether the scrum team is executing committed tasks properly.

* He is also responsible to increase the efficiency and productivity of the team so that they can achieve the sprint goal effectively.
* Responsible for the Scrum process
* Removes impediment
* Facilitates scrum events
* Facilitates communication

### **How do you describe a scrum team?**

the team is a group of people who are sharing the same goal , moving to the same direction , who trust each other and who will effectively communicate and collaborate with each other to build great product. There should be no star individual but a star team.

### **What are the responsibilities of a Scrum Master?**

### Responsible from the scrum process

### Tracking and monitoring team and events

### Understanding requirements properly

* Work to reach the project goal
* Process checking master and quality master
* Improving the performance of the team
* Lead the meetings and resolve issues
* Team Communication and reporting

### **What is User Story?**

*(Note: basically, a user story is just a requirement)* User story is a short simple description minimum **shippable** product. It is product owner responsibility. Anyone can write it.

It normally looks like this: As **<end-user>** I want to do **< action>** So that I can **<benefit>**.

As amazon user I should able to login, so I can buy stuffs online

### **What is an Epic?**

Epic is a big user story that you cannot complete in one sprint.

For example, as a user I want to buy online so I don’t have to visit the local store. This story is too big, and it cannot be completed in one sprint. So, we can call it Epic instead of user story. It should be divided to multiple user stories like:

As a customer I want to be able to login so I can view my account.

As a customer I want to be able to search for a product so I can buy them.

As a customer I want to be able to proceed to checkout so I can pay for the item that I am going to buy.

As a customer I want to be able to logout so I can protect my account.

As you can see< As a customer I want to be able to buy...> can be divided to multiple user stories. The team can pick one or more user story in every sprint.

**Agile experience in your most recent project?**

Our sprint is 2 weeks and we have release every 3 sprints as a release cycle

We have —- people in my team. —- developers, —- automation and — functional testers also 1 SM and 1 PO.

We start a sprint with Sprint Planning Meeting and

we discuss about the team’s priority features and product backlog items and

we learn the part of the application which we are going to developed.

Choosing story based on *velocity* and *capacity*

**Velocity**: Number of story points delivered/demo in a sprint. For example: if team planned 30 story points (Business value); worth of user stories in a sprint and able to deliver as planned then team’s velocity is 30

**Capacity**: Total number of available hours for a sprint is Team’s capacity. Calculates holiday and PTO hours

This meeting is held every week and lasts for almost 1 hour. We get general idea than we do Sprint Grooming meeting for giving some estimates for the tasks.

Team, SM, and PO get together to ensure work items are relevant and useful

Ask questions to P.O of user stories

Re-define acceptance criteria

Writing new stories

Breaking epics into user stories

Understand the story to give proper estimation/to prevent under/over estimate

### **How do you estimate?**

Based on my experience and complexity of the story and it is something I worked on before.

After sprint starts, we do Daily Standup Meeting

everyday morning and we discuss what did we do yesterday, what will we do today and is there any blocker.

Just we synchronize info about the sprint.

End of the sprint, usually we do Sprint Demo/Review Meeting .

It is just to show customer what we build sprint (PO can put feedback)

As an SDET in my team, I have done presentation sometimes and go over through the functionalities in the conference room.

Client or stakeholders or business people they ask questions what they don’t know.

After Sprint Demo, we do Sprint Retrospective Meeting .

In sprint Retro, we talk about what was good in last sprint, what kind of mistakes we made.

We go over them and make sure that we don’t make the same mistakes again.

If we did something good and improvements, we would continue doing it.

This meeting that is held at the sprint review meeting or at the end of the sprint; it lasts for 2-3 hours.

### **What is Acceptance criteria?**

Acceptance criteria is the way that we know the user story is successfully developed or not. Basically user story done based on AC and working as expected.

### **Test Case?**

Test case is a specific condition to check against the Application Under Test. It has information of test steps, prerequisites, test environment, and outputs.

Test case describes the functionality and test steps.

Test Case ID

Step number

Description of the functionality

Expected result

Actual Result

### **What are the steps you take to automate?**

Learn the functionality

Reading requirements

Knowledge transfer session with B.A

Ask teammates

Manually test it

Making sure I understand each step properly

Understand expected results

Automate it

Create POM pages

Add necessary elements/methods I am going to use and add PageFactory design pattern

Create a driver class with Singleton pattern

### **When do you do automation in your sprint?**

When developers are done with their part

When code is deployed to QA/test environment

When testing framework is set up

When all manual tests are done

Smoke tests are passing

### **What is Test Plan?**

Test plan is a word document that described the testing scope

High level test cycle

Defect life cycle

Entrance Criteria (defines what all need to start the testing)

Exit Criteria (defines what the testing is finished)

### **What are the tables in test plans?**

Test design, scope, test strategies , approach are various details that Test plan document consists of.

Test case identifier

Scope

Features to be tested

Features not to be tested

Test strategy & Test approach

Test deliverables

Responsibilities

Staffing and training

### **Who approves test cases?**

Generally QA lead. In my company developer team lead approves my test cases because of I have no test lead.

### **What are the functional testing types?**

Unit Testing

Smoke testing

Sanity testing

Integration Testing

System Testing

Regression Testing

UAT (user acceptance testing)

Other test type is **Smoke test.**

It is basically application is alive or not.

To Check if the testing Environment is UP or Down.

Smoke test can be scheduled multiple times a day to give timely report to the team.

It should be take short time. Just has most important test cases. Like log in log out. Clicks modules.

### **Explain the difference between bug severity and bug priority.**

Bug severity refers to the level of impact, bug priority refers to the level of urgency in the need for a fix.

**Priority is how soon we need to fix it. Product manager decides // low - medium - high - immediate**

**Severity is how bad the defect is. Test engineer decides // low - minor - major - critical**

### **What is defect? Bug**

When the expected result does not match the actual result, it is defect.

### **What is defect?**

When the expected result does not match the actual result, it is defect.

### **What is Defect Life Cycle (DLC)?**

New 🡪 Assigned 🡪 Open 🡪 Fixed 🡪 Retested 🡪Close

### **What to do when you find a defect?**

If I find a defect, before reporting it I reproduce the bug that I need to make sure that is a valid defect.

If it is a small issue, I will go to the developer desk, and he can fix it right away.

If it is a big issue, then I open my JIRA and log the defect.

If I am not sure it is bug or not, I will talk to SME (subject matter expert it means the person who knows the application better than anyone).

### **If developer says not a defect, what to do?**

I always make sure that it is a real defect that’s why I reproduce it.

I take screenshots and give all the steps to reproduce the defect.

Actually, one of my biggest challenges that I faced in my current project is that.

### **What is the different between Automation Testing and manual testing?**

The process of performing testing automatically which reduces the human intervention this is automation testing.

The automation testing is carried out with the help of some automation tool like QTP, Selenium, WinRunner etc.

In automation testing we use a tool that runs the test script to test the application; this test script can be generated manually or automatically. When testing is completed then tools automatically generate the test report and report

### **When will you automate?**

If it is taking a lot of manual effort. I run at least once manual and after that I automate it.

Automation is good for most repetitive functionality

### **When will you NOT automate?**

If functionality keeps changing

If functionality is used only once during the entire project

**Ad-hoc test** cannot be automated.

### **What do you understand by Daily stand-up?**

The daily stand-up is an everyday meeting (most preferably held in the morning) in which the whole team meets for almost 15 minutes to find answer to the following three questions –

What was done yesterday? What is your plan for today?

Is there any impediment or block that restricts you from completing your task?

The daily stand-up is an effective way to motivate the team and make them set a goal for the day.

**What is story point? How do you estimate story point in your company?**

**story point is a number** that tells the team about the difficulty level of the story.

User story size based on Effort, complexity and doubt.

Usually done before sprint planning meeting, during release planning. But we do in sprint planing meeting in my company.

We use fibonacci sequence numbers in my company

Estimating days times depend on company. In my company 1 point equals 1 day.