Interview

SOFTSKILL

Tell me about yourself?

- Tell me about your technical skill?
- What did you learn about Java?

What are the benefits of Automation Testing?

- Saves time and money. Automation testing is faster in execution.
- Reusability of code. Create one time and execute multiple times with less or no maintenance.
- Low-cost maintenance. It is cheaper compared to manual testing in a long run.
- Automated testing is more reliable
- It is mostly used for regression testing. Supports execution of repeated test
  cases.
- Maximum coverage. It helps to increase the test coverage.

Can you describe your daily activity?

- About your studies on QA Ingenieur.

Why are you looking for a job? Why did you apply for this position?

Where do you see yourself 5 years from now?

What are your strengths?

> For example;

- I am a very detail-oriented person. I can prioritize my job according the deadline.
- I am also very much dedicated person towards my job.
- I am also honest person and I have the skills and expertise in QA process.

What is your weakness?

How do you handle conflict?

> Example;
> Nothing is personal. Everyone thinks company’s benefits so I would like to explain my concern and his/her explanation makes sense for me. - Of course, I can do the things which is most helpful to my company. So, I try to communicate with his/her, and I would try to understand the concern. Because everyone have the same goal and wants to get job done successfully.

How do you handle stress?

> Example;
> ▪ One of our sprints my developer deployed code very late time and I didn’t have much time to get job done. But I was working so hard worked extra hours and especially nights and I was finished my task on time.
> ▪ My first approach is trying to calm down and work extra hours. I try to react to situations, rather than to stress. That way, the situation is handled and doesn't become stressful. Mostly these have helped me to handle stress. I also eat chocolate.
> ▪ Also, in scrum environment we working as a team. I always maintain good communication and relationship with my colleagues. So, they trust me, and they can communicate with me very easily. I always avoid miscommunication and my team believe me every time.
> ▪ Sometimes some requirements are not understandable, so I try to figure out and I try to understand the requirements. In the beginning of the application I spend extra effort to understand functionality. Sometimes it takes time to understand.

Can you work under pressure?

> Example;
> Good pressure, such as having a lot of assignments to work on, or an upcoming deadline, helps me to stay motivated and productive. Of course, there are times when too much pressure can lead to stress; however, I am very skilled at balancing multiple projects and meeting deadlines, which prevents me from feeling stressed often. For example, I once had three large projects due in the same week, which was a lot of pressure. However, because I created a schedule that detailed how I would break

> down each project into small assignments, I completed all three projects ahead of time and avoided unnecessary stress.

What do you like the most about testing?

> Example;
> Testing is fun job for me because you are very important person to the client and end users. I love testing because as end user I want to buy better product that is peas of art and defect free. Also, I am helping others to make sure their product has top quality. Imagine if you are testing the safety features of the Toyota Camry you are almost saving 100 of lives by doing your job and loving your job.

Why should we hire you?

> Example;
> First of all, I've done thorough research into this position and have read your job description and combined with the information you have so helpfully provided me during this interview, I can CONFIDENTLY say that I'm well qualified for this position. Meaning, I have all the technical and non-technical expertise, as required and preferred in the job description to not only succeed in this position but also to thrive.....

What is Software Testing?

❖ It is the process of executing a program or application with the intent of finding software
bugs/defects using functional and automation tools. Or we can also say that it is the process of
validating/verifying a software application.
❖ Testers should have a “test to break” approach instead of “test to pass” so that they can find the
bugs.

What is Requirement?

❖ The software requirements are description of features and functionalities of the target system.
Requirements convey the expectations of users from the software product.
❖ The process to gather the software requirements from the client, analyze and document them is
known as requirement engineering.

❖ The goal of requirement engineering is to develop and maintain a descriptive System
Requirements Specification document.

What is the Software Requirements Specification (SRS)?

❖ A software requirements specification (SRS) is a detailed description of a software system to be
developed with its functional and non-functional requirements.
❖ The SRS is developed based on the agreement between customer and contractors. It may include
the use cases of how the user is going to interact with the software system.
❖ The software requirement specification document is consistent with all necessary requirements
required for project development.
❖ To develop the software system we should have a clear understanding of Software system. To
achieve this we need continuous communication with customers to gather all requirements.

What is the Software Development Life Cycle (SDLC)?

❖ SDLC defines the phases in building of software or application.
❖ The Phases are:
➢ Project Planning
➢ Requirement Gathering
➢ Design (how the application will be
built)
➢ Coding or Developing
➢ Testing
➢ Production (deployment or releasing
the product)
➢ Maintenance

What is the Software Testing Life Cycle (STLC)?

STLC defines the phases in testing of software or application.

1. Requirement / Design Analysis
2. Test Planning
   a. Test Plan

    b. Test Estimation
    c. Test Schedule
3. Test Case Development (Designing)
    a. Test Cases / Test Scripts / Test Data
    b. Requirements Traceability Matrix
4. Test Environment Setup
5. Test Execution
    a. Test Results
    b. Defect Reports
6. Test Closure Activity (Reporting)
    a. Test Results (Final)
    b. Test Metrics
    c. Test Closure Report

What is a tester's main responsibility?

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

How Many Environments Do You Have?

1. Development Environment
    a. Unit testing
    b. Less stable than test environment
2. Test Environment
    a. Manual testing happens here
    b. Replicates the production environment
    c. Changes are deployed in intervals
    d. Automated smoke tests are run here
    e. Automation tests are run here
    f. Automated Integration tests run here
3. Pre-production Environment
    a. UAT environment
    b. Demo happens here
    c. load/performance testing happen here
    d. Changes are deployed in big intervals
    e. Automated major regression tests here (before release)
    f. Very stable
4. Production environment

Black Box Testing

❖ In Blackbox Testing, the tester only knows what the software is supposed to do, he doesn't care what is going on inside the box, how it works and what the logic is. He only cares if the software is doing its job.
❖ Testing smartphones by using it. Can I make calls? Can I browse on the web? Testing the application like an end-user.

White Box Testing

❖ Whitebox Testing is the software testing method in which internal structure is being known to the tester who is going to test the software.
❖ Testing smartphone chips, memory etc.
❖ Testing the codes.

Gray Box Testing

What is Functional Testing?

❖ In functional testing basically the testing of the functions of a component or system is done. It refers to activities that verify a specific action or function of the code.

What is Non-functional Testing?

❖ Performance testing is testing that is performed, to determine how fast some aspect of a system performs under a particular workload. For example,
➢ Can 2000 concurrent - user login to the application at the same time?
➢ Can a user move to the next page in 1second?
❖ Load Testing
❖ Stress Testing
❖ Endurance Testing
❖ Virtual Users
❖ Transaction Response Time

Unit Testing

❖ A unit is the smallest testable part of an application like functions, classes, procedures, interfaces.
❖ Unit tests are basically done by developers to make sure they eliminate the bug before testers find it.

❖ Normally Testers don’t perform Unit Tests, everyone is responsible for their own Unit Test.

Component Testing

❖ Component testing is a method where testing of each component in an application is done separately.
❖ Component testing is also known as module and program testing. It finds the defects in the module and verifies the functioning of software.
❖ Before integrating the components of the software the single component has to be working fine.
❖ Amazon is a huge website which has separate components like: My Account, Search, Payment, Prime, etc. Testing each of them separately!

Integration Testing

❖ Integration testing tests after integrating multiple components if the system works fine or not.
❖ Single component might work fine by itself but when integrated with multiple components it might fail.
❖ For example, testing Amazon’s all components together.

Acceptance 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.
❖ Also known as UAT User Acceptance Testing.
❖ The Acceptance testing will be performed after QA testing. In my current project it is done by the UAT team. After the UAT team performing the acceptance testing the code will go to production.

Regression Testing

❖ If any new functionality is added we have to make sure the new functionality did not break existing functionality, unintentionally.
❖ If there is any bug fixed we have to make sure the bug fixed did not break other functionality that was working fine.
❖ If there is any change in the code we have to make sure the developer did not break the other codes unintentionally.

Smoke Test

❖ Smoke testing is the initial testing process exercised to check whether the software under test is ready/stable for further testing.
❖ Before putting the entire team to testing effort, the system needs to be stable enough or worth testing.
❖ Also called Sanity Check. In Fannie Mae it is called Shakeout.

Positive Testing

❖ Positive testing can be performed by testing the application with valid input.
❖ If you try to login with a valid username and password it is positive testing.

Negative Testing

❖Negative Testing can be performed on the system by providing invalid data as input.
❖It checks whether an application behaves as expected with the negative input.
❖This is to test the application that does not do anything that it is not supposed to do so.
❖How many children do you have? 3.5 (trying to input this value to the application is negative test)
❖ Login with invalid username and password.

Software Testing Life Cycle

❖ Software Testing Life Cycle refers to a testing process which has specific steps to be executed in a sequence to ensure that the quality goals have been met.
❖ In the STLC process, each activity is carried out in a planned and systematic way. Each phase has different goals and deliverables.
STLC Phases:
■ Requirements Analysis
■Test Plan Creation
■Test Case Creation
■Environment Setup
■Test Execution
■Test Cycle Closure

Requirement Analysis:

❖ Analyze and study the requirements. Have brainstorming sessions with other teams and try to find out whether the requirements are testable or not.
❖ This phase helps to identify the scope of the testing.
❖ If any feature is not testable, communicate it during this phase so that the mitigation strategy can be planned.
❖ PO or team lead assigns you user stories to test , or you may choose user stories to test. You need to read the user story carefully and make sure to understand it well.

Test Plan Creation:

❖ Describes the scope, approach, resources, schedule, test items, features to be tested and not tested, tasks, and contingencies for the entire testing process.
❖ It identifies test items, the feature to be tested, the testing tasks, who will do each task, and any risks requiring contingency planning.
❖ It is written by a team lead or tester.
❖ A detail of how the test will proceed, who will do the testing, what will be tested, in how much time the test will take place, and to what quality level the test will be performed.

Test Case Creation

❖ A Test Case is a documentation which is a set of actions executed to verify a particular feature or functionality of your software application.
❖ Test cases are written by test engineers.
❖ They are written formats: excel sheet format, Jira with Gherkin Language.
❖ Test case describes the functionality and test steps.
➢ Test Case ID
➢ Step number
➢ Description of the functionality
➢ Expected result
➢ Actual Result
❖ Best Practices to Write Test Cases
➢ Test Cases need to be simple
➢ Create a Test Case with the End User in Mind.
➢ Avoid test case repetition
➢ Ensure 100% Coverage
➢ Test Cases must be identifiable

What do you 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 it 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 if it is a bug or not, I will talk to the SME (subject matter expert it means the
person who knows the application better than anyone).

What is Waterfall Methodology?

❖ Waterfall project management entails mapping out a project into distinct, sequentialphases, with each new phase beginning only when the prior phase has been completed.
❖ The waterfall system is the most traditional method for managing a project, with team members working in a linear fashion towards a set end goal.
❖ Each participant has a clearly defined role and none of the phases or goals are expected to change.
❖ Waterfall project management works best for projects with long, detailed plans that require one phase to be done before another can start.
❖ These projects require a single timeline and changes are often discouraged and costly.

What is Agile Methodology?

❖ Agile is a process by which a team can manage a project by breaking it up into several stages and involving constant collaboration with stakeholders and continuous improvement and iteration at every stage.
❖ The Agile methodology begins with clients describing how the end product will be used and what problem it will solve. This clarifies the customer's expectations from the project team.
❖ Once the work begins, teams cycle through a process of planning, executing, and evaluating — which might just change the final deliverable to fit the customer's needs better.
❖ Continuous collaboration is key, both among team members and with project stakeholders, to make fully-informed decisions.

What is Scrum?

❖ Scrum is an agile process that allows us to focus on delivering the highest business value in the shortest time.

❖ It allows us to rapidly and repeatedly inspect actual working software (every two weeks to one month).
❖ The business sets the priorities. Teams self-organize to determine the best way to deliver the highest priority features.
❖ Every two weeks to a month anyone can see real working software and decide to release it as is or continue to enhance it for another sprint.

Product Owner:

❖Define the features of the product.
❖Decide on release date and content.
❖Be responsible for the profitability of the product .
❖Prioritize features according to market value.
❖Adjust features and priority every iteration, as needed.
❖Accept or reject work results.
❖Leading the team to the right direction.
❖Motivating team with clear direction.
❖Consistently communicates with the client and stakeholders to make sure the Development team is in the right business direction.
❖ Being a bridge between the Development team with Client and Stakeholders.

Scrum Master:

❖Responsible for enacting Scrum values and practices
❖Removes Impediment
❖Ensure that the team is fully functional and productive
❖Enable close cooperation across all roles and functions
❖Shield the team from external interferences
❖Represents management to the project
❖Do everything possible to make sure Development Team performs in highest level
❖Coordinates the Collaboration among the team members
❖Serve to the team, he has no authority over the team

Scrum Development Team:

❖ This is a group that works together for creating and testing incremental releases of the final product.
❖ Formed by the collective contribution of individuals who perform for the accomplishment of a particular project.
❖ The team is bound to work for the timely delivery of the requested product.

Scrum Artifacts:

Product Backlog:

❖ This refers to what remains on the "to be done" list. Or it is a list of work that has to be done in the project.
❖ During a product backlog grooming session, the development team works with the business owner to prioritize work that has been backlogged.
❖ The product backlog may be fine-tuned during a process called backlog refinement.
❖ Change is welcome on the product backlog.

Sprint Backlog:

❖ Sprint Backlogs are tasks that a scrum team needs to complete in 2 weeks sprint or 4 weeks sprint.
❖ These are divided into time-based user stories.

Burnout Chart:

❖ It is a visual representation of the amount of work that still needs to be completed.
❖ A burn-down chart has a Y axis (work) and an X axis (time).
❖ Ideally, the chart illustrates a downward trend, as the amount of work still left to do over time burns down to zero.

Product Increment:

❖ This refers to what's been accomplished during a Sprint -- all the product backlog items -- as well as what's been created during all previous Sprints.
❖ The product increment reflects how much progress has been made.

Scrum Ceremonies:

Sprint Planning Meeting:

❖ Discuss and decide what can be realised in the next sprint and how this should be done.
❖ Team needs to know how much capacity and commit some points (each point represents some hours (individually).
❖ Jointly establish a Sprint Goal Discuss and decide what can be realised in the next Sprint and how this should be done.

Daily Stand up (Daily Scrum) Meeting:

- Everyday but the first day of the sprint and around 15 mins
- Same Place and time with Full Team
- Run by SCRUM MASTER
- Provide Update on the sprint backlog work progress
- Focus on 3 Main Questions
  - ➢ What did you do yesterday?
  - ➢ What will you do today?
  - ➢ Do you have any blocker (impediment)?

Sprint Demo (Review) Meeting:

❖ Usually it happens at the end of the Sprint (like after the Every 2 sprint) § Usually everyone will join (+ customer, stakeholders ...)
❖ All new functionalities of the app will be shown
❖ Usually run by the developer
❖ Automation engineer can DEMO their automation script as well

Sprint Retro (Retrospective) Meeting:

❖ Will be after every DEMO (ideally)
❖ Scrum Team will be present in the meeting
❖ Opportunity for the Scrum Team to inspect and improve itself.
➢What went well?
➢What went wrong?
➢How can we improve?
➢Start doing more of what?
➢Stop doing what?
➢Continue doing what?

What is Epic?

❖ Epic is a big user story that you cannot complete in one sprint or we can say that it is the biggest user story which is almost everything about one project. >For Example:

> As a customer I should be able to buy from Amazon, so I don't have to go to local stores.

```
Above user story is everything about amazon.com. It includes Login, Logout, Search Product, View Product, Basket, Credit Card, Shipping
```

```
Method and Return functionalities.
```

What is Feature?

❖ Feature is multiple user stories that you can put together and it becomes a stand alone component of one application.

> As an Amazon Prime member I should be able to stream movies online, so I don't have to use Netflix.
> It is not totally away from amazon, it is under amazon project, but it can be stand alone component of amazon.com

What is a User Story?

❖ Note: basically, a user story is just a requirement.
❖ User story is a short simple description of a minimum shippable product.
❖ 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 Acceptance Criteria?

❖ Acceptance criteria is the way that we know the user story is successfully developed or not.
❖ When BA or PO write user stories, s/he writes AC together.
❖ Statements of requirements that are described from the point of view of the user to determine when a story is "done" and working as expected

> Example User Story:

```
        As a customer I should be able to pay with a credit card so
I don't have to pay cash.
Acceptance Criteria:
                User should be able to pay with a Visa card.
                User should be able to pay with Mastercard.
                User should be able to pay with Amex.
User should be able to pay with any above credit card by entering
following information:
Card Holder Name, Card Number, Expiration Date and 3 digit security
code in the back.
```

What is STLC? What are the steps of the STLC?

```
-   Requirement Analysis
-   Test Plan Preparation
-   Test Case Creation
-   Environment Setup
-   Test Execution Test
-   Cycle Closure
```

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 where quality of software or product ensures. SDLC has vast and vital role in complete development of a 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

When the testing starts?

▪ 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 first place. With the wrong requirement it is impossible to build bug free application.

Why 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 buy delivering bug free and user-friendly application.

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 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.

When you run regression test?

```
-   before release
-   after major bug fix
-   after major new functionality
-   where we keep test scenarios and where we as a team take
decision which will be executed more than once, in one sprint you
test some scenarios.
```

How do you ensure that your regression tests are effective?

- The regression tests should be wide and detailed enough to allow catching defects. You can also eliminate duplicate test cases, merge test cases and automated tests as feasible.

How do you run your regression? How often, how many VMs, how many days, how many tests?

▪ Regression is scheduled before every release and we release twice a year (Spring release and fall release).
▪ Regression also happens when there is a major bug fix.
▪ Around 500 feature files and 1300 scenarios.
▪ Regression tests are kicked off by jenkins. Tests are executed on the jenkins server (VM). My Linux server is RedHat.
▪ The latest run look more than 12 hours.

Another answer is;

- I have built a suite of regression tests. They are feature files with regression tag. And I have a job in jenkins that kicks of the regression tests. It uses the maven command to trigger the test. The maven command includes that tag name: mvn test -D cucumber.options =”--tags @Regression”.
- At the end of the execution, jenkins generates HTML report with detailed tests steps and screenshots

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 Integration Testing?

▪ Integration testing is black box testing. Integration testing focuses on the interfaces between units, to ensure that units work together to complete a specific task.
▪ The purpose of integration testing is to confirm that different components of the application interact with each other. Test cases are developed with the purpose of exercising the interfaces between the components.

What is Stress Testing?

▪ Stress testing tests the software with a motive to check that the application do not crashes if we increase the stress on the application by increasing the large number of users working on the application.

▪ We can also apply the stress on the application firing the lots of process which cannot be handled by the application.
▪ We perform the stress testing on the application evaluate the application capabilities at or beyond the limits of its specified requirements to determine.
▪ Generally, this is a type of performance testing performed in a very high level of load and stress condition.

What is Acceptance testing?

```
-   The Acceptance testing will be performed after QA testing. In my current project 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)
```

What is the difference between UAT (User Acceptance Testing) and System testing?

```
• System Testing: System testing is finding defects when the system undergoes testing as a whole, it is also known as end to end testing. In such type of testing, the application undergoes from beginning till the end.
• UAT: User Acceptance Testing (UAT) involves running a product through a series of specific tests which determines whether the product will meet the needs of its users.
```

What types of Test cases?

▪ I cover different scenarios
o Positive
o Negative
o Boundary Value Analysis

Who approves test cases?

- The approver of test cases varies from one organization to the next.
- In some organizations, the QA lead may approve the test cases while another approves them as part of peer reviews.

Who writes test plans and test cases?

```
• Test plans are typically written by the quality assurance lead while testers usually write test cases.
```

What is Requirement Traceability Matrix (RTM)?

```
• RTM is used to make sure that all test cases cover the requirement or not. It is like excel sheet
```

What are the functional testing types?

```
-   Unit Testing
-   Smoke testing
-   Sanity testing
-   Integration Testing
-   System Testing
-   Regression Testing
-   UAT (user acceptance testing)
```

Explain the difference between bug severity and bug priority.?

▪ Bug severity refers to the level of impact that the bug has on the application or system while bug priority refers to the level of urgency in the need for a fix.
▪ Usually the severity is defined in terms of financial loss, damage to environment, company's reputation and loss of life. Priority of a defect is related to how quickly a bug should be fixed and deployed to live servers

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).

What is increment? Explain.

▪ An increment is the total of all the product backlogs items completed during a sprint.
▪ Each increment includes all the previous sprint increment values as it is cumulative.
▪ It must be in the available mode in the subsequent release as it is a step to reach our goal.

What is a Test Plan? who writes? How often?

- it is a doc, prepared by a QA (QAs contribute in a meeting)
  - which has project info, feature description, role & responsibilities, schedule
    testing strategy -> testing types / tools / env , risk & mitigation , approves
- one test plan for one release

What are the Test case types?

```
        -   Happy/Positive test case ⟶ test a function with valid data, test how it should respond
```

- negative test case --> test a function with INVALID data, or no data

```
× Failure ›
```

Error: You exceeded your current quota, please check your plan and billing det ails.