Interview Story

- Put tabs/tickets in sublime notes and you can read them/ follow the points during the interview
- When we present to the client in demo meeting, we do not present the technical part; just UI
- Slack is likes discord, where teams communicate
- Make sure delete all bookmarks related to cybertek etc
- As SDETS we create test plans in Jira for sprints; but do not deal with whole project test plan

1. Why are you looking for a job?

   Make sure to say thank you for this time and opportunity.

   I've been with this company on contract for the past 2.5 years; and at this moment I am looking for new opportunities for professional and personal growth. And I am looking for a full time opportunity

   Other choice: I love my team, but the company is relocating to the other state. They require me to be in the office twice a week but at this moment I don't feel comfortable to move with the company.

2. Tell me about yourself?

   First of all, I would like to thank you for your time and for the opportunity to be here today. Shortly about me. I've been an SDET for the past 5 years. I started as a manual tester, and I was very lucky to have wonderful team who were willing to dedicate their time and share their knowledge and helped me to become an SDET. During these years I've been working in logistics and communication industries. Right now I am working on …….. I have been successfully testing web based applications and client server applications. Mostly I have been working with Windows operating system. Java is my main programming language and I have been exposed to JavaScript with Selenium just a little bit. I mainly build projects in

Maven. So far worked mainly with IntelliJ IDE but I was exposed to Eclipse before as well. I have experience building testing frameworks using TestNG, junit, Selenium, Cucumber. Framework design – page object model, BDD cucumber framework. For repositories I've been using GitHub. For continuous integration I work with Jenkins. Project management tools in our team are Jira Jira xray for test management.

I've been working in Agile, SCRUM environment. Well exposed to SCRUM ceremonies. I have a good understanding of SDLC, I am proficient in testing automation and manual types. I have experience with testing types such as, smoke, regression, functional testing, end to end, API. I am always willing to learn something new and interact with a team. Very detail oriented. If you have any more specific questions, please let me know.

3. What is software testing?

   The software testing is the ongoing process of checking if the software application or website is working as it's supposed to. It improves the reliability of the application and higher quality, with less error's product is being delivered to the customer.

4. As a tester, what are your responsibilities?

   To perform functional software testing, so we could deliver a high quality, bug free application. I do write the test plan and test cases/ scenarios. I also determine the severity of the defect of the bug when I find one.

   I also perform Ad hog testing (random testing without any plan).

   I am responsible for Smoke Suite, Regression suite, framework maintenance, sprint automation testing. I was responsible for system testing, that includes regression and smoke, UI/API, DB scenarios as well.

5. Tell me about your project?

   There are several projects under the company, but the application I've been working on is the NextBase CRM project. It is a web application for customers to organize the easy collaboration between team members, as well as manage the documentation flow, reporting and task assignments. It is an internal application where different users with different roles and permissions can create the end-to-end workflows. Also, the application enables users to share files, create calendars etc.

My team and I were working on Time and Reports Module. It's the combo of Absence, work time reports, meeting etc.

My other most recent project was Fleet Management for a VyTrack company. The module I was working on was the Fleet module. VyTrack company specialized in vehicle deliveries, vehicle rent for business, replacement vehicle, and damage management, vehicle management, Insurance, and registration. The purpose of the project was customer account management, vehicle status management, calendar, log call, event creation.

The Fleet module was for contract creation, create or delete vehicle info, update odometer and fuel cost, report any damage cost.

6. Tell me about your team?

   My team that I've been working with consisted of 3 developers, 2 testers, SM, PO and 1 BA. Eight people.

   Per project, we had 4 teams. 1 lead QA per whole project.

   We performed the smoke tests every morning at 5 AM, 20 scenarios, 25 minutes (17 test cases or 8 scenarios) in QA environment. Once major functions would be developed and added, we usually would add new scenarios; typically, it happens after every sprint. Smoke test is performed automatically; Jenkins's tool was used. If smoke test would fail, all testers were responsible to check if the failure is real, maybe the test was flaky, or the environment was down. We would also reach out to the peer testers, and check manually, and if it confirms that the failure is real, we notify the whole team and we inform the developers and we all agree we would create the bug report. 1 smoke suite per whole project.

We had a very sophisticated team; we did not create the bug report right away. Usually, we would check if that could've been the environment issue, the test data issue, or the scenario issue.

Major Regression testing for our project is done every release. We also did minor regression testing at the end of each sprint; so, we would test existing and new features functioning together before the demo.

Regression testing was scheduled by our DevOps team in Jenkins.

Regression was done manually by all testers under the project and automatically. Usually, 80% is automated and the rest is manual. For our 715 scenarios it took about 3 hours of automated testing and 4 to 5 days of manual testing. And we had 2 virtual machines running in parallel. Regression testing was done in staging environment.

We would also run the regression whenever there would be the hot fix as well.

7. What kind of meetings did you have within your company?

   For our meetings, we would usually stick to the big 4: Sprint Planning, Daily stand-ups, Sprint reviews, Sprint Retrospective.

   Daily stand-ups, every morning at 9.30 am, for 15 minutes or less; After daily stand ups, sometimes we would hold the Sidebar meetings for 10 to 15 minutes only if needed. Our Scrum master hosts them.

   Sprints: 2 weeks, we would usually start on Wed a new Sprint.

   Product backlog refinement/grooming meeting would be done before the Sprint planning. The whole team attends, and we made sure that AC criteria is clear, everything has been prioritized before the sprint planning.

   Sprint planning- usually the whole team is present, Backlog refinement prepared before the Sprint planning meeting. About 2 hours or less. The Doneness agreement is completed with PO and the Team. Usually, all teams of the project would meet to avoid the confusion. Our team used card game to decide point values.

Retrospective: looking back at our Sprint what could be done better, what we'll do again. SM and dev team. After Sprint Review. 2 hours. Each team member reflects on our sprint. We use Easy Retro.

8. When did you join the project?

   I joined when the project was existing already for a little while.

   When I joined the project already existing for a few years. And it's an ongoing project.

9. What were the roles in your company? Who did what?

   PO would get the user stories ready and would prioritize work schedule. Also, he would set the release dates. Also, PO would communicate between the team and the client.

   SM would calculate teams' capacity and velocity. Facilitates the meetings.

   BA with PO would gather the user stories, prioritize them and do the verification of documentations etc.

10. Strengths: Great team worker, responsible, get work done on time, quick learner, open minded, like to improve and learn something new, ambitious. I adapt to changes well. Good work ethics.

11. Weaknesses: Sometimes being too detailed oriented; if I start something I always want to finish it so sometimes I'm impatient and trying to get it done as fast as possible; hardworking, so sometimes hard to find work life balance. So I am trying to have a good plan and have a checklist what needs to be done and I try to stick to it to avoid the burnout and I try to create a healthy balance between work and personal life.

My other weakness is public speaking, I get very nervous. So I try to do demos and to actively participate during meetings to overcome that fear and to become more comfortable.

12. What are some of the achievements of yours in your company?

    I have encouraged the company to create the naming convention manual for the projects to keep everything more organized and it was also very helpful for the new team members.

    I also love mentoring; whenever we would get a new team member on board, I would make sure to make him/her comfortable in our team.

    Also, I was able to establish friendly relationship and effective communication with developers and peer SDETS.

13. What are your achievements in your project?

14. What is feature?

    Application withing application. For example Amazon prime, you can stream the movies.

15. Tell me about your day-to-day activities?

    My shift starts at 8am, first thing first I double check the results of my smoke test, make sure that the application is up and running. Then I am focusing on my emails which I received yesterday after I left my desk, to make sure that I am not missing anything from yesterday. Then I am opening my Jira board, basically getting ready for my daily stand up which occurs at 9.30 am every day. At stand up as a team we discuss what we did yesterday what we are working on right now and more importantly if there is anything that holding me back to do my job, impediments. After stand- up I go back to my desk and depending on which part of sprint our team is my tasks differ. Let's say if it is early sprint, I would participate in sprint planning meeting where we go over the user stories and pointing them. Also, I would work on maintaining my framework, participate in test scenario reviews.

I might start working on my test cases and scenarios.

Towards the mid sprint where the code is being released, I would've completed my automation and start execution. If it is end of the sprint, I would be busy for getting reports ready and participate in demo meeting. We also have retro meetings at the end of the sprint. Almost every Tuesday we have lunch and learn session where I share my coding and automation experience with my fellow manual testers, and every Friday we have code review meeting among automation testers of the project. As I am Java certified I also participate in developers code review meetings just to get an idea of the code. So, this is basically my day, oh I actually forgot about coffee :) I do not start my day without a cup of coffee.

During early sprint we have sprint planning meeting, we analyze the user stories and acceptance criteria, we do points, create test plans in Jira x-ray, I maintain my framework, prepare test scenarios and review test scenarios.

During mid sprint we are automating and executing scenarios.

At late sprint we have demo and retro meetings, test result creations.

Extra's- lunch and learns, brown bag sessions, and mentoring.

We were utilizing the Microsoft Teams for meeting and communication.

16. Tell me about your responsibilities?

Well, As I am an SDET, my main responsibility is test automation so we could create a high quality, bug free applications. I will try to break it down. One of my main responsibilities is to automate, run and maintain smoke test suite. I am also responsible for maintaining, automating test scenarios for regression suite and for sure executing it which will include reporting as well. As a cross functional team member, I am responsible for reviewing test scenarios with my peer manual test team members and figuring out good candidates for automations. Also reviewing my code with automation testers in order make sure that we are following the best practices for our team. As we are implementing Agile, I am attending to Daily stand ups, where we are talking what we are doing what we did and are the any impediments. I am attending Sprint planning in every sprint where we discuss user stories, give points(Fibonacci), and assign it to team members and demos, where I usually present what as a team we have accomplished.

17. What if the developer delays but you must present on time?

    I always notify the team during our stand-up meetings so that there are no surprises. I make sure that I work ahead with whatever I have at the moment so that when the code is ready, I'm still on time. I also can ask for at least a partial release of the code; I might be able to get the look and feel of UI of the app and start working on the locators and stuff.

    Things can always go wrong…the main principal is to maintain time management, work ahead, keep an open communication with developers, notify the PO and other team members of any impediments; in worst cases, PO can re-prioritize that part for the next Sprint to be done the first day. Sometimes, there is only so much we can do, but we must learn from the mistakes, improve, and adapt to those situations.

18. Where do you see yourself in 5 years?

    I love being a tester. In my opinion it is a great career, it very valuable position, because a good tester is like a good health insurance plan for a company. But I like growing. I see myself being more technically advanced, maybe learning more about mobile testing, improve technical skills. Also, if needed and if I would become very familiar and comfortable with the project, I would consider helping with leading projects etc.

19. If regression testing fails, who fixes that?

    We have a separate team for that.

20. Which software tools and applications have you been working with?

IntelliJ

GitHub

Jira

Cucamber

Zoom or Teams

Slack or Discord

Selenium

Java

21. Requirement vs Acceptance criteria?

    The requirement comes from the client. Acceptance criteria must be passed with testing, in order to meet the requirement. The requirements are coming from the client.

22. What is defect?

    Variance between expected vs actual after the product goes into production.

    We create a defect(report) once malfunction or error is found.

23. Where do you get your test data?

    From business analyst.

24. DoD and DoR?

    DoR- agreement, PO and Team, Which user stories in backlog will be moved to sprint backlog. Beginning of sprint.

    DoD- towards the end of sprint. If everything was accomplished to consider that it's done.

25. Increment in soft skills – what was accomplished

26. What is failure?

Inability of the application to perform its function.

27. What is Bug?

    Coding error that causes the program to malfunction.

28. Testing vs Debugging?

    Testing is the process of verification and validation, looking if there is a bug.

    Debugging is actually fixing the bug or error.

29. Is a 100% testing possible?

    Of course, the answer is no. There are always scenarios that we haven't considered to check, there are always some bugs there. But we are doing our best to find the defects or bugs and to maintain everything as stable and safe as possible. And there is always an opportunity to improve the application, that why our developers work hard to add new features and new functionalities, but our job is to make sure everything works well together.

30. Is Java fully object oriented?

    No, because Java has primitive data.

31. Difference between DoD and DoR?

    DoR- Definition of Ready. PO explains the User Story to the Dev team; It's an agreement between the PO and the team. They agree that user stories are ready to be pushed to the Sprint backlog. It's done in the beginning of the Sprint during the grooming or Sprint planning. Acceptance criteria is clear, reviewed and pointed, hours are given.

    DoD- it the checklist that proves that everything was completed by the dev team as discussed in the beginning of the sprint. All requirements have met the expectations. Acceptance criteria is fulfilled. PO accepts the user stories.

32. Tell me about the environments in your project?

    We have Dev environment, QA/ Test environment, Staging environment and Production environment. We also have demo environment. Smoke suite is run in QA environment and Regression in staging environment.

33. Which is the most stable and which is the least stable environment?

    The most stable is production and staging; Development environment is the least stable; there are always constant changes pushed to the code.

34. What is epic?

    It is a major user story that will be divided into smaller tasks.

35. When are new scenarios added to the smoke suite?

    After every sprint there might be some changes in smoke suite. When major functionalities are added to the app, the more scenarios are added. All testers under the project usually decide.

    We used Fibonacci as pointing system. 8 is the largest.

36. How is decided if something needs to be automated or just manually tested?

    Well, we usually decide that as a team during our Sprint planning meeting. Some of the criteria are- if there is no frequent repetition of test cases needed, we can perform manual testing. Also, if we are checking how user friendly the application is, manual testing is the best way to go. For automation testing, it is the most suitable when test cases need to run repeatedly and for a long period of time. Usually, it is also more reliable. That is why, the best way to test, when we incorporate both. Also, Ad hog testing doesn’t have to be automated. Typically, if the user story is 3 or more points, it has to be automated. But usually everything gets tested manually first in most cases.

37. Which phase in SDLC is the longest phase?

Testing is the longest phase.

38. What are inputs in SDLC?

Client input → SRS (PO and BA makes SRS and passes to design team)

SRS→ SDS and blueprint (developers work with SDS and SRS)

For testers→ SRS and application itself (we test based on SRS and application itself).

Sprint Review/ Demo: PO and Stakeholders are present. We would show what was accomplished during the Sprint. Last day of the Sprint. 2 hours. PO accepts or rejects the task based on DOD.

The artifacts of Agile: Product backlog – the list of ideas for the project and description of Definition of Done; Sprint backlog – set of stories that the team has agreed to work on during the sprint; Product increment (We use burndown chart) that demonstrates how efficient the team is in completing items.

Definition of Ready agreement between PO and team if the user story is ready to be added into the sprint backlog.

39. Tell me about the Bug Life Cycle?

Bug life cycle is part of the STLC, Test Execution. We determine the severity(testers) and priority (BA and PO) of the bug.

a. Tester find the bug.
b. Test lead or Project lead verifies the bug.
c. Developer fixes the bug.

Bug status

a. New
b. Assigned if approved
c. Open- when working on it
d. Rejected- invalid bug
e. Deferred- will be fixed later

f. Completed/ Fixed
g. Re-tested
h. Closed

40. What you like about Agile: feedback from the users, accepts changes, focuses on result not as much on paperwork. The team is self-organized, everyone has input into the project.

41. The negative side of Agile: Sometimes if feels like we spend lots of time in meetings instead of doing work. But I do understand that we need a good plan in order to do a good job.

42. End to end scenario: analyze requirements, list down how every system needs to respond, design test cases, run tests. Replicates a live scenario from start to finish.

43. End to end scenario is System testing; Testing everything from beginning to end. Example: Let's say testers have to verify the functioning of a Gmail account. The following features have to be tested:

    1. *Type URL into the address bar to launch the Gmail login page.*
    2. *Log into account with valid credentials.*
    3. *Access Inbox. Open Read and Unread emails.*
    4. *Compose a new email.*
    5. *Reply to and forward an existing email.*
    6. *Open Sent items folder. Check emails there.*
    7. *Open Spam folder. Check emails there.*
    8. *Log out of Gmail by clicking 'logout'.*

44. Testing methods?

As an SDET, I was performing the system testing; functional testing, such as smoke tests, regression, API. Manual and Automated.

45. What are some disadvantages of Agile?

Well, as I mostly like Agile, but of course there are some things that are challenging. Even though, I like any time changes acceptance in the requirements, but I also consider it to be the most challenging because it makes it really difficult sometimes to meet the deadlines. Also, I think lack of documentation takes place among disadvantages. As PO writes user stories, may have an idea about a new feature but may not be aware of the specifics. This means they cannot write a good set of AC.

Sometimes seems like it's too many meetings as well.

46. Positive things about Agile?

Communication with customer; it's easier and cheaper to make changes or fixes in earlier stages of development. Can make changes while creating something. Flexible.

47. How to achieve better test coverage?

Automation as much as possible. Having a good plan. Sticking to the plan.

48. How do you get involved with your team members?

I have worked in several different teams within my company and overall have a great experience and great relationships. We openly communicate with each other. Mutual respect, being nice and approachable, helping others, and being able to ask for help when needed comes a long way. We've also always tried to actively participate at all meetings.

Our SM was amazing! She would always make sure that our team environment is healthy and positive and productive. She would organize the bi-weekly "non-official

20 minute coffee breaks" where we could talk about not work related stuff and just connect with each other; share hobbies, pictures, mom's stuff. It was always fun. As a team, we would often enjoy our lunches together as well (eating with camera's on, on Zoom and Teams).

In my opinion, only a good and strong team can bring out a great quality product. It is so great to work in a nice atmosphere and go to work with a smile and confidence.

49. Where do you document your test cases and what are different types of documents that you prefer?

    We do our management through Jira (or Jira x-ray); attach screenshots- the results of the manual test cases, spreadsheets with steps and results.

    For automation, we store the code in GitHub. We make the branch from the user story.

    If there are production defect fixes, we document those in word document and attach it to Jira.

    Write steps in Jira xray, then copy to word.

50. Can API testing be done manually?

    Yes, we use postman. And for automation we use rest assured.

51. What kind of exceptions have you run in and how you solved them?

    NoSuchElement exception (due to not enough time for the element to appear on the page) is one of the most common ones; so I use implicit wait command to avoid it. Sometimes element is not on the page due to developers changing something. Could truly be wrong locator.

    StaleElement Exceptions is also common; due to the web element has been entirely deleted or no longer attached to the DOM.

    Solutions: Refreshing the page, try and catch block.

52. What kind of bugs have you found in your project and how did you solve them?

53. What information should be included if we find a bug?

54. How do you handle exceptions?

Try and catch blocks.

55. How do you handle dynamic elements?

I inspect the element and analyze how it is changing, normally there could be constant part of the id, name, or text and other parts that are dynamic and changing. If so, I write custom xpath using “starts-with”, “end-with”, “contains”, methods. For example, //h3[starts-with(@id,’sunny’)] //h3[ends-with(text(),’day’)]; But if all attributes seem to be changing, I would look at the parent elements to see if I can locate the parent then navigate to child.

56. Which locator are you using the most?

The most common locator that I use in Selenium is XPATH. We can use any attribute value, locate parent then navigate to element that we need. We can also locate elements by index, text, etc. It is very flexible locator in Selenium. The advantages over CSS are- CSS cannot use indexes, locate by text, and navigate from child to parent.

57. In CSS, what does dot, and hash mean (“.”, “#”)?

“.” → class name

“#” → id

CSS also uses ^, $, *

58. When cannot you use class name as a locator?

If the class name has spaces in it.

59. OOP concepts/ in your last project?

Explained each concept. Encapsulation is for hiding data, getter and setter. Inheritance- improving reusability. Abstraction – abstract class and interface; abstract methods. Polymorphism. How we use them? Encapsulation in POJO class. We declare variables as private and declare getters and setters (API framework). Inheritance base page class, test base class- we want to use one driver to set the browser and reuse it in other test classes. Test base and base page are abstract, we do not create any objects there, those are just parent classes, it's only the super class. Interface in the framework, we usually do not create ourselves. Interfaces can be inherited to other classes, we want to achieve multiple inheritance (take screen shot, List is interface, Set is interface). We cannot create an object from interface. Polymorphism- in framework when parent is referenced to child. We can test in different browsers. WebDriver driver = new ChromeDriver(); Polymorphism we have used in the driver class. Also, List and ArrayList. Map being referenced to HashMap, Set referenced to HashSet.

60. What kind of collections are used in Java?

Array, Collection, Map- collection structure. Only tell what you actually have used. For example, List, in Selenium the list of elements. Set- getWindow.handles(). In Selenium, avoid duplicate element avoiding, Set. Map- doesn't have inheritance with collection.

61. What version of Java and Selenium have you been using?

Java8; Selenium 3.141

62. How do you write your codes in your framework?

In reusability, readability easy maintainability manner. So it could be updated in one place.

63. Where do you use inheritance in automation?

Test base classes that extend to the test class. I add @BeforeMethod and @AfterMethod to the test base class.

64. Tell me, when would you use Array, ArrayList, or LinkedList?

    Arrays when I'm working with fixed size data type that doesn't change. It is fast and simple for basic operations. ArrayList is the most commonly used in collections. I use it with dynamic data type. For example, in Selenium, I use List <WebElement> to find element. LinkedList is being used with dynamic size data type and when I need to preform adding or removing values from the LinkedList. It is faster then ArrayList.

65. What is the difference between ArrayList and LinkedList?

    Both implement List interface. Both are indexed, ordered(sorted), and both can have duplicates. ArrayLists use dynamic array objects but LinkedLists use Node objects. So ArrayLists are better or faster for sorting and accessing data. But LinkedLists are more useful for data manipulation-for example for adding and deleting data.

66. Why is the main method static?

    If it's not static, it will be instance method. And every code we would need to make an object. Also, we wouldn't be able to use static members.

67. What is constructor?

    It is a special method that is used to initialize the objects. It is called when an object of the class is created.

68. Can the final method be overloaded?

    Yes. But it cannot be overridden; we cannot change the implementation of final method.

69. Difference between instance and local variables.

    Instance variable is declared within the class but outside the method. Local variable is within the method or constructor.

70. How does encapsulation work with inheritance?

    Private information does not get inherited. But the child class can inherit public getters and setters.

71. Can we override static method?

    No, we cannot. But we can hide them.

72. What are two ways to achieve abstraction in java?

    Abstraction class and interface.

73. Check vs uncheck error?

    Checked == Compile time error, like thread.sleep(); we have to handle it right away. It will mark red.

    Unchecked == runtime error. It will show in console, like, no such element.

74. Who is the parent of all classes?

    Object class.

75. Throws vs throw in exceptions.

    Throw is the keyword that makes an exception, causes an exception.

    Throws is allowing and ignoring an exception.

76. Parents of exceptions?

    Exception is the parent of checked exceptions. Run time exception is the parent of unchecked exceptions. Throwable is the class of all exceptions and errors in java.

77. What is collection framework?

    It’s an architecture or a group of interfaces and classes that define how the objects would be stored and manipulated.

78. Collection vs Collections?

    Collection is an interface that all Collection objects implement.

    Collections is the utility class that has methods to work with collection objects. Like Arrays class is the utility class for Array objects.

79. Difference between array list and linked list.

    Both implement List interface. Array lists work with arrays internally. Linked lists work with nodes internally.

80. How would you access an element in the Set?

    You would have to loop with for loop.

81. What is Stack?

    Stack is the child class of Vector class. Follows LIFO. Methods push(), peek(), pop();

82. Difference between static and public methods?

    Static methods can be accessed without the object. Public methods must be accessed by an object. Static belongs to a class. Instance belongs to an objects.

83. What are non- access modifiers in java?

    Final and abstract; we cannot create objects of them.

84. What is a static block?

It is a code block that allows only the static members. Used to initialize the static variables. Runs once when class is loaded into memory. It’s like a constructor but for static members.

We use static block in configuration reader class because we only want to open properties file and load to properties object only once before we read values.

We are adding the method that needs to run ONCE only.

85. What cannot be static?

    Package and constructor.

86. Difference between static and instance method?

    Static methods don’t need an object to call. Instance methods need an object to be called.

87. How we print an object?

    To String method.

88. Constructor, 4 rules.

    We cannot inherit constructor, we cannot, use this must be on the first line, can only call one constructor at a time.

89. What is Junit and TestNG?

    Both are testing framework for unit testing. Similar, but TestNG is more powerful and it’s newer. Junit is good for regression testing, it’s open-source framework.

    TestNG unit, end-to-end, and integration testing. Grouping test cases we are only able to do with TestNG.

90. How do you implement OOP concepts within your framework?

For example, Inheritance: We use Test Base class for Before and after methods. Then we inherit that class to other classes and we do not have to repeat the codes over and over again.

Polymorphism we implement for WebDriver driver = new ChromeDriver(); As a reference for chrome driver object we are using the reference of the web driver.

Abstraction: We use abstract classes to help organize our subclasses. We use Page Object Model layer of the framework. We have Base Page where we have all common members for every page written in that class.

Also, Test Base is abstract class; it has set up and tear down methods and all test classes extend from this class.

Encapsulation we use for our Driver class in utils. We make our driver instance private and static (private static WebDriver driver;).

So, it wouldn't be reachable from outside of the class.

91. How do you apply Collection in automation?

    For example, I use Set class when I need to print out all window handle id's.

92. What was your challenge in your project?

    My application uses many AJAX calls. And I kept getting different exceptions. So I used Explicit waits to handle the synchronization issue.

    AJAX →

    Your page does not reload/refresh completely. It only reloads partially. Only certain section of the page reloads.

    Also, non- technical challenge, when I was just starting to work on my project, I kept finding the bugs in the code and had disagreements with our developer on them and I had issues understanding the requirements. Then I realized that the requirements were not specific enough. So in one of the retro meeting this was brough up and we collaborated as a team and we were able to solve that issue. Or requirements were more detailed, clear, and better organized.

93. How do you get your credentials for testing?

    It depends, usually I get them in spread sheets where all usernames and passwords are saved. For personal info like names, addresses, emails etc. I use library java faker. It helps generating values on the fly programmatically.

94. Where do you your test data?

    I store my URLs, credentials in configuration. properties file.

    Right click on project name, new, file, configuration properties.

    I do not hardcode values into my automation code.

95. What is Singleton?

    It's the pattern that ensures that we are only creating one object per class and all classes share and use the same object. For example, our Driver. Java class in utils package. It ensures we only have one single web driver object, and all classes use the same web driver/ browser.

    We must make private constructor (new keyword will not work), and public static getter method.

96. Javascript Executer usage?

    I use JavaScriptExecutor interface to scroll the page.

97. How do you perform right click, double click, drag and drop actions in automation?

    I use Actions class. First, I create an object from Actions class. The I call the method. And at the end I add perform() or build().perform() methods.

    Ex: Actions actions = new Actions(driver);

    Actions.moveToElement(element).perform();

98. What is POM?

    It is Page Object Model. It is a dedicated java class for each page in our application. And all elements within this class are instance variables on this class. And actions that we do in that class are actually our methods for java page object class. In POM we use annotation @FindBy to locate elements.

POM is great for reusability, maintenance, readability, and it is used as a shared repository and maintenance for the team for web elements, helps to avoid Stale Element Exception.

99. Why do we use Base Page class in pages?

Base page is beneficial. We add all common web elements, methods or anything that's common for all pages. We also create a constructor and initialize (page factory init elements) all elements so we do not have to do it in all other pages. The constructor of Base Page will automatically run first. It is an abstract class.

100. What exceptions are possible in implicit wait and explicit wait?

Explicit wait – time out exception.

Implicit wait – no such element exception.

101. What is stale element exception?

"Old" element exception. Element is located but it is not attached to DOM anymore. We need to re-navigate or re-fresh page.

102. What is TestNG and Cucumber?

It's an automation testing framework. UI testing. Cucumber is not automation tool; it's BDD (tool that implements BDD).Cucumber is used with Junit or TestNG to run and do assertions.

103. Null pointer exception?

It is a run time exception that is indicating that variable is not pointing to any object.

104. Where do you store sensitive test data?

The sensitive information I do not store in configuration properties. That is usually stored in excel files or databases and that is encrypted.

Questions to Ask At the Interview

1. What do you expect from me in the first three months of work?
2. Are you providing any training?
3. What is the culture of your team? Is team participating in any additional team building activities.
4. If I would have difficulties getting the task complete, who would be able to help me?