INTERVIEW QUESTIONS

1. Tell me about yourself

- Most recent Company, Project and Technologies used the in the past project. Previous experiences and technologies used.
- Overview of all the technologies and tools.
- Achievements (Ex: Created Test Automation framework from scratch, Introduced BDD into a normal agile company).
- Other activities (Ex: Attending Agile meeting, mentoring new corners or interns in the company).

2. Tell me about your past experience

- Details about the company, details about the client, details about the project and how that will be used by End-user and Client Nam, if applicable.
- Development used in past project and day to day activities related to manual testing (Ex: following Agile BDD).

3. Day-to-Day Activities in Agile BDD:

- Attending Sprint plan meetings.
- Providing Test estimates and Task identification. Attending Daily Stand up meetings.
- Communicating with Developers, BA's and Product Owners,
- Creating Feature files based on acceptance criteria and reviewing that with BA
- Elaborating scenarios to include Un-Happy path scenarios and Edge cases
- Automating the feature files and getting the failed test results
- Helping developers to create Unit and integration test cases
- Performing exploratory testing, <cross browser testing UI Testing (Look and feel) and executing automating Regression Tests.
- Attending Retrospective aril Shay/ case meetings.
- Planning, Design, Implementing and maintaining Automation Framework.
- Identifying any Defects and' communicating that with BA's and developers and making sure that the identified 'bugs are fixed.
- Moving'the stories in Kanban Board and putting the stories into Done status

4. Tell me about your automation experience, tell your activities in your previous automation role and test automation process

- Identifying the Scope for automation and providing automation estimates
- Creating automation task in the story and providing the automation estimates Generating Step definition files and creating automation Test data
- Identifying the locators and saving in the Page factory file.
- Identifying the global variables (Ex: URL's, Database connections into a .properties file)
- Creating Page objects and Page object methods
- Performing dry run and identifying any potential automation issues

- Committing the code into GitHub and run the newly added scenarios in Continuous integration Server(Ex: Jenkins Configuration)
- Generating Reports and Tagging Scenarios with any automation failure(Ex: Automation issues ,Bugs)

Manual Testing

1. How to identify Test Scenario, Test case, Test steps and Test Data?
2. What is Test plan and Test Strategy?
3. What are the different types of Testing?
4. What is Testing life cycle?
5. What is Defect management and Defect (Bug) life cycle?
6. What is Regression Testing?
7. What are the differences between Waterfall and V-model?
8. What are the advantages and disadvantages of Waterfall and V-model?
9. What are the differences between V-model and Agile Scrum(Advantages and disadvantages of each)

S. No.	V-Model	Agile Scrum
1.	Change Request is not possible	Change Request is possible
	Large development cycles and last minute testing	Shorter cycles, testing throughout the process
3.	Huge requirement documentation. And, if any gaps are found, nothing can be done to fix	Requirements broken into product backlogs, Sprint backlogs and user stories (Acceptance crieteria)
4.	No Visibility of the status of the project	Agile uses KANBAN board to provide the
5.	No meeting and no roles are defined	Meetings and roles are defined. Meeting :Sprint plan meeting, Daily stand up meetings, Retrospective and Showcase meetings Roles: **PO,** Testers, BA, Developers
6.	Test Plan, High level •RS, BR5 documents and Defect reports 'etc.... are documented	No documentation, except Acceptance Criteria and user stories. And Interaction among the team will fix any kind of issues (Gap in the application, minor defects and any blocking issues) by communication.

10. Testing Process in Agile Scrum

- Actively participating in pre-sprint activities and attending Sprint plan meetings.
- Providing estimate andtask identifications,
- Reviewing Acceptance criteria, reviewing technical specification documentation, Ul documentation and creating test cases and test data.
- Executing Test cases on daily builds and working against the tasks created in JIRA
- Adding comments on JIRA Stories, and raising bugs in JIRA.
- Closing the stories and moving the stories to UAT status.
- Performing Regression testing at the end of the sprint.
- Maintaining the test cases as per the acceptance criteria and updating the test cases and regression pack, whenever a new story is added or a change in requirement comes in.

11. What are different challenges in Agile?

- Less documentation
- Maintaining Test cases
- More frequent Regression Testing

12. What are the disadvantages in Agile?

- Test planning is not possible at the system level and integration level. Because, the testing is done on the story level, not at the bigger picture.
- Lack of documentation and fully signed documentation.
- Last minute testing, before the end of the sprint. It will be very hard because of lack of proper testing process.
- If defects are not logged, performing regression testing and tracking the changes will be Impossible Automation Theory

Automation Theory

1. How to identify the test cases which should be automated? The following test cases should be automated:

- Test cases which arerepeatedly executed (at least 3 times)
- All End-user scenarios
- Test cases which are testing the different 3rd party integration systems (Ex: Post code validation functionality while filling the address fields)
- Test cases which contains more Data combination's (Ex different currencies for e-commerce website, different types of card payments in e'-commerce website)

2. Which test cases cannot be automated (which are worth automating, but which will have some technical constraints to automate)?

- Test cases which have the elements Of very dynamic in nature (Ex: Gaming applications)
- Test cases which will not supported by automation tools (Ex: flash player, silver light, Image based application and etc.)

- Dynamically changing UI 'objects which cannot be captured using existing locating techniques
- Command line based application like Mainframes, UNIX server cannot be automated

3. Which Test cases should not be automated (Test cases which are not worth of automating)?

- Extreme test cases which may not be executed more than once, if they are passed.
- UI test cases for which UI is not stable
- Basic validation testing which can be executed quickly by manual testing (Ex: blank username, blank password, providing special chars)
- Look and feel test cases which tests are visual part of the applications (Accessibility testing (Ex: Screen size, color contrast, resolution, page reader))
- Applications which are very stable and don't have much changes in future (chances of failing the test case is minimal in future)

4. What are the Challenges in Automation?

- Change of UI (How to capture Using x-path)
- Synchronization of the application and application response Vs test execution (Using Implicit wait and Explicit wait)
- Maintenance of automation framework if the test case are more than Musing Page object model framework)
- Generating enough detailed level of reports(using User defined reports and Log4j logging)

5. What is the Test Automation process? (or) How do you implement Test Automation process from scratch?

<u>Requirement Gathering Phase:</u>

- Identifying the scope of Test Automation
- Extracting the automatable test cases from Regression or Smoke Test Suite
- Identifying the Test data source (Ex: Database, Random Data, Excel sheet, Feature files, Properties file, Third party services, Hard coded data)
- Identifying the different components in the test automation scope (Ex: Browsers, API Services, Database, Excel sheets, Interfaces(.dat files,.csv files, .txt files,.xml files, .json files, Rest and Soap services))
- Selecting Preferred programming language (Ex: java/c#)
- Version controlling system where the automation code will be stored(Ex: GitHub, SVN,CVS)
- Preferred Unit testing framework(Ex: Junit, Test NG,MS Test, R-Spec, NUnit, XUnit, Punit)
- Preferred Build tools (Ex: Maven, Ant, Nuget, RVM, Symfony)
- Preferred folder Structure (Ex: quick start Arche type)
- Test execution mechanism (Ex: Jenkins, Windows Task scheduler), through command prompt, as part of development build(CI), running locally using IDE or sauce labs)
- Preferred mapping mechanism between manual and automated tests (Ex: Excel sheet, cucumber, fitness, Key word driven framework

- Preferred Reports (Ex: sample user defined reports, cucumber reports, Test NG reports ,
- Email able Reports, Jenkins Selenium plugin Reports, screen shot Reports, Build Monitors)
- Preferred test framework configuration parameters(Ex: command line ,environment variables, Jenkins, .properties file,pom.xml file, global constraints)
- Preferred test execution tools (Ex: selenium grid, Sauce-labs, virtual machines, headless
- testing tools)

Analysis Phase:

- Identify the best set of stools which fits to the company process, budget, application,language, other preferred values from above
- Create a sample dummy project with the best combination of the above and create a automation test plan documentation which includes the below sections

I. Scope
II. Types of Testing
III. Environments
IV. Testing Resources
V. Test Automation Frameworks
VI. Risk Analysis
VII. Maintenance Procedures
VIII. Test Deliverables

- Create a Skelton project based on maven and generate folder structure, dependencies, .properties file and any run time parameters.
- Add the following dependencies as per Test plan
  - Junit
  - Cucumber
  - Selenium
  - Log4J
  - Poi in case of data driven test framework
- Create a different levels of test suites as Regression Test Suite and Smoke Test Suite, resource folder for Cucumber Feature files.
- Create a Page Object Model design document including interface, abstract classes, Base Page, Base Test and Browser factory etc...
- Review the document with the Development Manager and get the sign off
- Create Automated testing task stories in JIRA

Implementation Phase:

- Create a cross browser specific driver class which returns remote Web Driver based on browser selection done from commandline arguments
  $mvn clean test -Dbrowser=firefox-Dtest=smoketest -Denv=test
- Create a common Utility class which holds all the reusable Utils functions (Ex: getDataTime())
- Create a user defined Web Driver objects which adds more Web Driver functionalities (Ex: uniquelTSolutionsDriver, which have methods like uniqueITSolutionsDriver.doubleClick (), uniqueITSolutionsDriversightClick())

- Create a user defined assertions which generates details report Report.asserttrue() which will fail and give a details report and screenshots.
- Implement all the page object methods
- Create feature files and Test data

Maintenance Phase:

- Changing the locators, if any UI changes are happening in the application
- Remove any older functionalities
- Improve the framework on the go

Selenium Web Driver

1. What are the differences between Selenium IDE and Web Driver?

- Selenium IDE is a record and playback tool.
- Selenium IDE can run my linear test cases.
- Selenium IDE cannot fit into automation framework.
- SeleniumIDE19n.prgive dynamic data
- Selenium jp,r90, handle dynamic content.
- Web DriyerAquires selenium server behind the screens to run the tests.
- Web Driver uses the advantages of the programming languages.
- Web Driver is a jar file, which uses other java dependencies like Maven, Jonit, Cucumber, TestNG etc..., to create Test framework.

2. How to perform the Cross-Browser testing using Web Driver?

We can perform Cross browser testing in the following ways:

- Saving all the browser names in an Array, and parameterising the tests and sending the browser name dynamically (Junit parameterisation).
- Creating TestNG.xml file and sending the browser name from external source lEx: Parallel and sequential tests in TestNG)
- Recommended way is to send the browser name as command line argument and create different test jobs in Jenkins or Windows batch commands.
- Give the browser name in excel sheet and create data driven tests.
  For all the above, driver should be Remote Web Driver and send the browser name dynamically.

3. What are the locators in Web Driver?

4. What is the difference between CSS and X-Path?

CSS	X-Path
Is much faster	Is very slow
Don't use DOM	It uses DOM
It selects multiple elements using N-th element	It selects using arrays (Arrayl,Array2)

Don't have robust inbuilt functions	It contains very robust inbuilt functions (contains, end with, Ancestor, Parent, and Begins With etc...)

5. What are the synchronizationmechanisms in Web Driver?

Implicit and Explicit Wait Commands

Implicit Wait	Explicit Wait
hich applies for all the elements throughout the session	Is used to wait for a specific element
Cannot wait for specific conditions (Attribute enable **and** disable)	It waits for specific conditions (Attribute enable **and** disable)

6. How do you handle the alerts in Web Driver?

- Using Alert object and perform command
- We can use exception handling if the alert is un-predictable

7. What is the difference between findElement'and findElements?

- Find Element: will return one Web Elements and throws exception, if the element is not found.
- Find Elements : will return list of Web Elements and return zero based list objects, if element is not found

8. What are the different types of objects in WebDriver?

- Driver, Web Element, Select, Wait, Capabilities, By and etc…

9. What are the differences between JUnit and TestNG?

JUnit	TestNG
Before, After annotations	BeforeMethod and AfterMethod annotations
No data provider annotation	@Dataprovider annotation
	Groups are possible, parallel and sequential tests are possible, testing,xml can remotely control the test execution process by sequence id
	E-mailable Reports

10. What is cucumber JVM? (IBM Interview Question)

- It's a unit testing tool which provides the below functionalities
- It understand .feature files based on the gherkin format
- It generated step definition snippets if it's not developed
- It provides annotation Given, When, Then, But , Before and After annotations

- It provides run file with options (report format, tags, glue, features....)
- It provides html reports or json reports
- It provides tagging mechanism
- It also supports example and data tables

11. How to work with frames in selenium?

- Using **driver.switchTodeafult, driver.switchToframe and driver.switchToWindow** commonds

Agile BDD

1.What are the differences between agile and BDD?

Agile	BDD
Agile is **more** of a shorter version of V-Model, where the development and testing is happening **in** parallel, based on the acceptance	BDD is an extension of TDD, where the behaviour and tests are written before the development
Manual test will be written in excel sheets and not obliged to automate	Tests are written in cucumber feature files and should be 100% automated
Developers write code based on assumptions and tester will find the defects	Developers write the code to pass the test cases written by the tester and the quality of the code should be 100% accurate
Regression testing will be done at the end of the sprint or last week of the sprint	Regression testing is not separately required , because every story is automated
Manual tests are required forUAT, performance, integration and etc...	Only exploratory testing is required which gives high chance of finding the defects

2. What is TDD?

- TDD is a test driven model which are used from long time back where developer writes white box and unit and integration test cases before the code and executes the code after every line of the code is written
- Developer will do refactoring of the code to optimize the quality of the code while relying on the unit tests

3. What are the challenges in BDD?

- Writing 100% failed test cases
- 100% automation testing
- Covering all the scenarios in feature files
- Lack of test management tool
- Writing feature files other than acceptance criteria (Covering Negative and Edge Case scenarios)

4. What is Exploratory testing in BDD?

- Testing the application randomly apart from feature files only with the intension of finding the bugs
- This is very important testing type, which is giving maximum chances to find the defects

Continuous Integration

What is CI?

2. How to configure Jenkins?

- Create a job in Jenkins
- Give the path of GitHub or local Directory
- Give the maven command which includes browser name, environments and to
- Give the triggering mechanism (Specific Time, Build Dependency, GitHub code:ssh)
- Give the reports (using Junit, Selenium, Cucumber(Cucumber report.$)

3. What are the advantages of the CI?

- Faster feedback
- Quality code in GitHub
- Detailed email able reports
- Visibility of the status of the build by Build, Monitors
- Headless automation testing
- Cross browser automation testing

4. What is headless testing?

5. What are the nightly Builds?

- Nightly builds are the regression test suites which will be running after the deployment on test environment without a human intervention.

6. What is selenium: Grid?

- Selenium grid is hub of selenium servers which are linked with specific environment and browsers
- We can link different browsers from different Operating System into one single hub and all the selenium tests will be optioned to the selenium grid and selenium grid will diverts the traffic into specific environment
- Selenium grid will help us to run sequential and parallel tests across the different browsers