**Tasks**

1. Imagine the following situation. You need to establish a QA process in a cross-functional team. The team builds a front-end application using REST APIs.
2. **Where would you start? What would be your first steps?**

Firstly, I will get requirement document. I will attend calls with Business Users and development team to understand the process. Based on that I will come up with strategy and Test plan(Time required to test, phases, Risks, dependencies, constraints, assumptions and tools required to test this)

**2. Which process would you establish around testing new functionality? How would you want the features to be tested?**

If it is a new feature I will start with Manual testing. If the scenarios are working as expected, then I will establish the Automation to do the regression testing for every patch up.

1. **Which tools would you suggest using to help your team with a daily work?**

**Manual Testing:**

* To maintain everyday status, and to maintain the synchronization I would suggest HP dashboard to update their respective tasks.
* I would suggest Jira for writing test cases and for reporting new defects.

**Automation Testing**

* Selenium IDE – For web browser automation I would suggest selenium IDE with Java.
* XPath Helper Wizard – This is an extension for creating xpaths, which are short and easy which is very time saving for creating xpaths manually
* Log4j – I would suggest log4j for creating the logs and for creating console messages.
* Jenkin – For software development process automation and extension of the technical capabilities of teams with continuous data transfer and continuous integration I would suggest this tool
* BitBucket – For cloud source code repository service I would suggest using Git and Mercurial

1. **If you would do a test automation which techniques or best practices would you use the application?**

**Technique & best practices**

* Firstly I would suggest to check which Test Cases to be Automated
* I would suggest to select the Right Automation Testing Tool based on conditions
* I would suggest dividing Automated Testing Efforts among team by using appropriate Quality Test Data
* Create Automated Tests that are Resistant to Changes in the UI by creating the relative xpaths rather than absolute xpaths
* Maintaining a standard repository of reusable test cases for your application will ensure that most common bugs will be caught more quickly.
* Reusing the test cases help to save money on resources to write repetitive tests.

**These are the basis scenarios to check the coverage and most frequently used testing conditions**

* Max length should be set for all the text boxes
* Password field should be masked with asterisks (\*\*\*\*\*)
* Left and right trimming should be done for Password field
* Login credentials in UPPER case should not be treated as invalid
* Forgot Password link should be present on the form
* Reset button should clear data from all the text boxes in the form
* Validation message should be shown when special characters are entered in the username field, or when invalid username and/or password is entered or the fields are left blank
* User should be redirected to Login page if the Login URL is bypassed
* Clicking on ‘Log out’ should take the user back to Home Page
* If the user clicks on “Remember me” option, he should be redirected to appropriate page on next login
* User should be redirected to appropriate page for Forgot password option
* Clicking on Sign Up should take the user to registration page
* Browser compatibility for the search functionality should be verified

1. How would you test search UI functionality of your favorite website (e.g. https://medium.com ,https://www.google.de )?
2. **Choose your favorite website which has search functionality.**

For this, I have selected Make my trip website( <https://www.makemytrip.com> ). I’m also providing the Maven project folder which I developed using the selenium

1. **Document several test cases.**

**PFA sheet on the same**

****

1. **Implement one or two automated tests based on the test cases.**

**PFA zip folder having automated TC’s**



**Implementation steps:**

* I’ve created the maven project using commands through command prompt and connected this project to eclipse using commands through command prompt
* And imported this Maven project to Eclipse
* I’ve created the TC’s under src/test/java,

I’ve created the page objects under src/main/java/page objects

I’ve created the basic browser validations, properties file,log4j file under src/main/java/resources

* I’ve updated the POM.xml with dependencies(selenium,log4j,TestNG) and edited the file with testing suite XML resources, log4j resources
* I’ve created the TestNG XML file
* Then I’ve executed the TC’s using Maven commands (mvn compile) and (mvn test) through command prompt
* I’ve validated the results in logs folder and test-output/index.html path

For continuous integration, we can even try Jenkins(I didn’t use this for my project to make it simple)