Test Automation Strategy

1. Scope

<-- { Describe the scope of the project in high level } --!>

> Example using application https://api.octoperf.com/
> Automation for octoperf will cover functionalities such as creating workspace and edit workspace. As UI Automation tool Selenium Web Driver and BDD approach will be used to cover features developed.

2. Test Strategy

<-- { Describe the testing approach to achieve E2E automation testing } --!>

1.0 - Test Strategy

1. Strategy
2. Test Schedule
3. Planning

2.0 - Test Development

1. Test Plan
2. Test Scripts
3. Test Data

3.0 - Execution

1. Defects
2. Test Reports
3. Test Metrics

4.0 - Defect Management

1. Bug Fixing
2. Bug Verification
3. Bug Tracking

5.0 - Delivery

1. Requirement Verification
2. Regression
3. Document Release

1.1 Strategy

> *<-- { Describe in a few bullet points what is the automation strategy, Framework, Repository, Documentation developed } --!>*

1.2 Schedule

> *<-- { Describe Schedule for Automation testing execution such as Smoke Test, Regression Test, Pre production release, Production Release schedules } --!>*

1.3 Planning

> *<-- { Trace a road map for each component to be developed such as timeframe for Automation Framework, Structure Page Object Model notate here how you plan to achieve it } --!>*

2.1 Test Plan

> *<-- { Add here the test plan you developed for your project } --!>*

2.2 Test Scripts

> *<-- { Describe where your test scripts will be centralized } --!>*

2.3 Test Data

> *<-- { Add information such as where is data coming from, are you generating your own data?*
>
> *is the developer Providing you test data, is your framework generating raw data? Decribe in high level for visibility of product} --!>*

3.1 Defects

> *<-- { Describe the defect life cycle for your project } --!>*

3.2 Test Reports

> *<-- { Add the different types of reports generated by Automation Framework, Also where are you storing the reports, in the Cloud or maybe local repository } --!>*

3.3 Test metrics

> *<-- { Describe any type of metrics based on each sprint to help product understand how much allocation is given to manual testing and Automation testing } --!>*

4.1 Bug Fixing

> *<-- { Describe appoach to track bugs weather its automated or manual step } --!>*

4.2 Bug Verification

> *<-- { What steps are taken for verification describe User Story Validation for Defect Ticket } --!>*

4.3 Bug tracking

> *<-- { What tool is used to track defects based on your project } --!>*

5.1 and 5.2 Requirement verification Regression

> *<-- { Defenition of DONE should be describe for verification of requirements. When is regression happening what is the coverage perecentage detail here in high level } --!>*

5.3 Document Release

> *<-- { When is regression happening what is the coverage perecentage detail here in high level } --!>*

3. Tools

Tools	Repository	Manual or Automation	Description
Selenium WebDriver	<git repository>	Automation	• BDD • Page Obeject Model • Maven • TestNG • etc…

4. Schedules

> *<-- { Detail Test execution schedules, such as Regression, smoke along with the repositories } --!>*

5. Environment

> *<-- { Detail if this will run in QA, DEV, PRODUCTION environment }*

6. Deliverables

> *<-- { Detail in a documentation the deliverables and versions for each release. }*

7. Risks

> *<-- { List here possible outcomes that can cause Risks for deployment such as environment issues, team dependencies, framework support and test coverage }*

8. Reports Results

> *<-- { Describe reports for automation testing such as html reports, json reports also where are this reports found for visibility of product }*

9. Code Reviews

> *<-- { Detail the process for Code Reviews. will it be performed after every commit or will it be 1 on 1 code review with a team member }*