

Hi, Myself Vijay Kumar, coming from Chennai

I have 3.5 years of experience as software engineer in ABC Technologies, and

More familiar in tools and technologies such as Selenium-Java, Maven, cucumber, POM, Data driven, Git, Jenkins etc.

Coming to my project,

My project name is Bo Concept

It is E-commerce domain project

It has various module as Product, Service, Find store, Contract

We follow Agile - Scrum Methodologies in our project,

We are a team of 9 in my project including dev, tester and manager and

We follow 2 weeks of sprint

Daily Activities:

We are using JIRA defect tracking tool to manage our scrum board

My sprint starts with **Sprint Grooming Meeting**

In which we will discuss on taking the user stories

from product backlog to sprint backlog

based on priority and velocity of the team (total hours that team can spend for that week

- it is based on capacity of the individual) and also

based on user story points (sizing - simple, moderate and complex),

we will move stories to **work In-progress** from **To-Do list**

We have **Daily Scrum Call**

where we will discuss

what we did yesterday,

what we are going to do today and

Any impediments if any.

After the end of 2 weeks of sprint, we will have **Sprint Demo Meeting**

where we will demonstrate the task which we have completed for the sprint,

if the task or story is completed, we will move it to **Done** status,

if it is not completed, we will continue that in next sprint

Finally, we will have **Sprint Retrospective Meeting**,

where we will discuss

what went well on last sprint and

if any improvements need

It is like increment iterative model

My project is built on **Maven** (Dependencies Management and It creates Folder Structure) and integrated with **selenium web driver** (to automate our web application by using programming language **Java**) and we follow **cucumber BDD framework** (**Gherkin** plain English language script which makes manual tester, managers and client to easily understand about the test scenarios) and we also follow **POM** (Page object design pattern (for reusable and maintainable of web locators where we collect all the locators for each page and store in it separate class and using **page factory** we initialize the web elements)) and with **singleton design pattern** (a class must ensure that only single instance should be created and single object can be used by all other classes wherever required) and also we have **library class** (Base class) where we will have reusable methods – Selenium-Java actions and we will maintain **property file** where we define which browser to execute, application URL and implicit wait (which we can change it dynamically instead of changing it in our code) and we will have **driver files** in it and finally we have **report folder** where we collect html, json and xml reports with screenshot and we use cucumber extent report for our project. Once execution is completed, we will maintain our code in **GitHub repository** using git commands and we integrate it in **Jenkins** to do continuous testing whenever deployment happens or to run at any particular time using build triggers (build periodically or POLL SCM (when new commits)) using maven goal (clean test -Dcucumber.options=--tags @test) to execute our project and using **cucumber reports** we will see html reports with screenshots and history trends and we will **email** to clients.

Project Structure:

POM - Page object model

BASE CLASS - library file

Feature file - test script

Step definition - selenium java method for each step

Runner - Runner class where we mention feature file path, step definition, reports, and tags

Resources - driver and property file

Reports - html report

Helper - framework support class

Parallel testing - maven clean test -P Featured (we mention in pom.xml)

Maven:

My project is built on **Maven**

Dependencies Management and It creates Folder Structure

Selenium WebDriver

It is integrated with **selenium web driver**

Java

To automate our web application by using programming language **Java**

Cucumber

In **cucumber BDD framework**, we use **Gherkin** plain English language script which makes manual tester, managers and client to easily understand about the test scenarios

POM – Page Object Model

Page object design pattern is used for reusable and maintainable of web locators

Here we collect all the locators for each page and store it in separate class

Using **page factory**, we initialize the web elements

Singleton design pattern

A class must ensure that only single instance should be created and, single object can be used by all other classes wherever required

Library class - Base class

where we will have reusable methods

This is Selenium-Java actions

Property file

Each Parameter in this file are stored as Key-Value Pair

We can change it dynamically instead of changing it in our code

Like browser to execute, application URL and implicit wait.

Driver Folder

This has different driver execution files to test our test scripts

Like Chrome, IE, Firefox

Report folder

Where we collect html, json and xml reports with screenshot and we use cucumber extent report for our project.

GIT

Once execution is completed, we will maintain our code in **GitHub repository** using git commands.

Jenkins

We integrate it with **Jenkins** to do continuous testing whenever deployment happens or to run at any particular time using build triggers (build periodically or POLL SCM (when new commits))

Using maven goal (clean test -Dcucumber.options=--tags @test) to execute our project and using **cucumber reports** we will see html reports with screenshots and history trends and we will **email** to clients.