

BDD Automation using Cucumber.JS, Selenium and JavaScript

- Create a new folder, **CucumberJSSelenium**, this will be our Project root
- Open this folder via VSCode
- In the VSCode terminal initiate a new NPM project – **npm init**. This will create a package.json file in the CucumberJSPlaywright folder.
- Install Cucumber -- **npm install --save-dev @cucumber/cucumber**
- Install Selenium -- **npm install selenium-webdriver --save-dev**
- Install WebDriver Manager -- **npm install webdriver-manager --save-dev**
- Install Chai Assertion Library -- **npm install --save-dev chai**
- Create a folder, **features**. This will contain all the project files.
- Create the feature files in the features folder or create multiple folders under **features/** which will then have the feature files.
- Create a subfolder, **features/step_definitions**. This folder will have all the step definitions
- Create a subfolder, **features/reports**. This folder will store the reports generated after the test run
- Create a subfolder **features/support** and create the **hooks.js**, **methods.js**, **configs.js** and **selectors.js** files
- In the Project root (outside the features folder), create a **cucumber.js** file

Running commands:

- **.\node_modules\.bin\cucumber-js** – This will run all the feature files
- **.\node_modules\.bin\cucumber-js --format html:.\features\reports\Report.html** – This will run all the feature files and store the report to the reports folder
- **.\node_modules\.bin\cucumber-js .\features\feature_addition\add2.feature --format html:.\features\reports\Report.html** – This will run the add2.feature file and store the generated report to the reports folder
- **.\node_modules\.bin\cucumber-js -tags "@new" --format html:.\features\reports\Report.html** – This will run the scenario with @new tag and store the generated report to the reports folder

Gluing feature files to step defs in VSCode :

- Install Cucumber (Gherkin) Full Support via the VSCode Store (Reload after installation)
- Open File → Preferences → Settings
- Go to Workspace → Open settings.json (Icon on top right)
- Add the below code

```
{  
  "cucumberautocomplete.strictGherkinCompletion": true,  
  "cucumberautocomplete.syncfeatures": "features/*/*feature",  
  "cucumberautocomplete.steps": "features/step_definitions/*.js"  
}
```

- Reload VSCode and now you can navigate from Feature files to step def

- Reference : <https://stackoverflow.com/questions/55363967/navigating-from-feature-file-to-step-definition-any-plugins>

Folder Structure:

