# Framework Design Document

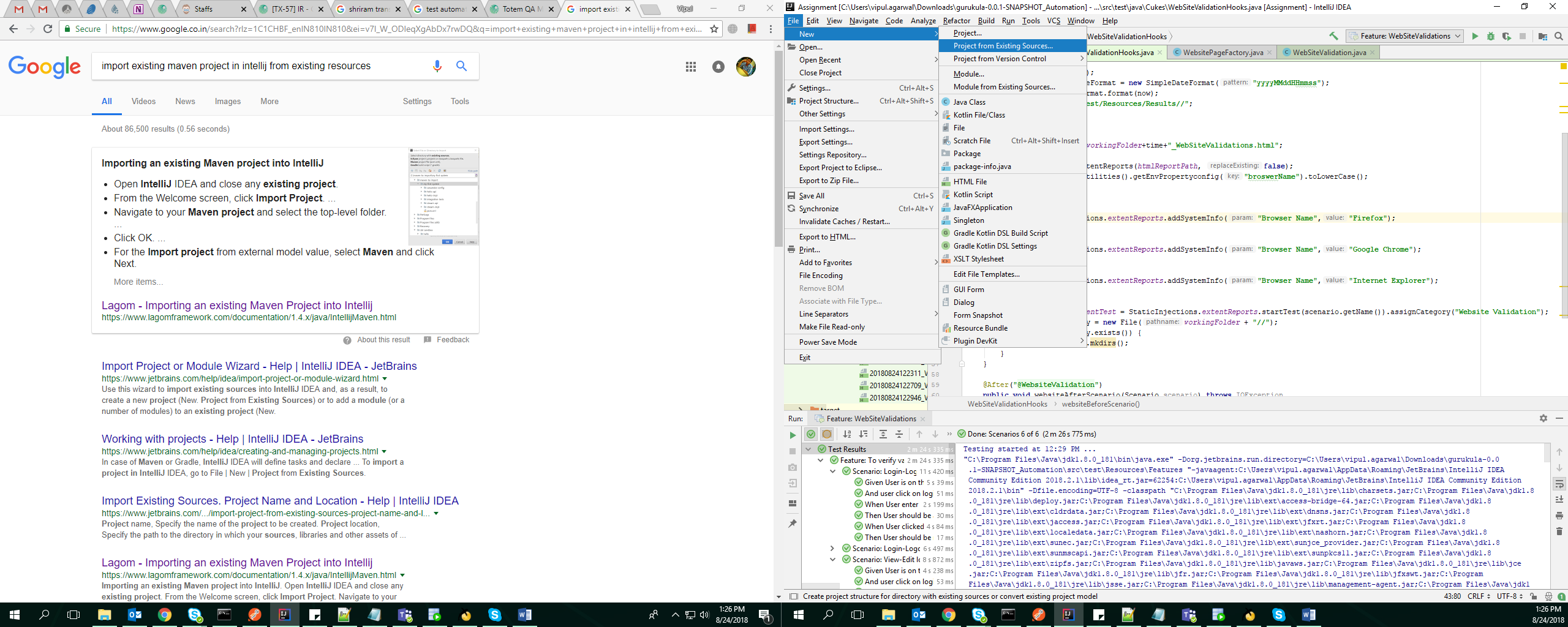
1. **Language** - Java 8
2. **Test** **Driving** **Tool** - **Cucumber**- Cucumber is a software tool used by computer programmers for testing other software. It runs automated acceptance tests written in a behavior-driven development (BDD) style.[8] Central to the Cucumber BDD approach is its plain language parser called Gherkin. It allows expected software behaviors to be specified in a logical language that customers can understand. As such, Cucumber allows the execution of feature documentation written in business-facing text.
3. **Result** - **Extent** **Report** - Extent Report is a HTML reporting library for Selenium WebDriver for Java which is to a great degree simple to use and makes excellent execution reports. We can use this tool within our Cucumber automation framework.
4. **Build Management Tool** - **Maven** - Maven is a [build automation](https://en.wikipedia.org/wiki/Build_automation) tool used primarily for [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) projects. Maven addresses two aspects of building software: first, it describes how software is built,[[clarification needed](https://en.wikipedia.org/wiki/Wikipedia:Please_clarify)]and second, it describes its dependencies.
5. **Web Automation Tool** - **Selenium** - Selenium WebDriver accepts commands (sent in Selenese, or via a Client API) and sends them to a browser. This is implemented through a browser-specific browser driver, which sends commands to a browser and retrieves results.

## Pre- conditions

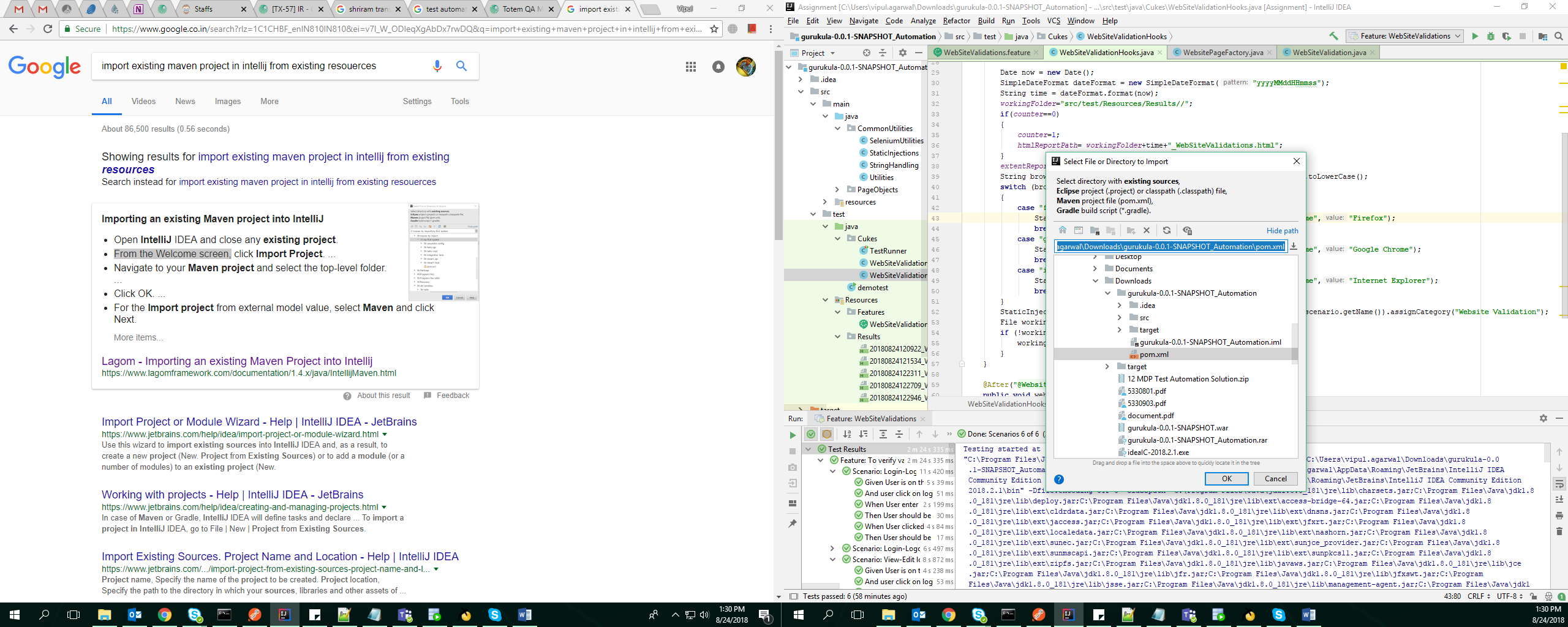
1. JDK 1.8
2. IntelliJ IDEA (Community Version)

## Getting Code Base

1. Extract the Project GurukulAssignment.zip in any folder
2. Open IntelliJ IDEA and close any existing project.
3. From the Welcome screen, Choose option Project from Existing Sources as shown below.



1. Select pom.xml from the directory where project has been extracted in first step

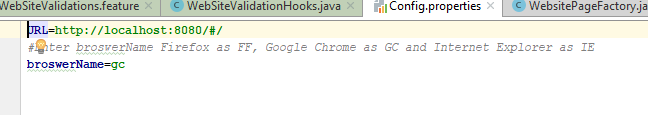


1. Project will start downloading all the dependencies and plugins

## URL and Browser Configuration

As shown below

1. Go to path src\main\resources\Config.properties
2. URL of website can be changed from URL property
3. TO select which browser to launch for automation use browserName property



## Execution

1. Go to folder “**src\test\Resources\Features**”
2. Open **WebSiteValidations.feature** file
3. To run all the scenarios in this feature file, Right Click on the feature file and select Run **‘Feature: WebSiteValidations’**
4. To run one scenario at a time right click on any of the scenario and click on Run **‘Scenario: abc’**

## Test Results

1. Go to src\test\Resources\Results folder
2. Open the recently created (based on timestamp) .html file.

