TEST AUTOMATION

EPAM Systems

A Training Report

Submitted in partial fulfilment of the requirements for the award of degree of

BACHELOR OF TECHNOLOGY

COMPUTER SCIENCE AND ENGINEERING

LOVELY PROFESSIONAL UNIVERSITY

PHAGWARA, PUNJAB

FROM JAN 2023 TO APR 2023

SUBMITTED BY
Soveet Rout
11902267

SUBMITTED TO
SAKSHI
Assistant Professor

Student Declaration

To whom so ever it may concern

I, Soveet Rout, 11902267, hereby declare that the work done by me on "**TEST AUTOMATION**" from **JAN 2023** to **APR 2023**, under the supervision of **SAKSHI** and Lovely Professional University, Phagwara, Punjab, is a record of original work for the partial fulfilment of the requirements for the award of the degree Computer Science and Engineering.

Name: Soveet Rout
Registration number: 11902267

Soveet Rout

Signature of Student
Dated: 01/05/2023

Table of Contents

CHAPTER 1
ABOUT THE COMPANY

1.1 Company's Vision and Mission

Our teams of technologists, strategists and designers transform our customers' business through a combination of engineering expertise, design thinking and business consulting. The vision statement for EPAM Systems Inc. is its strategic plan for the future – it defines what and where EPAM Systems Inc. Company wants to be in the future. The vision statement for EPAM Systems Inc. is a document identifying the goals of EPAM Systems Inc. to facilitate its strategic, managerial, as well as general decision making processes.

The vision statement of EPAM Systems Inc. is brief and to the point. This means that the company has not used long dialects and dialogues to delivers its opinion ad stance to the public and relevant stakeholders. The vision statement should be brief and comprehensive – it should communicate the essence of the business, and its future plans to help the stakeholders understand its business philosophy and business strategy.

The vision statement of EPAM Systems Inc. should be brief but should be holistic in nature. This means that the visions statement should be complete in its description and information of what the company desires, and how it plans to achieve its long term goals strategically. The vision statement should be a comprehensive statement identifying the company's core strengths, which would enable it to achieve its futuristic goals.

The mission statement for EPAM Systems Inc. is a public document that details the values and strategic aims of EPAM Systems Inc. The mission statement of EPAM Systems Inc also identifies the purpose of the organization existence, highlighting the services and the products it offers. Further, the mission statement also identifies the organization's operational goals for EPAM Systems Inc. the processes the company uses to achieve those, the target customer groups, and the region where the company operates.

The mission statement of EPAM Systems Inc. focuses on addressing issues of customer satisfaction. The mission statement of EPAM Systems Inc. has identified its target customer groups, and also identified their needs and demands. The mission statement reflects on how its products and services work towards increasing customer satisfaction for its target customers.

The mission statement of EPAM Systems Inc. is based on its integral strengths and competencies. This is important for EPAM Systems Inc. as the mission statement will highlight the different systems and processes as well as strategic tactics that the company uses to achieve its organizational and strategic goals. The achievement of the goals will depend on how well EPAM Systems Inc. makes use of its core competencies.

The mission statement for EPAM Systems Inc. is also realistic and clear. This means that EPAM Systems Inc. has used simple, string, and easily understood words and phrases in the drafting of its mission statement. Clarity is important so that the mission statement is understood by all relevant stakeholders of EPAM Systems Inc. Company. EPAM Systems Inc.'s mission statement is also realistic, which makes it able to achieve various set goals and targets.

The mission statement of EPAM Systems Inc. is motivational in that it works towards inspiring the employees and the workforce towards giving their optimal best performance towards the goal achievement of EPAM Systems Inc. The mission statement of EPAM Systems Inc. is also inspirational in that it develops the need for growth and progress in individuals – for the betterment of not only the company but also for their own selves.

1.2 Origin and Growth of Company

In 1993, Arkadiy Dobkin and Leo Lozner founded EPAM, a global software engineering services company, in New Jersey, USA and Minsk, Belarus.

In 2002, EPAM was ranked publicly for the first time as a fast-growing company by Deloitte & Touche.

In 2012, EPAM was listed on New York Stock Exchange under moniker EPAM, becoming the first company from Belarus on NYSE.

Timeline

In 2004, EPAM acquired Fathom Technology, a software development services company based in Budapest, Hungary, expanding its offshore services beyond North America. A couple years later, EPAM secured an equity investment from Siguler Guff to fund its competitive growth plans.

In 2006, EPAM acquired VDI, a software development services company with delivery centers in Russia, which expanded the company's presence in the CIS region. That year, EPAM CEO Arkadiy Dobkin was named one of the Top 25 Most Influential Consultants of the Year by Consulting Magazine.

In late 2012, EPAM made two acquisitions – Thoughtcorp, which expanded its service offerings in Agile, business intelligence and mobile, and Empathy Lab, which established a digital engagement practice focusing on customer experience, design and eCommerce.

EPAM made two acquisitions in 2018 to expand its service offerings: Continuum (now EPAM Continuum) and TH_NK to add consulting capabilities and develop its digital and service design practices. Also that year, EPAM launched InfoNgen®, a text analytics and sentiment analysis enterprise software product that uses artificial intelligence.

The company also productized TelescopeAI®, an artificial intelligence-based platform for IT operations and workforce management, which won a 2019 Big Innovation Award presented by the Business Intelligence Group.

In 2019, EPAM joined the Blockchain in Transport Alliance (BiTA). That year, the company launched EPAM Continuum, its service for business, experience and technology consulting.

The company also launched EPAM SolutionsHub, a catalogue of its software products, accelerators and open source platforms. As part of its SolutionsHub launch, EPAM also released the Open Source Contributor Index (OSCI), a tool that ranks the top open source contributors by commercial organization.

In August 2021, EPAM expanded its presence in Latin America through the acquisition of Colombia-based S4N, a software development services firm specializing in the design and development of modern software products and enterprise platforms.

In July 2021, EPAM acquired CORE SE, a professional service provider specializing in IT strategy and technology-driven transformations, to further expand its Western European footprint in the DACH region.

In May 2021, EPAM acquired Just-BI, a Netherlands-based consultancy specializing SAP/S4HANA and enterprise data and analytics program management.

EPAM acquired Israel-based cyber-security services provider White-Hat Ltd. in May 2021.

In April 2021, EPAM acquired PolSource, a Salesforce Consulting Partner with more than 350 Salesforce specialists across the Americas and Europe.

In December 2021, EPAM joined the S&P 500.

In May 2021, EPAM Systems ranked 1,804 on the Forbes Global 2000 list.

CHAPTER 2
PROJECTS UNDERTAKEN

2.1 Spotify-UI Test

For my project report, I conducted a UI test for Spotify, focusing on the login process, playlist creation, adding songs, playing and pausing songs, and deleting both songs and playlists.

1. Login process: I first tested the login process of Spotify, making sure that users can enter their username and password correctly and access their account. I also checked for any error messages or bugs that may occur during the login process.
2. Playlist creation: I then proceeded to test the playlist creation feature of Spotify. I created a new playlist and ensured that the name was entered correctly. I also tested the addition of a description and cover art. I then checked that the playlist appeared in the user's library.
3. Adding songs: Next, I tested the ability to add songs to a playlist. I selected a song and added it to the newly created playlist. I then checked that the song was added correctly and appeared in the playlist.
4. Play and pause: I tested the play and pause feature of Spotify. I clicked on the play button to start playing the song and then clicked on the pause button to stop the song. I then checked that the play and pause buttons worked correctly and the song played and paused as expected.
5. Deletion of song and playlist: Finally, I tested the ability to delete both songs and playlists. I removed a song from the playlist and checked that it was no longer in the playlist. I then deleted the playlist entirely and ensured that it was removed from the user's library.

Overall, I found that the UI of Spotify was intuitive and easy to use. The login process was straightforward, and playlist creation and management were simple. Adding songs, playing and pausing songs, and deleting songs and playlists were also easy to do. There were no significant issues or bugs that I encountered during the test.

2.1.1 Technologies Used

For my project report, I used several technologies to perform UI testing for the Spotify application. These technologies include Maven, Java, Selenium Web--Driver, Jenkins, TestNG, and the Page Object Pattern.

1. Maven: Maven is a build automation tool that helps in managing project dependencies, build processes, and deployment. It simplifies the development process by automating the building of projects and managing libraries and dependencies. I used Maven to build and manage the project dependencies of my UI testing framework.

2. Java: Java is a popular programming language used for developing enterprise applications, web applications, and mobile applications. I used Java to write the test scripts for my UI testing framework. Java is a robust language with an extensive library of APIs and tools, making it an excellent choice for UI testing.

3. Selenium WebDriver: Selenium WebDriver is a popular open-source tool used for automating web browsers. It supports different browsers, including Chrome, Firefox, and Safari, and provides a set of APIs for interacting with web elements. I used Selenium WebDriver to automate my UI testing for the Spotify application.

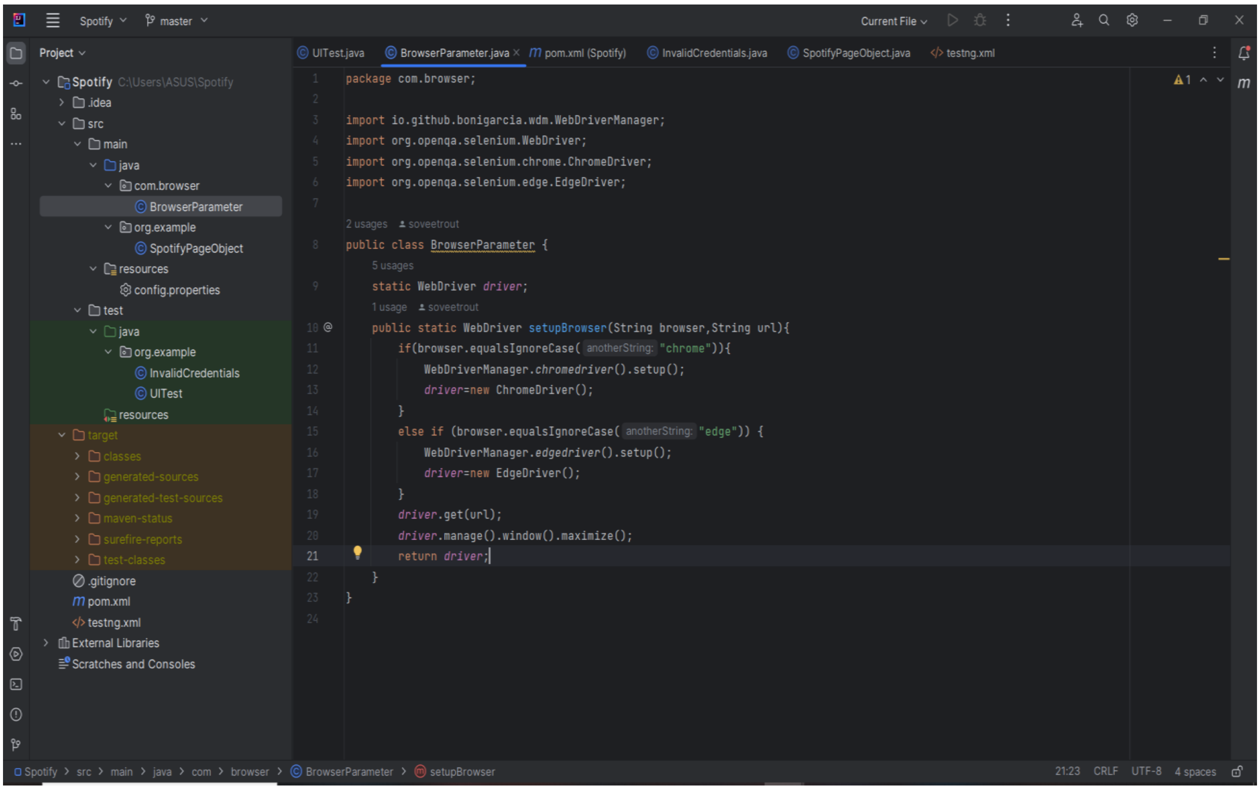
4. Jenkins: Jenkins is an open-source automation server that helps in building, testing, and deploying software applications. It provides a wide range of plugins that enable continuous integration and continuous deployment. I used Jenkins to set up and run automated tests as part of the CI/CD pipeline.

5. TestNG: TestNG is a testing framework that provides a wide range of features for automating unit tests, integration tests, and functional tests. It supports different test types, including data-driven tests, parameterized tests, and dependency tests. I used TestNG to create and execute test cases for my UI testing framework.

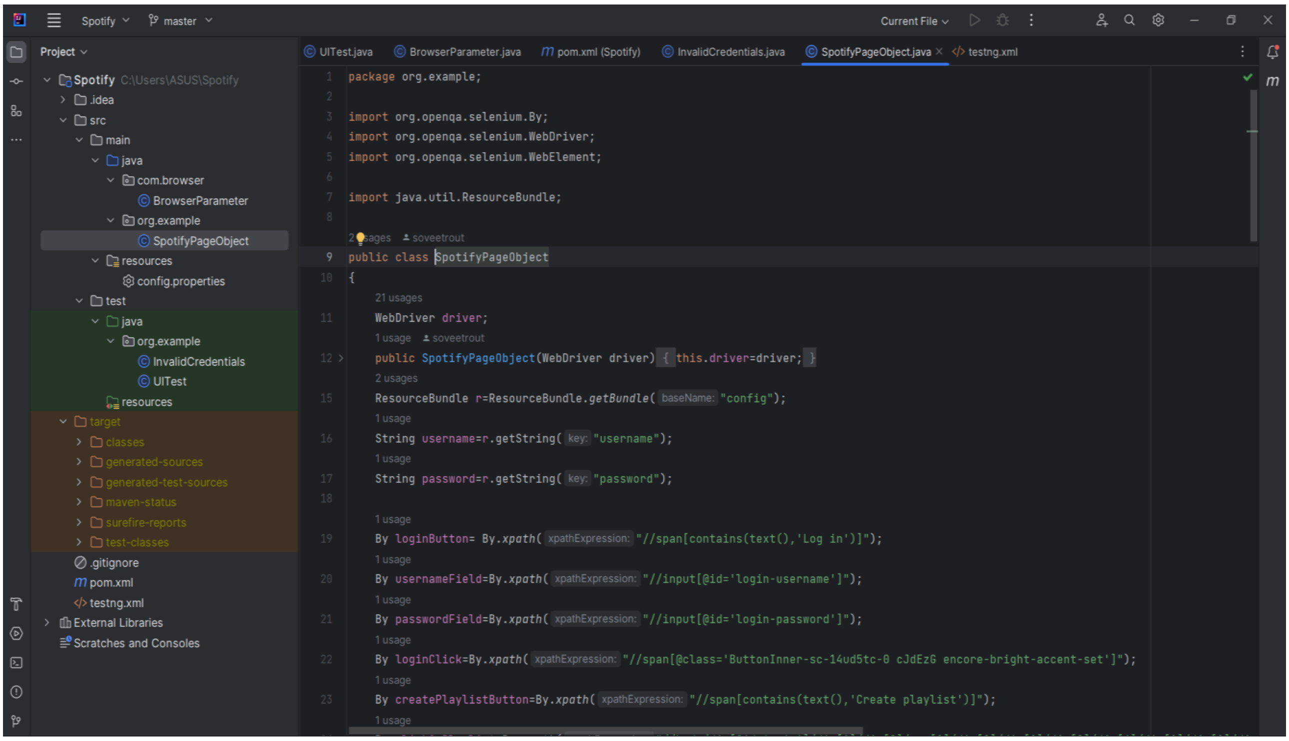
6. Page Object Pattern: The Page Object Pattern is a design pattern used for implementing UI testing frameworks. It helps in creating a modular and maintainable code structure by separating the page objects from the test scripts. I used the Page Object Pattern to structure my UI testing framework, making it easier to maintain and update.

In summary, I used Maven, Java, Selenium WebDriver, Jenkins, TestNG, and the Page Object Pattern to create a robust and reliable UI testing framework for the Spotify application. These technologies helped me automate the testing process, reducing the time and effort required to test the application manually.

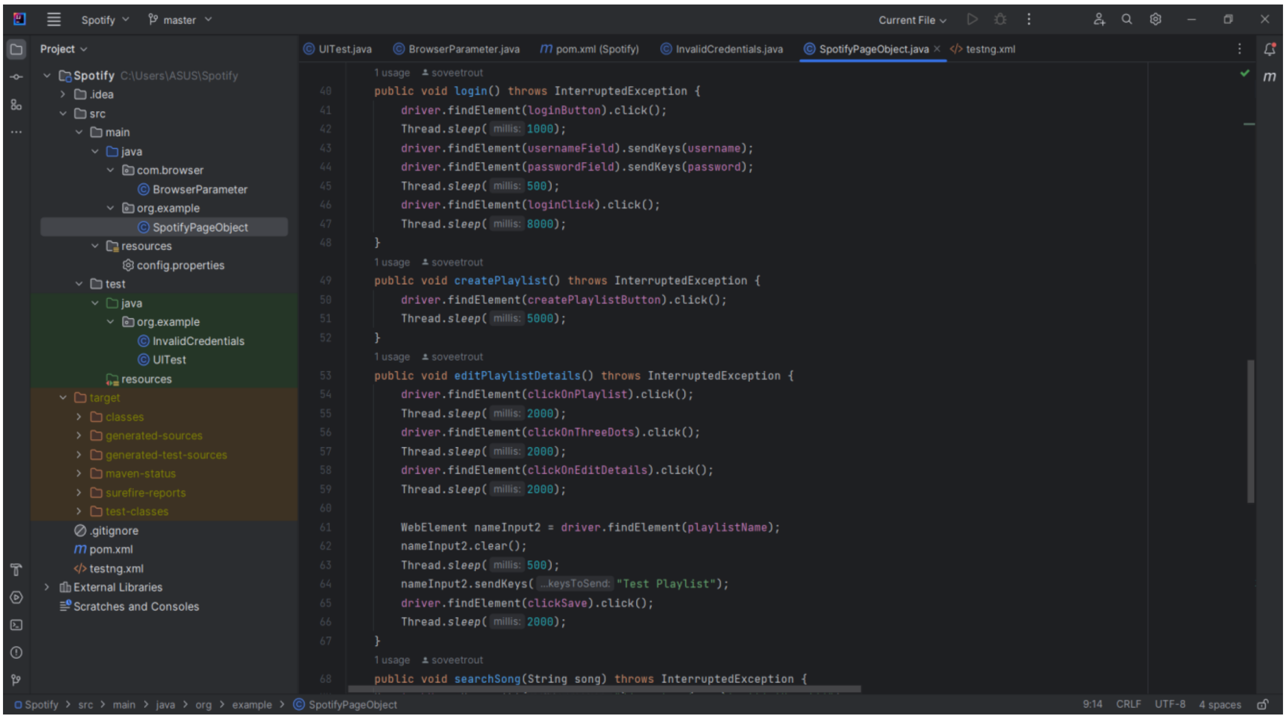
2.1.2 Snapshots



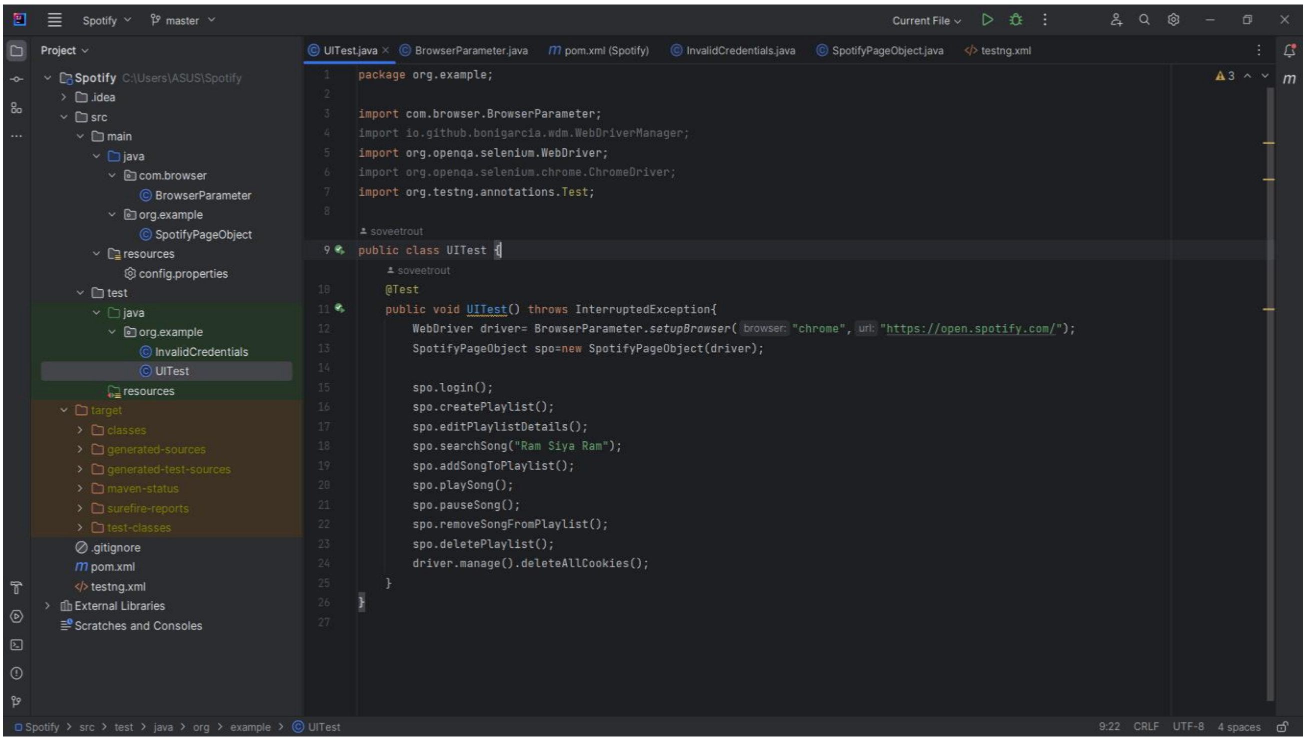
```java
package com.browser;

import io.github.bonigarcia.wdm.WebDriverManager;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.edge.EdgeDriver;

public class BrowserParameter {
    static WebDriver driver;
    public static WebDriver setupBrowser(String browser,String url){
        if(browser.equalsIgnoreCase("chrome")){
            WebDriverManager.chromedriver().setup();
            driver=new ChromeDriver();
        }
        else if (browser.equalsIgnoreCase("edge")) {
            WebDriverManager.edgedriver().setup();
            driver=new EdgeDriver();
        }
        driver.get(url);
        driver.manage().window().maximize();
        return driver;
    }
}
```



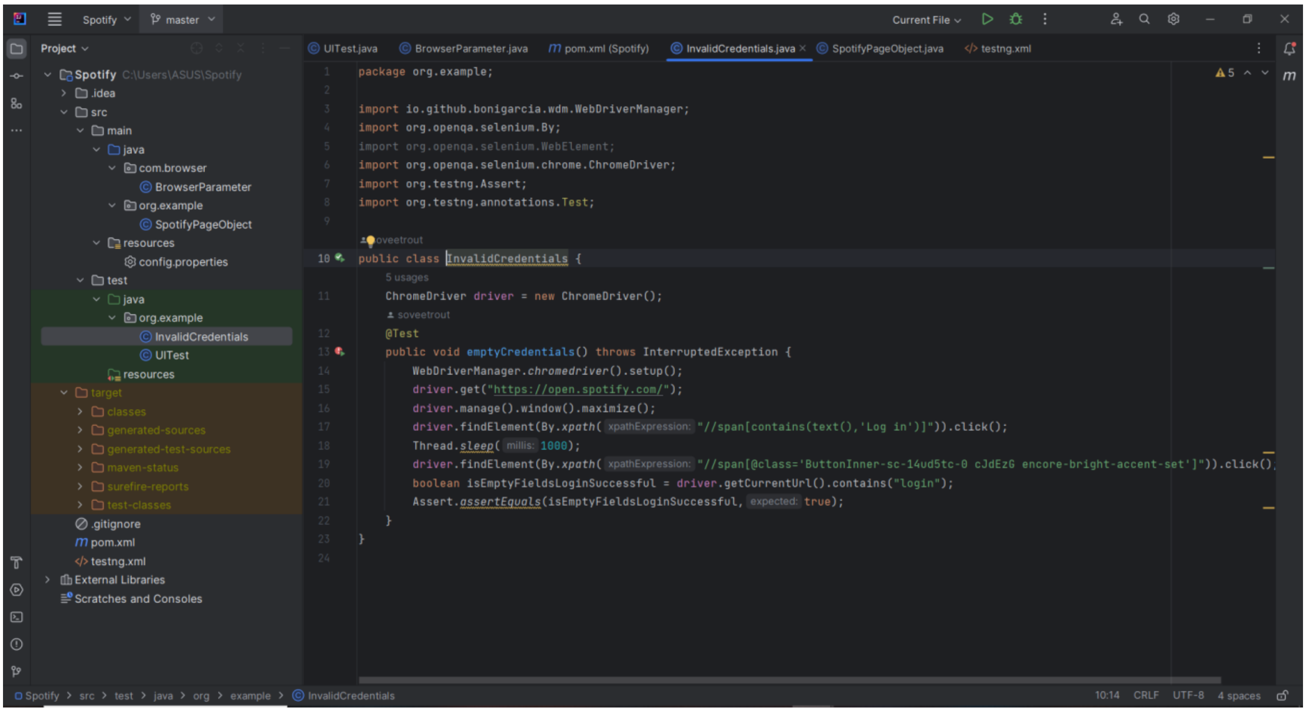
```java
package org.example;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;

import java.util.ResourceBundle;

public class SpotifyPageObject
{
    WebDriver driver;
    public SpotifyPageObject(WebDriver driver) { this.driver=driver; }
    ResourceBundle r=ResourceBundle.getBundle("config");
    String username=r.getString("username");
    String password=r.getString("password");

    By loginButton= By.xpath("//span[contains(text(),'Log in')]");
    By usernameField=By.xpath("//input[@id='login-username']");
    By passwordField=By.xpath("//input[@id='login-password']");
    By loginClick=By.xpath("//span[@class='ButtonInner-sc-14ud5tc-0 cJdEzG encore-bright-accent-set']");
    By createPlaylistButton=By.xpath("//span[contains(text(),'Create playlist')]");
```



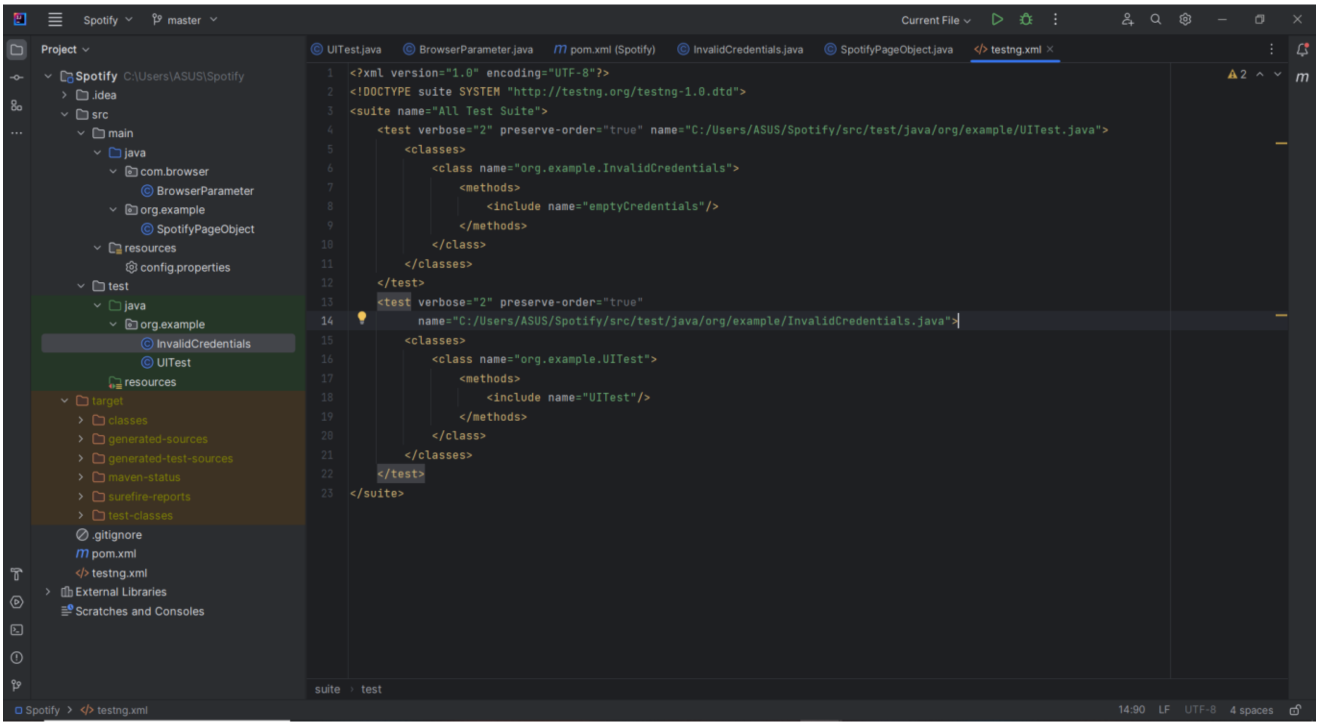
```java
    public void login() throws InterruptedException {
        driver.findElement(loginButton).click();
        Thread.sleep(1000);
        driver.findElement(usernameField).sendKeys(username);
        driver.findElement(passwordField).sendKeys(password);
        Thread.sleep(500);
        driver.findElement(loginClick).click();
        Thread.sleep(8000);
    }
    public void createPlaylist() throws InterruptedException {
        driver.findElement(createPlaylistButton).click();
        Thread.sleep(5000);
    }
    public void editPlaylistDetails() throws InterruptedException {
        driver.findElement(clickOnPlaylist).click();
        Thread.sleep(2000);
        driver.findElement(clickOnThreeDots).click();
        Thread.sleep(2000);
        driver.findElement(clickOnEditDetails).click();
        Thread.sleep(2000);

        WebElement nameInput2 = driver.findElement(playlistName);
        nameInput2.clear();
        Thread.sleep(500);
        nameInput2.sendKeys("Test Playlist");
        driver.findElement(clickSave).click();
        Thread.sleep(2000);
    }
    public void searchSong(String song) throws InterruptedException {
```



```java
package org.example;

import com.browser.BrowserParameter;
import io.github.bonigarcia.wdm.WebDriverManager;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.Test;

public class UITest {
    @Test
    public void UITest() throws InterruptedException{
        WebDriver driver= BrowserParameter.setupBrowser( "chrome", "https://open.spotify.com/");
        SpotifyPageObject spo=new SpotifyPageObject(driver);

        spo.login();
        spo.createPlaylist();
        spo.editPlaylistDetails();
        spo.searchSong("Ram Siya Ram");
        spo.addSongToPlaylist();
        spo.playSong();
        spo.pauseSong();
        spo.removeSongFromPlaylist();
        spo.deletePlaylist();
        driver.manage().deleteAllCookies();
    }
}
```



```java
package org.example;

import io.github.bonigarcia.wdm.WebDriverManager;
import org.openqa.selenium.By;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.Assert;
import org.testng.annotations.Test;

public class InvalidCredentials {
    ChromeDriver driver = new ChromeDriver();
    @Test
    public void emptyCredentials() throws InterruptedException {
        WebDriverManager.chromedriver().setup();
        driver.get("https://open.spotify.com/");
        driver.manage().window().maximize();
        driver.findElement(By.xpath( "//span[contains(text(),'Log in')]")).click();
        Thread.sleep( 1000);
        driver.findElement(By.xpath( "//span[@class='ButtonInner-sc-14ud5tc-0 cJdEzG encore-bright-accent-set']")).click()
        boolean isEmptyFieldsLoginSuccessful = driver.getCurrentUrl().contains("login");
        Assert.assertEquals(isEmptyFieldsLoginSuccessful, true);
    }
}
```



```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">
<suite name="All Test Suite">
    <test verbose="2" preserve-order="true" name="C:/Users/ASUS/Spotify/src/test/java/org/example/UITest.java">
        <classes>
            <class name="org.example.InvalidCredentials">
                <methods>
                    <include name="emptyCredentials"/>
                </methods>
            </class>
        </classes>
    </test>
    <test verbose="2" preserve-order="true"
          name="C:/Users/ASUS/Spotify/src/test/java/org/example/InvalidCredentials.java">
        <classes>
            <class name="org.example.UITest">
                <methods>
                    <include name="UITest"/>
                </methods>
            </class>
        </classes>
    </test>
</suite>
```

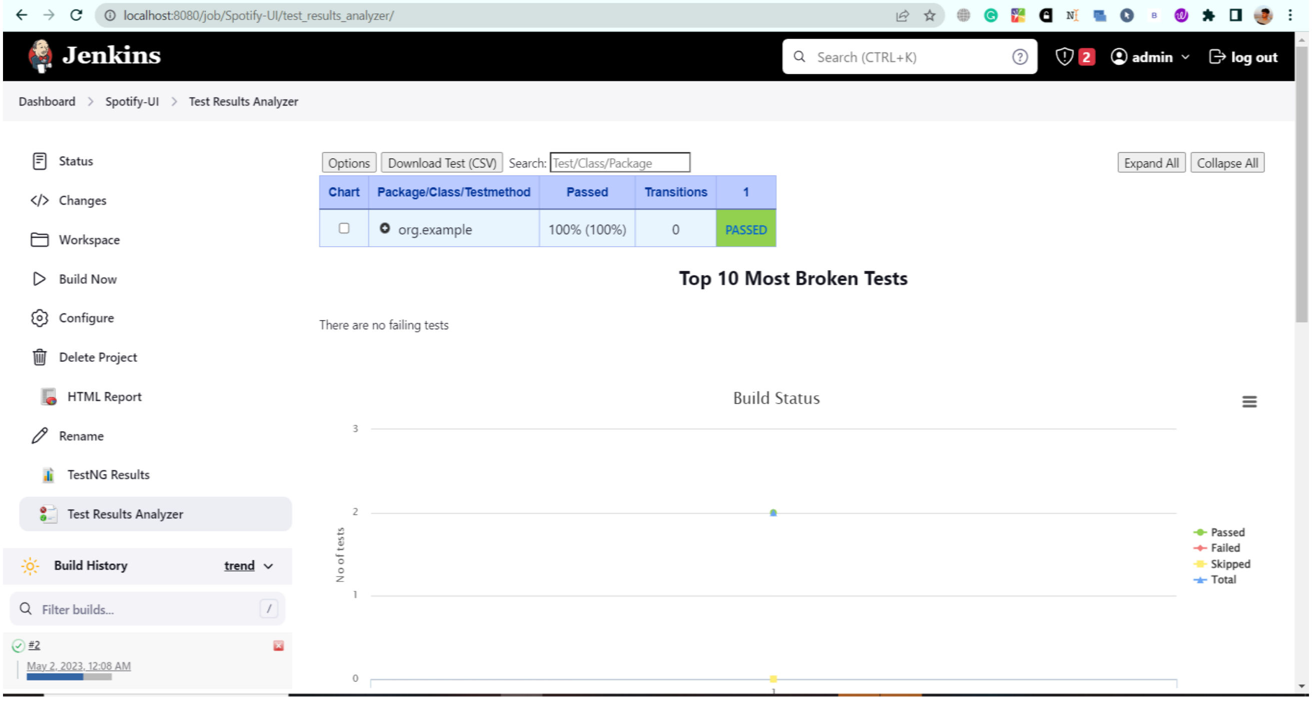
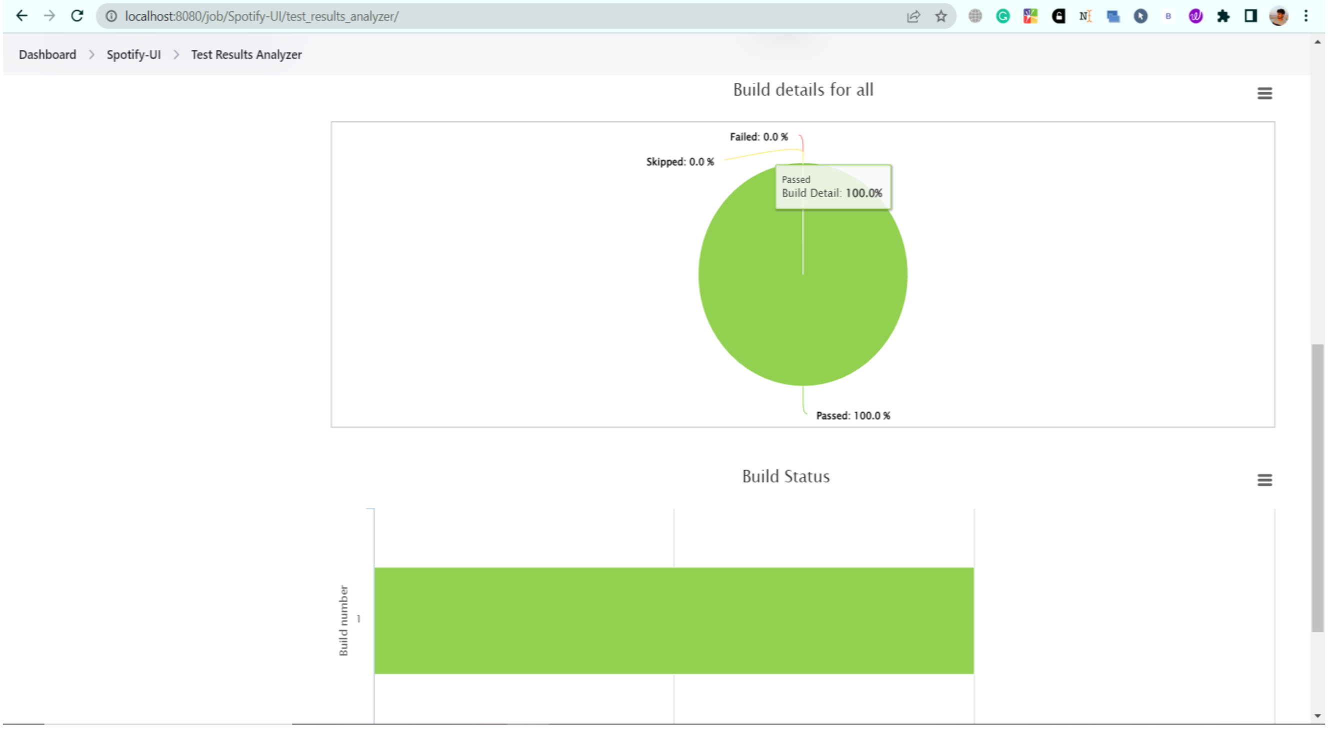


Chart	Package/Class/Testmethod	Passed	Transitions	1
□	org.example	100% (100%)	0	PASSED

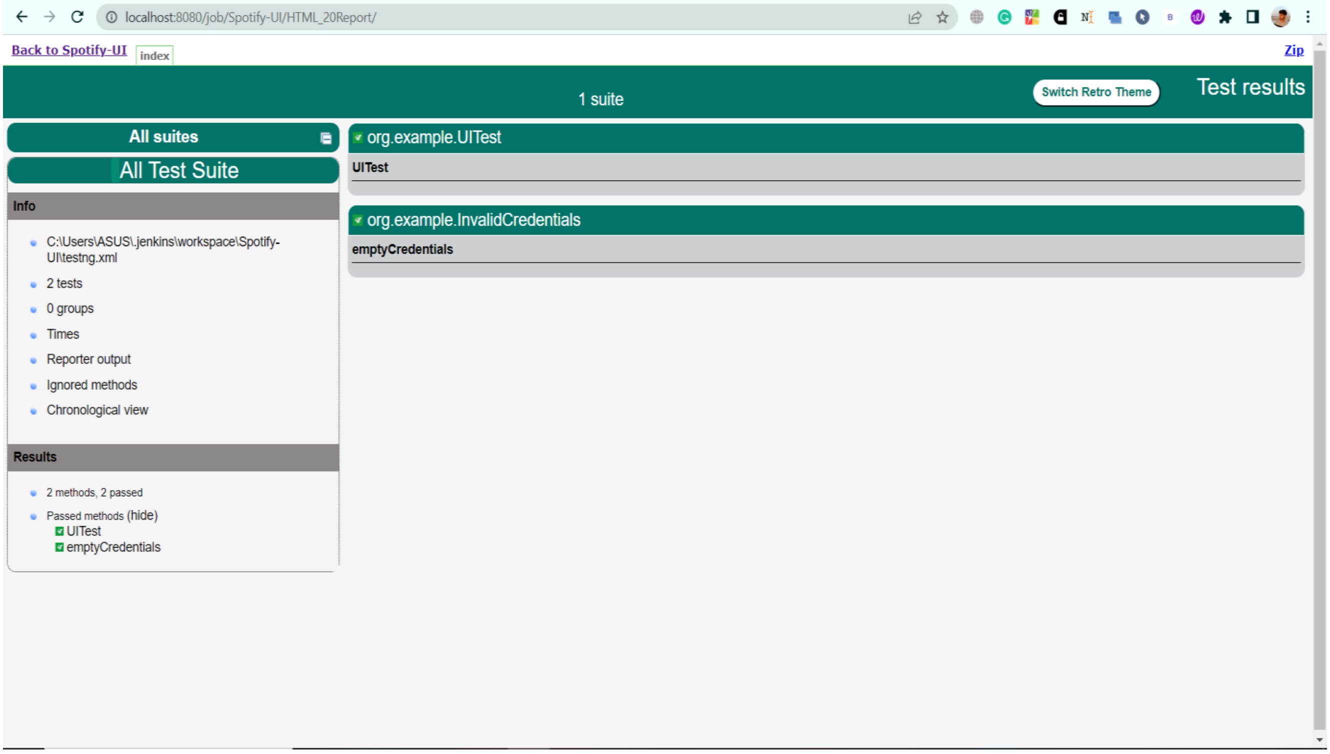
Top 10 Most Broken Tests

There are no failing tests



localhost:8080/job/Spotify-UI/test_results_analyzer/

Dashboard > Spotify-UI > Test Results Analyzer



localhost:8080/job/Spotify-UI/HTML_20Report/

Back to Spotify-UI index Zip

1 suite

Switch Retro Theme

Test results

All suites

All Test Suite

Info

- C:\Users\ASUS\.jenkins\workspace\Spotify-UI\testng.xml
- 2 tests
- 0 groups
- Times
- Reporter output
- Ignored methods
- Chronological view

Results

- 2 methods, 2 passed
- Passed methods (hide)
  ☑ UITest
  ☑ emptyCredentials

✓ org.example.UITest

UITest

✓ org.example.InvalidCredentials

emptyCredentials

2.2 Individual Tasks

1. The project focused on automated testing, utilizing various technologies such as Maven, Java, Selenium WebDriver, TestNG, and Jenkins.

2. Several tasks were completed as part of the project, including installing Maven, downloading a test project from GitHub, and changing the Junit version.

3. The main objective was to create automated tests for website functionality, such as creating new paste and using the Google Cloud Platform Pricing Calculator.

4. A framework was developed to facilitate the automation of tests, including WebDriver management for browser connectors, Page Object/Page Factory for page abstractions, Models for business objects of the required elements, and Property files with test data for at least two different environments.

5. The framework also included XML suites for smoke tests and other tests, an option for running with Jenkins and browser parameterization, test suite, environment, and screenshot capture in case of test failure.

6. Jenkins was used to set up continuous integration, which involved creating a task to clone the project and launch tests from the project in the Java directory using the mvn test goal. Build triggers were set up to perform the task every 5 minutes.

7. The overall goal of the project was to demonstrate proficiency in automated testing using various technologies and tools commonly used in industry settings.

2.2.1 Technologies Used

For my project report, I used several technologies to perform UI testing for the Spotify application. These technologies include Maven, Java, Selenium Web--Driver, Jenkins, TestNG, and the Page Object Pattern.

1. Maven: Maven is a build automation tool that helps in managing project dependencies, build processes, and deployment. It simplifies the development process by automating the building of projects and managing libraries and dependencies. I used Maven to build and manage the project dependencies of my UI testing framework.

2. Java: Java is a popular programming language used for developing enterprise applications, web applications, and mobile applications. I used Java to write the test scripts for my UI testing framework. Java is a robust language with an extensive library of APIs and tools, making it an excellent choice for UI testing.

3. Selenium WebDriver: Selenium WebDriver is a popular open-source tool used for automating web browsers. It supports different browsers, including Chrome, Firefox, and Safari, and provides a set of APIs for interacting with web elements. I used Selenium WebDriver to automate my UI testing for the Spotify application.

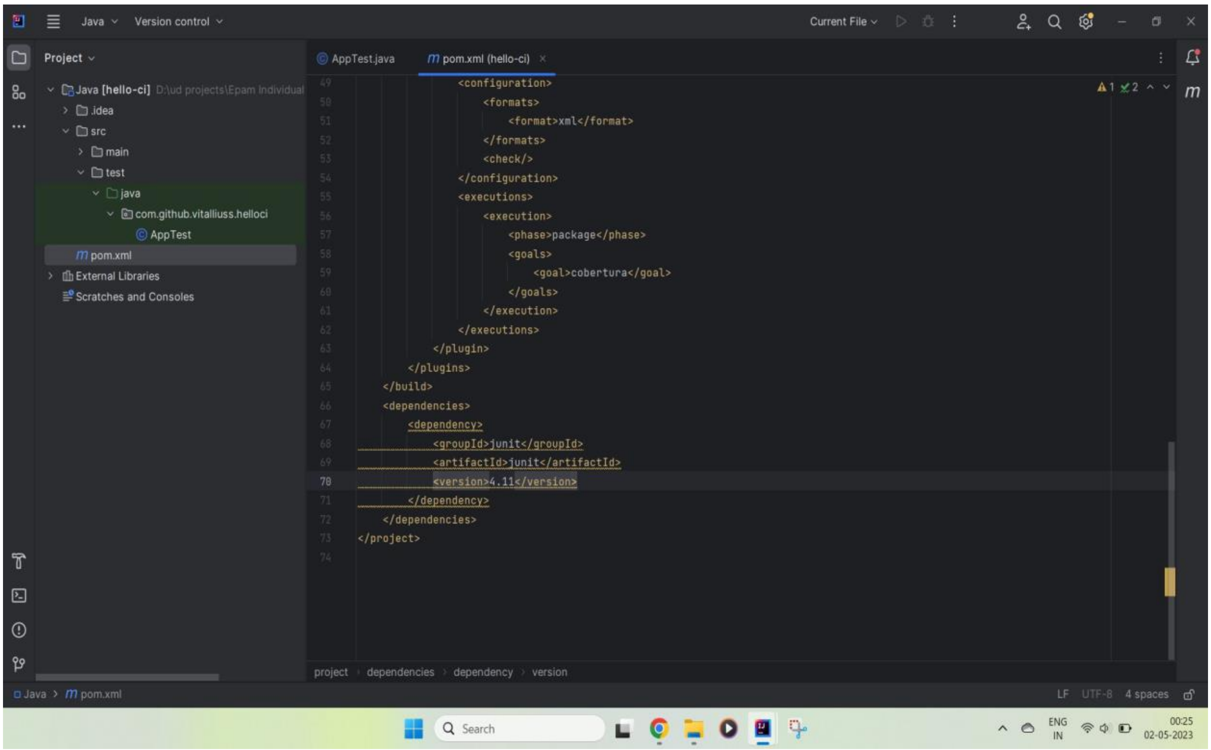
4. Jenkins: Jenkins is an open-source automation server that helps in building, testing, and deploying software applications. It provides a wide range of plugins that enable continuous integration and continuous deployment. I used Jenkins to set up and run automated tests as part of the CI/CD pipeline.

5. TestNG: TestNG is a testing framework that provides a wide range of features for automating unit tests, integration tests, and functional tests. It supports different test types, including data-driven tests, parameterized tests, and dependency tests. I used TestNG to create and execute test cases for my UI testing framework.

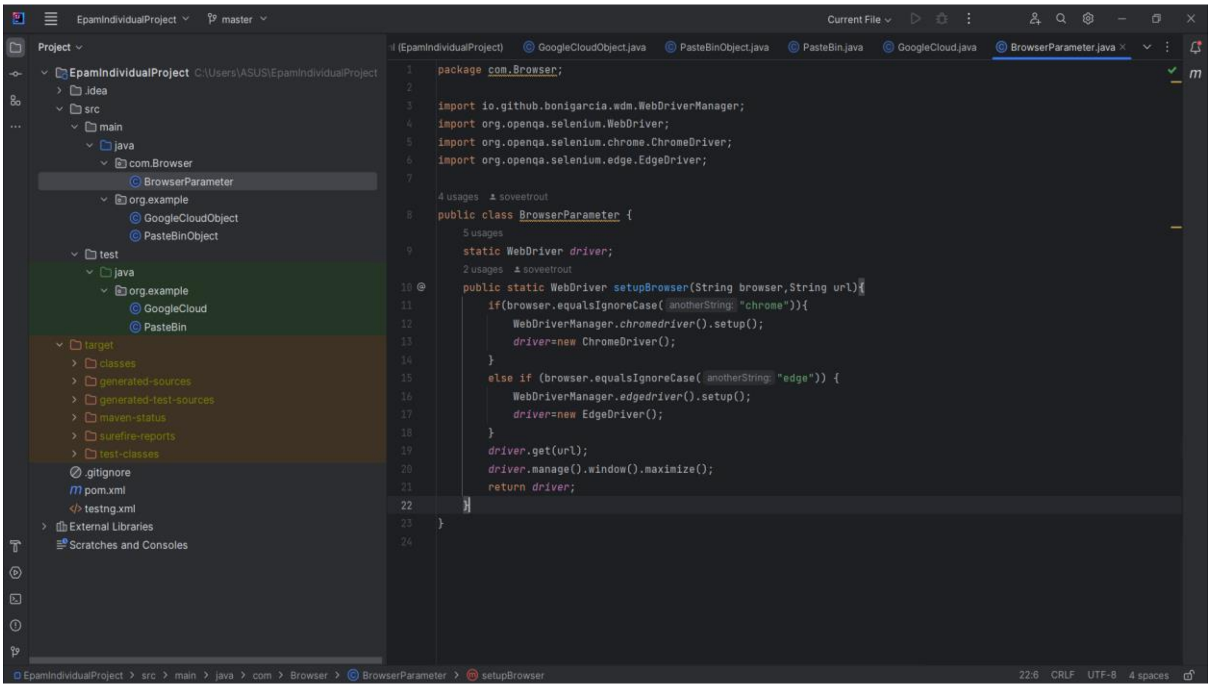
6. Page Object Pattern: The Page Object Pattern is a design pattern used for implementing UI testing frameworks. It helps in creating a modular and maintainable code structure by separating the page objects from the test scripts. I used the Page Object Pattern to structure my UI testing framework, making it easier to maintain and update.

In summary, I used Maven, Java, Selenium WebDriver, Jenkins, TestNG, and the Page Object Pattern to create a robust and reliable UI testing framework for the Spotify application. These technologies helped me automate the testing process, reducing the time and effort required to test the application manually.

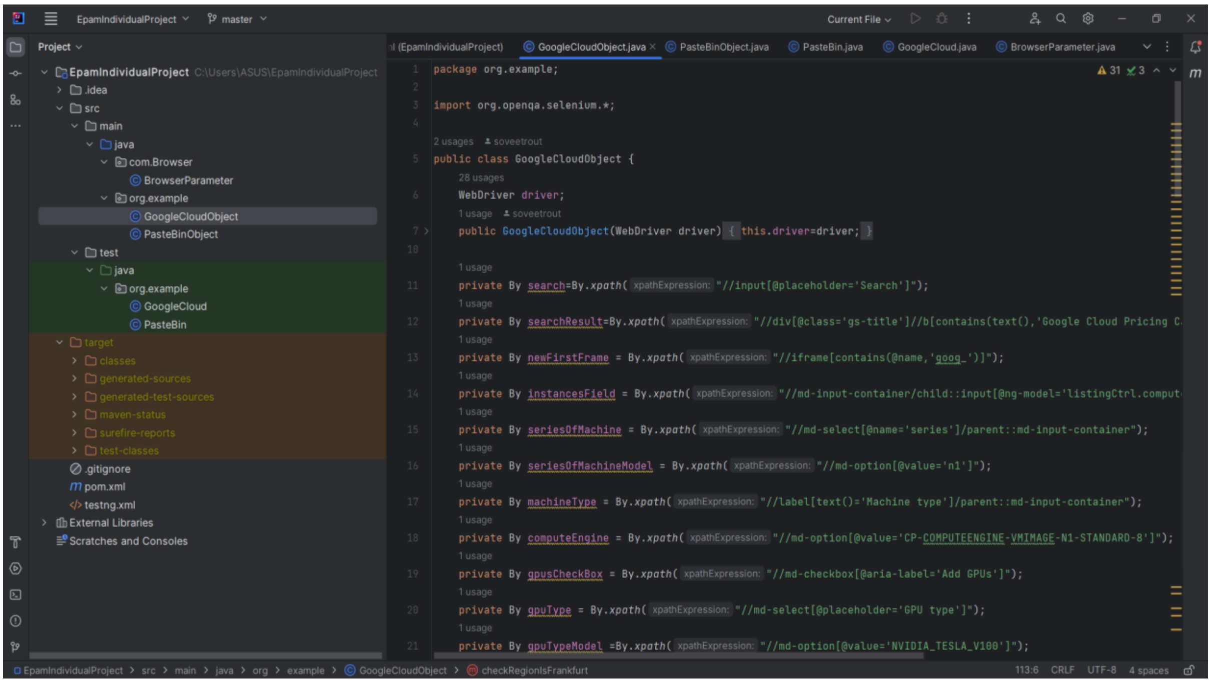
2.2.2 Snapshots



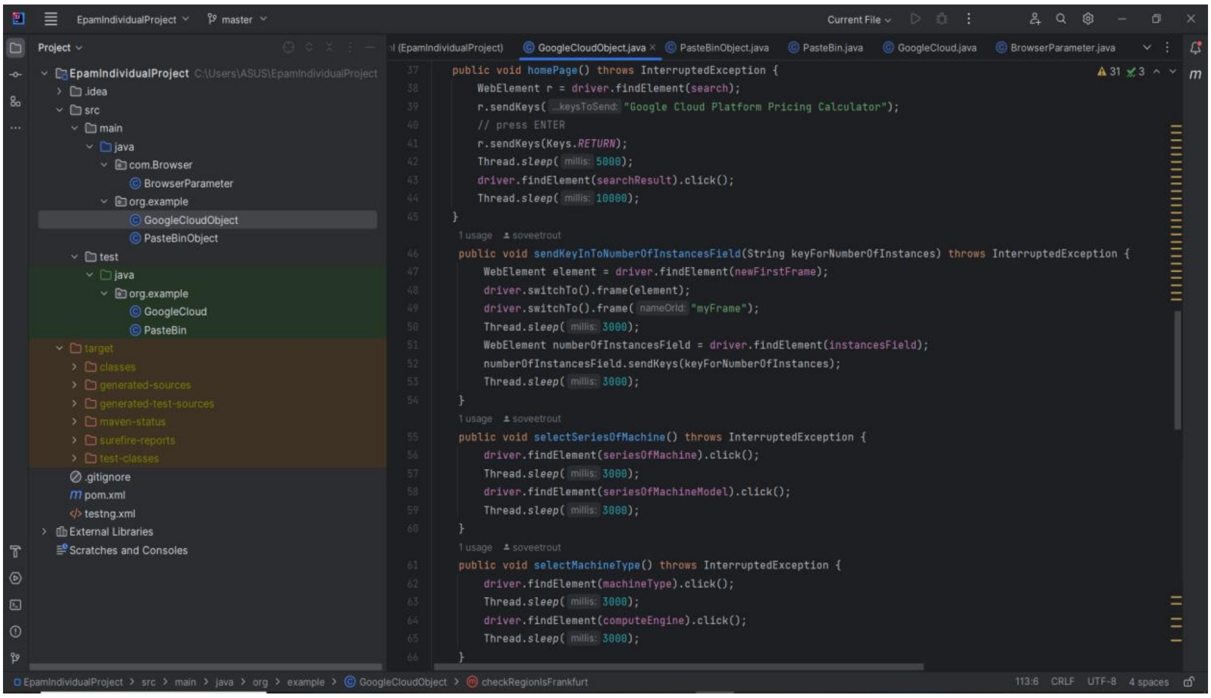
```
                <configuration>
                    <formats>
                        <format>xml</format>
                    </formats>
                    <check/>
                </configuration>
                <executions>
                    <execution>
                        <phase>package</phase>
                        <goals>
                            <goal>cobertura</goal>
                        </goals>
                    </execution>
                </executions>
            </plugin>
        </plugins>
    </build>
    <dependencies>
        <dependency>
            <groupId>junit</groupId>
            <artifactId>junit</artifactId>
            <version>4.11</version>
        </dependency>
    </dependencies>
</project>
```



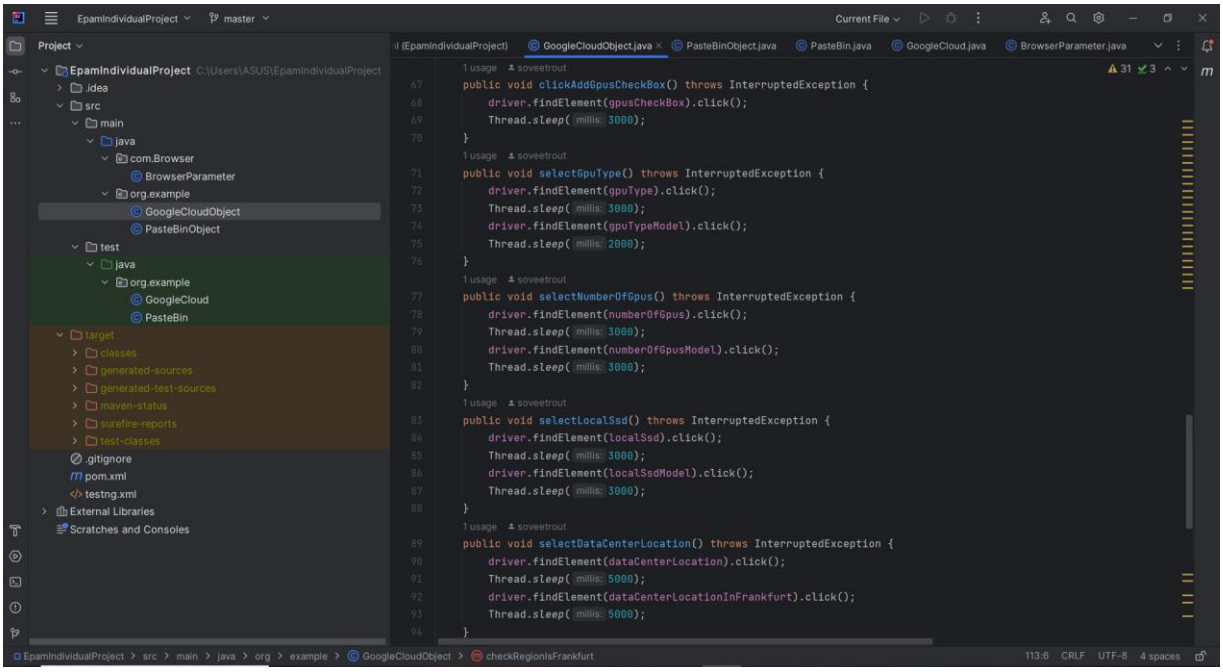
```java
package com.Browser;

import io.github.bonigarcia.wdm.WebDriverManager;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.edge.EdgeDriver;

public class BrowserParameter {
    static WebDriver driver;
    public static WebDriver setupBrowser(String browser,String url){
        if(browser.equalsIgnoreCase("chrome")){
            WebDriverManager.chromedriver().setup();
            driver=new ChromeDriver();
        }
        else if (browser.equalsIgnoreCase("edge")) {
            WebDriverManager.edgedriver().setup();
            driver=new EdgeDriver();
        }
        driver.get(url);
        driver.manage().window().maximize();
        return driver;
    }
}
```



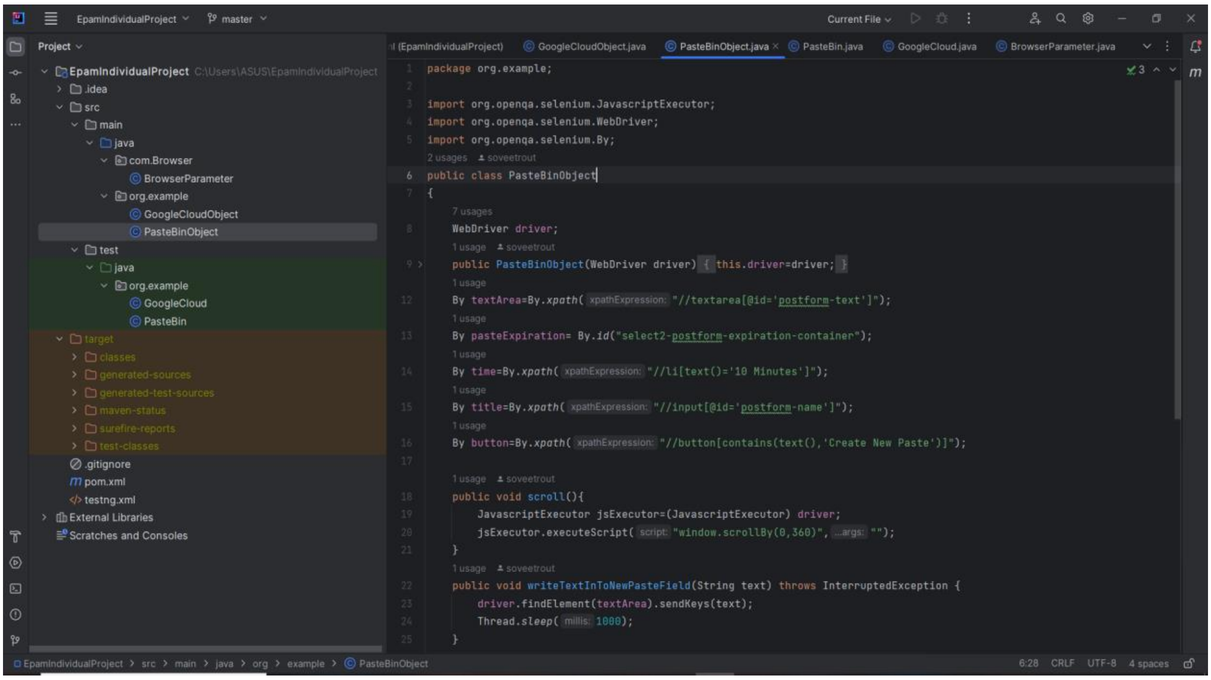
```java
package org.example;

import org.openqa.selenium.*;

public class GoogleCloudObject {
    WebDriver driver;
    public GoogleCloudObject(WebDriver driver) { this.driver=driver; }

    private By search=By.xpath("//input[@placeholder='Search']");
    private By searchResult=By.xpath("//div[@class='gs-title']//b[contains(text(),'Google Cloud Pricing C
    private By newFirstFrame = By.xpath("//iframe[contains(@name,'goog_')]");
    private By instancesField = By.xpath("//md-input-container/child::input[@ng-model='listingCtrl.comput
    private By seriesOfMachine = By.xpath("//md-select[@name='series']/parent::md-input-container");
    private By seriesOfMachineModel = By.xpath("//md-option[@value='n1']");
    private By machineType = By.xpath("//label[text()='Machine type']/parent::md-input-container");
    private By computeEngine = By.xpath("//md-option[@value='CP-COMPUTEENGINE-VMIMAGE-N1-STANDARD-8']");
    private By gpusCheckBox = By.xpath("//md-checkbox[@aria-label='Add GPUs']");
    private By gpuType = By.xpath("//md-select[@placeholder='GPU type']");
    private By gpuTypeModel =By.xpath("//md-option[@value='NVIDIA_TESLA_V100']");
```



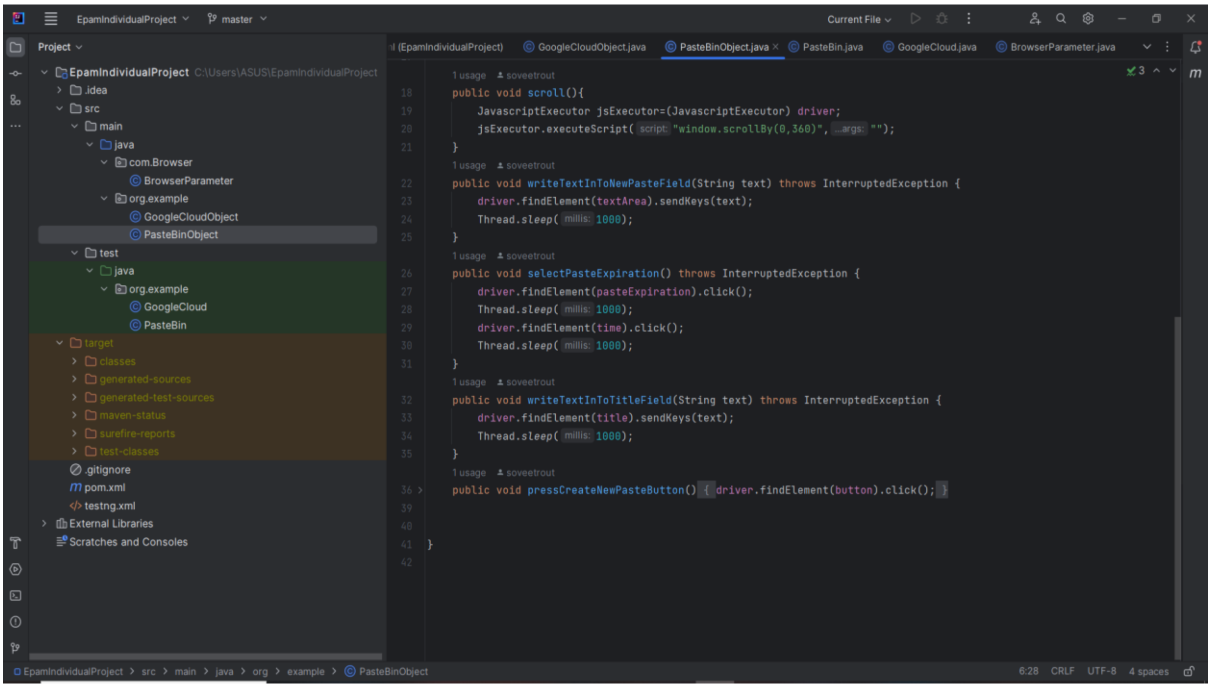
```java
public void homePage() throws InterruptedException {
    WebElement r = driver.findElement(search);
    r.sendKeys( ...keysToSend: "Google Cloud Platform Pricing Calculator");
    // press ENTER
    r.sendKeys(Keys.RETURN);
    Thread.sleep( millis: 5000);
    driver.findElement(searchResult).click();
    Thread.sleep( millis: 10000);
}
 1 usage  soveetrout
 public void sendKeyInToNumberOfInstancesField(String keyForNumberOfInstances) throws InterruptedException {
     WebElement element = driver.findElement(newFirstFrame);
     driver.switchTo().frame(element);
     driver.switchTo().frame( nameOrId: "myFrame");
     Thread.sleep( millis: 3000);
     WebElement numberOfInstancesField = driver.findElement(instancesField);
     numberOfInstancesField.sendKeys(keyForNumberOfInstances);
     Thread.sleep( millis: 3000);
 }
 1 usage  soveetrout
 public void selectSeriesOfMachine() throws InterruptedException {
     driver.findElement(seriesOfMachine).click();
     Thread.sleep( millis: 3000);
     driver.findElement(seriesOfMachineModel).click();
     Thread.sleep( millis: 3000);
 }
 1 usage  soveetrout
 public void selectMachineType() throws InterruptedException {
     driver.findElement(machineType).click();
     Thread.sleep( millis: 3000);
     driver.findElement(computeEngine).click();
     Thread.sleep( millis: 3000);
 }
```



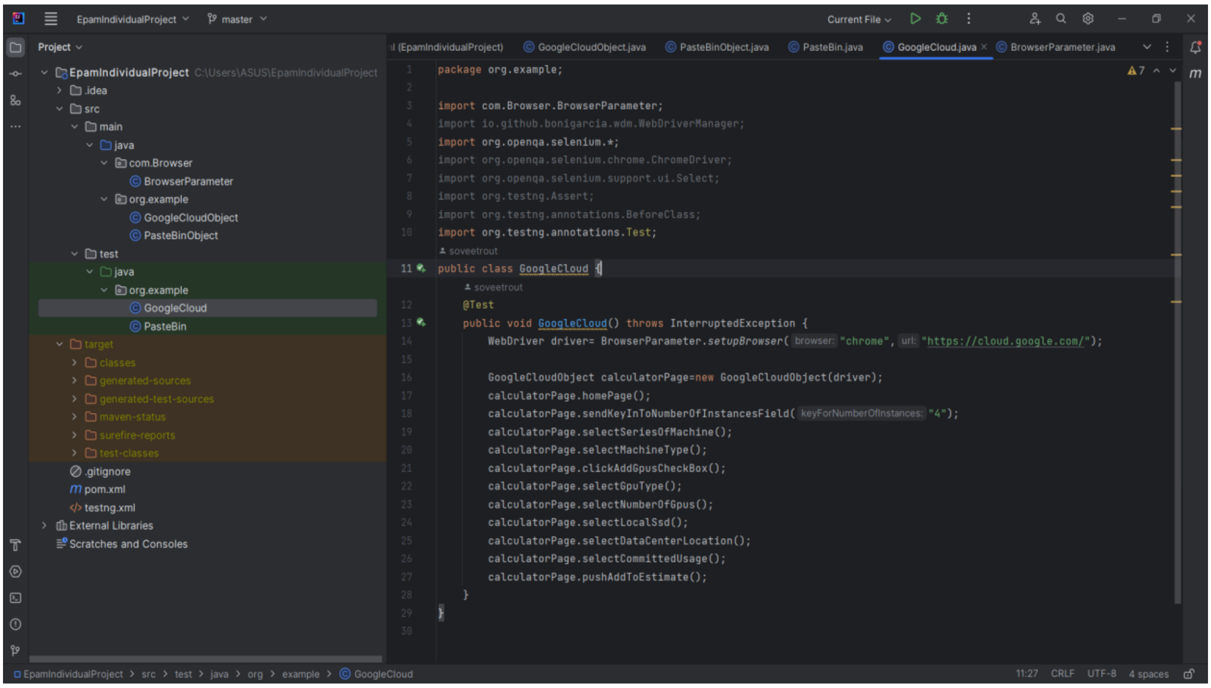
```java
 1 usage  soveetrout
 public void clickAddGpusCheckBox() throws InterruptedException {
     driver.findElement(gpusCheckBox).click();
     Thread.sleep( millis: 3000);
 }
 1 usage  soveetrout
 public void selectGpuType() throws InterruptedException {
     driver.findElement(gpuType).click();
     Thread.sleep( millis: 3000);
     driver.findElement(gpuTypeModel).click();
     Thread.sleep( millis: 2000);
 }
 1 usage  soveetrout
 public void selectNumberOfGpus() throws InterruptedException {
     driver.findElement(numberOfGpus).click();
     Thread.sleep( millis: 3000);
     driver.findElement(numberOfGpusModel).click();
     Thread.sleep( millis: 3000);
 }
 1 usage  soveetrout
 public void selectLocalSsd() throws InterruptedException {
     driver.findElement(localSsd).click();
     Thread.sleep( millis: 3000);
     driver.findElement(localSsdModel).click();
     Thread.sleep( millis: 3000);
 }
 1 usage  soveetrout
 public void selectDataCenterLocation() throws InterruptedException {
     driver.findElement(dataCenterLocation).click();
     Thread.sleep( millis: 5000);
     driver.findElement(dataCenterLocationInFrankfurt).click();
     Thread.sleep( millis: 5000);
 }
```



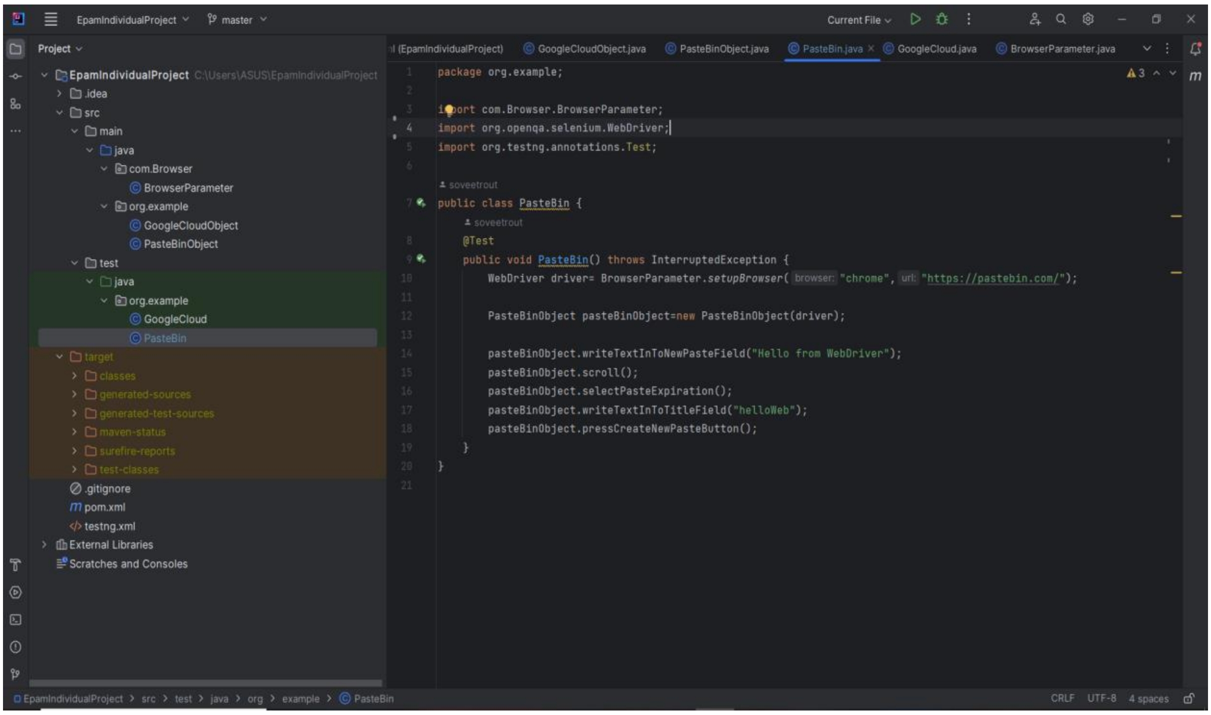
```java
package org.example;

import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.By;
public class PasteBinObject
{
    WebDriver driver;
    public PasteBinObject(WebDriver driver) { this.driver=driver; }
    By textArea=By.xpath("//textarea[@id='postform-text']");
    By pasteExpiration= By.id("select2-postform-expiration-container");
    By time=By.xpath("//li[text()='10 Minutes']");
    By title=By.xpath("//input[@id='postform-name']");
    By button=By.xpath("//button[contains(text(),'Create New Paste')]");

    public void scroll(){
        JavascriptExecutor jsExecutor=(JavascriptExecutor) driver;
        jsExecutor.executeScript("window.scrollBy(0,360)", "");
    }
    public void writeTextInToNewPasteField(String text) throws InterruptedException {
        driver.findElement(textArea).sendKeys(text);
        Thread.sleep(1000);
    }
```



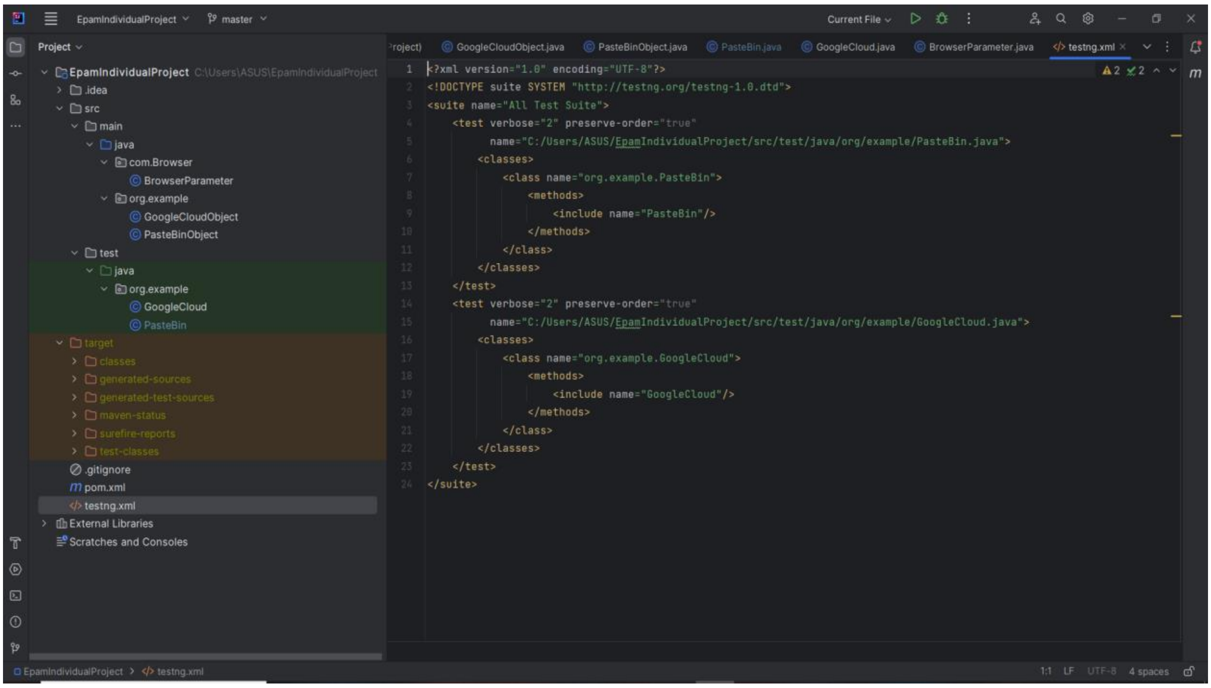
```java
    public void scroll(){
        JavascriptExecutor jsExecutor=(JavascriptExecutor) driver;
        jsExecutor.executeScript("window.scrollBy(0,360)", "");
    }
    public void writeTextInToNewPasteField(String text) throws InterruptedException {
        driver.findElement(textArea).sendKeys(text);
        Thread.sleep(1000);
    }
    public void selectPasteExpiration() throws InterruptedException {
        driver.findElement(pasteExpiration).click();
        Thread.sleep(1000);
        driver.findElement(time).click();
        Thread.sleep(1000);
    }
    public void writeTextInToTitleField(String text) throws InterruptedException {
        driver.findElement(title).sendKeys(text);
        Thread.sleep(1000);
    }
    public void pressCreateNewPasteButton() { driver.findElement(button).click(); }


}
```



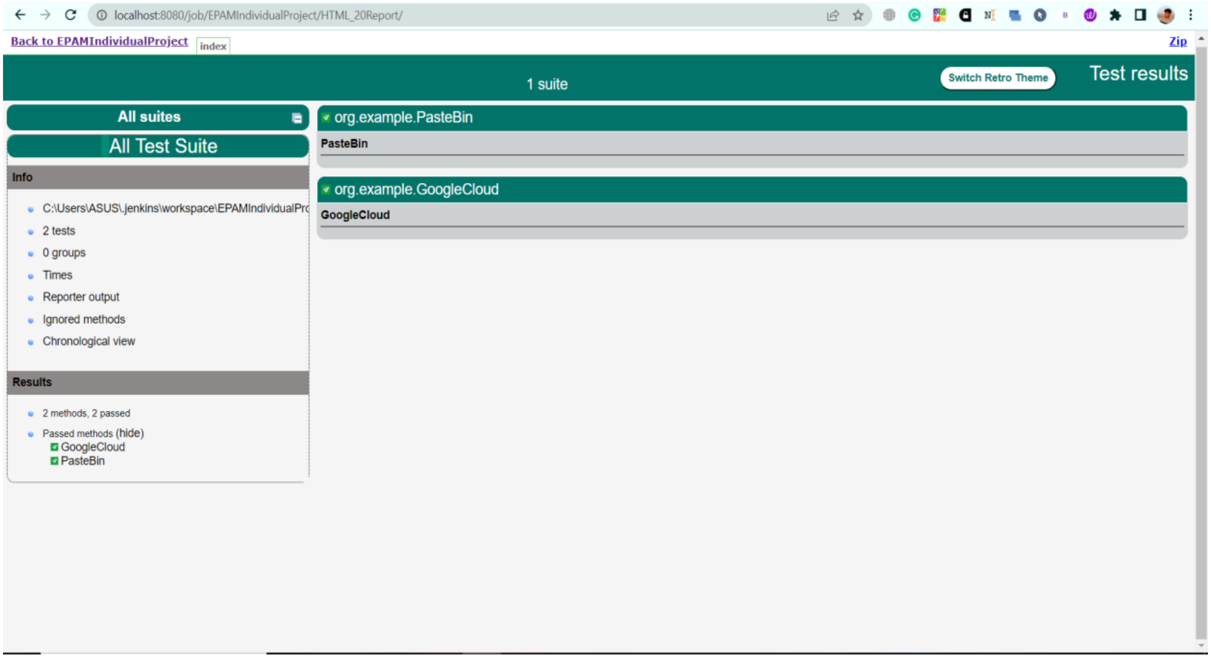
```java
package org.example;

import com.Browser.BrowserParameter;
import io.github.bonigarcia.wdm.WebDriverManager;
import org.openqa.selenium.*;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.support.ui.Select;
import org.testng.Assert;
import org.testng.annotations.BeforeClass;
import org.testng.annotations.Test;
public class GoogleCloud {
    @Test
    public void GoogleCloud() throws InterruptedException {
        WebDriver driver= BrowserParameter.setupBrowser( browser: "chrome", url: "https://cloud.google.com/");

        GoogleCloudObject calculatorPage=new GoogleCloudObject(driver);
        calculatorPage.homePage();
        calculatorPage.sendKeyInToNumberOfInstancesField( keyForNumberOfInstances: "4");
        calculatorPage.selectSeriesOfMachine();
        calculatorPage.selectMachineType();
        calculatorPage.clickAddGpusCheckBox();
        calculatorPage.selectGpuType();
        calculatorPage.selectNumberOfGpus();
        calculatorPage.selectLocalSsd();
        calculatorPage.selectDataCenterLocation();
        calculatorPage.selectCommittedUsage();
        calculatorPage.pushAddToEstimate();
    }
}
```



```java
package org.example;

import com.Browser.BrowserParameter;
import org.openqa.selenium.WebDriver;
import org.testng.annotations.Test;

public class PasteBin {
    @Test
    public void PasteBin() throws InterruptedException {
        WebDriver driver= BrowserParameter.setupBrowser( browser: "chrome", url: "https://pastebin.com/");

        PasteBinObject pasteBinObject=new PasteBinObject(driver);

        pasteBinObject.writeTextInToNewPasteField("Hello from WebDriver");
        pasteBinObject.scroll();
        pasteBinObject.selectPasteExpiration();
        pasteBinObject.writeTextInToTitleField("helloWeb");
        pasteBinObject.pressCreateNewPasteButton();
    }
}
```



```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">
<suite name="All Test Suite">
    <test verbose="2" preserve-order="true"
          name="C:/Users/ASUS/EpamIndividualProject/src/test/java/org/example/PasteBin.java">
        <classes>
            <class name="org.example.PasteBin">
                <methods>
                    <include name="PasteBin"/>
                </methods>
            </class>
        </classes>
    </test>
    <test verbose="2" preserve-order="true"
          name="C:/Users/ASUS/EpamIndividualProject/src/test/java/org/example/GoogleCloud.java">
        <classes>
            <class name="org.example.GoogleCloud">
                <methods>
                    <include name="GoogleCloud"/>
                </methods>
            </class>
        </classes>
    </test>
</suite>
```


localhost:8080/job/EPAMIndividualProject/HTML_20Report/
Back to EPAMIndividualProject
index
Zip
Switch Retro Theme

1 suite

Test results

All suites

All Test Suite

Info

- C:\Users\ASUS\.jenkins\workspace\EPAMIndividualPrc
- 2 tests
- 0 groups
- Times
- Reporter output
- Ignored methods
- Chronological view

Results

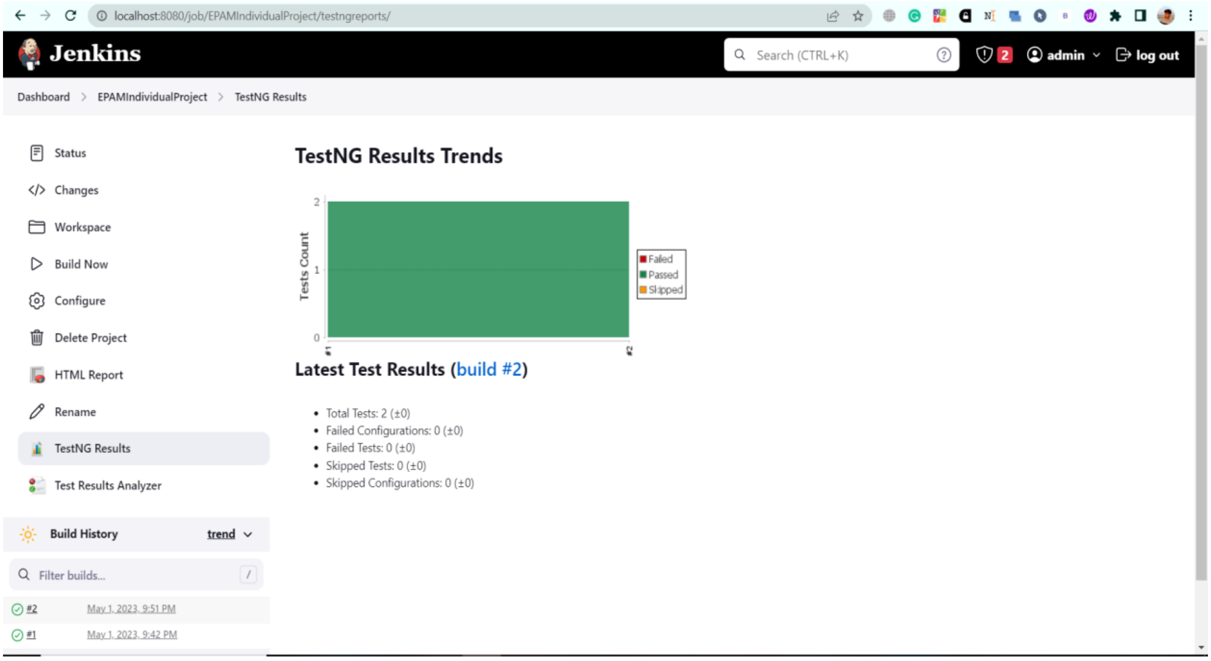
- 2 methods, 2 passed
- Passed methods (hide)
  ☑ GoogleCloud
  ☑ PasteBin

☑ org.example.PasteBin

PasteBin

☑ org.example.GoogleCloud

GoogleCloud



TestNG Results Trends

Latest Test Results (build #2)

- Total Tests: 2 (±0)
- Failed Configurations: 0 (±0)
- Failed Tests: 0 (±0)
- Skipped Tests: 0 (±0)
- Skipped Configurations: 0 (±0)

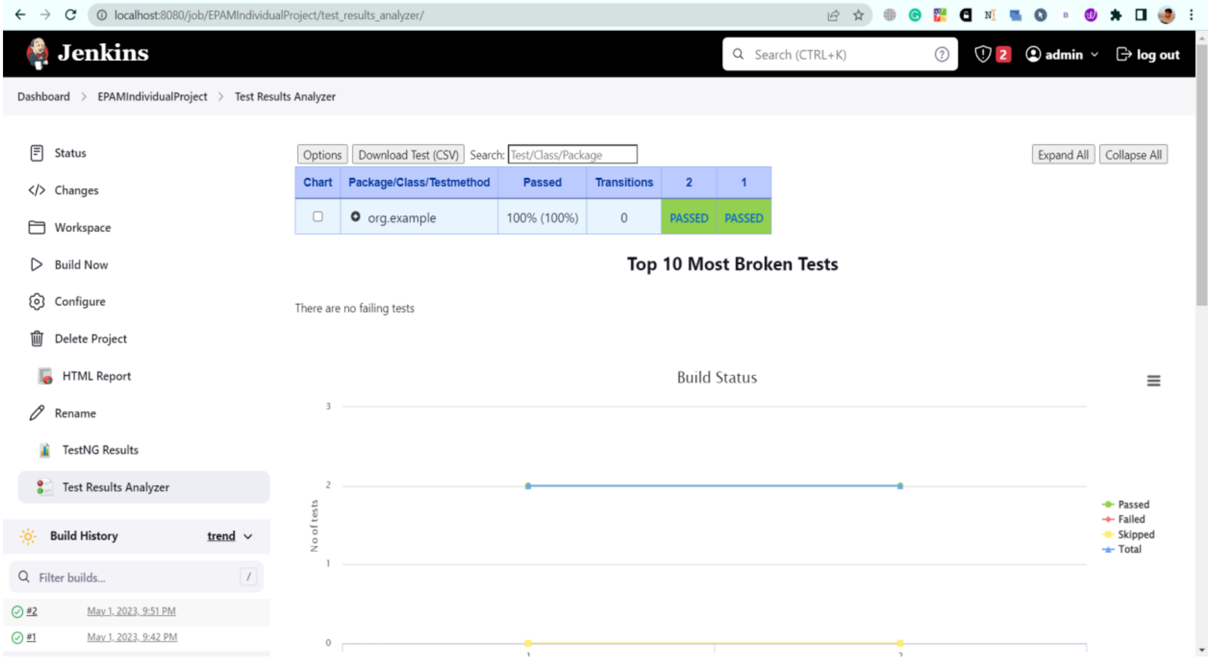
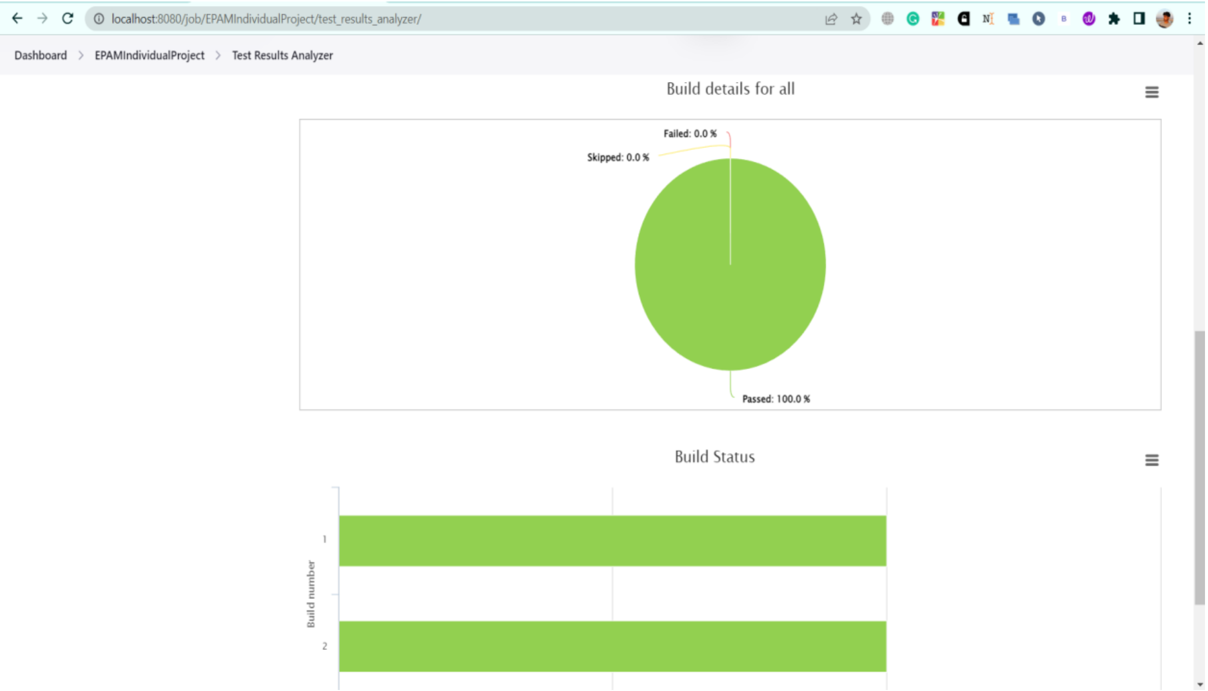


Chart	Package/Class/Testmethod	Passed	Transitions	2	1
□	org.example	100% (100%)	0	PASSED	PASSED

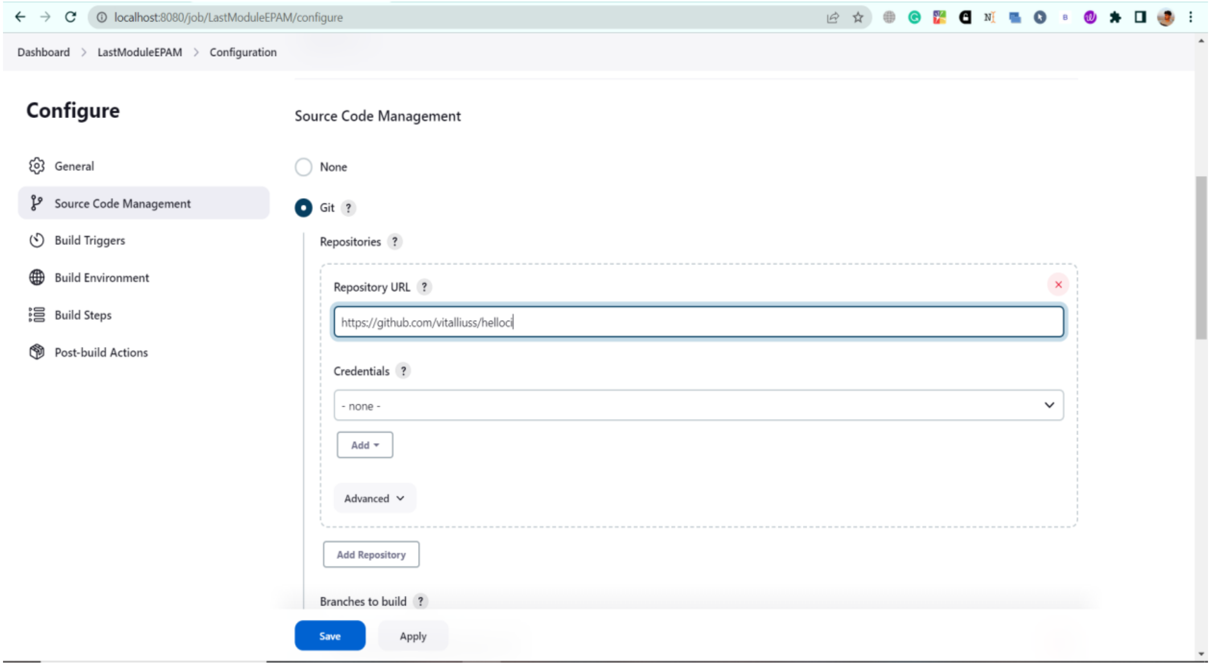
Top 10 Most Broken Tests

There are no failing tests



localhost:8080/job/EPAMIndividualProject/test_results_analyzer/

Dashboard > EPAMIndividualProject > Test Results Analyzer



localhost:8080/job/LastModuleEPAM/configure

Dashboard > LastModuleEPAM > Configuration

Configure

Source Code Management

○ None

● Git ?

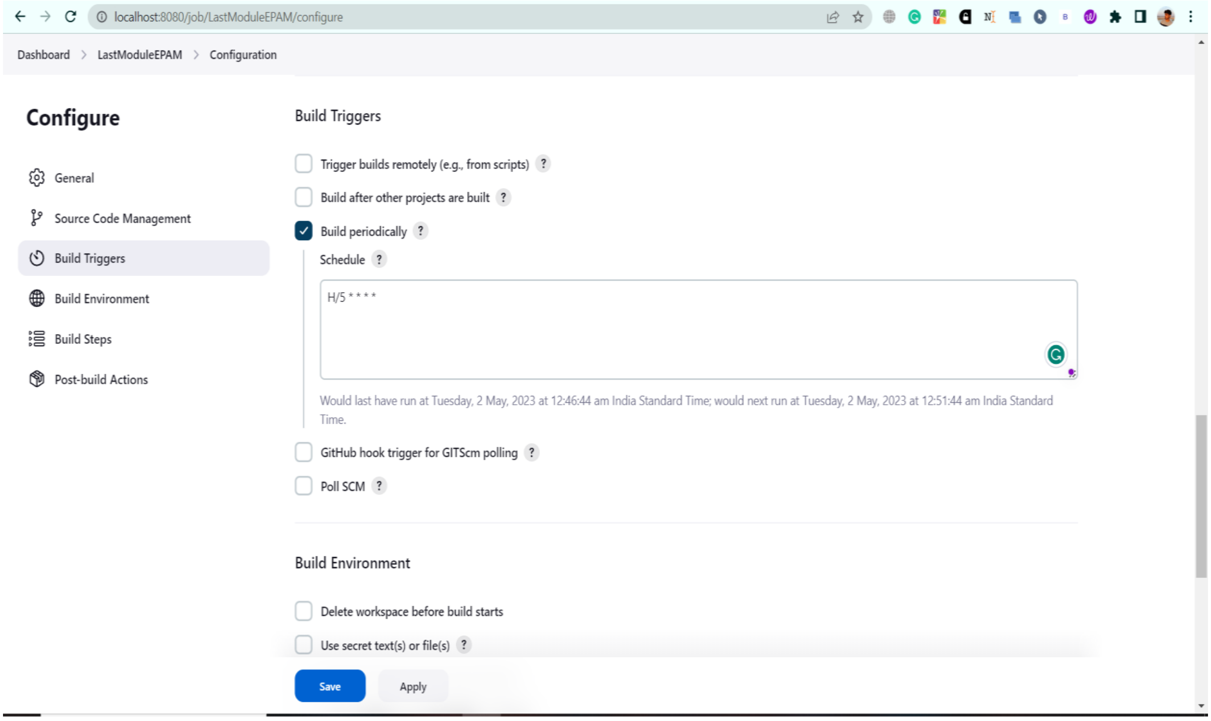
Repositories ?

Repository URL ?

https://github.com/vitalliuss/helloci

Credentials ?

- none -



localhost:8080/job/LastModuleEPAM/configure

Dashboard > LastModuleEPAM > Configuration

Configure

Build Triggers

□ Trigger builds remotely (e.g., from scripts) ?

□ Build after other projects are built ?

☑ Build periodically ?

Schedule ?

```
H/5 * * * *
```

Would last have run at Tuesday, 2 May, 2023 at 12:46:44 am India Standard Time; would next run at Tuesday, 2 May, 2023 at 12:51:44 am India Standard Time.

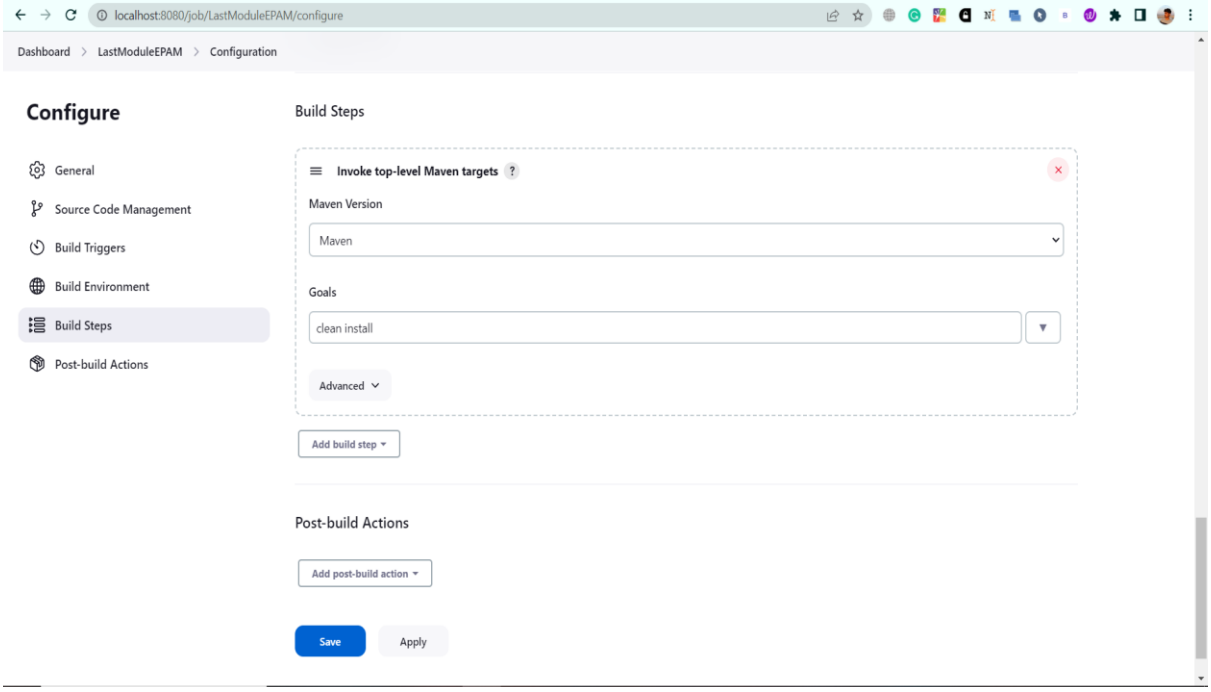
□ GitHub hook trigger for GITScm polling ?

□ Poll SCM ?

Build Environment

□ Delete workspace before build starts

□ Use secret text(s) or file(s) ?



localhost:8080/job/LastModuleEPAM/configure

Dashboard > LastModuleEPAM > Configuration

Configure

Build Steps

Invoke top-level Maven targets ?

Maven Version: Maven

Goals: clean install

Post-build Actions

CHAPTER 3
CONCLUSION

The completion of the Maven practical task, as well as the subsequent tasks, showcases a strong grasp of the Maven build tool and the ability to use it efficiently. The task involved downloading a test project and collecting it with Maven, changing the junit version, and ensuring that the new library version was added to the repository. These tasks demonstrate proficiency in managing dependencies and building projects with Maven.

The Webdriver module's practical tasks demonstrate a solid understanding of Selenium WebDriver, framework unit tests, and Page Object concepts. The "I can win" task involved automating the creation of a new paste on a service like Pastebin with specific attributes, while the "Hurt Me Plenty" task automated the use of the Google Cloud Platform Pricing Calculator with specific parameters. Both tasks demonstrate a solid understanding of the Page Object model and the ability to create efficient, maintainable code.

The successful completion of the Framework practical task involves the development of a robust automation framework for the "Hurt Me Plenty" task. The framework includes a WebDriver manager for managing browser connections, page abstractions using Page Object/Page Factory, models for business objects of the required elements, and property files with test data for at least two different environments. The framework also includes XML suites for smoke tests and other tests, screenshot capture when a test fails, and the ability to run with Jenkins, parameterize browsers, test suites, and environments.

The Spotify UI test involved testing various functionalities of the Spotify web application, including login, playlist creation, adding songs, playing and pausing music, and deleting songs and playlists. The test demonstrates proficiency in using Selenium WebDriver, TestNG, and Page Object pattern to automate various tasks, ensuring that the application functions as expected. The successful completion of the test shows that the test cases were comprehensive, and the application meets the necessary requirements.

Overall, the completion of these tasks and the successful execution of the Spotify UI test demonstrate a strong understanding of various technologies such as Maven, Selenium WebDriver, TestNG, Jenkins, and Page Object pattern. These skills are essential for developing and maintaining robust automation frameworks, ensuring that web applications meet the necessary requirements and function as expected.