**MAIN PROJECT DOCUMENTATION**

**PROJECT TEAM NAME: “Spidey Testers”**

**Team no: 01**

HOLIDAY TRACKER

|  |  |
| --- | --- |
| **TEAM MEMBERS** | |
| **NAME** | **EMPLOYEE ID** |
| Ajay D | 2082237 |
| Dhanush K | 2080522 |
| Nithiyashree S | 2080462 |
| Pradeep S | 2082215 |
| Renuga K | 2080805 |
| Soundarya S | 2081256 |

**1.WEBSITE:**

<https://be.cognizant.com>

Graphical user interface, text

Description automatically generated

**2.TOOLS AND TECHNOLOGY USED:**

***1.Selenium with Java in Eclipse***: Selenium is an open-source tool that automates web browsers. Eclipse is the most used IDE by the Java developers. And it is used for any programming language for which a plugin is available. Selenium provides a playback tool for authoring functional tests developed using java, the Eclipse platform can be used to develop rich client applications, integrated development environments and other tools**.**

***2.TestNG (Test Next Generation***): TestNG framework is an open-source automated testing framework and here NG means Next Generation. TestNG is inspired from Junit and NUnit, but it has a new functionality that makes this framework more powerful and easier. TestNG provides parallel execution of test methods. It has support for parameterizing test cases using **@** Parameters annotations.

***3.Maven:*** Maven is a build automation tool used primarily for Java projects. It is used to check the compilation issues between framework components whenever multiple test engineer integrates their files into the same framework. It provides ‘build success’ message if no compilation issues in the framework or else provide ‘build failure’ message. Maven has new features like dependency, which is used to download the dependency jar from the internet before the test execution.

***4.Page Factory with POM:*** Pagefactory is a class provided by selenium webdriver to support page object design patterns. The initElements is a static method of Pagefactory class which is used to initialize all the webelements located by **@FindBy** annotation. The advantage is that it makes your framework more structured and maintainable.

***5.POI:*** Apache POI, where POI stands for **(Poor Obfuscation Implementation)** is an API that offers a collection of Java libraries that helps us to read, write, and manipulate different Microsoft files such as excel sheets, power-point, and word files. Apache POI uses certain terms to work with Microsoft Excel.

***6.EXTENT REPORT:*** Extent Reports is an open-source library used for creating visually attractive reports during test automation. It produces HTML-based documents in graphs, pie charts, screenshots, custom logs, and test summaries in Java and .Net. Testers can track multiple test case runs in a single test suite with Extent Reports

**3.WORK DONE:**

**1**.The code is written in JAVA and is automated in different browsers with the help of Selenium. The project and dependencies are managed with the help of Maven.

**2.**POM is used to create classes for each page therefore reducing usage of same code for various objects.

**3.**Input data is obtained from an excel file with the help of Apache POI.

**4**.Extent Reports are generated by testing the requirements.

**5**.Screenshots are taken where it is necessary.

**4.PROJECT STRUCTURE:**

**Text

Description automatically generatedGraphical user interface, text, application

Description automatically generated**

**5. STEPS TO EXECUTE:**

***1.Display the User Profile/Details in Be Cognizant HomePage****.*

* Login into <https://be.cognizant.com>
* Print the User Profile/Details*.*

Graphical user interface, text

Description automatically generated

***2.Search Holiday Tracker in the Start Searching****.*

* Click “*Start Searching*” search box.
* Enter Holiday Tracker on Search Box and Click Search Button.

Graphical user interface, text

Description automatically generated

* Click on Holiday Tracker link

Graphical user interface, text, application, email

Description automatically generated

***3.Display the Holidays by clicking view all****.*

* Click “*view all*”.

Calendar

Description automatically generated

* Holidays will be displayed and to be printed

Graphical user interface, text

Description automatically generated

***4.Display the Locations of any of the Holiday****.*

* Close the Holiday tab.
* Search for Republic day on the Search Box

Calendar

Description automatically generated

* Click Enter
* Locations are displayed and to be printed and verified

Graphical user interface, application

Description automatically generated

**6.*Once Everything is Setup:***

* Go to TestNG.xml.
* Right click on anywhere on the editor>>Run As>>TestNG Suite.
* The automation will take place, the reports will be generated and screenshot is captured.

***7.Steps to view the Screenshot:***

* Right click on the screenshots folder.
* Click on Refresh button.
* Click on .png file.

**6.PREREQUISITES:**

1.Initially we need to create a maven project.

2.We have included the following maven dependencies:

* Selenium Java>>3.141.59, Apache POI>>3.17
* Apache POI Based on OPC and OOXML Schemas>>3.17

3.We created methods in the project.

4.Data driven concept is used for the pages to read data.

5.These are the basic requirements of the project.

**7.DEVELOPMENT PROGRESS:**

1.On the first day, we have implemented the different parts of the automation separately.

2. We executed the entire automation inside a single main function, in the next day.

3. On day 3, We executed the same existing code inside Maven.

4. On day 4, We implemented the same code using TestNG.

5. On day 5, We implemented POM.

6. We used Apache POI on the same day.

7. The whole project integration part was completed by day 6.

8. We completed the report part on day 7.

**8.EXTENT REPORT:**

A screenshot of a computer

Description automatically generated with medium confidence