



Automated HR Management System Using Selenium WebDriver and Python

¹Shivani Bhavsar, ¹Preeti Mandge, ¹Srushti Patel, ¹Mrunal Nikumbe, ²Akash Koli

¹Student, Department of Computer Engineering D. N. Patel College of Engineering, Shahada, India

²Asst. Professor, Department of Computer Engineering D. N. Patel College of Engineering, Shahada, India

Abstract: The process of automatically performing operations on a web browser is called Web Browser Automation. In this study Human Resource Database Management System is automated using Selenium WebDriver. WebDriver is an API which has functions to control Browser. One can use the functions in writing a script that controls the elements of a browser in the way they want. Python language is selected for writing the scripts. Django and MySQL database is selected for the development of Human Resource Database Management System. In corporate offices the work an employee has to perform on his desk that involves web base applications which is repetitive, tedious and time-consuming process. This work is being reduced using automation tool Selenium WebDriver.

Index Terms - Web Automation, Selenium, WebDriver

I. INTRODUCTION

Nowadays, corporate offices and several traditional front and back offices are shifting from desktop applications to web based applications. Considering all the work an employee has to perform is repetitive. This work can be reduced by using automation tool, selenium web driver. Selenium web driver is a component of open-source tool Selenium. In this paper we have automated Human Resource Database Management System. The API used in python helps you in connecting to the browser through the Selenium. The binding of python and Selenium offers a simple API to write functional tests by utilizing Selenium WebDriver in a perceptible way.

Several tools can drive the web browser the way a real user would do like navigating to different pages, interacting with the elements of the page and capturing some data. This process is called *Web Browser Automation*. [1]
Some of the common use cases of web browser automation could be:

- Automating the manual tests on a web application
- Automating the repetitive tasks like scrapping information from websites
- Filling up the HTML forms, doing some administrative jobs, etc. [1]

1.1 Problem Definition

Managing employee database is one of tasks performed by HR in corporate offices which is repetitive and tedious task. He repeatedly has to perform CRUD operations on the database which is time consuming. But this task can be reduced using automation tools. Selenium is one of the best automation tools. The basic objective of this project is to automate the operations on the web-based application.

II. LITERATURE SURVEY

2.1 Related Work

Marcus Atish D Rozario built Human Resource Database Management System which consists of modules like Employee Management, Payroll Generation, Employee Grade Management. [2]

A.S. Syed Navaz, A.S. Syed Fiaz, C. Prabhadev, V. Sangeeth, S. Gopalakrishnan proposed a Human Resource Management System which consists of modules like Employee Details, Payroll, Training, Performance, Resignation, Resume tracking. [3]

Earlier the admin or the user who is accessing the website had to performed all these activities manually which was time consuming. Because every single time he had to input and submit the data. There was also a chance of filling wrong data. For things like this we have tool "Selenium". It is automated testing tool used to perform these activities automatically. It enters same data in the form via code and tool all the time instead of manual entry.

The purpose of automation is to perform a wide variety of tasks. Browser automation tools can automate your Web browser to perform repetitive and error-prone tasks. Automation not only saves time but it also improves the efficiency and quality of output. It also reduces the usage of resources used in traditional methods.

2.2 Project Objective

- Freeing up HR staff
- Quality output
- Increase in productivity
- Simple, efficient processes

2.3 Components of Selenium

Selenium has been in the industry for a long time and used by automation testers all around the globe. Let's check the four major components of Selenium [4]

- Selenium IDE
- Selenium RC
- Selenium Web driver
- Selenium GRID

2.4 What is Selenium WebDriver?

Selenium WebDriver is a framework that allows automation testing. It allows testing across various browsers. It can execute multiple tests over multiple browsers on multiple OS. Web Driver makes it possible to write a test script in Linux and run it in Windows. [5]

III. PROPOSED SYSTEM & METHODOLOGY

3.1 Experimental Procedure:

1. We have built a Human Resource Database Management System using Django framework, MySQL database and Visual Studio Code IDE. The application consists of features like manage Employee, Department and Project Table, send email to all employees present in your database at once or specific employee, display employee performance report, project progress and employee working hours with the help of graphs, generate employee performance report.
2. All the above operations of the application are automated using Selenium WebDriver. For that we have written the automation script in python language.
3. The data required to perform CRUD operations on tables in application is stored in the excel sheet. The data in excel sheet get fetch by automation script.
4. To start the automation, first we need to start the local server required to run the application which is in-built in Django framework. The server will be started with the help of below command in VS Code IDE.
"python manage.py project runserver"
5. After starting the server, we need to run the automation script. As we run the script, the menu will be displayed in the console. The menu contains below options:
 - Manage Employee
 - Manage Department
 - Manage Project
 - Email to all Employees
 - Report Generation
 - Statistics

The first three options contain sub options as below:

- Add
- Delete
- Edit

The automation will be performed according to the choice made.

6. After the selection of the option, the application will be opened automatically in the chrome browser and the operation will be performed accordingly.

7. To perform the operation, the WebDriver is being used. It is a programming interface which helps us to interact with the web elements like buttons, input box etc.

8. After the completion of the specified operation, the web browser will be closed automatically.

3.2 Modules:

- Manage Department Details
- Manage Project Details
- Manage Employee Details
- Send Email to all Employees
- Report Generation
- Statistics

3.3 Data Flow Diagram:

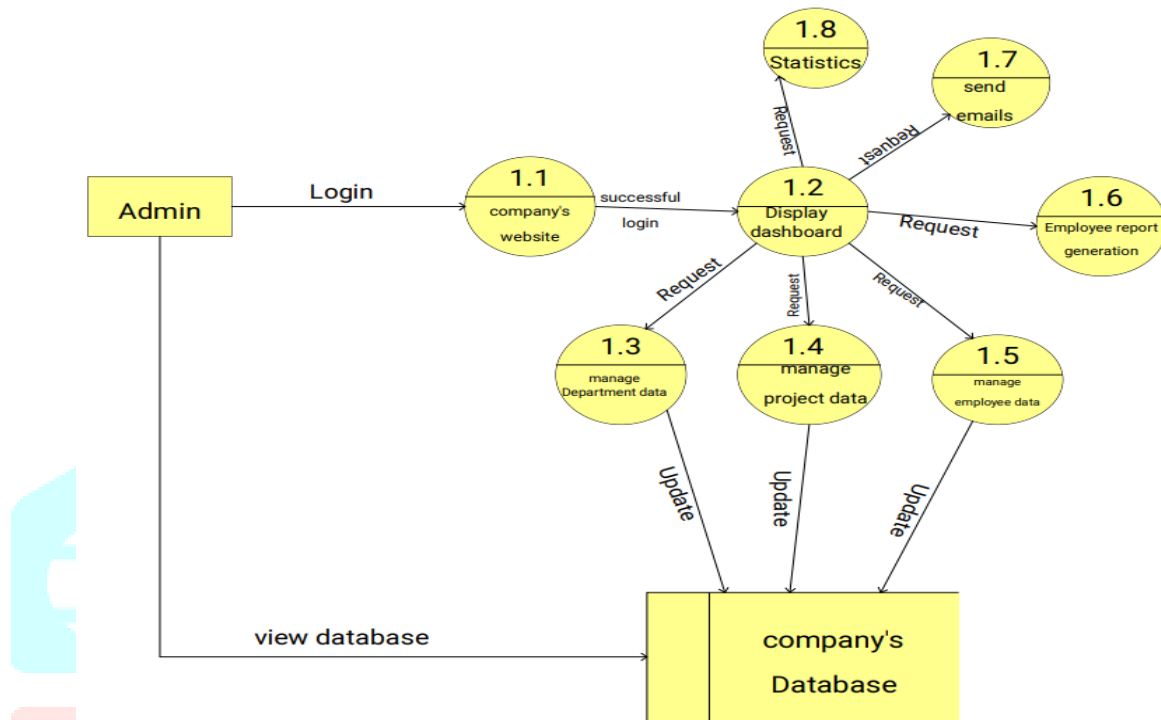


Fig. 3.3: Data Flow Diagram

IV. IMPLEMENTATION

4.1 Web Page Inspection Tools

Google Chrome provides an in-built feature to analyze pages and elements. This is very similar to Firebug. You can move the mouse over a desired element on the page and right-click to open the pop-up menu, then select Inspect element option. This will open Developer Tools in the browser, which displays information similar to that of Firebug. Chrome Developer Tools also provide a feature where you can get XPath for an element by right-clicking on the desired element in the tree and selecting the Copy XPath option from the pop-up menu. The Chrome add-in "XPath Helper" is a great tool for verifying XPath selectors found through the Copy XPath option, as it highlights the element that corresponds to the XPath selector in the browser. [6]

Table 4.1: Web Inspection Tools

Strategy	Description
By ID	Locates an element using the ID attribute
By name	Locates an element using the Name attribute
By class name	Locates an element using the (CSS style) Class attribute
By tag name	Locates an element using the HTML tag (for example “Select”, “Input”, “List”, “Button” etc.). The “Input” tag has multiple different types (such as “RADIO”, “CHECKBOX”, “TEXTBOX”, “PASSWORD”) that need to be further filtered by using the “type” attribute.
By link text	Locates link using its text
By partial link text	Locates link using its partial text
By CSS	Locates element using the CSS selector. The by CSS method is similar to the By XPath method in its usage, but the difference is that it is slightly faster
By XPath	Locates element using XPath query. However, this is the least preferable locator strategy due its slow performance, as XPath provides the ability to search elements bi-directionally (downwards in the hierarchy, as well as upwards).

4.2 Locating Elements

Locating elements in Selenium WebDriver is done by using the “findElement()” and “findElements()” methods provided by WebDriver and WebElement class. The “findElement()” method returns a WebElement object based on a specified search criteria or throws up an exception if it does not find any element matching the search criteria. The “findElements()” method returns a list of Web Elements matching the search criteria. If no elements are found, it returns an empty list. The “findElement()” and “FindElements()” methods throw up the “NoSuchElementException” exception when they fail to find the desired element using the specified locator strategy. [6]

Table 4.2: Locating Elements

Method	Result
get()	get() method in Selenium is used for opening the URL
maximize_window()	maximize_window() method is used for maximizing the browser window
find_element_by_name()	The 'Inspect Tool' in Chrome is used for locating the required element on the web page with this one of the method.
send_keys()	The send_keys() method is used for entering the sample text in the text box with id match to it
click()	Once the elements are located, the click() method is used for performing the necessary operation.
quite()	The quite() Selenium method is used for closing the browser window under focus. This will release the resources held by the WebDriver instance.

V. RESULT

Our implemented project satisfies all the expected objectives as of mentioned above. As the website is completely automated, it saves a lot of time of the user, the chances of errors get minimized to 0% and also this system maintains the data efficiently. So, user can focus on the working of the form instead of entering the data manually all the time.

VI. CONCLUSION

As per the basic objective of this paper, we have automated employee database operations and various other operations like report generation, email sending etc. performed by HR which can help the HR/user to reduce manual task which is tedious in nature. It enters same data in the form via code and tool all the time instead of manual entry. Thus, we can say that automation of processes or more specifically automation of websites can help us to improve efficiency.

REFERENCES

- [1] Stack Abuse, Definition of Web Browser Automation, <https://stackabuse.com/web-browser-automation-with-selenium-and-java>
- [2] Marcus Atish D Rozario, Project on Employee Database and Payroll Management System, Department of Computer Applications, RCC Institute of Information Technology
- [3] A. S. Syed Navaz, A. S. Syed Fiaz, C. Prabhadevi, V. Sangeetha, S. Gopalakrishnan, Human Resource Management System, Department of Computer Applications, Muthayammal College of Arts & Science/ Periyar University, India
- [4] Tutorials Point, Components of Selenium, <https://www.tutorialspoint.com/explain-the-components-of-selenium>
- [5] BrowserStack, What is Selenium WebDriver, <https://www.browserstack.com/guide/selenium-webdriver-tutorial>
- [6] BrowserStack, Web Page Inspection Tools, <https://www.browserstack.com/guide/locators-in-selenium>