



Automation Testing BLUESTONE

Project based experience learning program

**BLUESTONE Automation Testing Project**

**About Bluestone:-**

Established in 2011, BlueStone is India's leading destination for high quality fine jewellery with strikingly exquisite designs. We aim at revolutionizing the fine jewellery and lifestyle segment in India with a firm focus on craftsmanship, quality and customer experience. In a short span of time, BlueStone has built a large family of loyal consumers in India and abroad.

We house more than 8000 unique designs for you to choose from. All these designs are crafted to perfection with utmost care giving you the flexibility to customize the product's gold purity and colour or diamond clarity to suit your needs.

Our stores have been instrumental in spreading the shine of BlueStone and bringing us closer to you. With world class experience, friendly staff and the dazzling beauty of exquisite jewellery, every store is a sparkling gem.

With an award-winning design team that pays great attention to detail, each of our products are a symbol of perfection. With cutting edge innovation and latest technology, we make sure the brilliance is well reflected in all our jewellery.

We also offer a 30 Day Money Back guarantee, Certified Jewellery and Lifetime Exchange. You can also experience luxury shopping from the comfort of your home with our complimentary Try At Home service.

BlueStone is a web-based e commerce application and it deals with all type of jewellery items and it has both mobile app support for customers to track orders and easy access.

**Table of Contents**

* Introduction
* Project Overview
* Getting Started
  + Prerequisites
  + Installation
  + Project Setup
* Project Structure
* Writing Test Cases
* Running Tests
* Test Reporting
* Troubleshooting
* Conclusion

**Introduction**

The BLUESTONE project is a Python automation project that utilizes the Selenium framework for automating web interactions. This documentation provides an overview of the project, instructions for installation and configuration, details about the project structure, and examples of test cases.

**Project Overview**

The BLUESTONE project aims to automate various tasks on the Bluestone website. It leverages Selenium, a powerful web automation tool, to interact with web elements and perform actions such as clicking buttons, filling forms, and extracting data.

**Milestone 1: Getting Started**

Explain the prerequisites and setup required to run the project.

**Activity 1:-Prerequisites**

Before getting started with the BLUESTONE project, ensure you have the following prerequisites:

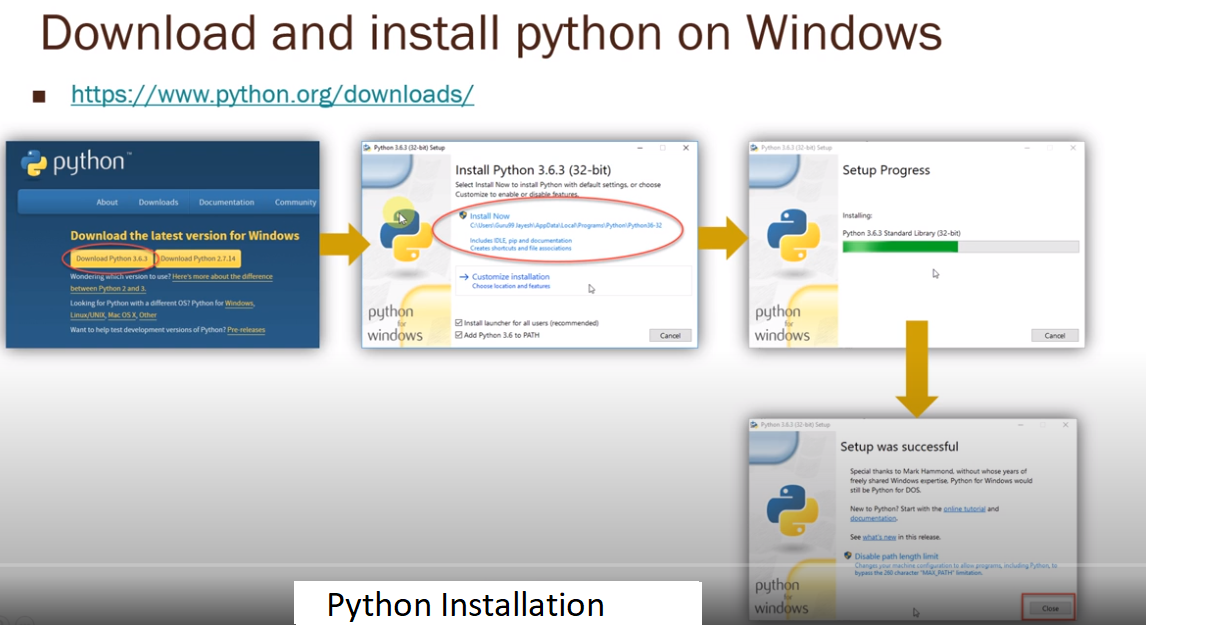
List the software and tools that need to be installed before running the project, such as

* **Python (3.x)**
* **Selenium (3.x)**
* **Pytest (Latest Version)**
* **Any other dependencies**

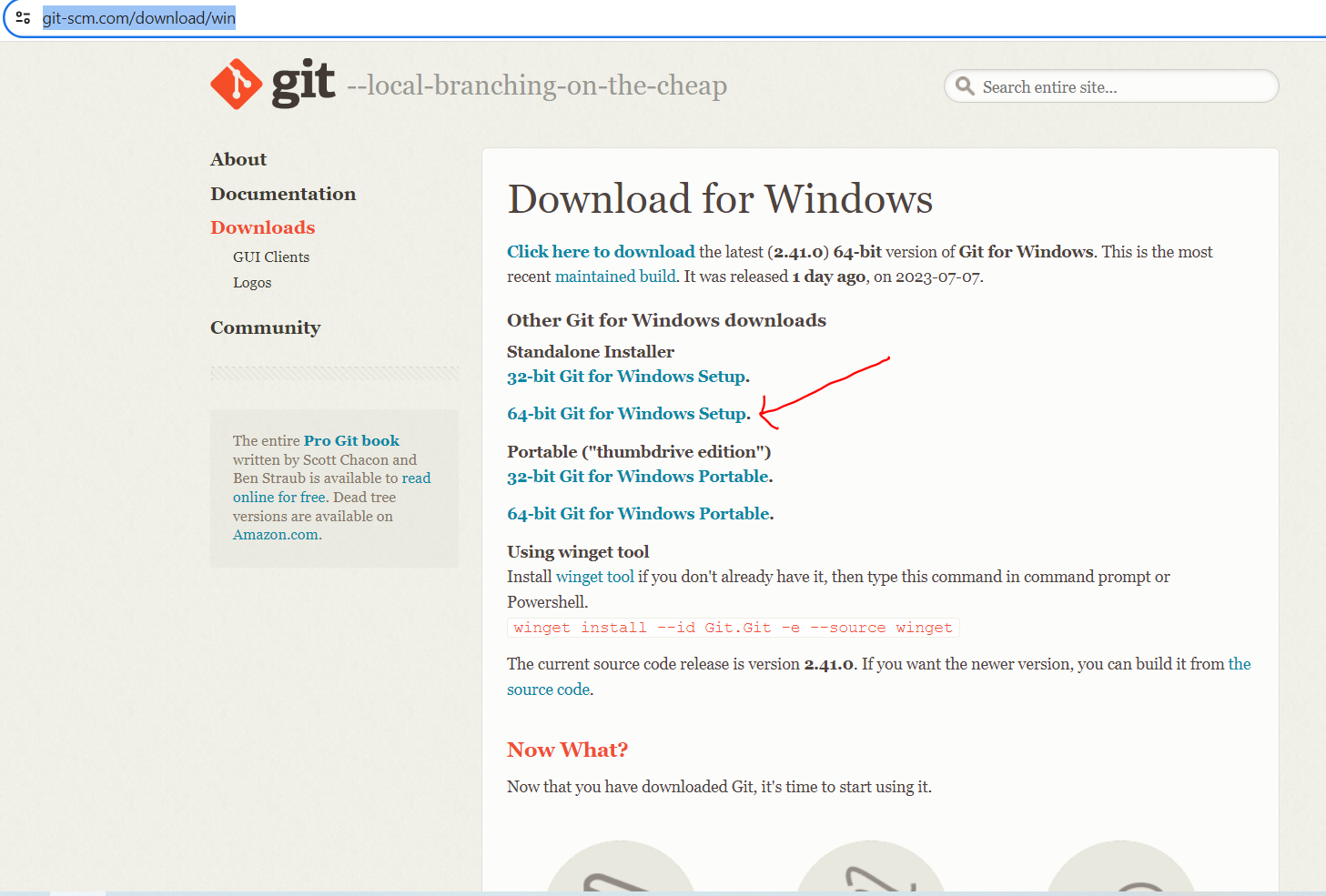
**Activity 2:-Installation**

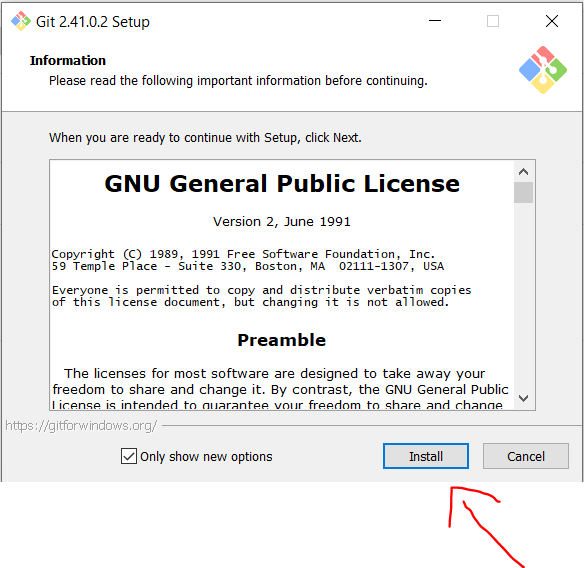
To install the necessary dependencies for the BLUESTONE automation project, follow these steps:

**Activity 2.1:-** Install Python: Visit the Python website ([https://www.python.org](https://www.python.org/)) and download the latest version of Python. Follow the installation instructions for your operating system.



**Activity 2.2:-** Download and install **gitbash** from <https://git-scm.com/download/win>



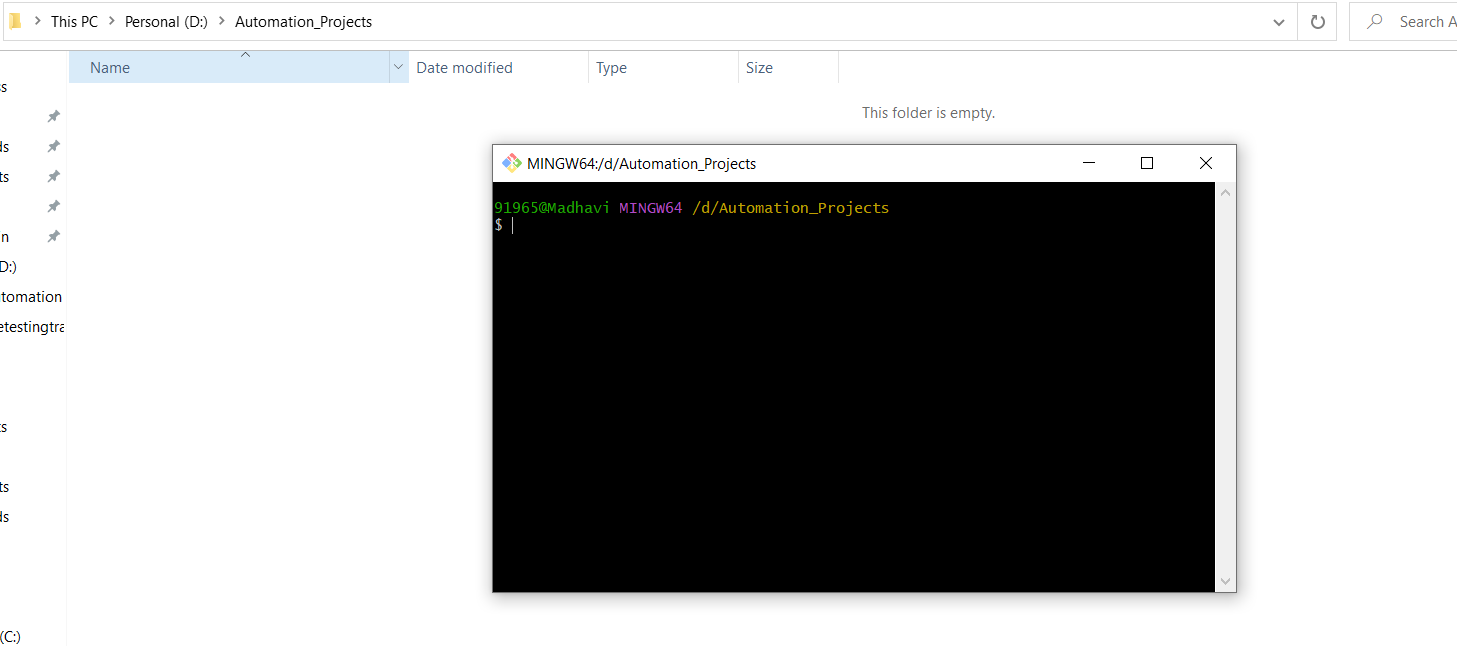


**Activity 2.3 :-**

Go to the desired location where you want to create the folder, such as Desktop.

Folder name like eg…(Automation\_Testing\_Project)

Open the folder-🡪 Right click with in the folder and select open gitbash here, you will get pop up like below and clone the repository based on the next steps provided below.



**Activity 3:- Project Setup**

Explained how to set up the project locally, including steps like:

**Activity 3.1:-** Clone the repository from github.

**Activity 3.2:-** Clone the BLUESTONE project repository from [repository URL].

<https://github.com/viswatesting123/SmartBridge_Automation_Projects.git>

**Activity 3.3:-** Install all dependencies using **requirements.txt**

**Requirements.txt containing all software’s/libraries required for our projects. We don’t need to install separately.**

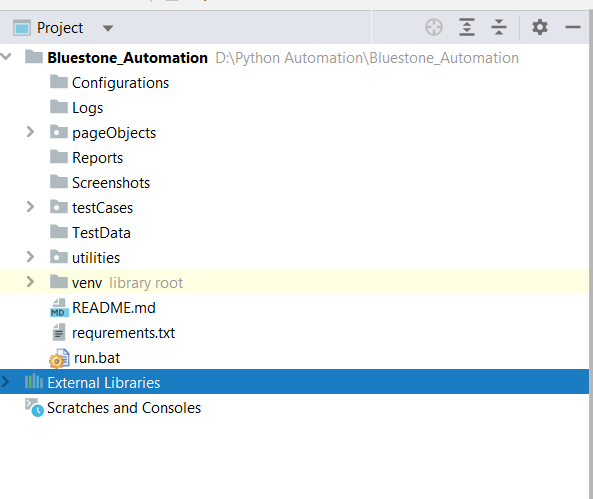
**Activity 3.4:- RUN the command from the terminal:**



**Activity 3.5:-** Configuration settings (if any)

**Milestone 2: Project Structure**

**Activity 1:-**Explained the structure of the project, including directories and files, and their purposes. For example:

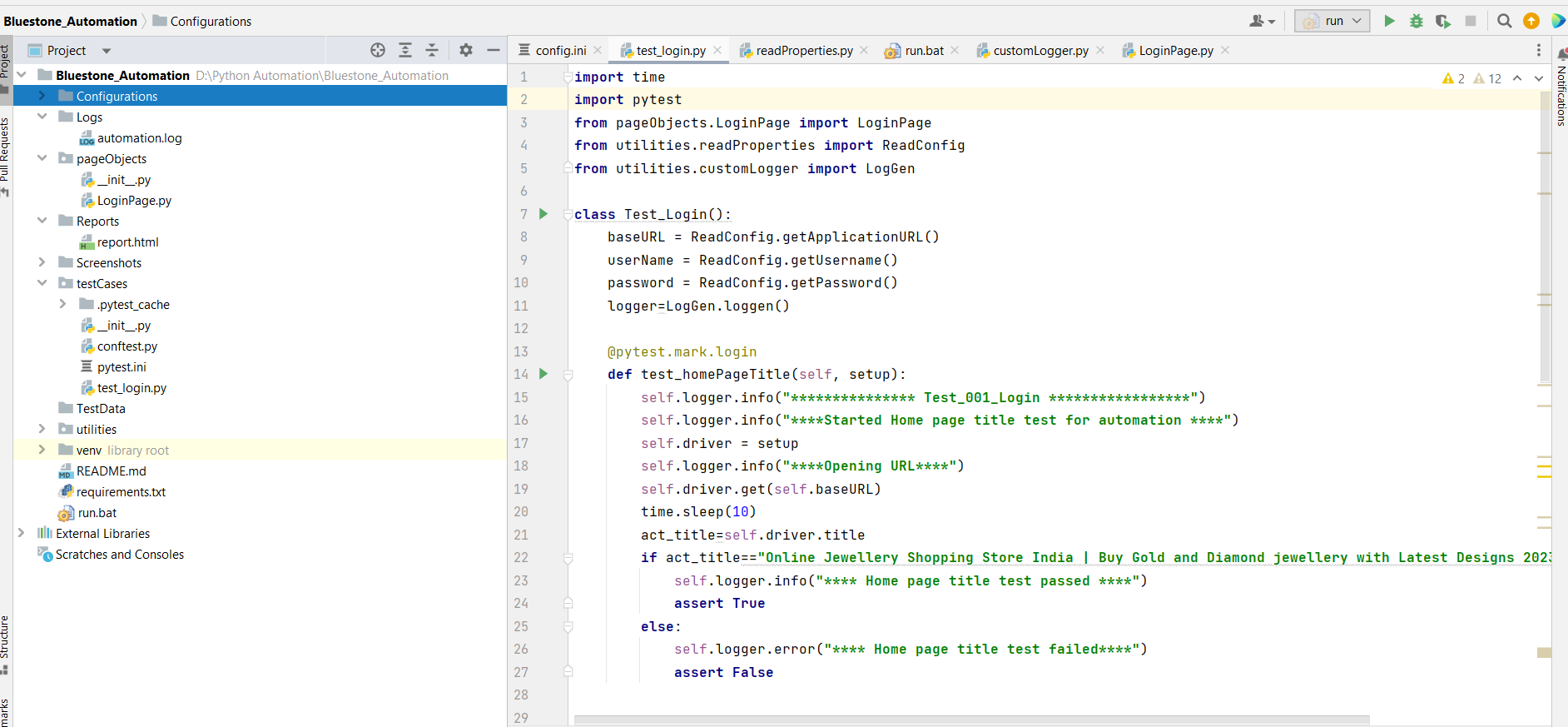


* **testcases/**: Directory containing test scripts
* **pageObjects/**: Directory containing page object models
* **utils/**: Directory containing utility functions
* **reports/**: Directory containing test reports
* **testData/**: Directory containing test data (**ex:excel,csv,text,json files**) related to project
* **Configuration/**: Configuration file for storing settings
* **Screenshots/**: Project testcases screenshots.
* **Logs**/: Directory maintain the project logs.
* **Run**.**bat**/: Now, you can double-click the **run.bat** file to execute the Python script specified in the command. The script will run in a Command Prompt window.

**Activity 2:-Writing Test Cases**

Explained how to write test cases using Python, Selenium, and pytest. Provide examples of different types of test cases, such as:

* Test case 1: Home Page functionality

Screenshot of the code: 

* Test case 2: Register new user functionality
* **Add new testcases as per Manual testcases prepared.**

Include best practices for writing maintainable and readable test cases.

**Milestone 3: Running Tests**

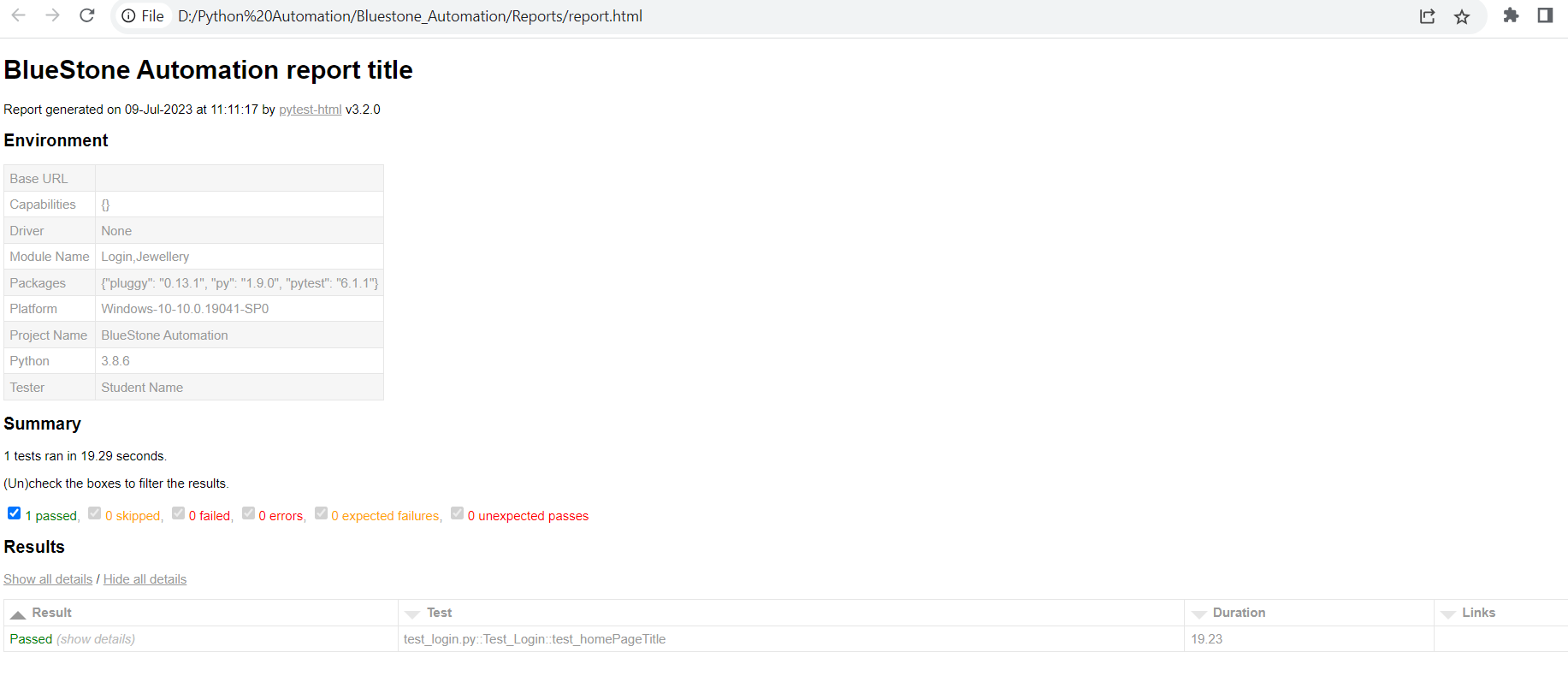
Explained how to run the test cases using pytest. Include information about command-line options and flags that can be used to customize test execution.

**Activity 1:- Run**.**bat**/: Now, you can double-click the **run.bat** file to execute the Python script specified in the command. The script will run in a Command Prompt window.

**Milestone 4: Test Reporting**

Explained how test reports are generated and where they are stored. Include information about any tools or libraries used for reporting, such as pytest-html.

**Activity 1:-reports/**: Directory containing test reports



**Troubleshooting**

List common issues that may occur during project setup or test execution and provide possible solutions.

**Conclusion**

The **BLUESTONE** automation project provides a framework for automating various tasks within the **BLUESTONE** web application using Python, Selenium, and Pytest. With this documentation, you should be able to set up the project, execute tests, and extend/customize it according to your needs.