



## 1. Introduction

**BrowserStack** is a cloud-based cross-browser testing platform that enables developers and testers to test their websites and mobile applications across 3,000+ real devices and browsers. It eliminates the need for setting up complex device labs and virtual machines.

## 2. Why Use BrowserStack?

- Access to **real devices** for accurate testing.
- No need for emulators or simulators.
- Supports **manual and automated testing**.
- Wide browser/device/OS combinations.
- Ideal for **agile** and **DevOps** workflows.

## 3. Key Features

Feature	Description
Real Device Cloud	Test on physical Android and iOS devices.
Live Testing	Interactive browser/device access.
Automate	Selenium, Cypress, Playwright, and Appium support.
Local Testing	Test local servers behind firewalls.
CI/CD Integration	Works with Jenkins, GitHub Actions, GitLab, etc.
Visual Testing	Screenshots and layout comparison tools.
Parallel Test Execution	Run multiple tests simultaneously to save time.
Debugging Tools	View logs, screenshots, videos, and console output for each test.
<b>Geo-location Testing</b>	Evaluate if your UI is usable for all users.

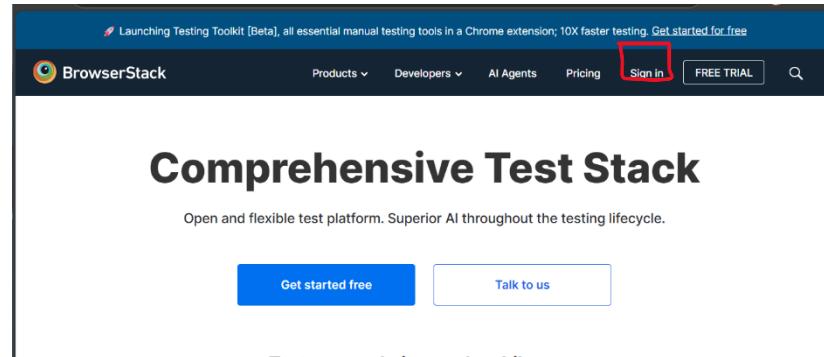
## 4. Types of Testing Supported

Testing Type	Tools Used / Services
Manual Web Testing	Live
Manual App Testing	App Live
Selenium Testing	Automate
Appium Testing	App Automate
Cypress Testing	Automate
Playwright Testing	Automate
Visual Regression Testing	Percy
Localhost Testing	BrowserStack Local binary

## 5. Setting Up BrowserStack

### Step 1: Create account

1. Go to <https://www.browserstack.com/>
2. Click on Sign Up (top right).
3. Use your **email** and set a password (or sign in with GitHub Google).
4. After signup, go to <https://www.browserstack.com/automate>
5. You will find:
6. **Username**
7. **Access Key**



or

**Authentication & Security**

Email Address	pasindiyachavindi13@gmail.com
Linked account	 Google
Username	pasindiyachavindi_aqrLHz 
Access Key	DJvpMvmYRtraEPy2CLHu 

[Edit details](#)

## Step 2: Install Python and Required Packages

1. Install Python (if not already installed):

Download: <https://www.python.org/downloads/>

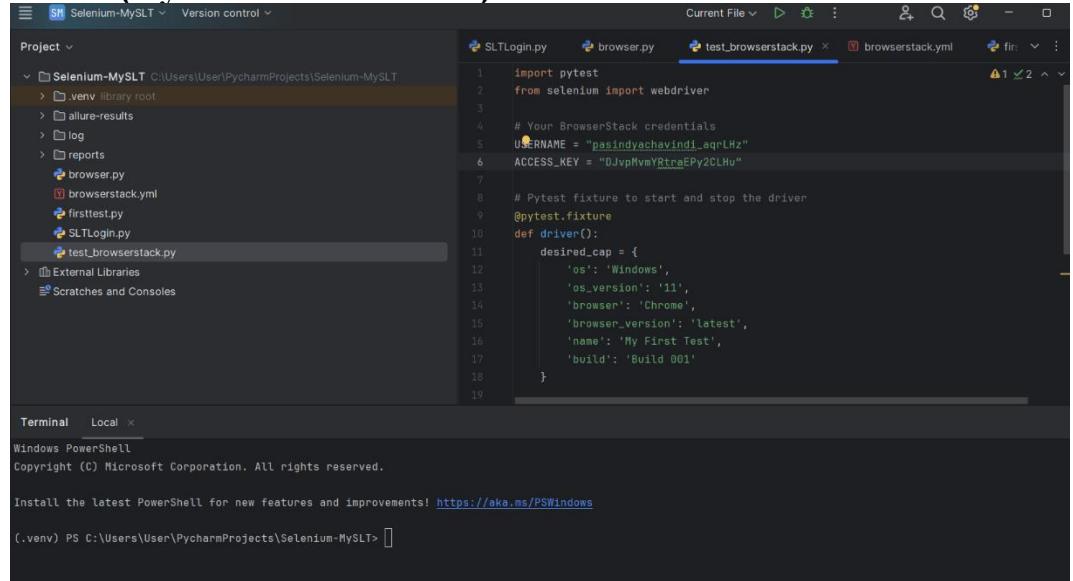
2. Install packages:

Open your **command prompt or terminal** and type:

**Pip install selenium pytest**

## Step 3: Write Your First BrowserStack Test (Python + Selenium)

1. Open PyCharm or VS Code.
2. Create a file called test\_browserstack.py
3. Copy this code:



The screenshot shows the PyCharm IDE interface. The project structure on the left displays a folder named 'Selenium-MySLT' containing files like 'browser.py', 'firsttest.py', 'SLTLogin.py', and 'test\_browserstack.py'. The code editor on the right contains the following Python script:

```

1 import pytest
2 from selenium import webdriver
3
4 # Your BrowserStack credentials
5 USERNAME = "pasindiyachavindi_aqrLHz"
6 ACCESS_KEY = "DJvpMvmYRtraEPy2CLHu"
7
8 # Pytest fixture to start and stop the driver
9 @pytest.fixture
10 def driver():
11     desired_cap = {
12         'os': 'Windows',
13         'os_version': '11',
14         'browser': 'Chrome',
15         'browser_version': 'latest',
16         'name': 'My First Test',
17         'build': 'Build 001'
18     }
19

```

Below the code editor, the terminal window shows the command line environment:

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

(.venv) PS C:\Users\User\PycharmProjects\Selenium-MySLT>

```

## Step 4: Run the Test

Open your terminal or command prompt in the project folder and run:

**pytest test\_browserstack.py**

The test will run on BrowserStack's cloud (you won't see it on your screen, it happens online).

## Step-by-Step Setup Using SDK (browserstack-sdk)

### Step 1: Install SDK and All Required Packages

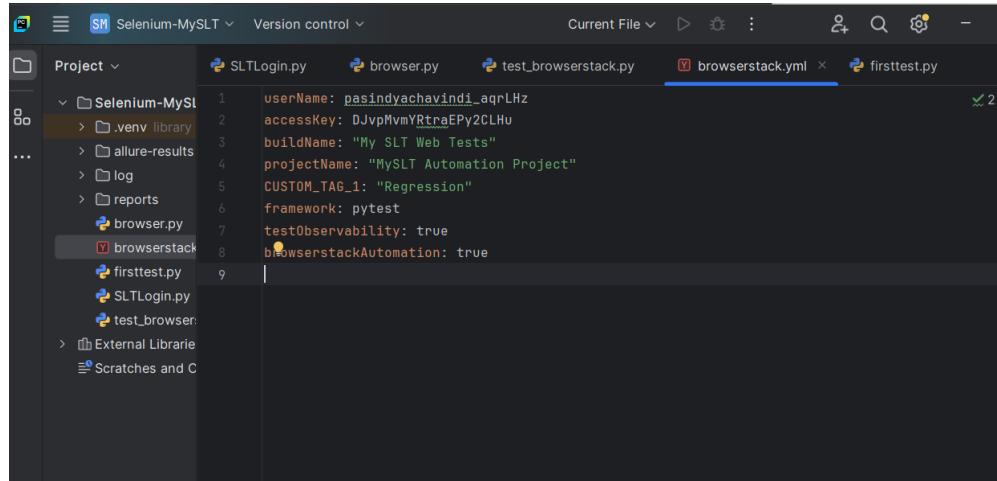
In your terminal:

**pip install browserstack-sdk selenium pytest allure-pytest**

```
(.venv) PS C:\Users\User\PycharmProjects\Selenium-MySLT>
(.venv) PS C:\Users\User\PycharmProjects\Selenium-MySLT>
(.venv) PS C:\Users\User\PycharmProjects\Selenium-MySLT> pip install browserstack-sdk selenium pytest allure-pytest
Requirement already satisfied: browserstack-sdk in c:/users/user/pycharmprojects/selenium-myslt/.venv/lib/site-packages (1.29.1)
Requirement already satisfied: selenium in c:/users/user/pycharmprojects/selenium-myslt/.venv/lib/site-packages (4.33.0)
Requirement already satisfied: pytest in c:/users/user/pycharmprojects/selenium-myslt/.venv/lib/site-packages (8.4.1)
Requirement already satisfied: allure-pytest in c:/users/user/pycharmprojects/selenium-myslt/.venv/lib/site-packages (2.14.3)
Requirement already satisfied: psutil in c:/users/user/pycharmprojects/selenium-myslt/.venv/lib/site-packages (from browserstack-sdk) (7.0.0)
Requirement already satisfied: pyyaml in c:/users/user/pycharmprojects/selenium-myslt/.venv/lib/site-packages (from browserstack-sdk) (6.0.2)
Requirement already satisfied: browserstack-local>=1.2.5 in c:/users/user/pycharmprojects/selenium-myslt/.venv/lib/site-packages (from browserstack-sdk) (1.2.12)
```

## Step 2: Create a File Named browserstack.yml

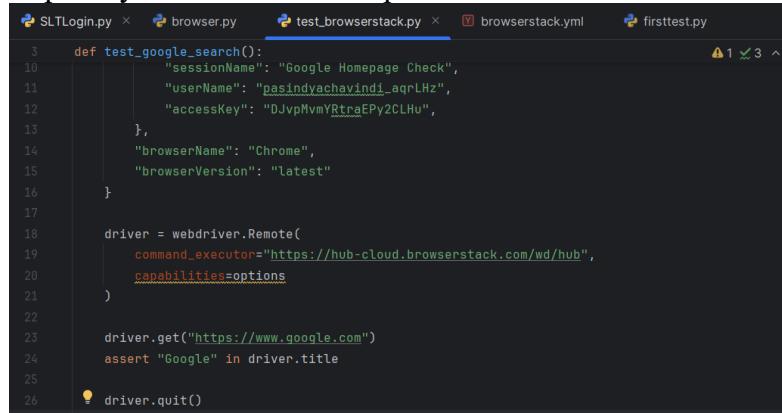
Inside your project folder (Selenium-MySLT), add this file:



```
1  userName: pasindyaachavindi_aqrLHz
2  accessKey: DJvpfNmYRtrnEPy2CLHu
3  buildName: "My SLT Web Tests"
4  projectName: "MySLT Automation Project"
5  CUSTOM_TAG_1: "Regression"
6  framework: pytest
7  testObservability: true
8  browserstackAutomation: true
```

## Step 3: Your test\_browserstack.py (Updated for SDK)

Replace your current test script with:



```
3  def test_google_search():
4      "sessionName": "Google Homepage Check",
5      "userName": "pasindyaachavindi_aqrLHz",
6      "accessKey": "DJvpfNmYRtrnEPy2CLHu",
7  },
8  "browserName": "Chrome",
9  "browserVersion": "latest"
10 }
11
12 driver = webdriver.Remote(
13     command_executor="https://hub-cloud.browserstack.com/wd/hub",
14     capabilities=options
15 )
16
17 driver.get("https://www.google.com")
18 assert "Google" in driver.title
19
20 driver.quit()
```

## Step 4: Run Test Using SDK Wrapper

Instead of using just pytest, use the SDK wrapper:

```
browserstack-sdk pytest test_browserstack.py --alluredir=reports
```

```
(.venv) PS C:\Users\User\PycharmProjects\Selenium-MySLT> browserstack-sdk pytest test_browserstack.py --alluredir=reports
2025-06-26T06:28:47Z [browserstack_sdk.__init__][INFO] - SDK run started with id: 4e93c37e-c264-4736-be9f-74e1bfc7e274
2025-06-26T06:28:55Z [browserstack_sdk.__init__][INFO] - Project and debug logs are captured by default. To disable, set di
```

Then result will be display like this

```
t_browserstack.py Test passed: test_browserstack.py::test_google_search jelsqkbeogm70ynozwidxficvjy0hhkom2lqtlyj
[100%]

===== 1 passed in 21.64s =====
5-06-26T06:29:53Z [SDKCLI][INFO] - Visit https://observability.browserstack.com/builds/jelsqkbeogm70ynozwidxficvjy0hhkomtlyj to view build report, insights, and many more debugging information all at one place!

5-06-26T06:29:53Z [browserstack_sdk.__init__][INFO] - View build on BrowserStack dashboard: https://automate.browserstack.com/dashboard/v2/builds/3f1cf6e4cc2e986cb21c9573673b08e31ec60769
5-06-26T06:29:53Z [browserstack_sdk.__init__][INFO] - SDK run ended for id:4e93c37e-c264-4736-be9f-74e1bfc7e274, testhub: jelsqkbeogm70ynozwidxficvjy0hhkom2lqtlyj
(venv) PS C:\Users\User\PycharmProjects\Selenium-MySLT>
(venv) PS C:\Users\User\PycharmProjects\Selenium-MySLT>
```

When script is running test steps are automatically recording on virtual desktop screen like this

The screenshot shows the BrowserStack Automate web interface. On the left, there's a sidebar with various project management and support links like 'Show Video Tour', 'Give Feedback', and 'View Documentation'. The main area is titled 'Build 1#1' and shows a 'Tests' tab selected. It displays two test runs labeled 'MySLT Automation Suite'. A message at the top says 'Your free plan comes with 100 mins! You have 96 mins remaining.' Below the tests, there's a warning about missing status information. To the right, the 'MySLT Automation Suite' window is displayed, showing a screenshot of a mobile application interface. The 'Run Overview' section includes 'Run information' (1m 22s, Chrome 137, Win 11) and 'Capabilities' (Local, OFF). On the far right, the 'Debug' section shows a timeline of test steps: '01:06 Click element' (0.018s), '01:11 Find element' (0.24ms), and '01:11 Click element' (0.022s). Each step has a corresponding screenshot and a detailed description.

## Step 5: View Your Results

Visit:

🔗 <https://automate.browserstack.com/>

Your free plan comes with 100 mins! You have **97 mins** remaining. [Get a Demo](#)

**Overview**  
Parallels 0/5 running | 0 queued

**Latest Builds** [All builds](#) [My builds](#)

- My SLT Web Tests #3** Regression  
Ran by 10 mins ago in MySLT Automation Project
- My SLT Web Tests #2** Regression  
Ran by 43 mins ago in MySLT Automation Project
- My SLT Web Tests #1** Regression  
Ran by 1 hr ago in MySLT Automation Project

**Test Executions Usage** 0  
BrowserStack Test Executions 0/∞  
External Test Executions 0/10k  
Resets on Jul 23, 2025  
Integrate all tests for complete visibility of your testing

**Notifications**  
No notifications yet  
Get started by [Configuring alerts](#) or [explore comments](#) to start receiving notifications.

[Talk to an Expert](#)

Your free plan comes with 100 mins! You have **94 mins** remaining. [Get a Demo](#)

**My SLT Web Tests #3** Regression  
Passed: 0 | QG Passed: 0 | Pasindya ch... | 21.73s | 42 mins ago | View metadata

**Build Summary** 1 Tests  
Passed: 1 | Failed: 0 | Skipped: 0

**Flakiness** 0

**Muted Tests** 0

**New Failures** 0

**Always Failing** 0

**Unique Errors** 0

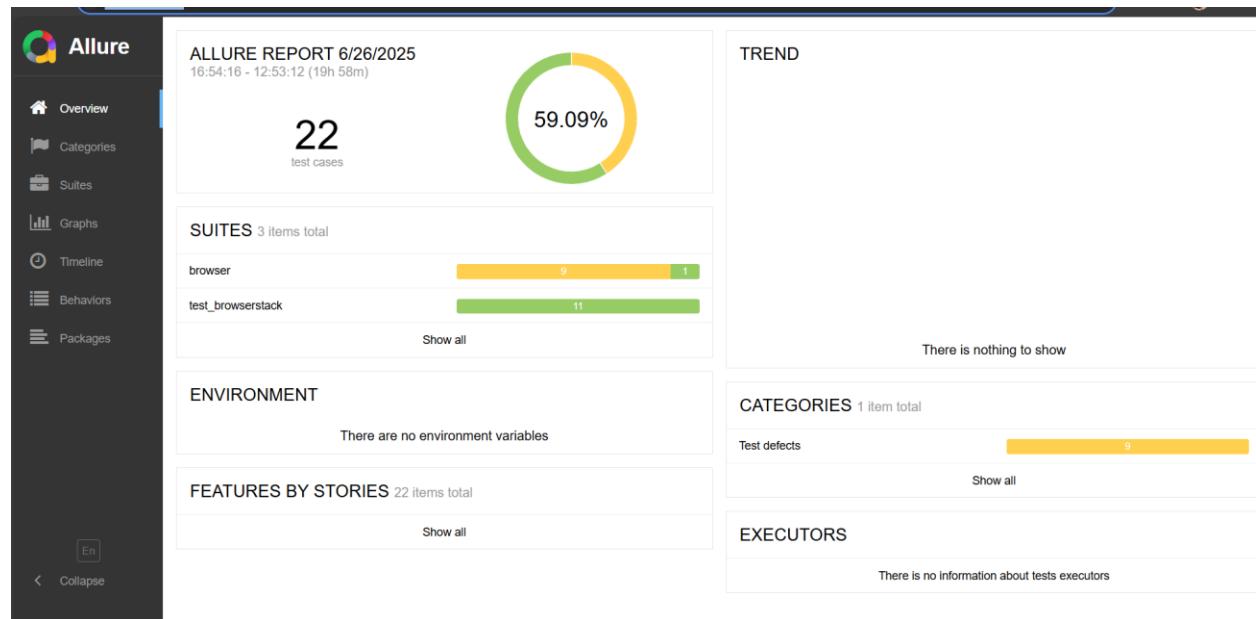
**Build History**  
Passed: 1/25 | Failed: 0/25 | Skipped: 0/25

**Alerts**  
Configure Alerts  
You have not configured any alerts  
[Get started](#)

**Build Stability**

You can get allure result also by entering this command on terminal :  
**allure serve reports**

```
-- Docs: https://docs.pytest.org/en/stable/how-to/capture-warnings.html
=====
10 passed, 1 warning in 85.29s (0:01:25)
(.venv) PS C:\Users\User\PycharmProjects\Selenium-MySLT> allure serve reports
Generating report to temp directory...
Invalid plugin directory C:\Users\User\scoop\apps\allure\current\plugins\junit-xml-plugin
Report successfully generated to C:\Users\User\AppData\Local\Temp\11793368392491593479\allure-report
Starting web server...
Server started at <http://127.0.0.1:51886>. Press <Ctrl+C> to exit
```



# Browserstack low code automation

## Key Features of low code automation

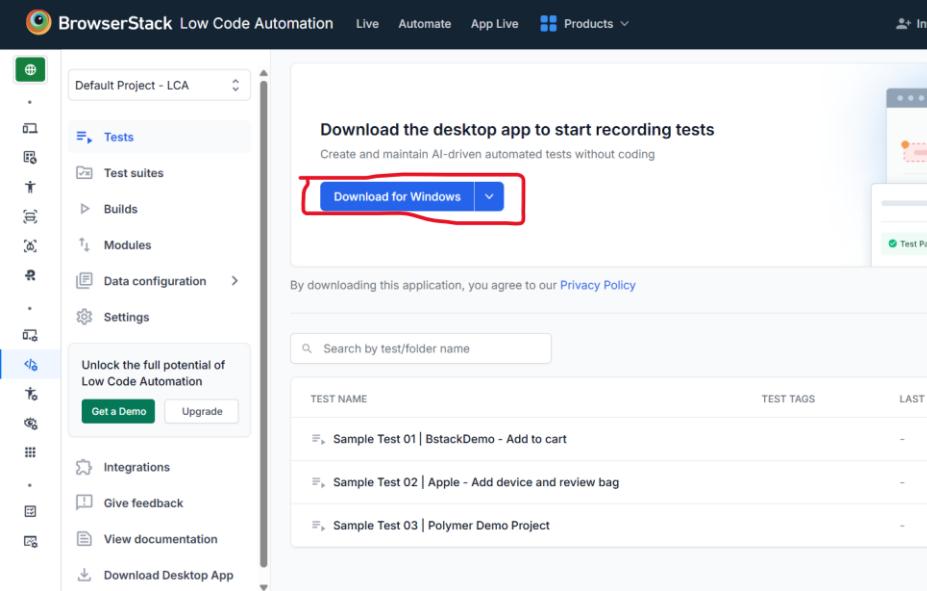
- No complex coding needed
- No learning curve
- No test infrastructure setup and maintenance
- Create resilient tests



## Step 1: Click on get start button

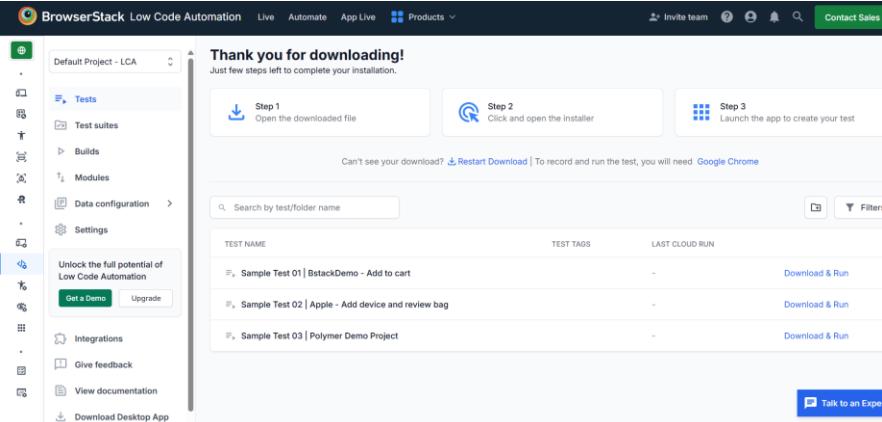
A screenshot of a web browser displaying the 'low-code.browserstack.com/introduction' page. The page has a dark header with the BrowserStack logo and navigation links for 'Live', 'Automate', 'App Live', and 'Products'. A 'Contact Sales' button is also visible. The main content area features a sidebar with icons for various tools like visual validation, text validation, and video recording. The central part of the page is titled 'Low Code Automation' and describes the service as 'Automate tests without writing a line of test code. No setup or training needed'. It lists three main features: 'Visual Validation', 'Text Validation', and 'Test Suites', each with a brief description and a small icon. At the bottom, there's a section for 'More than 50,000 customers trust BrowserStack' with logos for Microsoft, Google, and others, and two buttons: 'Learn More' and a large blue 'Get Started' button. A red rectangular box highlights the 'Get Started' button.

## Step 2: Click on download button



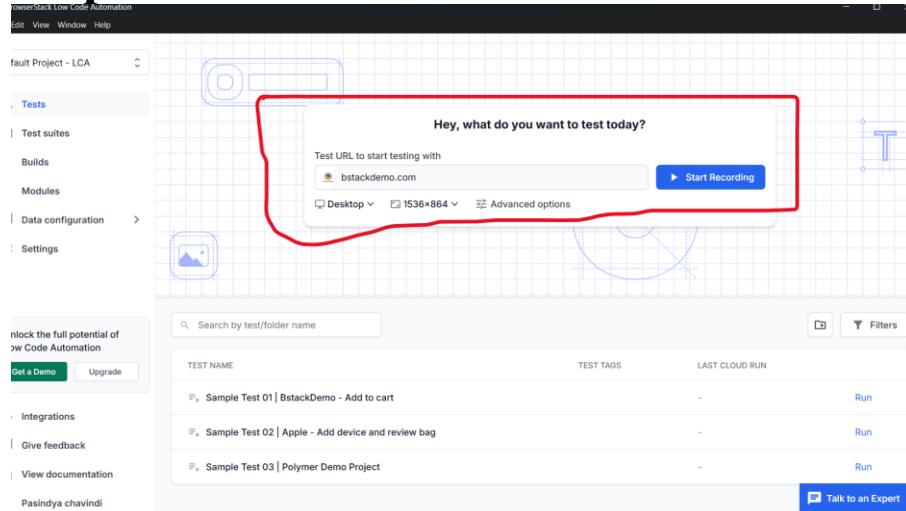
The screenshot shows the BrowserStack Low Code Automation dashboard. On the left, there's a sidebar with various project management and support links. The main area displays a message: "Download the desktop app to start recording tests" followed by a sub-instruction: "Create and maintain AI-driven automated tests without coding". A prominent blue button labeled "Download for Windows" is highlighted with a red box. Below the button, a note states: "By downloading this application, you agree to our [Privacy Policy](#)". To the right, there's a table listing three sample test cases: "Sample Test 01 | BstackDemo - Add to cart", "Sample Test 02 | Apple - Add device and review bag", and "Sample Test 03 | Polymer Demo Project".

## Step 3: then do these steps

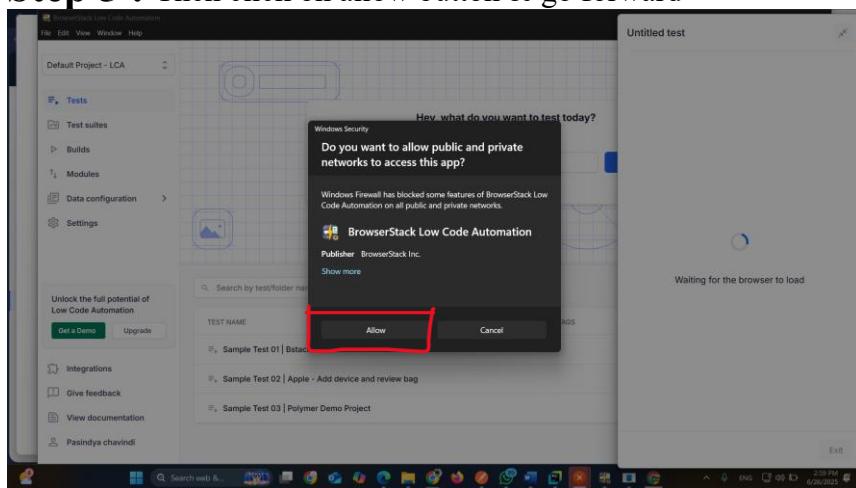


The screenshot shows the same dashboard after the download step. A message at the top says "Thank you for downloading!" with a note: "Just few steps left to complete your installation." Below this, three steps are outlined: "Step 1 Open the downloaded file", "Step 2 Click and open the installer", and "Step 3 Launch the app to create your test". A note below the steps says: "Can't see your download? [Restart Download](#) | To record and run the test, you will need [Google Chrome](#)". The table of test cases from the previous screen is still visible. At the bottom right, there's a blue button labeled "Talk to an Expert".

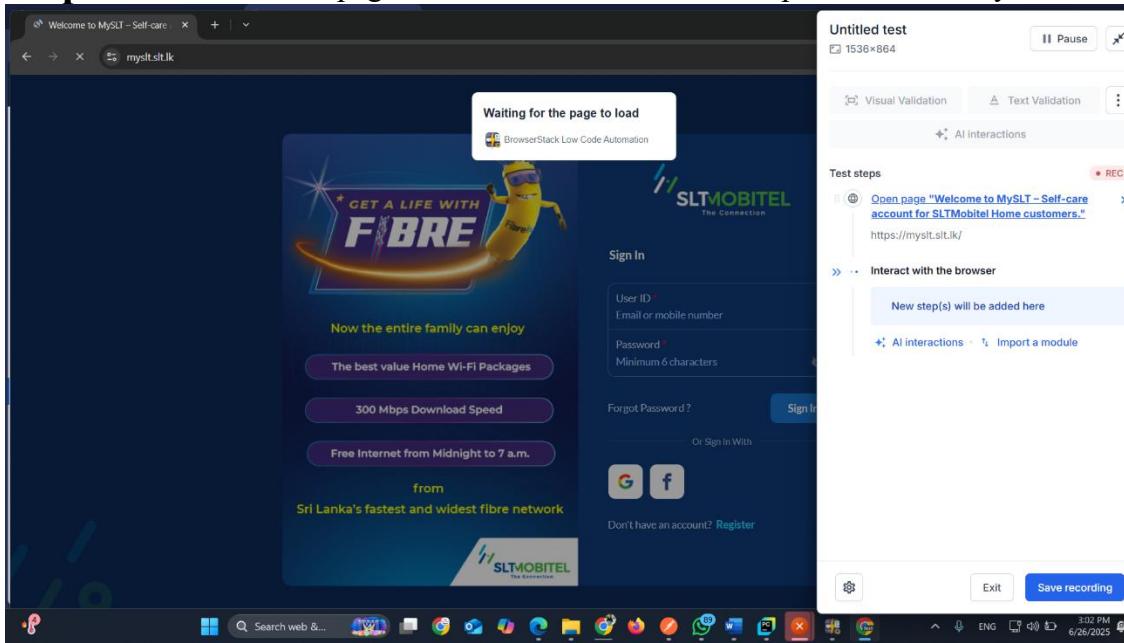
## Step 4: Type url in search section & click on start recording button



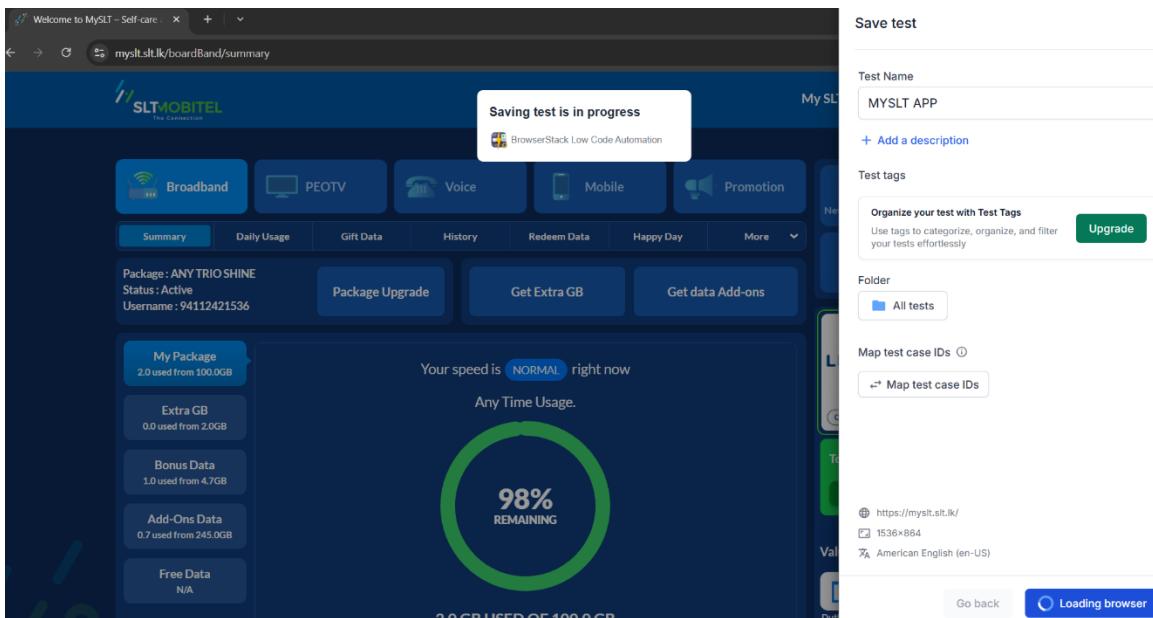
## Step 5 : Then click on allow button & go forward



**Step 6 :** After the web page related to entered URL will open automatically & It will record all the steps you do



**Step 7 :** Then the recorded steps will automatically save like this



## Step 8 : Finally test result will display top of the page

The screenshot shows the SLT MOBITEL Self-care account dashboard. On the left, there's a summary of usage: My Package (2.0 used from 100.0GB), Extra GB (0.0 used from 2.0GB), Bonus Data (1.0 used from 4.7GB), Add-Ons Data (0.7 used from 245.0GB), and Free Data (N/A). In the center, a large circular progress bar indicates "Your speed is NORMAL right now" with "98% REMAINING" and "2.0 GB USED OF 100.0 GB". At the top, there are tabs for Broadband, PEOTV, Voice, Mobile, and Promotion, with "Summary" selected. Below the tabs, it shows the package as ANY TRIO SHINE, status as Active, and username as 94112421536. Buttons for Package Upgrade, Get Extra GB, and Get data Add-ons are also present.

On the right, a "MYSLT APP" window is open, showing a test log. The log header says "Passed" with a green checkmark, a duration of "1m 16s", and 25 successful steps. A red box highlights the top section of the log. The log details 25 steps, each with a green checkmark indicating success:

- 16 Click on "Mobile" button
- 17 Click on "Voice" element
- 18 Click on "New Services" element
- 19 Navigate backward to "Welcome to MySLT – Self-care account for SLTMOBITEL H..." page
- 20 Click on "My SLT Portal" element
- 21 Click on "Bill" button
- 22 Click on "Voice" element
- 23 Click on "Call Forwarding" element
- 24 Click on "Ok" button
- 25 Click on "Broadband" element

At the bottom of the log window, there are buttons for "Edit test", "Go to test listing", and settings.