# SOFTWARE TESTING

## Project – Task 8 (*In-Browser Testing*)

**Group**: ST-G10

**Members**:

Sheraz Ahmad (L13-4049)

Asad Khalil (L17-4275)

Jawad (L17-4376)

# AUTOMATE IN-BROWSER TESTING

**In-Browser Testing**: Cross Browser Testing or In-Browser Testing, is a process to test web applications across multiple browsers. Cross browser testing involves checking compatibility of your application across multiple web browsers and ensures that your web application works correctly across different web browsers.

With **Selenium**, you can run your tests against any browser while Cypress only supports Chrome. So, if running against different browser vendors is an important feature for you Selenium wins. If you want to do browser testing, you should use cypress but if you want to run it on many browsers you should use selenium.

Selenium operates by running outside the browser and executes remote commands across the network. All the underlying commands are remote.

What you have is the test script running outside of the browser, executing remote commands into the browser. But it's never able to understand in the same run loop the reactions in the events that are being fired in the browser.

Where [Selenium Web Driver](https://testguild.com/selenium-webdriver/) runs remotely outside the browser, Cypress is the exact opposite; it runs inside the browser.

**Cypress** takes a different approach since it’s executed in the same run loop as your application. It also leverages a [Node.js](https://nodejs.org/en/) server to handle any task that needs to happen outside of the browser. Because of this, they claim to be able to give more consistent results since it’s able to understand everything happening inside and outside the browser.

This also affords you native access to every single object without having to deal with object serialization or over-the- wire protocols.

Basically, you pull your application into Cyprus.

As a result, it can synchronously be notified of everything that happens inside the browser. So, you have native access to every single Dom element.

Cypress also makes it easy to simply drop a debugger into your application or spec code, which makes it super easy to use the developer tools while you’re developing

Cypress is more developer-centric. It is aimed more towards making [TDD](https://testguild.com/2014/07/29/unit-tdd-and-bdd-testing-whats-the-difference/) development a reality with developers.

**References:**

[1] “JavaScript End to End Testing Framework,” *JavaScript End to End Testing Framework | cypress.io*. [Online]. Available: https://cypress.io/.

[2] “SeleniumHQ Browser Automation.” [Online]. Available: https://selenium.dev/.

[3] “What is Selenium? Introduction to Selenium Automation Testing.” [Online]. Available: https://www.guru99.com/introduction-to-selenium.html.

[4] “Cross Browser Testing, Compatibility Testing - AppPerfect.” [Online]. Available: http://www.appperfect.com/services/web-testing/cross-browser-testing.php.