**Playwright**

* Playwright is a powerful end-to-end testing framework that enables reliable automation for modern web applications.
* **Why Playwright?**

1. **Reliable End-to-end Testing:** 
   * Playwrigth’s Auto-wait capability ensure reliable and stable end-to-end testing for modern web applications, Even in the face of dynamic and complex user interactions.
   * [**Auto-waiting:**](https://playwright.dev/docs/actionability#:~:text=On%20this%20page-,Auto%2Dwaiting,-Introduction%E2%80%8B)

The playwright performs a range of actionability check on the elements before making actions to ensure these actions behave as expected. It auto-waits for all the relevant checks to pass and only then performs the requested actions. If required checks do not pass within the given **timeout**, action fails with the **TimeoutError.**

* + Before execution for that element playwright check some conditions are eligible on that element or not like it visible, stable or Enable or not

**A screenshot of a computer screen

Description automatically generated**

1. **Cross-Browser Compatibility:**
   * Playwright supports all major browsers, including Chrome, Edge, Firefox, Safari and Opera, allowing you to test your web applications across a wide range of browsers and platforms
2. **Multiplatform Support:**
   * Playwright works seamlessly on Windows, macOS, and Linux, and also supports native mobile emulation for Google chrome on Android and Safari on iOS, enabling comprehensive testing across different devices and operating systems.
3. **Multilingual Flexibility:**
   * Playwright provides language binding for JavaScript, TypeScript, Java, Python and C# allowing you to choose the programming language that bst fits your team’s preferences and expertise.

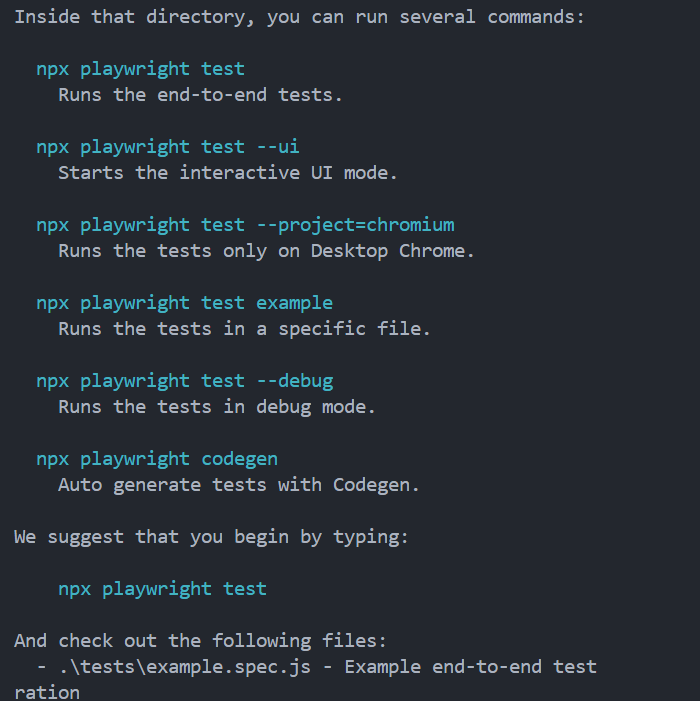
* **Playwright’s Advanced Features:**

1. **Tracing and Debugging:**
   * Playwright has built-in tracing and debugging capabilities, including automatic screenshots, test video recording and comprehensive logging , to simplify the process of identifying and resolving issues in your test suite.
2. **Network Interception:**
   * Utilize playwright’s API testing libraries to intercept and validate network calls within your web application, enabling you to test edge case scenarios and ensure the resilience of your application’s network interaction.
3. **Browser Context Management:**
   * Explore playwright’s browser context feature, which allows you to save and transfer browser state across your test suite, improving test efficiency and reducing the overhead of setting up the same browser state for each test case.
4. **Codegen Tool:**
   * It can generate code by recording your actions, saving your time and effort in creating initial test case and providing a starting point for further customization.

* **To create playwright project, we use following command**

npm init playwright

* + This command is creating playwright structure project with all playwright dependencies
  + **Playwright.config.json** file is important file which is also called as playwright test runner file



* **Sample test cases:**

A computer screen shot of a program code

Description automatically generated

We can manually create **page** using **browser** fixer or we can directly use **page** fixer to goto URL.

* To set which browser should execute we should mention in Playwright.config.json file.

A black background with red and green text

Description automatically generated

* To execute playwright project we use following command

**npx playwright test**

* Playwright by default run in headless mode means we won’t see anything of what is happening in the browser when our script runs. If we want to see browser execution then we should run in **headed** mode using following command

**npx playwright test –headed**

**or**

add in configuration file



* To generate test report:

**npx playwright show-report**

* To verify title of webpage we have assertion



It will check and verify title of webpage if match then **passes** otherwise **failed**

* **Note: webkit** is a default playwright specific browser engine.