**Documentation for Playwright Automation**

**Table of Contents**

1. **Cloning the Repository**
2. **Setting Up Playwright Framework**
   * **Using npm**
   * **Using npx**
3. **Setting Up Testing Extension in VS Code**
4. **Framework Structure Overview**

**1. Cloning the Repository**

**Prerequisites:** Ensure you have Git installed on your system. If not, download and install from <https://git-scm.com/downloads>.

**Steps:**

1. Open command prompt.
2. Navigate to your desired project directory.

“cd /path/to/your/project/directory”

1. Clone the repository using the provided Git URL.

“git clone https://your-git-repo-url.com/path -repo.git”

1. Change directory into your newly cloned repository.

“cd your-framework-repo”

**2. Setting Up Playwright Framework**

**Prerequisites:** Ensure you have Node.js (14.17.0 or later) and npm installed. Download from <https://nodejs.org/en/download/> if needed.

**Steps:**

1. Ensure you're in the project directory.

“cd /path/to/your/framework/repo”

1. Install dependencies, including Playwright.

“npm install”

**Note:** If you encounter issues, consider running npm install --force or updating your package.json to match the latest Playwright version.

1. Verify Playwright browsers are installed.

“npx playwright install”

**3. Setting Up Testing Extension in VS Code**

**Recommended Extension:** **Playwright Test for VSCode** by Microsoft

**Steps:**

1. Open VS Code.
2. Access the Extensions view by clicking the square icon on the sidebar or pressing Ctrl+Shift+X (Windows/Linux).
3. Search for "Playwright Test" in the Extensions Marketplace.
4. Select "Playwright Test for VSCode" by Microsoft and click **Install**.
5. Reload VS Code if prompted.
6. Open the Command Palette in VS Code with Ctrl+Shift+P (Windows/Linux).
7. Type "Playwright: Download Browser" and select the command to download browsers for your Playwright version.
8. To run your tests, use:

“Npm run test”

**4. Framework Structure Overview**

* **env/**: Environment configurations (e.g., URLs, usernames, passwords).
* **pages/**: Page Object classes containing web page elements' XPaths and Re-usable Functions.
* **tests/**: Test files where test functions are defined.
* **package.json**: Dependencies, scripts, and project metadata.

**5. Getting Started with Test**

1. Navigate to the tests/ directory.
2. Create a new test file (e.g., myFirstTest.ts).
3. Import necessary page objects and write your test using Playwright's API.

import { test, expect } from '@playwright/test';

import { MyPage } from '../pages/myPage';

test('My First Test',',