**Playwright Framework Setup**

Step 1:

Folder Structure:

1. Node\_modules is the folder which contains all the libraries. It contains not only the Playwright related libraries but also those libraries which are installed through npm command

Extensions:

1. ESLint: Integrates ESLint JavaScript into VS Code**(Search in ChatGPT)**

Step 2: Git Installation:

Access <https://git-scm.com/downloads/win> and download and install the .exe file

Step 3:

Create the required project structure by running the below commands in the Termninal:

Note: Copy all the commands at once and paste it in the Terminal and hit Enter

mkdir -p src/tests

mkdir -p src/pages

mkdir -p src/utils

mkdir -p src/config

mkdir -p src/api

mkdir -p src/reporting

mkdir -p src/logging

mkdir -p src/data

Step 4: Install Browser Engines by running the below command:

Npx playwright install

**Note:** **Need to understand the relationship between the browser and browser engine(Search in ChatGPT).**

**Step 5:** Creation of requirements.md file in the project structure(Sample file as shown below):  
  
A screenshot of a computer

Description automatically generated

Step 6: Page Object Chaining

Step 7: Git commit

From the project explorer, go to the working directory and open the command prompt

Enter the command git init

Verify if two files namely .git and .gitignore are present or not

Then, open the .gitignore file and paste the below content as it is:

Node\_modules/

/test-results/

/playwright-report/

/playwright/.cache/

Step 8: Create a new repository in Git

Once the new repository is created, run the below command so that the code present in the local project level can be pushed to Git

Step a: git remote add origin https://github.com/HudiniQA/Playwright\_Framework.git

Step b: git branch -M main ----------🡪 To create a main branch

Step c: git add .

Step d: git status

Step e: git commit -m "<Commit details>"

Step f: git push -u origin main

Step 9: Environment Management using dotenv file

1. Install dotenv using the below npm command in VSCode Terminal

npm install dotenv –save

1. Create separate .env files under config folder for each environment such as Staging, UAT etc and add the required details such as URL, login credentials
2. Select Command Prompt from the below option and run the below command:

A computer screen shot of a black screen

Description automatically generated

1. How to set the NODE\_ENV value:

Step 10: Encyption of credentials-**NOT COMPLETED-NEED TO WATCH THE VIDEO AGAIN**

We can encrypt/decrypt the credentials using CryptoJS library.

In order to install CryptoJS library, we need to run the below commands:

npm install crypto-js

npm install --save @types/crypto-js

Step 11: Implementing logging mechanism using Winston Logger

Install the below packages by running the below commands in the terminal:

npm i winston

npm i moment-timezone