CYPRESS PROJECT ON THE WEBSITE OF HSBC

Md Bashar Afzal

Description

1. We are going to make a Cypress application to automate the testing of the HSBC website (https://www.hsbc.co.in/)
2. We are going use the E2E (End-to-End testing) for this Automation Testing.
3. ***The Problem Statement*** **:**
    a. Visit the HSBC Home page (https://www.hsbc.co.in/).
    b. Check for possible checkpoint to make sure you are on the correct page. (HOME PAGE)
    c. Visit the login page.
    d. Check for possible checkpoint to make sure you are on the correct page. (LOGIN PAGE)
    e. Click on the help button.
    f. Close the help pop-up window.
    g. Enter the username detail from the fixture.
    h. Click on the checkbox.

Scope

Steps I will take to accomplish the Problem given to me **:**

1. Visiting the Home page

2. Checking through checkpoint to check whether I came to the right page.
   (To do that I need to have the basic knowledge of the page that I am going to be landed on.)
   For Example: - **i)** Some logo
   ii) Some Buttons
   iii) Some images
   iv) Title of the page
   These are some examples of elements that are present on a website to verify the page that we have landed on, is correct or not.

   The Checkpoints that I have put for checking whether we are on the Home page HSBC are:-
    a. Checking the presence of a “Log On” button.
    b. Checking the presence of the title.

3. After checking the presence of a "Log On" button as a checkpoint for homepage. We click on it and enter the new login page.

4. The new page that loads up, should also be checked for checkpoints to satisfy that we have landed on the page that we need to be in for further procedure. (As we now know what are checkpoints, we are going to look for the same in this page as well)

   <u>The Checkpoints that I have put for checking whether we are on the "Log On" page HSBC are :-</u>

   a. Checking the presence of a "Continue" button, but it should be disabled.
   b. Checking the presence of a textbox to enter the "USERNAME", but the textbox should be empty.
   c. Checking the presence of a "Remember Me" checkbox, which should be unchecked.
   d. Checking the presence of a "?" button for "HELP".

5. After checking all the checkpoints of the "Log On" page.

6. Grab the element for the "HELP" button and click on it.
7. On Clicking the help button a pop-up window will be displayed on the screen, with some user manual to understand the procedure of Logging In.

8. Grab the element of the pop-up window and close it, by clicking on it.

9. Make a 'fixture', storing the data in key-value pair. From we can access the data values that we need to pass/use on our webpage.

10. Grab the textbox element where we have to input the "USERNAME", and input username we are from the 'Fixture'.

- ## Structure

1. Have 1 TypeScript file for the Home Page automation.

   Does:

   a. Visiting the Home page of HSBC
   b. Does the 1st Checkpoint of verifying the Title.
   c. Does the 2nd Checkpoint of verifying the presence of "Log On" button.
   d. Clicks on the "Log On", that takes us to the "Log On" Page for further process.
   e. Export the class – 'Home'.

   Functions:

   a. **visit() –** used to visit the Home Page of HSBC.
   b. **checkTitleOfHomePage() –** Checks the 1st described checkpoint of the HomePage.
   c. **checkLogOnButton() -** Checks the 2nd described checkpoint of the HomePage.
   d. **clickLogOnButton() –** Clicks on the "Log On", to change and visit another page.

2. Have 1 other TypeScript file for the "Log On" Page automation.

   Does:

   a. Visiting the "Log On" page of HSBC
   b. Does the 1st Checkpoint of checking the "Continue" button, which should be disabled.
   c. Does the 2nd Checkpoint of checking the "USERNAME" textbox, which should be empty.
   d. Does the 3rd Checkpoint of checking the "Remember Me" checkbox, which should be unchecked.
   e. Does the 4th Checkpoint of checking the presence of the "HELP" button on the "Log On" screen.
   f. After completing the 4 "Log On" page checkpoints. Then proceed with the further steps.
   g. Click on the "HELP" button to open a pop-up window with the procedure instructed for new users.
   h. Click on the pop-up window to close it.
   i. Click on the textbox area to type the details we are getting from the 'fixture' file.
   j. Export the class – "LogOn".

Functions:

a. **visit() –** used to visit the Home Page of HSBC.

b. **checkContinueButtonStatus() –** Checks the 1st described checkpoint of the LogOnPage.

c. **getUsername() –** Grabs the textbox being displayed to enter the "USERNAME".

d. **checkUsernameIsBlank() -** Checks the 2nd described checkpoint of the LogOnPage.

e. **checkRememberCheckboxIsUnchecked() -** Checks the 3rd described checkpoint of the LogOnPage.

f. **checkHelpButton() -** Checks the 4th described checkpoint of the LogOnPage.

g. **clickToOpenHelp() –** Grabs the "HELP" button Web Element and clicks on it.

h. **clickToCloseHelp() –** Grabs the pop-up window Web Element came after clicking on the "HELP" button. Gets the close button of the pop-up and clicks on it, to close the pop-up.

i. **enterUsername(username : any) –** Gets the username from the fixture, which is passed as parameter and we use this function to type the received value into the textbox.

3. Have 1 '.cy.ts' file to Test the application.

Does:

a. Imports the two above defined '.ts' (TypeScript files), one for the Home Page components and the other for the LogOn Page components.

b. Defines the structure and organizes that the different Test scenarios, in the "DESCRIBE" method, it is a function with 2 parameters. A name and a callback function.

c. Every Test Scenario is defined inside the "IT" function. This function also has 2 parameters. A name and a callback function.

4. Make a JSON file inside "fixtures" folder. Containing the Key-value pairs for the values that we need in the describe method. Here, the username is coming from the JSON file.

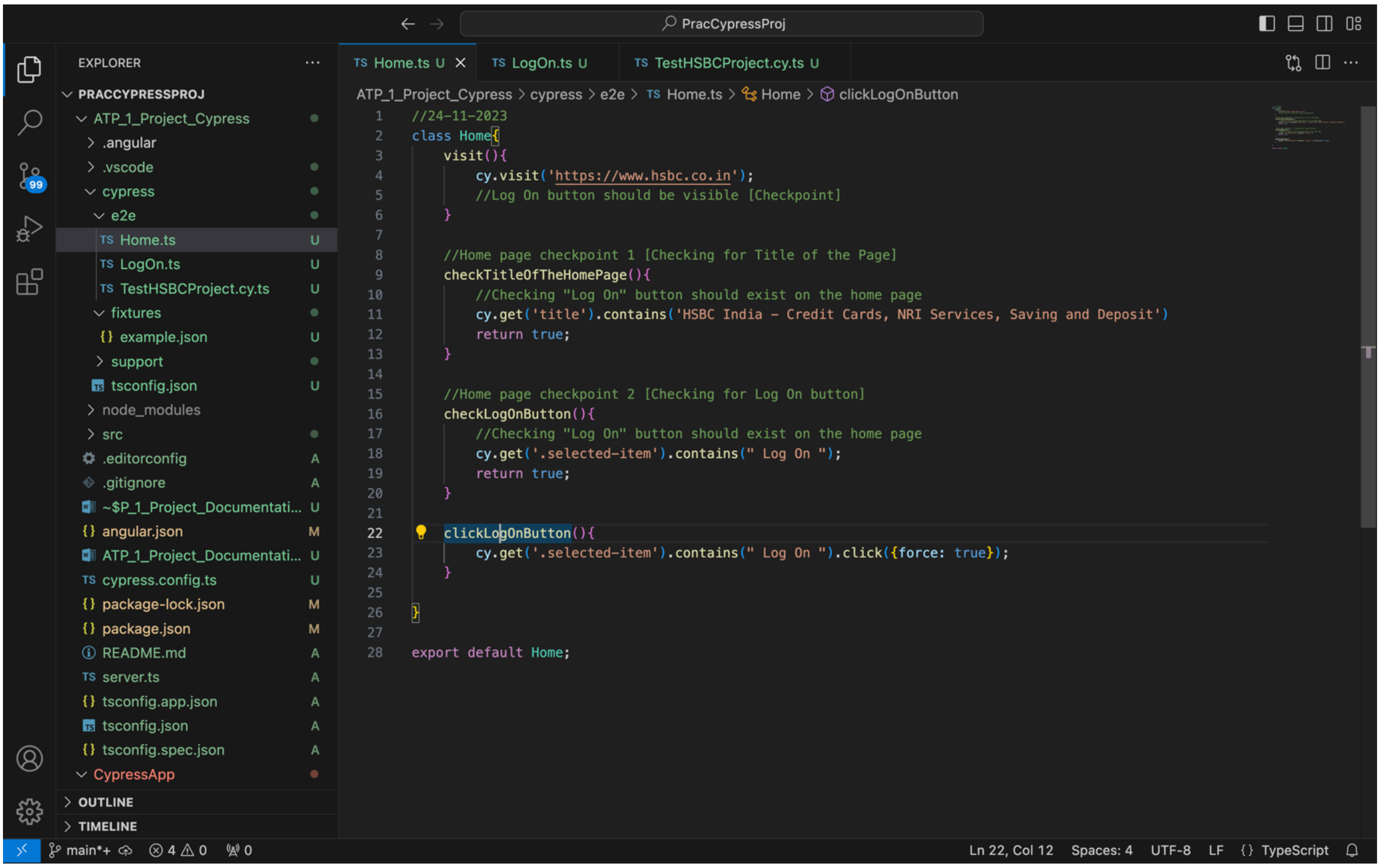
• Page Object Model

Page Object Model – It is an object design pattern that is popularly used in test automation for a better testing experience. In this technique, a Page class is created for each web page of the application. This Page class contains web elements and methods for action to be performed on these web elements.

1. We have made 2 separate file. One for the components and test to run in the Home page of the HSBC, and the second for the components and test to run in the Log On page of HSBC website.
2. **Home.ts ->** Contains the components, functions and checkpoints and test to run on the Home page.
3. **LogOn.ts ->** Contains the components, functions and checkpoints and test to run on the Log On page of the HSBC website.

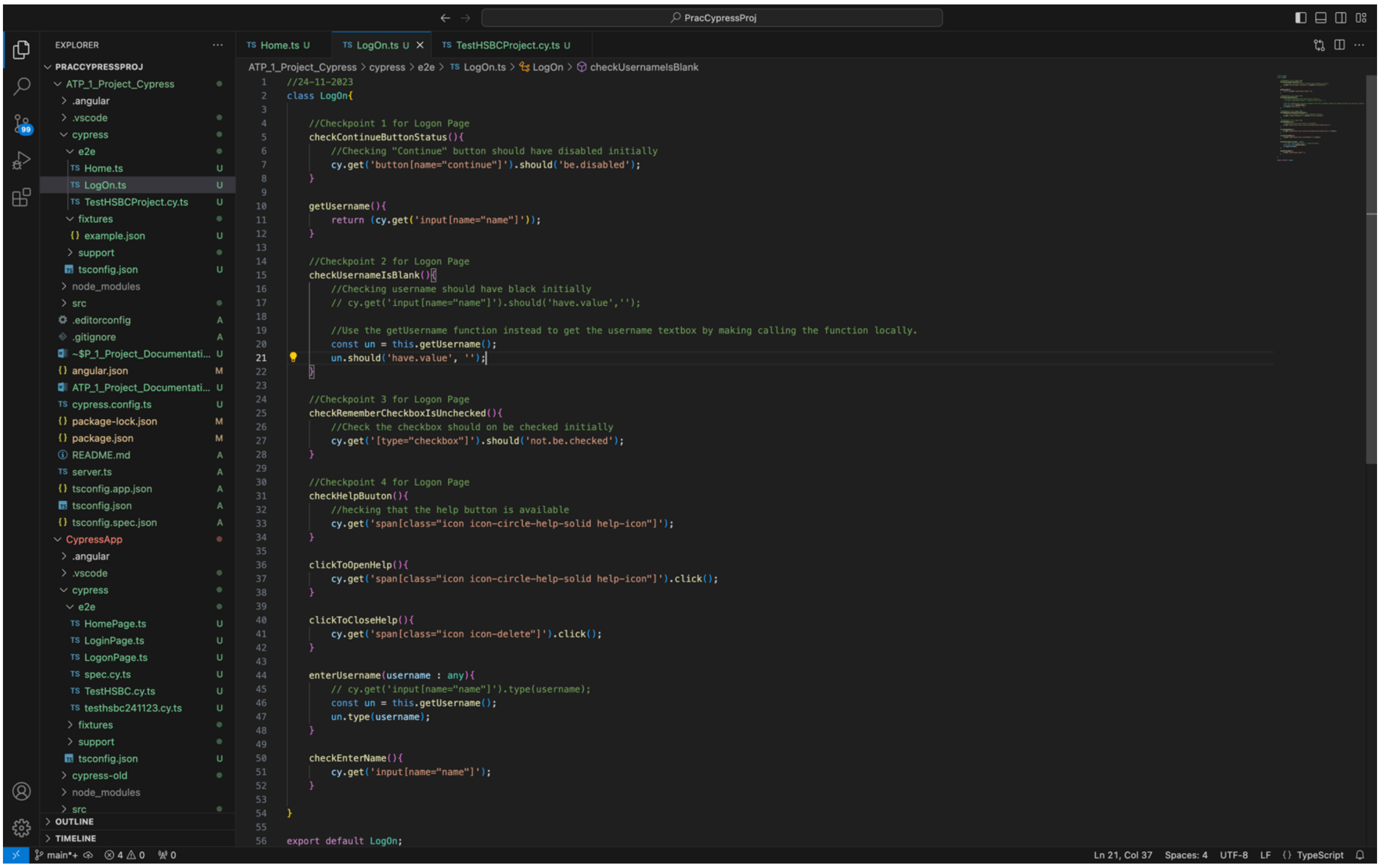
- **Screenshots**

1. Home.ts file (POM File 1 for Home)



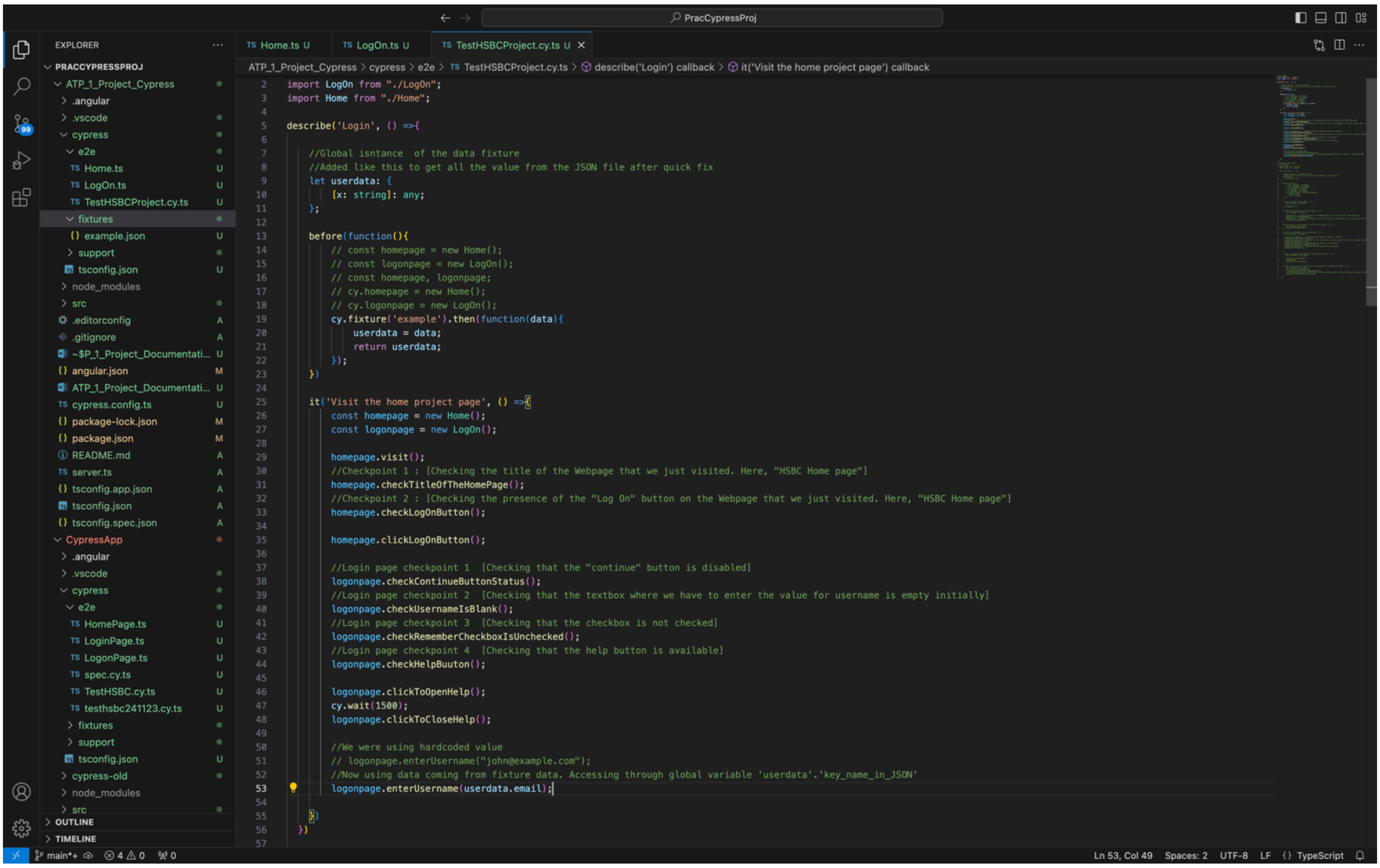
```
//24-11-2023
class Home{
    visit(){
        cy.visit('https://www.hsbc.co.in');
        //Log On button should be visible [Checkpoint]
    }

    //Home page checkpoint 1 [Checking for Title of the Page]
    checkTitleOfTheHomePage(){
        //Checking "Log On" button should exist on the home page
        cy.get('title').contains('HSBC India - Credit Cards, NRI Services, Saving and Deposit')
        return true;
    }

    //Home page checkpoint 2 [Checking for Log On button]
    checkLogOnButton(){
        //Checking "Log On" button should exist on the home page
        cy.get('.selected-item').contains(" Log On ");
        return true;
    }

    clickLogOnButton(){
        cy.get('.selected-item').contains(" Log On ").click({force: true});
    }

}

export default Home;
```

2. LogOn.ts file (POM File 2 for Log On)



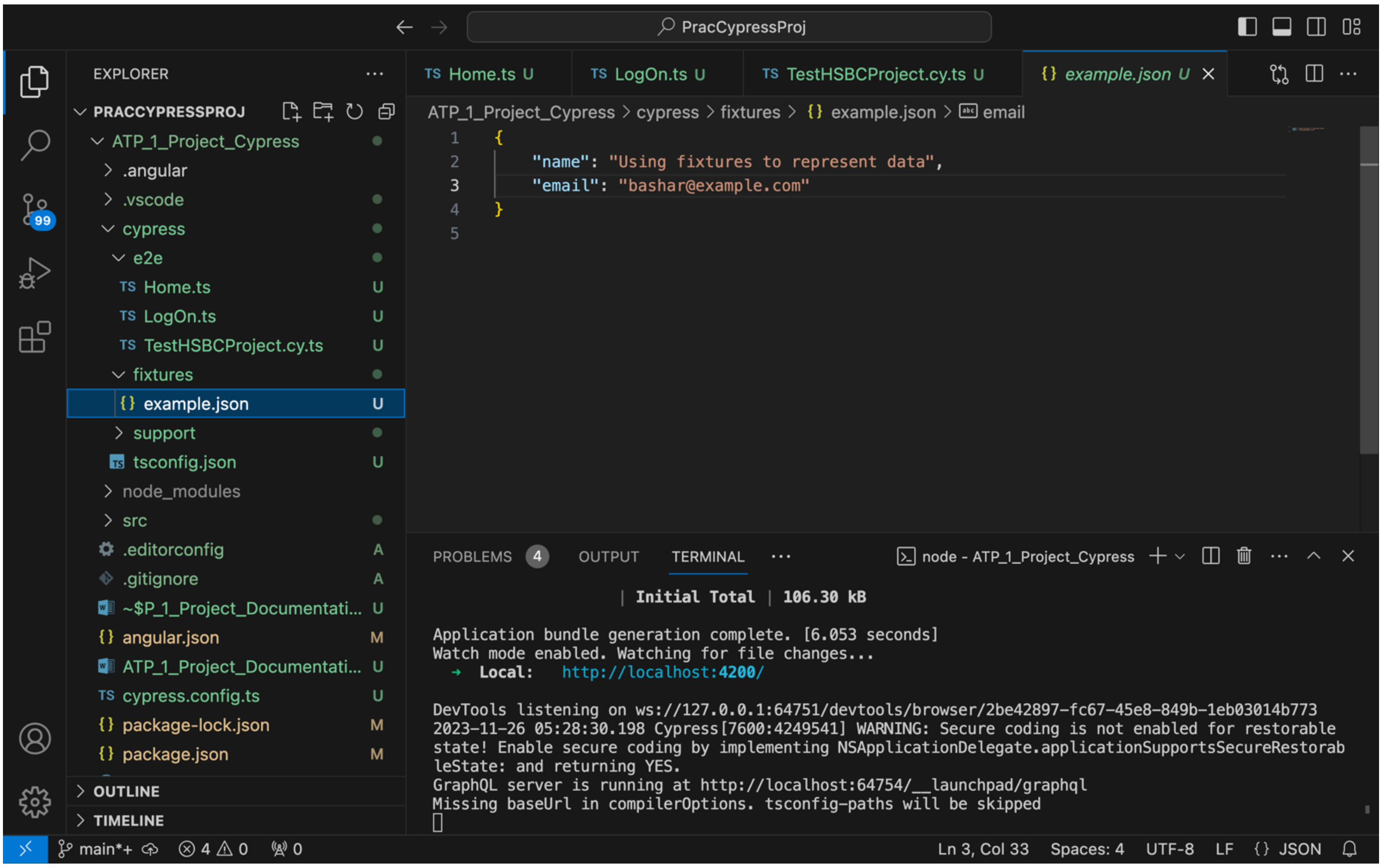
```
//24-11-2023
class LogOn{

    //Checkpoint 1 for Logon Page
    checkContinueButtonStatus(){
        //Checking "Continue" button should have disabled initially
        cy.get('button[name="continue"]').should('be.disabled');
    }

    getUsername(){
        return (cy.get('input[name="name"]'));
    }

    //Checkpoint 2 for Logon Page
    checkUsernameIsBlank(){
        //Checking username should have black initially
        // cy.get('input[name="name"]').should('have.value','');

        //Use the getUsername function instead to get the username textbox by making calling the function locally.
        const un = this.getUsername();
        un.should('have.value', '');
    }

    //Checkpoint 3 for Logon Page
    checkRememberCheckboxIsUnchecked(){
        //Check the checkbox should on be checked initially
        cy.get('[type="checkbox"]').should('not.be.checked');
    }

    //Checkpoint 4 for Logon Page
    checkHelpBuuton(){
        //hecking that the help button is available
        cy.get('span[class="icon icon-circle-help-solid help-icon"]');
    }

    clickToOpenHelp(){
        cy.get('span[class="icon icon-circle-help-solid help-icon"]').click();
    }

    clickToCloseHelp(){
        cy.get('span[class="icon icon-delete"]').click();
    }

    enterUsername(username : any){
        // cy.get('input[name="name"]').type(username);
        const un = this.getUsername();
        un.type(username);
    }

    checkEnterName(){
        cy.get('input[name="name"]');
    }

}

export default LogOn;
```

3. TestHSBCProject.cy.ts file



```typescript
import LogOn from "./LogOn";
import Home from "./Home";

describe('Login', () =>{

  //Global isntance  of the data fixture
  //Added like this to get all the value from the JSON file after quick fix
  let userdata: {
    [x: string]: any;
  };

  before(function(){
    // const homepage = new Home();
    // const logonpage = new LogOn();
    // const homepage, logonpage;
    // cy.homepage = new Home();
    // cy.logonpage = new LogOn();
    cy.fixture('example').then(function(data){
      userdata = data;
      return userdata;
    });
  })

  it('Visit the home project page', () =>{
    const homepage = new Home();
    const logonpage = new LogOn();

    homepage.visit();
    //Checkpoint 1 : [Checking the title of the Webpage that we just visited. Here, "HSBC Home page"]
    homepage.checkTitleOfTheHomePage();
    //Checkpoint 2 : [Checking the presence of the "Log On" button on the Webpage that we just visited. Here, "HSBC Home page"]
    homepage.checkLogOnButton();

    homepage.clickLogOnButton();

    //Login page checkpoint 1  [Checking that the "continue" button is disabled]
    logonpage.checkContinueButtonStatus();
    //Login page checkpoint 2  [Checking that the textbox where we have to enter the value for username is empty initially]
    logonpage.checkUsernameIsBlank();
    //Login page checkpoint 3  [Checking that the checkbox is not checked]
    logonpage.checkRememberCheckboxIsUnchecked();
    //Login page checkpoint 4  [Checking that the help button is available]
    logonpage.checkHelpBuuton();

    logonpage.clickToOpenHelp();
    cy.wait(1500);
    logonpage.clickToCloseHelp();

    //We were using hardcoded value
    // logonpage.enterUsername("john@example.com");
    //Now using data coming from fixture data. Accessing through global variable 'userdata'.'key_name_in_JSON'
    logonpage.enterUsername(userdata.email);

  })
})
```

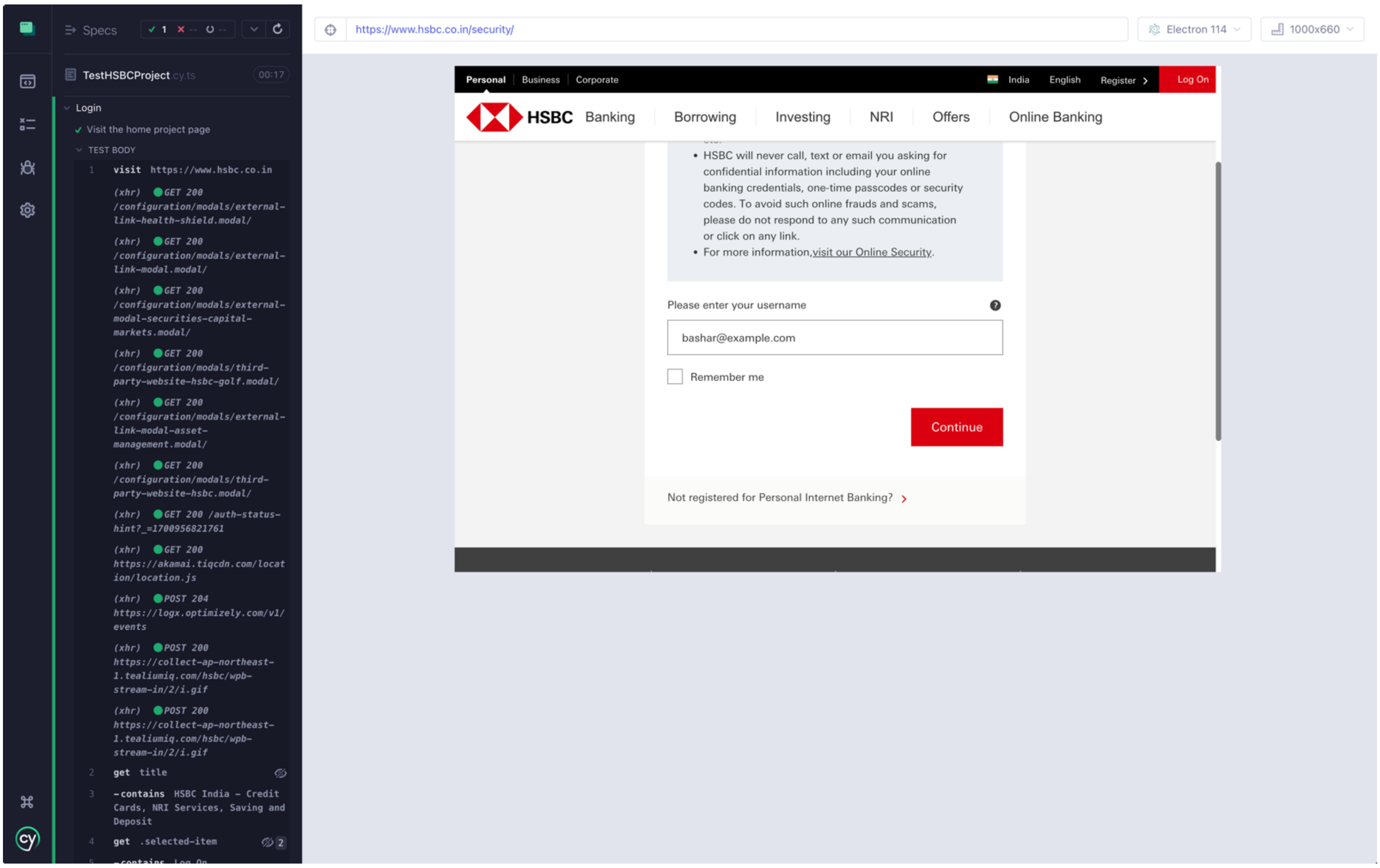
4. Example.json (FIXTURE File)



```json
{
    "name": "Using fixtures to represent data",
    "email": "bashar@example.com"
}
```

```
                | Initial Total | 106.30 kB

Application bundle generation complete. [6.053 seconds]
Watch mode enabled. Watching for file changes...
  ➜  Local:   http://localhost:4200/

DevTools listening on ws://127.0.0.1:64751/devtools/browser/2be42897-fc67-45e8-849b-1eb03014b773
2023-11-26 05:28:30.198 Cypress[7600:4249541] WARNING: Secure coding is not enabled for restorable state! Enable secure coding by implementing NSApplicationDelegate.applicationSupportsSecureRestorableState: and returning YES.
GraphQL server is running at http://localhost:64754/__launchpad/graphql
Missing baseUrl in compilerOptions. tsconfig-paths will be skipped
```

5. OUTPUT



TestHSBCProject.cy.ts 00:17

Login

✓ Visit the home project page

TEST BODY

```
1  visit  https://www.hsbc.co.in
   (xhr)  ●GET 200
   /configuration/modals/external-
   link-health-shield.modal/
   (xhr)  ●GET 200
   /configuration/modals/external-
   link-modal.modal/
   (xhr)  ●GET 200
   /configuration/modals/external-
   modal-securities-capital-
   markets.modal/
   (xhr)  ●GET 200
   /configuration/modals/third-
   party-website-hsbc-golf.modal/
   (xhr)  ●GET 200
   /configuration/modals/external-
   link-modal-asset-
   management.modal/
   (xhr)  ●GET 200
   /configuration/modals/third-
   party-website-hsbc.modal/
   (xhr)  ●GET 200 /auth-status-
   hint?_=1700956821761
   (xhr)  ●GET 200
   https://akamai.tiqcdn.com/locat
   ion/location.js
   (xhr)  ●POST 204
   https://logx.optimizely.com/v1/
   events
   (xhr)  ●POST 200
   https://collect-ap-northeast-
   1.tealiumiq.com/hsbc/wpb-
   stream-in/2/i.gif
   (xhr)  ●POST 200
   https://collect-ap-northeast-
   1.tealiumiq.com/hsbc/wpb-
   stream-in/2/i.gif
2  get  title
3  -contains  HSBC India - Credit
   Cards, NRI Services, Saving and
   Deposit
4  get  .selected-item  2
5  -contains  Log On
```

https://www.hsbc.co.in/security/

- HSBC will never call, text or email you asking for confidential information including your online banking credentials, one-time passcodes or security codes. To avoid such online frauds and scams, please do not respond to any such communication or click on any link.
- For more information, visit our Online Security.

Please enter your username

bashar@example.com

□ Remember me

Continue

Not registered for Personal Internet Banking? ›