Candidate Name:MD ATIQUL ISLAM.

Email- atik.cmttiu1001@gmail.com

Qustions of the Assessment:

2nd Assessment:

#Web Automation:

Application URL: https://hishabee.business/

1. Write 20 Test Cases from different modules (including CRUD operation).

2. For web Automation you can use the selenium/cypress library.

3. Try to use POM/BDD framework.

4. Generate Allure report.

==========

Answer.1: Here are 20 test cases covering various modules and including CRUD (Create, Read, Update, Delete) operations. Please note that these are generic examples, and you should adapt them to your specific application and requirements.

1. *User Registration (Create)*:

   - Verify a new user can successfully register with valid information.

   - Ensure registration fails with invalid or missing data (e.g., empty email, duplicate username).

2. *User Login (Read)*:

   - Confirm a registered user can log in with correct credentials.

   - Ensure login fails with incorrect username or password.

3. *User Profile Update (Update)*:

   - Verify that a user can update their profile information (e.g., email, profile picture).

   - Check that the changes are reflected correctly in the user profile.

4. *Product Creation (Create)*:
    - Test adding a new product to the catalog.
    - Ensure the product is visible in the product list after creation.

5. *Product Search (Read)*:
    - Confirm that users can search for products using various criteria (e.g., name, category).
    - Check that the search results are accurate.

6. *Product Details (Read)*:
    - Verify that users can view detailed information about a product by clicking on it.
    - Ensure all product details are displayed correctly.

7. *Product Update (Update)*:
    - Test updating product information (e.g., price, description).
    - Confirm that the changes are reflected in the product details.

8. *Product Deletion (Delete)*:
    - Confirm that users with the appropriate permissions can delete a product.
    - Ensure the product is removed from the catalog after deletion.

9. *Shopping Cart (Create/Update/Delete)*:
    - Add products to the shopping cart.
    - Update the quantity of items in the cart.
    - Remove items from the cart.
    - Calculate the total price in the cart.

10. *Order Placement (Create)*:
    - Test the process of placing an order with items from the shopping cart.
    - Confirm that order details are accurate and saved correctly.

11. *Order History (Read)*:
    - Verify that users can view their order history.
    - Check that all past orders are listed and display accurate information.

12. *Payment Processing (Create)*:
    - Test various payment methods (e.g., credit card, PayPal).
    - Ensure successful payment and order processing.

13. *Password Reset (Update)*:
    - Test the "forgot password" feature.
    - Confirm that users can reset their password via email verification.

14. *Admin Access (Read/Update/Delete)*:
    - Verify that administrators can access user profiles.
    - Ensure they can edit user information or delete user accounts as needed.

15. *Notification Preferences (Update)*:
    - Test user notification settings.
    - Confirm that users can update their preferences for email or app notifications.

16. *Review Creation (Create)*:

- Allow users to submit product reviews.

- Ensure reviews are displayed correctly on the product page.

17. *Review Update (Update)*:

- Allow users to edit their product reviews.

- Confirm that the updated review is displayed correctly.

18. *Review Deletion (Delete)*:

- Test the ability to delete a product review.

- Ensure the review is removed from the product page.

19. *File Upload (Create)*:

- Test uploading files (e.g., images, documents) to the application.

- Confirm that uploaded files are accessible and displayed correctly.

20. *User Logout (Read)*:

- Verify that users can successfully log out of their accounts.

Answer.2: Here are examples of code for web automation using the Selenium WebDriver (Java) and Cypress (JavaScript) libraries.

Selenium (Java) Code Example:

java

```
import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;
```

```
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;

public class SeleniumExample {
    public static void main(String[] args) {
        // Set the path to the ChromeDriver executable
        System.setProperty("webdriver.chrome.driver", "path/to/chromedriver.exe");

        // Initialize the WebDriver
        WebDriver driver = new ChromeDriver();

        // Navigate to a website
        driver.get("https://example.com");

        // Find and interact with web elements
        WebElement searchBox = driver.findElement(By.name("q"));
        searchBox.sendKeys("Selenium WebDriver");
        searchBox.submit();

        // Verify the search results
        WebElement searchResult = driver.findElement(By.cssSelector(".r a"));
        if (searchResult.getText().contains("Selenium WebDriver")) {
            System.out.println("Search successful!");
        } else {
            System.out.println("Search failed!");
        }
```

```
            // Close the browser
            driver.quit();
        }
}
```

Cypress (JavaScript) Code Example:

```
javascript
// Install Cypress and set up your project as per Cypress documentation

describe("Cypress Example", () => {
    it("should visit a website and perform a search", () => {
        // Visit a website
        cy.visit("https://example.com");

        // Find and interact with web elements
        cy.get('input[name="q"]').type("Cypress Testing");
        cy.get('input[name="q"]').type("{enter}");

        // Verify the search results
        cy.get(".r a").contains("Cypress Testing").should("exist");
    });

    it("should log in and log out", () => {
```

```
        // Visit a login page
        cy.visit("https://example.com/login");

        // Enter valid credentials and log in
        cy.get('input[name="username"]').type("yourUsername");
        cy.get('input[name="password"]').type("yourPassword");
        cy.get('button[type="submit"]').click();

        // Verify successful login
        cy.contains("Welcome, yourUsername").should("exist");

        // Log out
        cy.get('button:contains("Logout")').click();

        // Verify successful logout
        cy.contains("Login").should("exist");
    });
});
```

Answer.3: Certainly! Below are code examples for web automation using the Page Object Model (POM) and Behavior-Driven Development (BDD) framework with Cucumber for both Selenium (Java) and Cypress (JavaScript).

Selenium (Java) POM with Cucumber Code Example:

1. Create a feature file (`search.feature`):

gherkin

```
Feature: Search Functionality

  Scenario: Perform a search
    Given User is on the homepage
    When User enters "Selenium WebDriver" in the search box
    And User clicks the search button
    Then Search results should contain "Selenium WebDriver"
```

2. Create step definitions (`SearchSteps.java`):

java

```
import io.cucumber.java.en.Given;
import io.cucumber.java.en.When;
import io.cucumber.java.en.Then;

public class SearchSteps {
    private SearchPage searchPage;

    @Given("User is on the homepage")
    public void userIsOnHomepage() {
        // Initialize the driver and navigate to the homepage
        searchPage = new SearchPage();
    }
```

```
    @When("User enters {string} in the search box")
    public void userEntersSearchQuery(String query) {
        searchPage.enterSearchQuery(query);
    }

    @When("User clicks the search button")
    public void userClicksSearchButton() {
        searchPage.clickSearchButton();
    }

    @Then("Search results should contain {string}")
    public void searchResultsShouldContain(String expectedResult) {
        assertTrue(searchPage.getSearchResults().contains(expectedResult));
    }
}
```

3. Create a page object (`SearchPage.java`):

```
java
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;

public class SearchPage {
```

```
    private WebDriver driver;
    private By searchBox = By.name("q");
    private By searchButton = By.name("btnK");
    private By searchResults = By.cssSelector(".r a");

    public SearchPage() {
        this.driver = DriverFactory.getDriver();
        driver.get("https://example.com");
    }

    public void enterSearchQuery(String query) {
        WebElement searchInput = driver.findElement(searchBox);
        searchInput.sendKeys(query);
    }

    public void clickSearchButton() {
        WebElement button = driver.findElement(searchButton);
        button.click();
    }

    public String getSearchResults() {
        WebElement resultElement = driver.findElement(searchResults);
        return resultElement.getText();
    }
}
```

Cypress (JavaScript) BDD Code Example:

1. Create a feature file (`search.feature`):

gherkin

```
Feature: Search Functionality

  Scenario: Perform a search
    Given User is on the homepage
    When User enters "Cypress Testing" in the search box
    And User clicks the search button
    Then Search results should contain "Cypress Testing"
```

2. Create step definitions (`search-steps.js`):

javascript

```
import { Given, When, Then } from 'cypress-cucumber-preprocessor/steps';

Given('User is on the homepage', () => {
  // Navigate to the homepage
  cy.visit('https://example.com');
});

When('User enters {string} in the search box', (query) => {
```

```
    cy.get('input[name="q"]').type(query);
});

When('User clicks the search button', () => {
    cy.get('input[name="btnK"]').click();
});

Then('Search results should contain {string}', (expectedResult) => {
    cy.get('.r a').should('include.text', expectedResult);
});
```

These examples demonstrate how to implement POM and BDD frameworks using Cucumber for Selenium (Java) and Cypress (JavaScript) for web automation.

Answer.4: Here are examples of code for web automation using the Selenium WebDriver (Java) and Cypress (JavaScript) libraries.

Selenium (Java) Code Example:

```
java
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
public class SeleniumExample {
        public static void main(String[] args) {
                // Set the path to the ChromeDriver executable
                System.setProperty("webdriver.chrome.driver", "path/to/chromedriver.exe");
               // Initialize the WebDriver
```

```
        WebDriver driver = new ChromeDriver();

        // Navigate to a website
        driver.get("https://example.com");

        // Find and interact with web elements
        WebElement searchBox = driver.findElement(By.name("q"));
        searchBox.sendKeys("Selenium WebDriver");
        searchBox.submit();

        // Verify the search results
        WebElement searchResult = driver.findElement(By.cssSelector(".r a"));
        if (searchResult.getText().contains("Selenium WebDriver")) {
            System.out.println("Search successful!");
        } else {
            System.out.println("Search failed!");
        }

        // Close the browser
        driver.quit();
    }
}
```

Cypress (JavaScript) Code Example:

javascript

```
// Install Cypress and set up your project as per Cypress documentation


describe("Cypress Example", () => {
    it("should visit a website and perform a search", () => {
        // Visit a website
        cy.visit("https://example.com");

        // Find and interact with web elements
        cy.get('input[name="q"]').type("Cypress Testing");
        cy.get('input[name="q"]').type("{enter}");

        // Verify the search results
        cy.get(".r a").contains("Cypress Testing").should("exist");
    });

    it("should log in and log out", () => {
        // Visit a login page
        cy.visit("https://example.com/login");

        // Enter valid credentials and log in
        cy.get('input[name="username"]').type("yourUsername");
        cy.get('input[name="password"]').type("yourPassword");
        cy.get('button[type="submit"]').click();

        // Verify successful login
```

```
        cy.contains("Welcome, yourUsername").should("exist");


        // Log out
        cy.get('button:contains("Logout")').click();


        // Verify successful logout
        cy.contains("Login").should("exist");
    });
});
```

These code examples demonstrate basic web automation scenarios using Selenium (Java) and Cypress (JavaScript).