

Test Cases Document for Test Case Generation Application

Test Case 1: Code Input Functionality

1. Test Scenario: Verify that a user can input code correctly into the text area.

- **Test Steps:**
 - Open the application.
 - Locate the code input area (a large text area labeled "TYPE CODE HERE").
 - Type a valid code snippet into the text area.
- **Expected Result:**
 - The code is successfully entered into the text area, and no errors are displayed.

2. Test Scenario: Verify that an error message is displayed if the code text area is left empty.

- **Test Steps:**
 - Open the application.
 - Leave the code input text area empty.
 - Attempt to submit the form by clicking on the "Generate" button.
- **Expected Result:**
 - An error message should appear under the code input area saying, "Code snippet cannot be empty."

Test Case 2: Validation of Test Cases Inputs (Total, Positive, Negative)

1. Test Scenario: Verify that an error message is displayed if the total number of test cases is negative.

a. Test Steps:

- i. Open the application.
- ii. Enter a negative value in the "Total Test Cases" input field.
- iii. Enter valid values in the "Positive Test Cases" and "Negative Test Cases" input fields.

b. Expected Result:

- i. An error message should appear under the "Total Test Cases" input saying, "Total test cases cannot be negative."

2. Test Scenario: Verify that the sum of positive and negative test cases equals the total number of test cases.

a. Test Steps:

- i. Open the application.
- ii. Enter a total number of test cases (e.g., 10).
- iii. Enter 4 positive test cases and 7 negative test cases.
- iv. Click the "Generate" button.

b. Expected Result:

- i. If the sum of positive and negative test cases does not match the total, an error message should appear under the "Total Test Cases" input saying, "Sum of positive and negative test cases must equal total test cases."

3. Test Scenario: Verify that an error message is displayed if the number of negative test cases exceeds the total number of test cases.

a. Test Steps:

- i. Open the application.
- ii. Enter 5 test cases in the "Total Test Cases" field.
- iii. Enter 6 negative test cases in the "Negative Test Cases" field.
- iv. Enter 0 in the "Positive Test Cases" field.

b. Expected Result:

- i. An error message should appear under the "Negative Test Cases" input saying, "Negative test cases cannot exceed total test cases."

Test Case 3: Language Selection

- 1. Test Scenario: Verify that a user can select a programming language from the provided options.**
 - a. Test Steps:**
 - i. Open the application.
 - ii. Select one of the available languages (e.g., Java).
 - b. Expected Result:**
 - i. The selected language is correctly displayed in the input section.
- 2. Test Scenario: Verify that selecting a language triggers a re-validation of the inputs.**
 - a. Test Steps:**
 - i. Open the application.
 - ii. Enter a total number of test cases, positive, and negative test cases.
 - iii. Select a language (e.g., Python).
 - iv. Change the total or test case values.
 - b. Expected Result:**
 - i. The inputs are re-validated after the language is changed, and any new errors are displayed accordingly.

Test Case 4: Generate Test Cases Button

- 1. Test Scenario: Verify that the "Generate" button is disabled if any of the input fields are invalid.**
 - a. Test Steps:**
 - i. Open the application.

- ii. Enter invalid values in any of the required fields (e.g., leave the code field empty or input a negative number in "Total Test Cases").
 - iii. Observe the state of the "Generate" button.
- b. Expected Result:**
 - i. The "Generate" button should remain disabled until all input fields are valid.
- 2. Test Scenario: Verify that clicking the "Generate" button with valid inputs triggers the result.**
 - a. Test Steps:**
 - i. Open the application.
 - ii. Enter a valid code snippet.
 - iii. Enter valid values for "Total Test Cases," "Positive Test Cases," and "Negative Test Cases."
 - iv. Select a programming language.
 - v. Click the "Generate" button.
 - b. Expected Result:**
 - i. The test case generation process should begin, and the result (either a success message or generated test cases) should be displayed in the output section.
- 3. Test Scenario: Verify that an error message is shown if the server cannot process the request.**
 - a. Test Steps:**
 - i. Open the application.
 - ii. Enter a valid code snippet and valid values for "Total Test Cases," "Positive Test Cases," and "Negative Test Cases."
 - iii. Click the "Generate" button while the server is down or unreachable.
 - b. Expected Result:**
 - i. An error message should appear in the output section saying, "Unable to load server."

Test Case 5: Copy Output Result

- 1. Test Scenario: Verify that the user can copy the result to the clipboard.**

- a. Test Steps:**
 - i. After generating the test cases, locate the "Copy" button in the output section.
 - ii. Click the "Copy" button.
 - b. Expected Result:**
 - i. The result is successfully copied to the clipboard, and a message like "copied" should briefly appear next to the button.
- 2. Test Scenario: Verify that clicking the "Copy" button shows a brief "Copied" notification.**
 - a. Test Steps:**
 - i. Generate a result by entering valid inputs and clicking "Generate."
 - ii. Click the "Copy" button.
 - b. Expected Result:**
 - i. A "copied" message should briefly appear and then disappear after 2 seconds.

Test Case 6: Error Handling and Edge Cases

- 1. Test Scenario: Verify that an appropriate error message is shown when a non-code input is entered.**
 - a. Test Steps:**
 - i. Enter a non-code input (e.g., plain text or an invalid code snippet).
 - ii. Attempt to generate the test cases.
 - b. Expected Result:**
 - i. An error message should appear saying, "Irrelevant code cannot generate test cases."
- 2. Test Scenario: Verify that an error message is displayed for unexpected server errors.**
 - a. Test Steps:**
 - i. Simulate a server error (e.g., by stopping the backend server).
 - ii. Attempt to generate test cases.
 - b. Expected Result:**

- i. An error message should appear saying, "Unable to load server."

Conclusion

The above test cases cover various aspects of functionality for the code input and test case generation application, including input validation, error handling, language selection, and the generation of unit test cases. These tests ensure that the application is robust and user-friendly across different scenarios.