Test Cases Document for Unit Test Case Generation

Test Case 1: Input Component - Code Input Handling

- **Scenario:** Ensure InputComponent correctly captures code input.
  - **Steps:**
    - Enter code in the "TYPE YOUR CODE HERE" textarea.
    - Select a language (Java, Python, C#, JavaScript, Ruby).
    - Provide values for Total, Positive, and Negative test cases.
  - **Expected Result:** Input data is captured and passed to the OutputComponent.

Test Case 2: Output Component - Code Validation

- **Scenario:** Validate that OutputComponent checks the code's validity.
  - **Steps:**
    - Enter valid code in the "TYPE YOUR CODE HERE" textarea.
    - Click the "Generate Test Cases" button.
  - **Expected Result:** If the code is valid, proceed with test case generation; otherwise, display "IRRELEVANT CODE, CANNOT GENERATE TEST CASES."

Test Case 3: Output Component - Generate Unit Test Cases

- **Scenario:** Ensure OutputComponent generates test cases based on user input.
  - **Steps:**
    - Enter valid code, language, and test case values.
    - Click the "Generate Test Cases" button.
  - **Expected Result:** The correct number of positive and negative test cases is generated in the selected language.

Test Case 4: Output Component - Error Handling

- **Scenario:** Handle errors when input is invalid.
  - **Steps:**
    - Leave the code field empty or provide invalid input.
    - Click the "Generate Test Cases" button.
  - **Expected Result:** Display an error message such as "Code snippet cannot be empty" or "Invalid input."

Test Case 5: Output Component - Copy Functionality

- **Scenario:** Verify that the "Copy" button copies generated test cases.
    - **Steps:**
        - Generate test cases.
        - Click the "Copy" button next to the result.
    - **Expected Result:** The result is copied to the clipboard, and the button text changes to "copied" for 2 seconds.

Test Case 6: Input Component - Validation of Test Case Numbers

- **Scenario:** Ensure that test case numbers are validated (e.g., positive, negative, total).
    - **Steps:**
        - Enter invalid values (e.g., negative numbers for positive or negative test cases).
        - Check if the validation messages are displayed.
    - **Expected Result:** Appropriate validation messages are shown, e.g., "Total test cases cannot be negative."

Test Case 7: Output Component - Scenario-Based Test Case Generation

- **Scenario:** Verify that test cases are generated based on specific scenarios.
    - **Steps:**
        - Enter valid code, language, and test case details.
        - Click "Generate Test Cases."
    - **Expected Result:** The test cases should match the specified scenario.

Conclusion:

This test case document covers key functionalities such as **input handling**, **code validation**, **test case generation**, **error handling**, and **copy functionality**. These scenarios ensure the correct operation of both the **InputComponent** and **OutputComponent** for generating unit tests across multiple programming languages.