# Functional Testing

**Test Case 1:** Add New Post

**Objective:** Check that the meta box in the backend is loading, and the plugin renders correctly on the front-end.

**Steps:**

1. Navigate to WordPress dashboard > Post > New Post.
2. Verify the presence of the plugin's meta box in the backend.
3. Add a title and publish the page.
4. Check for the presence of the voting plugin on the front-end.

**Expected Result:** A new meta box should appear in the backend sidebar of the post, and the voting plugin should automatically appear on the front-end.

—------------------------------------------------------------------------------------------------------------------------

**Test Case 2:** Vote Yes

**Objective:** Verify that the front-end displays a highlighted "Yes" button, the text changes, and the results show 100% for "Yes" and 0% for "No."

**Steps:**

1. Visit the post on the front-end.
2. Click the "Yes" button.
3. Check the backend for updated results.

**Expected Result:** The votes should display 100% for "Yes" and 0% for "No." The "Yes" button on the front-end should be highlighted, and the text should change to a thank-you message.

—------------------------------------------------------------------------------------------------------------------------

**Test Case 3:** Refresh Page

**Objective:** Confirm that the front-end still reflects the previous vote after a page refresh.

**Steps:**

1. Visit the post on the front-end.
2. Click the "Yes" button.
3. Refresh the page.

**Expected Result:** The votes should still show 100% for "Yes" and 0% for "No." The "Yes" button on the front-end should remain highlighted, and the text should continue displaying the thank-you message.

—------------------------------------------------------------------------------------------------------------------------

**Test Case 4:** Check IP Check is Working

**Objective:** Verify that the front-end displays as if the voting was never done when the IP is edited in the database.

**Steps:**

1. Access the MySQL database and locate the wp\_post\_meta table.
2. Find the vote entry and edit the IP.
3. Refresh the front-end.

**Expected Result:** The plugin should display the original question with "Yes" and "No" buttons as if the voting was never performed.

—------------------------------------------------------------------------------------------------------------------------

**Test Case 5:** Vote No

**Objective:** Confirm that the front-end displays a highlighted "No" button, the text changes, and the results show 50% for "Yes" and 50% for "No."

**Steps:**

1. Visit the post on the front-end.
2. Click the "No" button.
3. Check the backend for updated results.

**Expected Result:** The votes should display 50% for "Yes" and 50% for "No." The "No" button on the front-end should be highlighted, and the text should change to a thank-you message.

—------------------------------------------------------------------------------------------------------------------------

**Test Case 6:** Change Value of vote “yes” to “yews”

**Objective:** Votes change from 50% and 50% to 0% and 100%

**Steps:**

1. Find the vote in mySQL
2. Edit the “yes” to “yews”
3. Check the frontend and backend for updated results.

**Expected Result:** The votes should display 0% for "Yes" and 100% for "No".

—------------------------------------------------------------------------------------------------------------------------

**Test Case 6:** Add New Page

**Objective:** Check that the meta box in the backend is missing, and the frontend doesnt have the plugin

**Steps:**

1. Navigate to WordPress dashboard > Page > New Page.
2. Verify the plugin's meta box in the backend is not there.
3. Add a title and publish the page.
4. Check that the voting plugin on the front-end is also missing.

**Expected Result:** The plugin meta box and front end should not be rendering.

—------------------------------------------------------------------------------------------------------------------------

**Test Case 7:** Proxy Detection

**Objective:** Ensure that the plugin detects proxy connections.

**Steps:**

1. Visit a post through a proxy

**Expected Result:** Front end plugin should not be rendering, and instead a message is shown.

—------------------------------------------------------------------------------------------------------------------------

**Test Case 8:** Browser Compatibility

**Objective:** Check if the plugin functions consistently across different browsers.

**Steps:**

1. Test the plugin on popular browsers (Chrome, Firefox, Safari, Edge).

**Expected Result:** The plugin should work without issues on all tested browsers.

—------------------------------------------------------------------------------------------------------------------------

**Test Case 9:** Ajax Requests

**Objective:** Ensure that Ajax requests are made and handled correctly.

**Steps:**

1. Vote on a post
2. Inspect network requests using browser developer tools

**Expected Result:** Successful Ajax requests should be visible in the network tab.