**Testing Story Creation and Completion Guidelines**

To maintain a consistent and effective testing workflow, please follow these guidelines when creating and completing testing tasks in our Kanban system.

**Testing Story Creation Format**

When creating a testing story, include the following components:

**1. Summary**

* Write a concise and descriptive title that specifies the test being performed.
  + Examples: "Regression Test for Login Module" or "Performance Test for API Endpoint."

**2. Description**

* Include the following details:
  + Explanation of what this test is testing if the summary isn’t enough.
  + **Test Type:** Specify the type of test (e.g., Unit Test, Integration Test, Regression Test, Performance Test).
  + **Test Criteria:** Describe the conditions for success. (For **manual** tests this should be a step-by-step list of actions the tester should take.)
  + **Dependencies:** Note any pre-requisites or related tasks.

**3. Labels and Tags**

* Use relevant labels to categorize the test.
  + Examples: regression, unit-test, performance, automation

**4. Priority**

* Assign a priority level based on the impact and urgency of the test (e.g., High, Medium, Low).

**5. Assignee**

* Assign the task to the responsible tester or leave it unassigned.

**6. Attachments**

* Include test scripts, test cases, or other supporting documentation as needed.

**Testing Story Completion Steps**

**1. Prepare for Testing**

* Review the test objective, scope, and criteria to ensure clarity.
* Set up the required testing environment, including tools, data, and configurations.

**2. Execute the Test**

* Perform the test according to the defined test criteria and document the results.
  + For automated tests, ensure logs and results are saved.
  + For manual tests, document steps and outcomes.

**3. Capture Evidence**

* Record evidence of the test execution, such as:
  + Screenshots of results.
  + Videos of workflows.
  + Logs from automated tests.

**4. Document Findings**

* Update the task with:
  + Test results (e.g., Passed, Failed, Blocked).
  + Details of any issues or bugs found, with links to corresponding Jira tickets.
  + Observations or suggestions for improvement.

**5. Move Story to "Done" or Notify Stakeholders**

* If the test meets its objectives and criteria, move the story to "Done."
* If issues are found, escalate them to the relevant team or stakeholders.
  + If the test failed, determine if it is an issue with the test or the code. (Verify the functionality by sending a message in testing or dev chat if unsure.)
  + If the test is the issue, set the test to **Broken**.
  + If the code is the issue, look for an existing bug for the issue (message chat if unable to find something.)
  + If an existing bug is found, link it to the test.
  + If the bug is new (it should be reproducible), create a bug ticket.

**6. Communicate Results**

* Share a message in the team chat that includes:
  + A link to the Jira story.
  + Example:

"Regression Test completed for Login Module. [Jira Story Link] All tests passed. Logs attached."