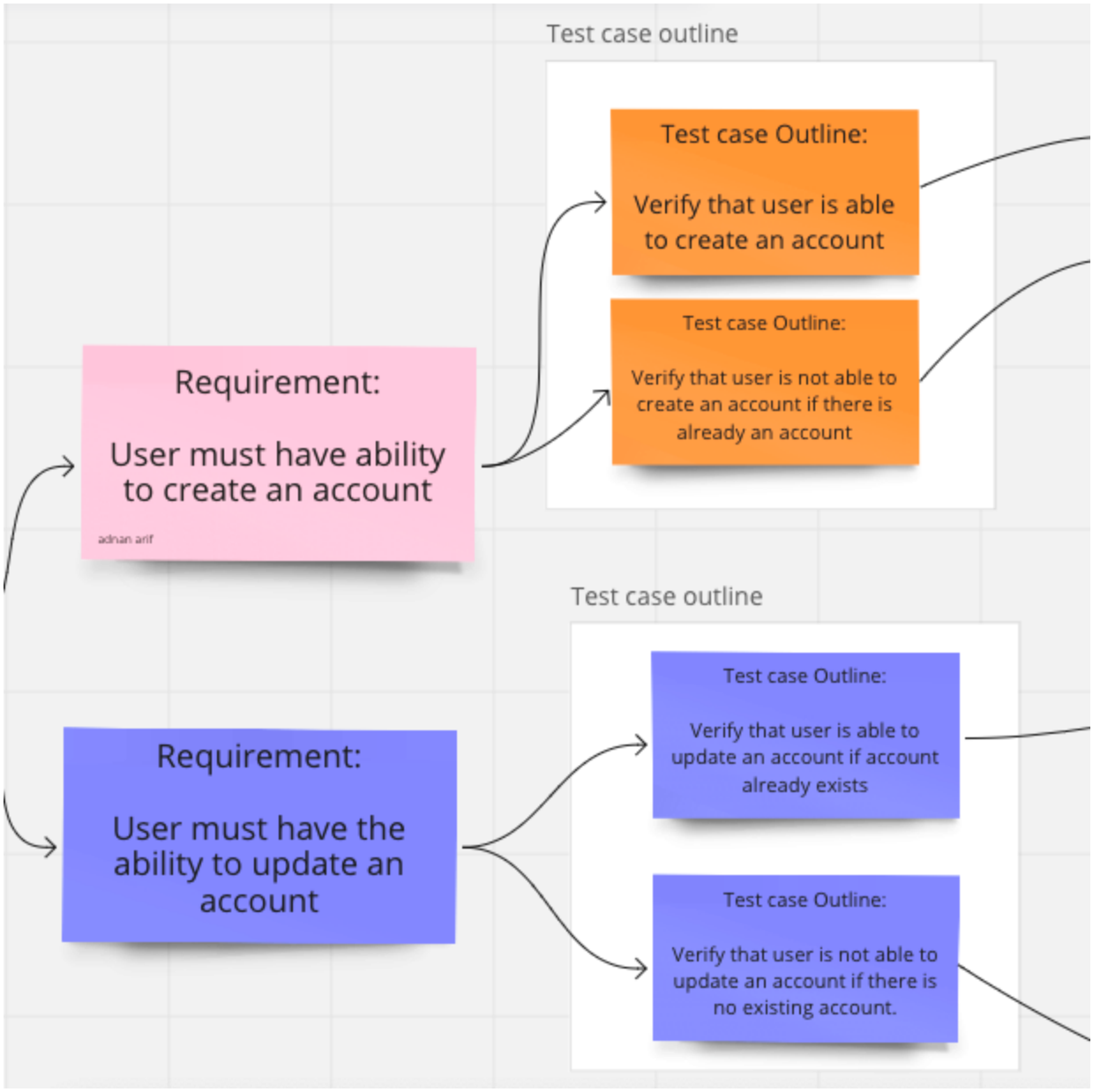
Test Case

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly. Test cases are typically used in software engineering and software testing to verify that a system meets its design specifications and behaves as expected.

Test cases are written to capture the expected outcomes of a particular test scenario, and they can be used to validate the accuracy and reliability of a system. Test cases should also be written to cover both positive and negative test scenarios. Test cases are often written by testers before any system functionality is written, and they can help developers understand the requirements and expectations of the system. Writing test cases is a great way to identify errors and other potential issues early in the development process. Ultimately, test cases help ensure the highest quality system possible.

Test cases specify how to test a system, software, or application through its paces. A test case is a distinct set of actions or instructions for a tester to follow in order to validate a specific expected result of a product's or application's functionality. If the test fails, testers are usually expected to communicate and create a defect.
Usually testers start writing test case outlines before putting them into test case template. Below is an example of requirement and how test case outline is created. Tetsters are expected to review test case outline with the team so that there are no invalid case. Team can also suggest more test case outlines that testers might have missed.



After test case outlines are approved, a QA person can move to putting test case outlines into proper test case template. See below an example where a ‘Create Account’ test case outline moved into test case template.

What is Test Case Creation:

Test case creation is the process of testing your product to make sure it works as you expect, including any interactions with other products or services. For example, a test case may verify that a user can complete a task without encountering any errors or problems. Test cases also help ensure that when new features are added, changes in functionality or appearance don't break existing functionality.

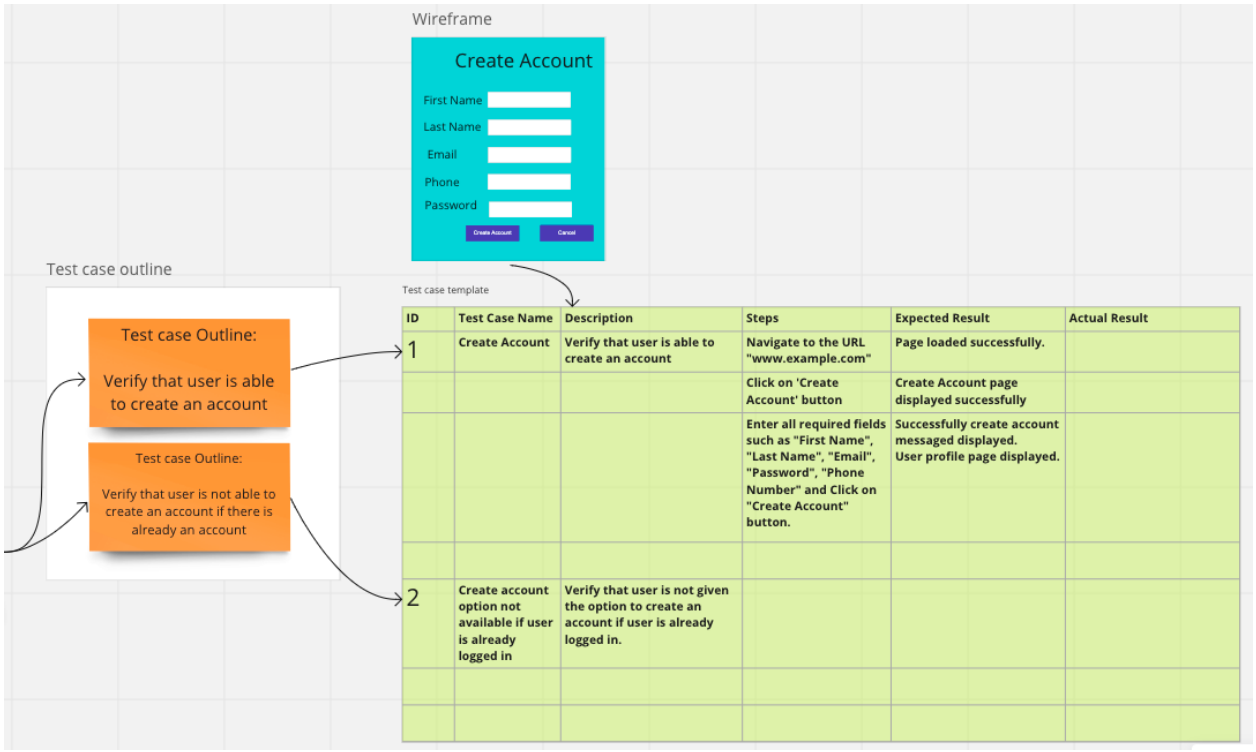
The goal of test case creation is to define the steps that you want your software to take, as well as the expected results for each step. You can use test cases to verify that all of your code works correctly and that there are no bugs in it. In addition, you can use test cases to find bugs in your code by writing a series of tests that highlight any problems with the code.

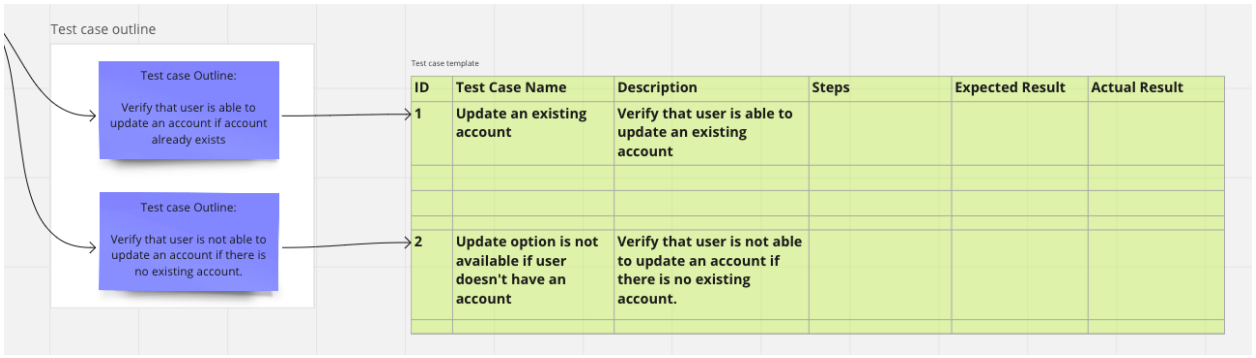
Test Case creation and Template:

A test case template is used to document exact steps to be taken in order to perform or execute a test case. Below are some of the main components of a test case template:

ID: ID is just a unique identification of a test case
Test Case Name: Name of the test case
Description: High Level description of the test case
Steps: Steps that should be taken to perform a test case
Expected Result: According to the requirement, what is the expected result of this step.
Actual Result: The actual result would be documented in this column after a test is performed.
Status: Passed/Failed
Defect: If a test is failed and the tester have created a defect. Defect ID would be documented here.

See below an example of how a test case outline is converted to test case template. First figure is about an example of a test case which is about ‘Create Account’ and the second figure is an example of a test case which is about ‘Update Account’.


Wireframe
Create Account
First Name
Last Name
Email
Phone
Password
Create Account
Cancel
Test case outline
Test case Outline:
Verify that user is able to create an account
Test case Outline:
Verify that user is not able to create an account if there is already an account
Test case template
ID
Test Case Name
Description
Steps
Expected Result
Actual Result
1
Create Account
Verify that user is able to create an account
Navigate to the URL "www.example.com"
Page loaded successfully.
Click on 'Create Account' button
Create Account page displayed successfully
Enter all required fields such as "First Name", "Last Name", "Email", "Password", "Phone Number" and Click on "Create Account" button.
Successfully create account messaged displayed.
User profile page displayed.
2
Create account option not available if user is already logged in
Verify that user is not given the option to create an account if user is already logged in.


Test case outline
Test case Outline:
Verify that user is able to update an account if account already exists
Test case Outline:
Verify that user is not able to update an account if there is no existing account.
Test case template
ID
Test Case Name
Description
Steps
Expected Result
Actual Result
1
Update an existing account
Verify that user is able to update an existing account
2
Update option is not available if user doesn't have an account
Verify that user is not able to update an account if there is no existing account.

Test Case Execution

Test case execution is the process of running and analyzing tests for a particular software project.

Test execution helps to ensure that the application meets the requirements specified in the test case. The output of a test case execution is a log which presents the results of the test case (e.g., whether it passes or fails).Test execution is an important part of the software testing process and should be done carefully and systematically.

To execute test cases, the tester should have the relevant test environment setup, test data, and expected results. The tester should also have a clear understanding of the test case and its purpose. After the test case is executed, the results should be compared to the expected results. If the actual results do not match the expected results, then the test case should be failed and the issue should be reported. After the end of the test execution activity, the result should be shared with the team.

QACop analysis - Here are some important points to consider while performing test execution activities.

1. Always make sure test cases are updated regularly and are valid.
2. Test data is valid. For example if you use an account number for a test case make sure the account number is still valid and available.
3. For every failed test case, create a defect and share with the team immediately.
4. Document all defects and share the report with the team everyday.
5. Take a screenshot of every failed test case and put it in the defect.
6. For reporting a defect follow a defect template. This will help the team understand the defect. A good defect template should have follow items
    a. Defect ID:
    A defect ID is a string that uniquely identifies a defect in your code. It's a way of tracking the status of an issue and making sure it gets fixed.

    b. URL The URL is the address that you can use to access a specific website or page on a site.

    c. Release (Optional)
    Release is the process of gathering, verifying and finalizing the requirements for a software application.

    d. Sprint (Optional)

    e. Defect Description What is the defect, explained in a short summary

    f. Steps to Reproduce:
    Provide steps to reproduce a defect

g. Expected Results:
   What were you expecting while testing

h. Actual Results:
   What actually happened.

i. Attachment:
   Screenshot of the defect.

j. Test Case ID (Optional):
   Provide test case id which failed because of the defect.

k. Requirement ID (Optional):
   Provide requirement ID which was not fulfilled because of the defect