Axis v2.0

User Acceptance Tests (UAT)

Summary

End-to-End testing is similar to user acceptance testing (UAT) because testers will replicate end-user behavior, like making a transaction through the website.
The aim of the tests is to ensure the website behaves the way it is expected.
Also, it helps to identify and track wrong behavior, which helps the developer at the time of fixing it.

Recommendation

Tests sheet

We will use a Google Spreadsheet to track the testing process [Axis v2.0 - E2E testing](#)

Once you open that spreadsheet, I recommend you to duplicate the only sheet called **Tests** several times, and use one sheet to follow all the steps for each course type.

Per instance, you can duplicate the **Tests** sheet and call it **FOET** and run all your tests with a FOET training record.

Then copy again the **Tests** sheet and call it **CA-EBS (DPR)**, another for BOSIET (OPITO) and so on.

The reason for that is because different course types generate different documents and require you to follow different steps that are not needed in some other certificates. Per instance:
- Expired certificate document upload which only works in FOET course. This step is not tested in other course types.
- OPITO file generation, certificate number and learner ID input, certificate document upload, etc., which only works in OPITO also cannot be tested with other course types.

For that reason, just duplicate the Tests sheet several times, rename those sheets with the name of the course to be tested and run all the tests for each of the different courses.

Test user

I will also recommend you to create a new user, so you can impersonate different roles by just changing its role whenever you need.

Per instance, create a user named **test** and set the role of **Admin** in order to perform the tests related to the **Admin** role, tests 4-8 in the UAT spreadsheet.

Then just change its role to **Frontdesk** to run the test 9 in the UAT spreadsheet.

Procedure

The UAT spreadsheet include 18 tests that shall be executed

Note that you need to impersonate each role in order to run the tests.
Each tests specify the following columns:
- **#** is the number that identifies of the test.
- **Activity**: Is the label of the test for easy reference.
- **Role**: Is the role that has to be impersonated to run the test.
- **Test steps**: What are the actions we need to execute for the test.
- **Expected behavior**: What it is expected to get as a result of each feature.
- **Test passed**: It's a dropdown that you have to set once the test is completed to reflect the results.
- **If failed, please provide details**: Here you need to write the details of the bug you found, or if the expected behavior was not fulfilled.

Conclusion

All tests shall be passed, but if for any reason a bug or not fulfilled expected behavior are found, we will address a code fix immediately.

Once a fix is implemented, we will run what is called an Integration Tests (I&T), which is to test all the involved components that might be affected by the that code fix (not the entire application)

Once all the tests have passed, we will be ready for product deployment and launch.