**Test Plan**

Version-1.0

1.0 Objective................................................................................................................

2.1  Reference Document....................................................................................................

2.2  Feature scope......................................................................................................

3.0    Test scope..............................................................................................................

3.1    Test prerequisites..........................................................................................

3.2    Definition of Done...........................................................................................

 4.0 Test Approach......................................................................................................

 5.0 Test Management...................................................................................................

5.1 Test Data Management....................................................................................

5.2 Test Case/Features Management....................................................................

5.3 Defect Management..........................................................................................

5.4 Reports Management.......................................................................................

6.0Risk, Assumption & Dependencies  .................................................................

1. **Objective –**

The key objectives are as follows:

* Determine the significance, or critical nature, of the application system to the business.
* Determine the types of tests required by each testing task.
* Identify the need for converted data from legacy systems or other sources.
* Determine the need for a systems integration test by identifying key system interfaces.
* Identify performance assurance requirements

**2.0 Reference Documents**

|  |  |
| --- | --- |
| **Document** | **Confluence page link** |
| Functional document |  |
| UI/UX |  |

**2.1 Feature scope**

|  |  |
| --- | --- |
| **Feature** | **Clickup id** |
|  |  |
| Core- Technical |  |
| Core- User Interface |  |
| Security |  |
| Branding |  |
| Architecture Design |  |
| Infrastructure |  |

**3.0 Test scope**

|  |  |  |
| --- | --- | --- |
| **Coverage** | **Details** | **High level test scenario** |
| Journey  configurations |  | Enable following functionalities   * Login * SignUp * Trade * Portfolio * Document * LogOut * Setting |
| Carolina Crypto **Sign Up** |  | * New to Carolina Crypto - Ability to sign up for Carolina Crypto (Name, number, email, security credentials) * Existing Carolina Crypto - Ability to sign up for Carolina Crypto (Name, Phone number, email, security credentials) |
| Carolina Crypto login |  | * Ability to log In and update Account details * Complete KYC of user * Ability to validate KYC |
| Trade |  | * Ability to display a ‘Brand View’ screen for Carolina Crypto where their customers can view/perform functionalities listed below * Buy/sell * Types of assets * Make changes in Amount * Make change in Fee * Request Price |
| Salary Advance |  | * Ability to view available salary for enrolled employers * Ability to drawdown salary for selected employer * Ability to view salary drawdown transaction history |

**3.1 Prerequisites of FQA( Functional QA Environment)**

When user story is selected to be delivered

* Define clearly enough that all members of the team understand what must be done
* Documented and agreed Acceptance Criteria in BDD format (Given When Then)
* Wireframes and Figmas - to be attached to the story.
* Key Architectural decisions and API Contract requirements identified and Design documented on confluence
* Monitoring requirement to be defined by BAs /  Architects  - Part of functional specification or as a task in the story. This could be also part of Acceptance Criteria.
* UI - Placeholder - TBC
* Accessibility - TBC
* Team should have clear definitions as to how to setup test Data.
* High level sub task has been identified and added to Jira
* Data model ready and understood by team and documented on confluence
* Identifying any dependencies or spikes if needed
* Estimated User Story
* Unit Testing done by Build before code promotion
  1. **Prerequisite of NP(Non Production Environment)**
* Prerequisite : User Story released to Non-Prod Env and exists the sprint
* Acceptance Criteria for each Jira issue is met
* Automated test scripts created and placed in GitLab
* Integration / Automation testing done and defects fixed if any
* Deployed to non-prod environment (if there is no separate story for integration)
* Accessibility implemented in code as per accessibility
* specifications (This must be in place once we start specifying accessibility specifications for UI stories)
* Monitoring is in place for stories and successfully tested and approved by SA
* NFT tasks executed successfully and results documented in Jira
* Story signed off by Product Owner and Solution Architect
* UI / UX sign-of for the stories where need

**4.0 Test Approach**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test types**  (type of test in scope) | **DETAILS** | **Who**  (Who is the owner) | **When**  (When are the tests executed) | **How**  (Approach used to test) | **Where**  (Environment where the test are executed ) | **What**  (Tools/Techno-logys used) |
| Unit testing |  | Developer | As a part of deployment to functional QA branch | Automated | Dev | Junit |
| API/Component testing |  |  | Whenever the component will developed | Manual/automation | FQA | Postman |
| Functional testing |  |  | When build release or QA validation | Manual/automation | FQA/non-prod | BDD |
| E2E testing |  | Maintainer | When unit and functional testing is completed | Manual/automation | Non-prod | Postman |
| Non Functional Testing |  |  | When all the features will be developed |  | NonProd | Jmeter |
| Security Testing |  |  | Each stage of the CI/CD pipeline |  | FQA/Non-prod |  |
| Accessibility testing |  | QA |  | Manual | FQA/non-prod | N/A |

**5.0 Test Management -**

**5.1 Test Data Management-**

We have to paste structure of project.

**5.2 Test Case/Features Management-**

Click here:-> [Test case formate](https://vlinkinc-my.sharepoint.com/:x:/g/personal/pankaj_kashyap_vlinkinfo_com/ETux0e6DxIlOpaF64vhN_ZkBqdRr1fHnvTQQTPRuTmNhVQ)

**5.3 Defect Management-**

Notion

**5.4 Reports Management-**

Shared/Google drive

**7.0 Test Risk, Assumption & Dependencies    -**

|  |  |  |
| --- | --- | --- |
| **Risk** | **Assumptions** | **Dependencies** |
| **RESOURCES**   * Not enough resources/clashing priorities with…. |  | * . |
| * Last minute changes in story,Test team may not be able to accomodate |  |  |
| * High defect turnaround time needs to be quicker to prevent increased testing time |  |  |

**8.0 Tools to be required for testing-**

* Notion
* Azure
* Miro
* Plaid
* Figma
* Selenium
* Eclipse
* Postman