## Test Plan - Soar

## 1. Objective:

* To continuously test and validate features during Agile sprints, ensuring that each iteration delivers functional, tested, and integrated code that meets user stories and business requirements

## 2. In Scope:

Functional testing of below user stories is coming under the scope of this test plan

* Registered User Login from a New Device
* Individual Investor - Upgrade to Premium
* Approvals Notifications | SMS
* Corporate investor - Commercial Registration Verification Via [X]

Integration with third-party APIs

## 3. Out of Scope:

* Security Testing
* Performance Testing

## 4. Test Approach/Strategy:

* Behaviour Driven Development (BDD), where tests will be written in plain language using Gherkin
* Automated testing with Selenium and repetitive tasks
* Manual testing for exploratory tests and complex scenarios
* Continuous Integration (CI) testing after every code commit

## 5. Test Environment:

* QA environment is updated with the latest code after every build
* Multiple browser configurations (Chrome, Firefox, Edge)
* Integrated with CI tools (Jenkins) to run automated tests after each code commit

## 6. Test Schedule:

Sprint Planning:

* Testing tasks are planned alongside development tasks

Daily Testing:

* Continuous testing during the sprint

Post-Sprint:

* Defect fixes and retesting performed after each sprint if needed, with bugs carried over to subsequent sprints if not resolved

## 7. Roles & Responsibilities:

QA Team:

* Write test cases, automate tests, and conduct manual testing

Developers:

* Write unit tests and fix defects

Scrum Master:

* Facilitates daily meetings, ensures communication between development and testing teams, and ensures testing is part of each sprint cycle

DevOps Engineer:

* Ensures CI/CD integration

## 8. Entry & Exit Criteria:

Entry Criteria:

* User stories defined and ready for development
* Test environment and Test data setup

Exit Criteria:

* All acceptance criteria for user stories met
* No open critical defects

## 9. Risk Management:

Risks:

* Changing requirements, tight sprint deadlines, incomplete test automation

Mitigation:

* Close collaboration with development, frequent backlog grooming, prioritizing high-risk areas, and ensuring automated tests cover critical workflows to reduce regression issues

10. Test Deliverables & Documentation:

* Test Plan
* Test Cases for user stories
* Automated Test Scripts
* Defect Reports
* Retrospective reports to improve future sprint testing

## 11. Test Cases and Scripts:

* Test cases are created for each user story, and scripts for automated testing are stored in a repository.

## 12. Metrics & Reporting:

Test Coverage:

* Percentage (%) of automated test cases

Execution Time:

* Duration for automated tests

Reporting Frequency:

* Regular status will be updated on daily stand-ups and Weekly progress will be updated to stakeholders

Defect Metrics:

* Opens versus closed defects and classification of the defects based on severity and priority will be updated