TEST PLAN FOR

MONEY LENS

Note that you can refine your testing plan as the project development goes. Keep the change log as follow:

ChangeLog

Version	Change Date	By	Description
1.0	Feb 27, 2025	Jashan	Set up the CI components with auto-build
1.0	Feb 28, 2025	Jashan	Tests added and synced with CI auto-build
1.1	Feb 28, 2025	Sahil	Added integration and unit tests
2.0	Mar 20, 2025	Ginelle	Switched to mongodb memory server instead of Jest mock library
2.1	Mar 20, 2025	Sahil	Workflow for CD set up
2.1	Mar 27, 2025	Jashan	Test CI
3.0	Mar 28, 2025	Aakash	JMeter load testing implemented
3.0	Mar 28, 2025	Sahil	SonarCloud security testing added

1 Introduction

1.1 Scope

Scope defines the features, functional or non-functional requirements of the software that **will be** tested.

Sign-In:

- New user created
- Sign In with existing user credentials
- Sign Out
- Wrong username or password
- Sign In with non-existing user credentials
- Change password

Plaid API Connection:

- Sign In with new user
- Connection with existing user

Dashboard:

- Correct user info
- Get transactions
- Get recent transactions
- Get monthly spending totals over previous year
- Get monthly earnings totals over previous year
- Get today’s spending total
- Display expected balance

Transactions:

- Get transactions
- Search transactions by keyword
- Search history

Analytics:

- Get transactions
- Get transactions by month
- Get transactions by date
- Get transactions by date range
- Get transactions by category
- Get transactions by debit
- Get transactions by credit

Goals:

- Add goal
- Edit goal
- Delete goal
- Add amount to goal

1.2 Roles and Responsibilities

Test Manager

- Develop the test plan, manage/monitor the testing execution, and coordinate with other team members to ensure quality test coverage.

DevOps Engineer

- Sets up the CI/CD pipeline, test automation and test environments.

QA Engineer

- Perform functional, UI and regression tests.

Integration Test Engineer

- Perform integration tests between services, modules and APIs. Ensure frontend and backend components work together correctly.

Developers

- Fixes reported issues and performs unit tests.

Name	**Net ID**	**GitHub username**	**Role**
Ginelle Temull	temullg	ginellego	Test Manager
Jashanjot Gill	gilljs5	Jashann	DevOps Engineer
Una Mayberry	mayberru	una-may	QA Engineer
Sahil Sharma	sharm56	Sahil-Sharma-603	Integration Test Engineer
Filip Karamanov	karamanf	FilipKaramanov	Developer
Aakash Chouhan	chouhana	Aakash812	Developer

2 Test Methodology

2.1 Test Levels

Unit Tests:

- Tests individual functions & modules
- At least 10 unit tests for each core feature

Integration Tests:

- Ensures API calls & DB queries work
- At least 10 integration tests total to cover core features

Acceptance Tests:

- Checks front end behavior
- A end-user test for each core feature
- Ask for real users to test the app

Regression Tests:

- Execute all unit and integration tests for each commit to the main branch

Load Tests:

- At least two requests associated with every core feature are included in the test load.
- Test the non-functional performance
    - Import 1000 transactions from 100 users within a minute

2.2 Test Completeness

Criteria:

- 100% back-end code coverage (mandatory for this project), all the back-end source code should be covered by test cases.
- No high severity defects
- Regression tests executed for major modules

3 Resource & Environment Needs

3.1 Testing Tools

Jest for unit and integration testing
Manual script for acceptance testing
GitHub Actions for regression testing
SonarCloud for security testing
JMeter for load testing

3.2 Test Environment

Hardware

CPU:	Multi-core processors
Memory:	8-16 GB
Storage:	256GB
Bandwidth:	Sufficient speed

Software

Frontend:	React, Next.js
Backend:	Node.js, Express.js
Database:	MongoDB, Firebase
Containerization:	Docker
Devices:	macOS, Windows
Browsers:	Chrome, Firefox, Edge, Safari
CI/CD:	GitHub Actions

4 Terms/Acronyms

Make a mention of any terms or acronyms used in the project

TERM/ACRONYM	DEFINITION
API	Application Program Interface
AUT	Application Under Test