TEST PLAN FOR

SAFEAT

ChangeLog

Version	Change Date	By	Description
V1	February 26, 2021	James Harvey	Added preliminary testing plan information

1 Introduction

Brief introduction of the scope, strategies, processes and workflow used for Safeat.

1.1 Scope

1.1.1 In-Scope

Sprint 3 target:

Back End: Integration and Unit tests for all endpoints. All in use endpoints should be covered.

Front End: Unit tests for logic layer, Integration tests for data layer, Acceptance tests for Login and Default routes. This includes all pages, components, and contexts.

1.1.2 Out-of-Scope

Unit tests for HTML layout.

1.2 Quality Objective

- Ensure the Application Under Test (AUT) conforms to functional and non-functional requirements
- Ensure the AUT meets the quality specifications defined by the user (TA)
- Bugs/issues are identified and fixed before go live
- All automated tests pass before pull request is merged
- Backend and frontend tested separately during AUT

1.3 Roles and Responsibilities

Name	Net ID	GitHub username	Role
James Harvey	harveyj5	Le-DarkOverlord	Full Stack Developer
Ezra Hall	halle3	ezrahall	Front End Developer
Matt Lewis	lewism36	Matthew9799	Back End Developer
Tahmidul Hasnain Zidaan	zidaant	thzidaan	Front End Developer
Michael Ukwenya	ukwenyam	ukwenyam	Full Stack Developer

2 Test Methodology

2.1 Overview

Safeat is following the Agile methodology, and testing is a part of that methodology. For testing we will focus on creating the automated tests as we code. It’s important to have working code but it’s also important to have strong tests that cover all layers of our project. Each feature will have user stories and each of those user stories must have at least one test associated with it, developed along side the code itself.

2.2 Test Levels

AUT:

Frontend:

- Unit
- Integration
- Acceptance
  - Due to time constraints, it is not possible to cover every route

Backend:

- Unit
- Integration

2.3 Test Completeness

Minimum requirements for test completeness:

- 100% test coverage of API endpoints
- 100% of Default and Login routes covered
- 100% test coverage for context API calls
- 2 Acceptance tests for Home, Login, Menu, Address pages
- Layouts components such as MenutItem, Cards need not be tested
- Unit tests for business layer related functions
- All tests automated and all test cases executed
- All open bugs relating to features in current sprint fixed
- All bugs discovered for future milestones or features fixed in next release

3 Test Deliverables

- Test Plan
- Back End test suites
- Front End test suites
- Test failure logs
- Bug Reports
- Changelog for suites (commit messages)

4 Resource & Environment Needs

4.1 Testing Tools

- GitHub Actions
- Node.js CI
- Python package CI
- .yml scripts
  - React: Using npm test
  - Python: Using pytest

4.2 Test Environment

Following **software** is required in addition to client-specific software.

- Windows 10
- GitHub Actions
- Node Version: [14.x]
- Python Version: [3.x]
- Web browser required
  - Chrome (recommended)
  - Edge
  - Safari

5 Terms/Acronyms

TERM/ACRONYM	DEFINITION
API	Application Program Interface
AUT	Application Under Test
Agile	Methodology for software development, emphasizes adapting to change over following a plan.
HTML	HyperText Markup Language (Web layout language)
React	JavaScript library for building user interfaces
Python	Interpreted language for backend web development
TA	Teaching Assistant