Sachin Kumar & Hiep Huynh

Abstract

This testing document contains the detailed information about the testing of the Employee Reimbursement System. The motive of this test is to uncover all the flaws of the system for all the user stories.

Test strategy

Employee Reimbursement System

**Revision and Signoff Sheet**

**Document History** - To maintain a list of changes being made

|  |  |  |  |
| --- | --- | --- | --- |
|  | Date | Author | Description of Change |
|  | 6/17/22 | Hiep | Rough Draft |
|  | 6/18/22 | Sachin | Finalize Document |
|  |  |  |  |

**Approvers List** - To track who has reviewed and signoff on the Test plan

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Role | Approver / Reviewer | Approval / Review Date |
| Sachin | Test Automation Engineer |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# INTRODUCTION

## Purpose

To Verify the Application performs as Intended. To uncover the bugs in the application and to ensure the implementation of all the user stories.

## Project Overview

This application is management system for employees to request reimbursements for business related expenses.

## Test Objectives

The objectives of the Tests:

* Assure all test passes
* Assure all features behave as intended

## Test Assumptions

**Key Assumptions**

* **Relevant Data to tests are present in the database for some tests to be functional**

## Test Principles

* Testing environment and data will emulate a production environment as much as possible.
* Testing will be a repeatable, quantifiable, and measurable activity.
* There will be entrance and exit criteria.

## Scope and Levels of Testing

### Exploratory

**PURPOSE**: the purpose of this test is to make sure critical defects are removed.

**SCOPE**: 2 weeks of the Initial Sprint

### Functional Test

**PURPOSE:**  Functional testing will be performed to check the functions of application. The functional testing is carried out by feeding the input and validates the output from the application.

**Scope:** 2 weeks of the Initial Sprint

# EXECUTION STRATEGY

### Entry Criteria

Entry criteria refers to desirable conditions to start test execution. Entry criteria is flexible and shall be determined by the test team.

### Defect Handling

known bugs that significantly reduced the performance of the application will be addressed before the end of the two week Sprint.

### Tools

### Python: will be required as Programming language and version is Python 3.10.4

**VSCode:** Will be required as an IDE.

**Pytest:** will be used for unit testing.

**Flask**: will be used for web framework to create the web applications in Python.

**AWS**: will be used to create database remotely on the server side.

**PostgreSQL:** will be used to link the server with local computer.

**SQLAlchemy:** will be used to execute SQL commands in python.

**Jinja2:** will be used as [web template engine](https://en.wikipedia.org/wiki/Template_engine_(web)) in Python.

**Postman**: Will be used for integration testing

**Behave & Selenium**: will be used for end-to-end testing and BDD

**Reports:** pytest-excel will be used to capture unit test results.

# TEST ENVIRONMENT

### Python virtual environment will be the environment for testing

# Test Approach

## Unit Test:

Unit test will be performed at programming level. Each module with higher probability of error and every user input will be tested at service level before inserting data into the database.

## Smoke Test:

Smoke Test will be performed by Testing happy path via selenium. Every user story will be checked against a happy path to recognize the major functionality of the application and will discover flaws of the major functionality of the system

## Regression Test:

All the functionalities will be tested for regression test suit to uncover the defects of the application. The whole application will be tested using the selenium and behave to record the front-end application testing.

## API Testing:

The APIs and databases will be tested using postman by injecting the input through http requests. Record the status code of every request and match the test results with the expected outputs.

# Test Plan

All the tests will be planned for given test approaches with all the possible test scenarios. The test plans will be recorded in separate MS excel files with test results. The test plan will make sure there are all possible and negative tests to cover all possible test cases. The unit test will be recorded using the automation framework of python and analyse the test result.

## Test Suite

The tests will be divided into test suits of feature files to execute the user stories to ensure all the features are included in the test suites.

## Test Results

The Test result will be captured in test plan files with log files and analyse the failure tests with its priority and severity level. The defects will go through bug life cycle and assign to developer to fix it in the given time frame.

# Requirement Traceability Matrix

All the requirements are given in the form of user stories and will evaluate thoroughly using requirement traceability matrix. The given test scenarios will record accordingly, examine and analyse the result of the major functionality. All the user stories required to have passed in testing.

# Acceptance Criteria

All the major functionality is working as intended. There should be no critical bug in the application that is having high severity. All the testcases with the high severity and high priority should be pass and having no effect of other modules. The application should pass all the smoke test scenarios and have tested with all the user stories with positive and negative test cases. The application should pass all the test cases of user scenarios given in the Requirement Traceability Matrix. All the methods must have at least one positive test and multiple negative test cases include boundary value analysis, equivalence partitioning and decision tables.