Preplanning Document

Revature, Project 2, Team B

Expense Reimbursement Web Application

Software Engineer Team : Max, Nick, Regina

**TABLE OF CONTENTS**

[Preliminary Problem Scenario Definition 3](#_Toc502677613)

[Preliminary Project Scope 3](#_Toc502677614)

Refrences…………………………………………………………………………………………………………………………………………………4

# Preliminary Problem Scenario Definition

1. The new software product will be an Expense Reimbursement System, The Expense Reimbursement System (ERS) will manage the process of reimbursing employees for expenses incurred while on company time. All employees in the company can login and submit requests for reimbursement and view their past tickets and pending requests. Finance managers can log in and view all reimbursement requests and past history for all employees in the company. Finance managers are authorized to approve and deny requests for expense reimbursement.

# Preliminary Project Scope

The project is for the purpose of creating a web application that serves as a user interface for employees, admins, and finance managers to submit or handle all reimbursement requests. This system will manage the process of reimbursing employees for expenses incurred while on company time. All registered employees in the company can login and submit requests for reimbursement and view their past tickets and pending requests. Finance managers can log in and view all reimbursement requests and past history for all employees in the company. Finance managers are authorized to approve and deny requests for expense reimbursement.

Functional Objectives:

* A new employee or new finance manager can request registration with the system
* An admin user can approve or deny first-time registration requests
* A registered employee can authenticate with the system by providing valid credentials
* An authenticated employee can view and manage their pending reimbursement requests
* An authenticated employee can view their reimbursement request history (sortable and filterable)
* An authenticated employee can submit a new reimbursement request
* An authenticated finance manager can view a list of all pending reimbursement requests
* An authenticated finance manager can view a history of requests that they have approved/denied
* An authenticated finance manager can approve/deny reimbursement requests
* An admin user can deactivate user accounts, making them unable to log into the system
* An admin user can reset a registered user's password

Operational Objectives:

* Basic validation is enforced to ensure that invalid/improper data is not persisted
* User passwords will be hashed by the system before persisting them to the data source
* Users are unable to recall reimbursement requests once they have been processed (only pending allowed)
* Sensitive endpoints are protected from unauthenticated and unauthorized requesters using JWTs
* Errors and exceptions are handled properly, and their details are obfuscated from the user
* The system conforms to RESTful architecture constraints
* The system's is tested with at least 80%-line coverage in all service and utility classes
* The system's data schema and component design are documented and diagrammed
* The system keeps detailed logs on info, error, and fatal events that occur
* UI and API builds and deployments are automated using a CI/CD pipeline
* API is deployed to AWS EC2 (via Elastic Beanstalk) as a Docker container
* UI is deployed to an AWS S3 bucket configured for static web hosting

REFRENCES

<https://github.com/Ultronomix/docs/blob/main/project-2-requirements.md>