1. Summary Statement

The Insurance Management System is designed to provide a comprehensive platform for managing various aspects of an insurance company's operations, including policy management, claims processing, user management, reporting and analytics, and security. This document outlines the business requirements for the development of IMS, which will facilitate the efficient management of policies, claims, and customer information.

1. Project Objectives

The primary objectives of this project are to develop and implement an Insurance Management System that streamlines insurance company operations, enhances customer service, and ensures data privacy and system security.

1. Purpose of the project

The project aims to address the following needs:

* Streamlining policy management and claims processing.
* Providing robust user management and authentication.
* Ensuring data privacy and system security.

1. Project Scope

The project scope includes:

* Policy Management: Creating, updating, and retrieving insurance policies.
* Claims Processing: Handling insurance claims and tracking their status.
* User Management: Managing user accounts, roles, and permissions.
* Security: Implementing authentication and authorization mechanisms.

5. Functional Requirements

5.1. Policy Management

Create policies with unique policy numbers, policy holder names, and premium amounts.

Support policy effective start and optional end dates.

Allow policy updates.

Retrieve policy details by policy ID.

5.2. Claims Processing

Create claims with unique claim numbers, descriptions, and incident dates.

Track claim status (open, in progress, closed) and add comments.

View and update existing claims.

5.3. User Management

Support user registration.

Enable user login with authentication tokens.

Define multiple user roles (admin, manager, agent).

5.4. Security

Securely hash and store user passwords.

Implement JWT-based authentication for API endpoints.

Authorize users based on their roles.

6. Non-Functional Requirements

6.1. Security

The system should adhere to industry-standard security practices.

Data should be encrypted during transmission and storage.

6.2. Scalability

The system should be able to handle a growing number of policies and claims.

Scalability should be achieved through a modular architecture.

6.3. Performance

The system should be responsive and provide quick access to information.

Response times for critical functions should be within acceptable limits.

7. Tools and Technologies

This Microservices application will be built using a technology stack that includes Java, MySQL, Spring MVC, Spring Microservices, Spring Cloud, and will be deployed on AWS EC2 instances. The API endpoints will be documented using Swagger.

A screenshot of a computer

Description automatically generated