Project Title: InsurAI – Corporate Policy Automation and Intelligence System
Name: Samir Shaikh

Product Backlog \ Requirements Document

---

1. Introduction

1.1 Purpose

The purpose of this project is to design and develop *InsurAI – Corporate Policy Automation and Intelligence System*. The system aims to automate core insurance operations such as policy management, renewals, and claims processing, while also providing intelligent customer support through an AI-powered assistant. This document describes the purpose and scope of the system, presents an overall product description, and specifies the functional and non-functional requirements to guide the design, development, and evaluation of the project.

1.2 Objectives

- Automate corporate insurance policy management
- Provide a user-friendly digital platform for customers and agents
- Enable secure authentication and authorization using JWT
- Offer AI-based assistance for policy queries and recommendations
- Ensure scalable and modular system architecture

1.3 Scope

InsurAI is designed as a corporate insurance policy management system enhanced with AI-based automation. It will support:

- Designed for corporates (not for individual policy buyers).
- Supports role-based access: Employee, Agent, HR Admin, System Admin.
- Core functions include: policy renewals, claim approvals, tax compliance, reporting, fraud detection, and analytics.
- Can scale for large organizations with thousands of employees.

1.4 Primary Roles

The system is built on a four-tier access model, ensuring that each user has a specific, secure interface for their tasks. This accurately reflects the system's role-based access control (RBAC) and operational workflow.

- Corporate Employee
- HR Admin
- Insurance Agent
- System Admin

1.5 Technologies to be used

Component	Technology/Tool
Frontend	React.js
Backend	Spring Boot
Database	MySQL & Supabase
AI/ML Services	Cohere API (Spring AI)
Authentication & Security	JWT
Version Control	Git & Github
Deployment	Vercel/Netlify & Render/Railway

2. Overall Description

Product Backlog $\longrightarrow$

Priority	Feature	Description
High	User Authentication	Secure login & registration
High	Policy Management	Add, view, update policies
High	Appointment Scheduling	Book and manage appointments
Medium	AI Assistance	Intelligent policy queries
Medium	Email Notifications	Automated emails
Low	Reports	Basic system reports

2.1 Product Perspective

InsurAI is a standalone, web-based application designed to automate and intelligently manage corporate insurance policies and related processes. The application provides separate interfaces and functionalities for customers, insurance administrators, and agents, ensuring role-based access and secure operations.

2.2 Product Functions

- **Customer Functions:**
  - Register/login to the portal
  - View available policies
  - Apply for policy and track renewal
  - Submit claims and track status
  - Interact with AI assistant for FAQs
- **Administrator Functions:**
  - Manage policies (add/update/remove)
  - Manage claims (approve/reject)

- View customer details and policy allocations
- Monitor system activity and basic reports

2.3 User Characteristics

- Customers: Insurance policyholders with basic computer literacy.
- Administrators: Company staff with training in policy and claims processing**.**

2.4 Constraints

- Time constraint
- Limited resources and workforce
- Dependency on third-party AI services for intelligent features

2.5 Assumptions and Dependencies

- Users have stable internet connectivity.
- AI service (Cohere API) is accessible during operations.
- The system is initially deployed as a web-based application only.

3. Specific Requirements

3.1 Functional Requirements

The system must support the following functions:

1. **Authentication & Authorization**
    - Secure login and registration using JWT.
    - Role-based access control for users, agents, and admins.
2. **Policy Management**
    - Create, update, and delete insurance policies.

- Search and view policy details.

3. **Customer Dashboard**
    - View active and expired policies.
    - Check claim status and premium details.
4. **Claims Processing**
    - Submit and track claims online.
    - Notify customers about claim status updates.
5. **Payments**
    - Process premium payments securely.
    - Generate receipts for transactions.
6. **Administrative Functions**
    - Manage user accounts and roles.
    - Update policy information.
    - Monitor system reports and performance.

3.2 Non-Functional Requirements

1. **Performance**
    - The system should handle multiple requests simultaneously with minimal delay.
2. **Scalability**
    - The architecture must allow easy scaling to support growing users and policies.
3. **Security**
    - Data encryption for sensitive information.
    - Strong authentication and authorization measures.
4. **Usability**
    - Simple and intuitive interface for customers, agents, and admins.

5. **Availability**
    - System should be available 24/7 with minimal downtime.
6. **Maintainability**
    - Code should be modular and easy to maintain or update.
7. **Time Constraint**
    - The system should be developed and deployed within the given project timeline.

References

- IEEE Std 830-1998, *IEEE Recommended Practice for Software Requirements Specifications*.
- Sommerville, Ian. *Software Engineering*, 10th Edition, Pearson, 2015.