Smart Health Insurance Management System

Functional Requirement Document

Contents

1. Introduction

This document describes the **end-to-end process flow** of the *Smart Health Insurance Management System*. It explains how different users interact with the system and how data flows from policy creation to claim settlement.

The system is designed as a **full-stack web application** using **ASP.NET Core Web API** (backend) and **Angular** (frontend), with secure role-based access.

---

2. Overview of functional Workflow

2.1 Customer

- Customer enrolls to the system and waits for admin approval
- Admin approves the user and assign a role, like a customer role.
- After activation of the user, customer can login to the system using their credentials
- Customer can browse the plans and buy them
- System redirect to a payment gateway for the 1st premium payment
- Customer can browse purchased plans
- Customer can submit claim for the medical expenses
- The submitted claim goes to respective hospital/provider
- Hospital/provider updates the treatment and billing details of the submitted claim
- Claim officer does the final review, after that they may approve or reject the claim
- If approved then payment initiated and coverage amount will be reduced.

2.2 System Admin

- System admin onboard Hospital/Provider, Agent, Claim Officer to the system
- Creates insurance plans with all the details of the plans
- Admin approves and assigns roles to the users.

2.3 Hospital/Provider

- View submitted claims
- Update treatment and billing details to the submitted claims
- Track claim status

2.4 Claims Officer

- Reviews queued submitted claims
- Verifies claim details against policy coverage
- Approves or rejects claims for the payment and a remark and updates the claim status

3. High-Level System Workflow

- User accesses the Angular web application
- User authenticates using login credentials
- JWT token is issued by ASP.NET Core API
- Role-based access controls determine allowed actions
- User performs business operations (policy, claims, payments, reports)
- Angular communicates with backend APIs via HTTP
- Backend processes requests and interacts with SQL Server
- Response is returned and displayed on UI dashboards

4. High-Level Backend Processing Flow

- Controller receives API request
- JWT token validated
- Request forwarded to service layer
- Business logic executed
- DTOs returned to controller
- Response sent to frontend

5. Frontend Processing Flow

- User interacts with Angular UI
- Angular service calls API
- HTTP interceptor adds JWT token
- API response received
- UI updated with data

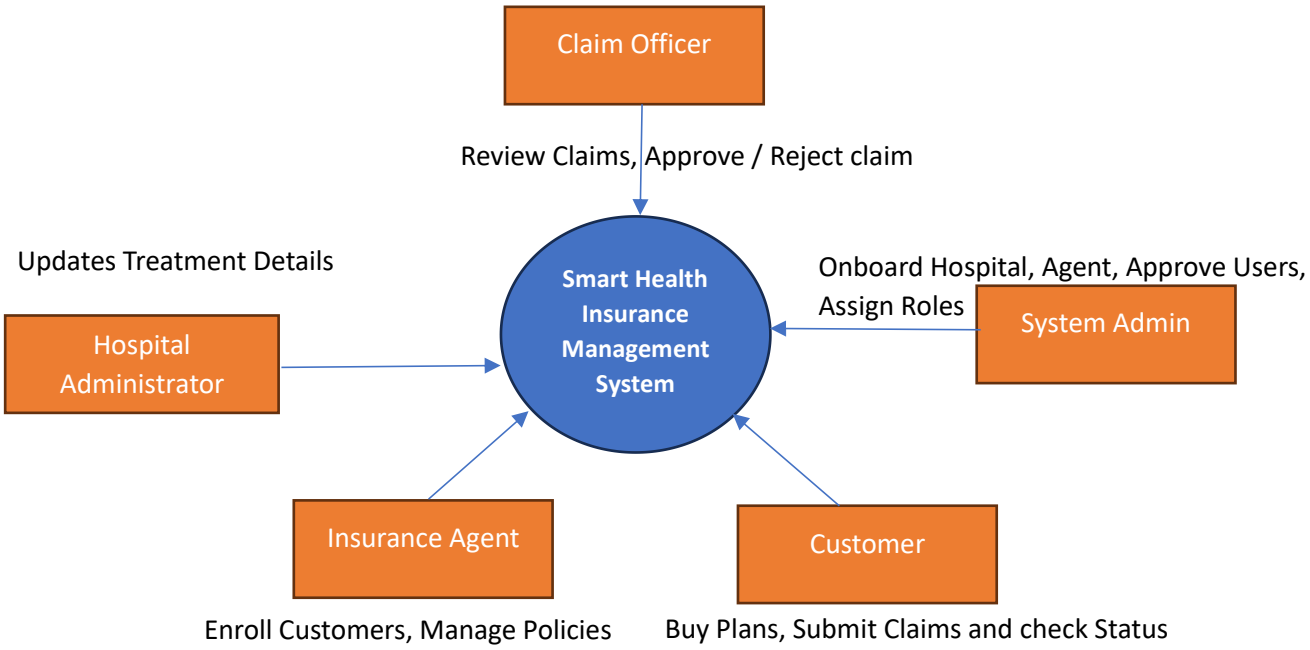
---

6. Conclusion

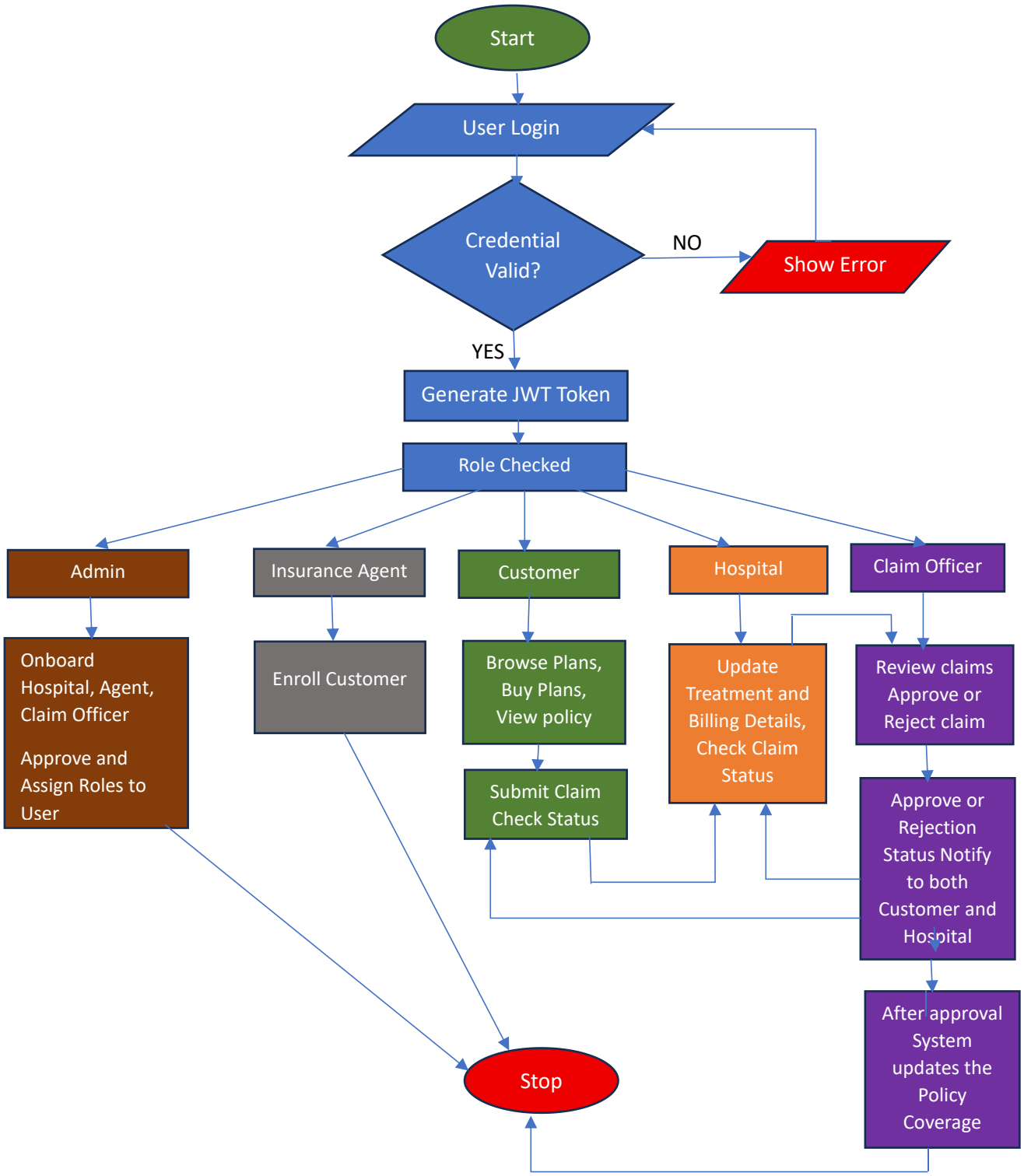
The Smart Health Insurance Management System provides a **secure, scalable, and efficient** solution for managing insurance policies and claims. The well-defined process flow ensures transparency, faster claim processing, and better coordination between customers, hospitals, and insurance staff.

7. Database Schema

8. System Architecture Diagram



9. System Flowchart



10. Requirement Traceability Matrix

Req ID	Requirement Description	Module / Screen	Role	Outcome
R1	Secure user authentication	Login Screen	All Users	Authorized access
R2	Role-based access control	Auth Guards / Dashboards	All Roles	Restricted access
R3	User approval workflow	User Management Screen	Admin	Controlled onboarding
R4	Role assignment to users	User Management Screen	Admin	Correct dashboard access
R5	Insurance plan management	Insurance Plans Module	Admin	Plan availability
R6	View insurance plans	Plans Listing Screen	Customer	Informed selection
R7	Purchase insurance policy	Buy Policy / Payment	Customer	Policy creation
R8	View purchased policies	My Policies Screen	Customer	Policy tracking
R9	Submit insurance claim	Claims Submission Screen	Customer	Claim initiation
R10	Update treatment details	Treatment Update Screen	Hospital Provider	Medical validation
R11	Review and approve claims	Claims Review Screen	Claims Officer	Claim decision
R12	Reject claims with reason	Claims Review Screen	Claims Officer	Audit clarity
R13	View claim analytics	Analytics & Reports	Claims Officer	Data-driven decisions
R14	Policy category analysis	Reports Dashboard	Claims Officer	Risk insights
R15	System monitoring & control	Admin Dashboard	Admin	System governance

11. LINQ USAGE OVERVIEW

Language Integrated Query (LINQ) is extensively used in the Smart Health Insurance Management System to perform efficient data querying, filtering, grouping, and aggregation on insurance-related data. LINQ enables readable, maintainable, and type-safe queries while interacting with the database through Entity Framework Core.

The system leverages LINQ to implement real-world business logic such as policy tracking, claim validation, reporting, and analytics.

11.1 Purpose of Using LINQ in the System

LINQ is used to:

- Simplify database queries
- Improve code readability and maintainability
- Implement business rules at the service layer
- Perform complex data analysis for reports
- Reduce SQL complexity by using strongly typed queries