Complete Project Report

Project Title: InsurAI – Corporate Policy Automation and Intelligence System

1. Introduction

Problem Statement

Large companies provide multiple insurance policies to their employees (e.g., health, accident, corporate benefits). Today, most of this work is managed by HR departments and insurance agents **manually**. This leads to:

- Delays in claim approval.
- Errors due to human processing.
- Lack of transparency for employees.
- Increased administrative costs.
- Higher chances of fraud going undetected.

Objective

To design and build an **automated, intelligent corporate insurance management system** that:

- Simplifies policy management.
- Speeds up claim approvals.
- Enhances transparency for employees.
- Reduces fraud through AI-based detection.
- Provides insights to HR and Admins.

Scope

- 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**.

2. Existing System (Current Practices)

- Employees submit forms manually or via email.
- HR and Agents manually verify documents and eligibility.
- Claims are often delayed due to paperwork and communication gaps.
- Employees cannot track the **status of their claims in real-time**.
- No proper fraud detection → companies lose money on false claims.

3. Proposed System – InsurAI Solution

A **web-based insurance management platform** with automation and intelligence.

Features of Proposed System:

- **Role-based logins** (Employee, Agent, HR, Admin).
- **Automation**:
    - Auto policy renewals.
    - Automatic claim eligibility checks.
    - Auto-generated reports (weekly, monthly, yearly).
    - Automated tax rule application.
- **AI & Analytics**:
    - Fraud detection: flagging duplicate claims, unusually high amounts, and suspicious patterns.
    - Predictive insights for HR/Admins.
- **Transparency**:
    - Employees can track claims and policies in real time.
    - Automated notifications and reminders (email/SMS).

4. System Users & Roles

Role	Responsibilities
Employee	Register/login, view policy details, submit claims, upload documents, track status.
Agent	Manage employee queries, update availability, assist employees in claim process.
HR Admin	Add employees, approve/reject claims, manage policies, generate reports.
System Admin	Manage all roles, assign permissions, configure tax rules, monitor fraud detection.

5. System Features (Detailed)

1. Authentication & Registration

- Secure login for Employees, Agents, HR, and Admin.
- JWT-based authentication with role-based access control.

2. Employee Portal

- Submit claims.
- Upload supporting documents.
- Track claim status in real time.
- View insurance policies & benefits.

3. Agent Management

- Agents register and set availability.
- Employees can request support from agents.
- Agent dashboard to track pending queries.

4. HR Dashboard

- Manage employee insurance data.
- Approve/reject claims.
- Generate scheduled reports (monthly/yearly).
- Review fraud alerts generated by AI.

5. Admin Dashboard

- Manage all system roles.
- Set tax rules, permissions, and company-wide policies.
- Oversee fraud detection alerts and system health.
- Configure automation rules.

6. Automation

- Auto-reminders for policy renewals.
- Auto-checks for claim eligibility.
- Scheduled reporting to HR/Admin.

7. AI Intelligence

- Fraud detection via anomaly detection (flagging duplicates, large unusual claims).
- Predictive analytics for HR (policy usage trends, employee health claim patterns).

8. Reports & Analytics

- PDF/Excel reports for claims, fraud alerts, policy usage.
- Visual dashboards (charts, graphs).
- Downloadable audit logs.

6. Modules Breakdown

1. **Login & Registration Module** – Secure entry, JWT authentication.
2. **Policy Management Module** – Add/update/view company insurance policies.
3. **Claim Management Module** – Raise claims, upload docs, auto-rule checks, HR approval/rejection.
4. **Agent Management Module** – Availability tracking, employee-agent interaction.
5. **Admin Dashboard Module** – Role management, tax rules, fraud alerts, audit logs.
6. **Reports & Analytics Module** – Automated reporting, fraud insights, visual dashboards.

7. Technology Stack

- **Backend:** Java Spring Boot (scalable, secure, enterprise-ready).
- **Frontend:** React.js (modern, responsive UI).
- **Database:** MySQL / PostgreSQL (structured insurance & user data).
- **Automation:** Spring Scheduler / Quartz (renewal reminders, report scheduling).
- **Authentication:** JWT (secure token-based login).
- **AI/Analytics:** Python ML models (integrated via REST APIs, optional for fraud detection).
- **Deployment (future):** Docker + Cloud (AWS/Azure).

8. Workflow Example

1. Employee logs in → Sees policies → Submits claim with documents.
2. System auto-checks claim eligibility.
3. Claim routed to HR → HR reviews → Approve/Reject.
4. Agent supports employee queries if needed.
5. Admin monitors fraud detection alerts + configures tax rules.
6. Reports auto-generated and sent to HR/Admin monthly.

9. UML Diagrams (Required)

- **Use Case Diagram** – Shows role interactions (Employee, Agent, HR, Admin).
- **Class Diagram** – Parent class (User), child classes (Employee, Agent, HR, Admin).
- **Activity Diagram** – Claim process workflow.
- **Sequence Diagram** – Flow between frontend (React), backend (Spring Boot), DB.
- **Deployment Diagram** – Shows components (React, Spring Boot, DB, AI).

10. Advantages

- **For Employees**: Transparency, faster claims, real-time updates.
- **For HR**: Reduced workload, automated reporting.
- **For Admins**: Strong system control, fraud detection, compliance.
- **For Company**: Reduced costs, better trust, scalable solution.

11. Limitations

- Currently web-only (mobile app in future scope).
- Fraud detection is rule-based; AI ML models will improve accuracy in later versions.
- External insurance company integration not yet implemented.
- Requires internet connectivity.

12. Future Scope

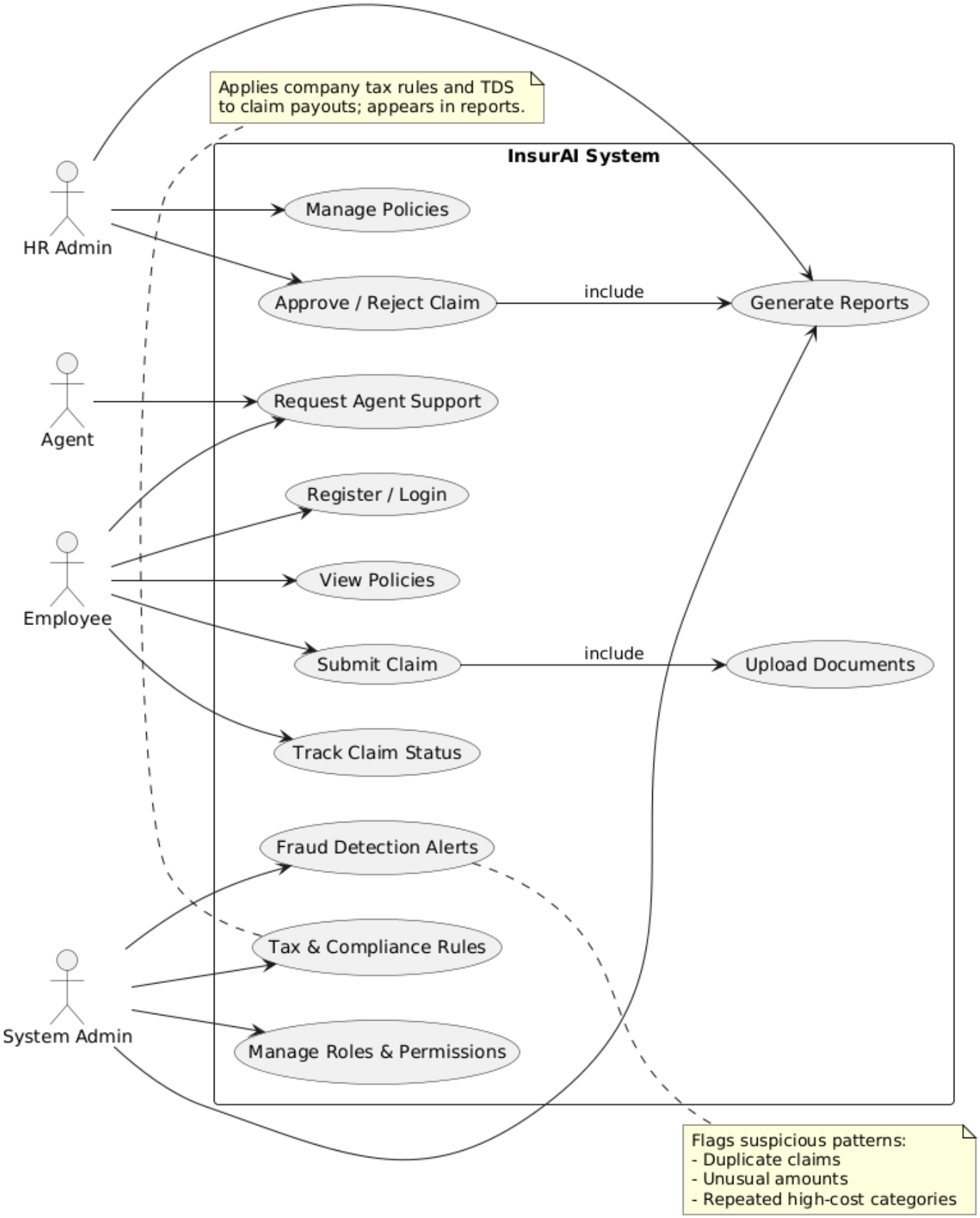
- Mobile app for employees (Android/iOS).
- Direct integration with insurance companies & hospitals.
- Agentic AI chatbot for employee queries.
- Blockchain for secure claim history tracking.
- Cloud deployment for large-scale use.

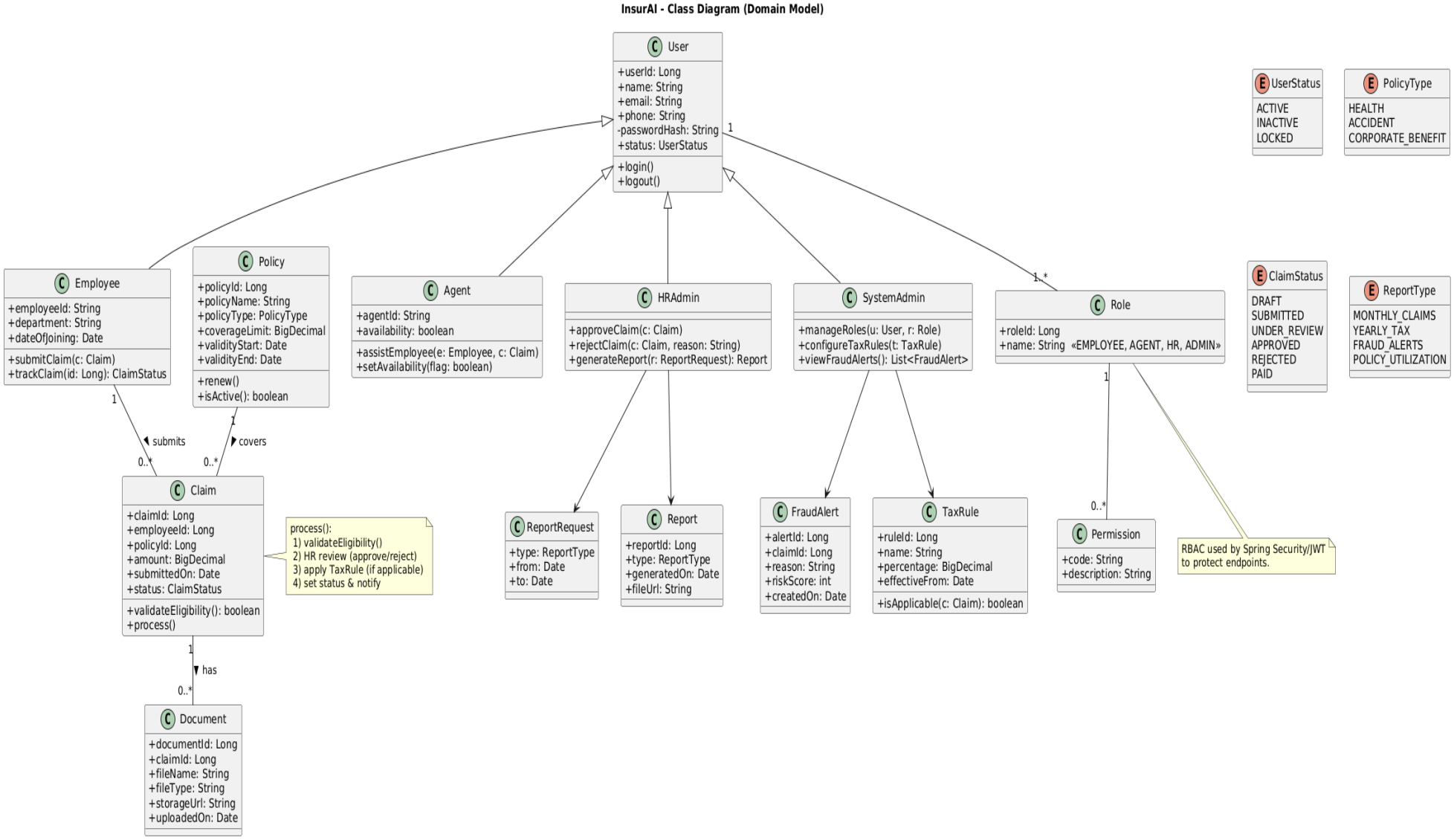
13. Conclusion

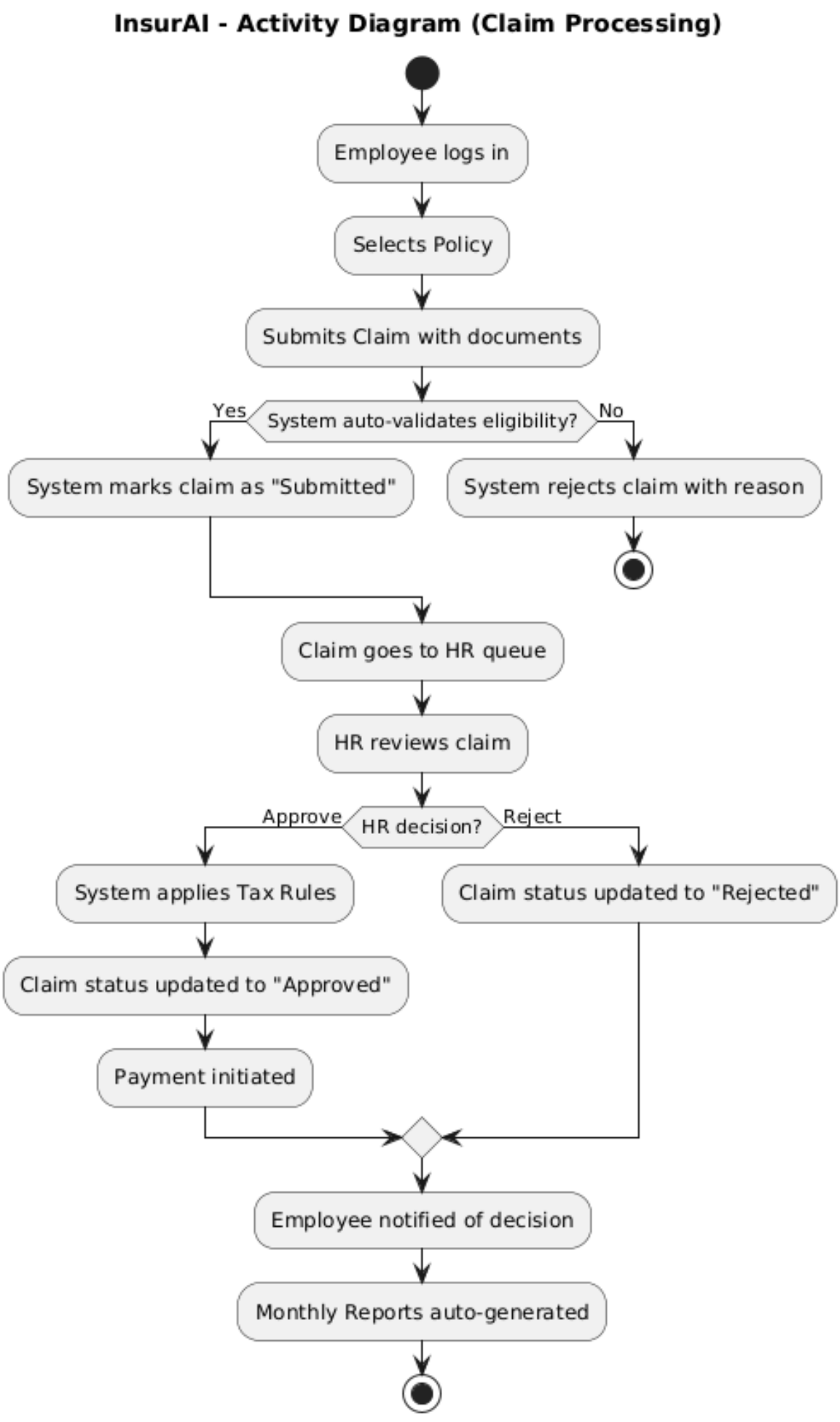
The **InsurAI – Corporate Policy Automation and Intelligence System** is a complete enterprise-grade solution for corporate insurance management. It combines **automation, role-based access, reporting, and AI intelligence** to improve efficiency, reduce fraud, and empower employees with transparency and self-service tools.

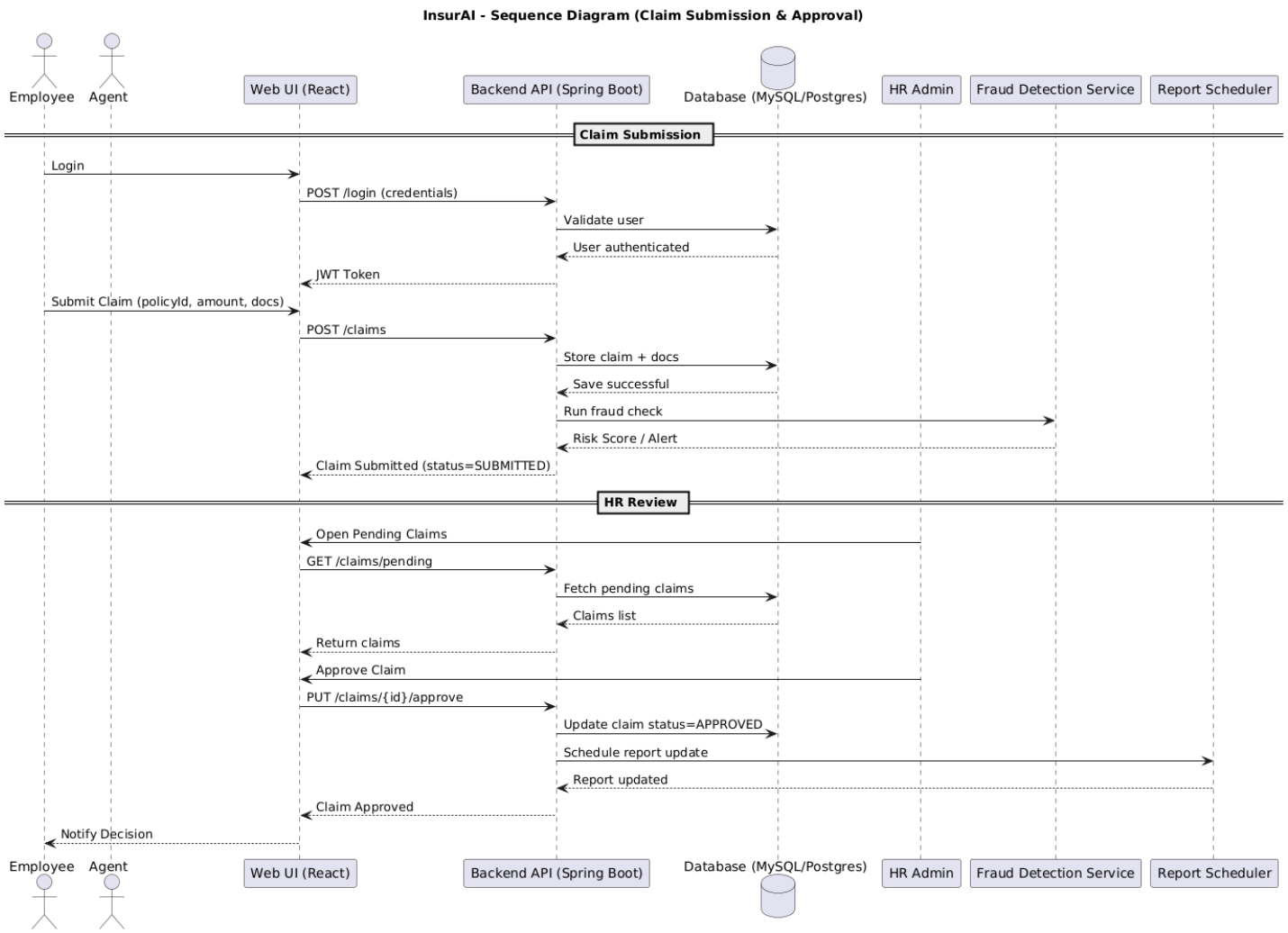
The system is **scalable, secure, and future-ready**, making it suitable for medium to large corporates aiming to modernize their insurance policy management.

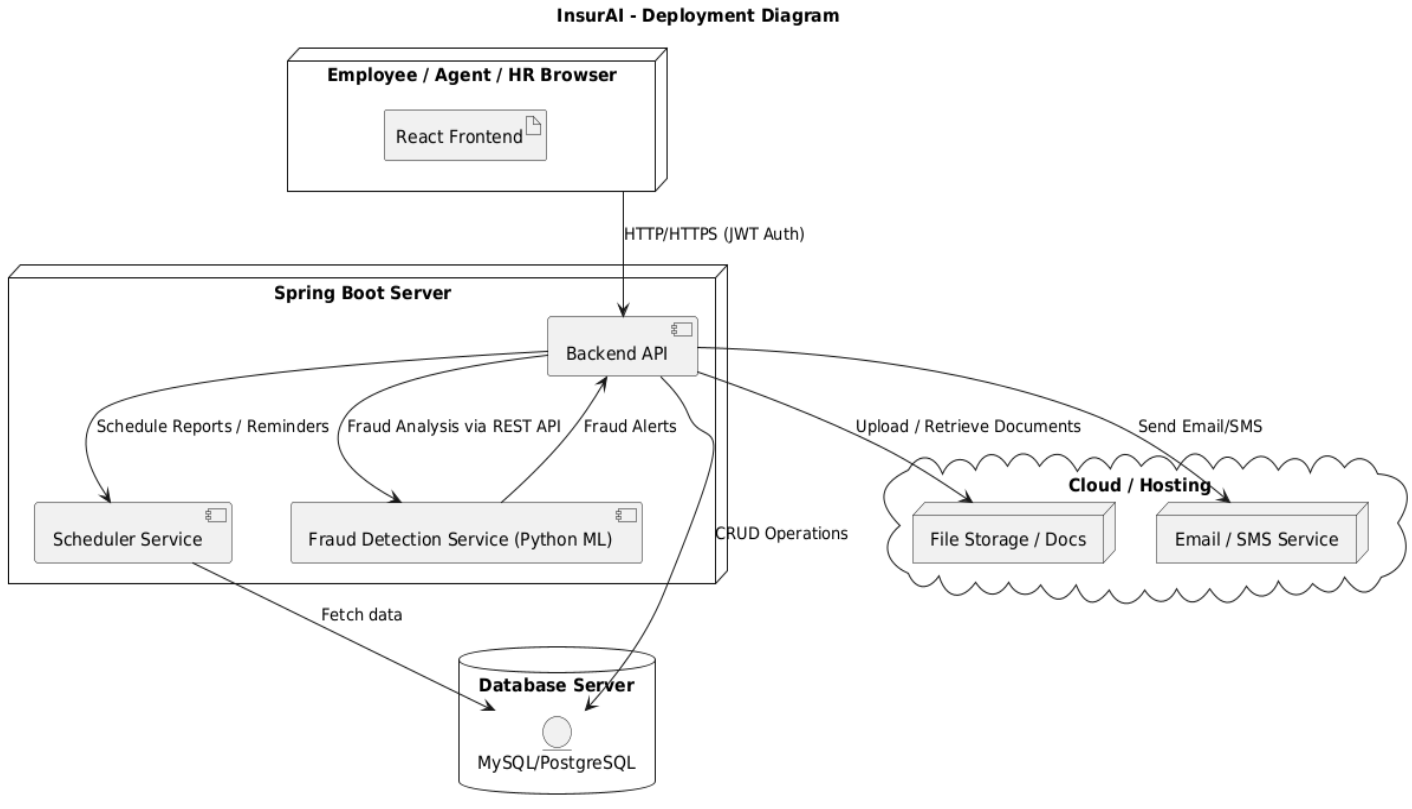
InsurAI - Use Case Diagram




InsurAI - Class Diagram (Domain Model)
User
+userId: Long
+name: String
+email: String
+phone: String
-passwordHash: String
+status: UserStatus
+login()
+logout()
UserStatus
ACTIVE
INACTIVE
LOCKED
PolicyType
HEALTH
ACCIDENT
CORPORATE_BENEFIT
Employee
+employeeId: String
+department: String
+dateOfJoining: Date
+submitClaim(c: Claim)
+trackClaim(id: Long): ClaimStatus
Policy
+policyId: Long
+policyName: String
+policyType: PolicyType
+coverageLimit: BigDecimal
+validityStart: Date
+validityEnd: Date
+renew()
+isActive(): boolean
Agent
+agentId: String
+availability: boolean
+assistEmployee(e: Employee, c: Claim)
+setAvailability(flag: boolean)
HRAdmin
+approveClaim(c: Claim)
+rejectClaim(c: Claim, reason: String)
+generateReport(r: ReportRequest): Report
SystemAdmin
+manageRoles(u: User, r: Role)
+configureTaxRules(t: TaxRule)
+viewFraudAlerts(): List<FraudAlert>
Role
+roleId: Long
+name: String «EMPLOYEE, AGENT, HR, ADMIN»
ClaimStatus
DRAFT
SUBMITTED
UNDER_REVIEW
APPROVED
REJECTED
PAID
ReportType
MONTHLY_CLAIMS
YEARLY_TAX
FRAUD_ALERTS
POLICY_UTILIZATION
1
1..*
submits
covers
0..*
Claim
+claimId: Long
+employeeId: Long
+policyId: Long
+amount: BigDecimal
+submittedOn: Date
+status: ClaimStatus
+validateEligibility(): boolean
+process()
process():
1) validateEligibility()
2) HR review (approve/reject)
3) apply TaxRule (if applicable)
4) set status & notify
has
Document
+documentId: Long
+claimId: Long
+fileName: String
+fileType: String
+storageUrl: String
+uploadedOn: Date
ReportRequest
+type: ReportType
+from: Date
+to: Date
Report
+reportId: Long
+type: ReportType
+generatedOn: Date
+fileUrl: String
FraudAlert
+alertId: Long
+claimId: Long
+reason: String
+riskScore: int
+createdOn: Date
TaxRule
+ruleId: Long
+name: String
+percentage: BigDecimal
+effectiveFrom: Date
+isApplicable(c: Claim): boolean
Permission
+code: String
+description: String
RBAC used by Spring Security/JWT
to protect endpoints.


InsurAI - Activity Diagram (Claim Processing)
Employee logs in
Selects Policy
Submits Claim with documents
System auto-validates eligibility?
Yes
No
System marks claim as "Submitted"
System rejects claim with reason
Claim goes to HR queue
HR reviews claim
HR decision?
Approve
Reject
System applies Tax Rules
Claim status updated to "Rejected"
Claim status updated to "Approved"
Payment initiated
Employee notified of decision
Monthly Reports auto-generated


InsurAI - Sequence Diagram (Claim Submission & Approval)
Employee
Agent
Web UI (React)
Backend API (Spring Boot)
Database (MySQL/Postgres)
HR Admin
Fraud Detection Service
Report Scheduler
Claim Submission
Login
POST /login (credentials)
Validate user
User authenticated
JWT Token
Submit Claim (policyId, amount, docs)
POST /claims
Store claim + docs
Save successful
Run fraud check
Risk Score / Alert
Claim Submitted (status=SUBMITTED)
HR Review
Open Pending Claims
GET /claims/pending
Fetch pending claims
Claims list
Return claims
Approve Claim
PUT /claims/{id}/approve
Update claim status=APPROVED
Schedule report update
Report updated
Claim Approved
Notify Decision


InsurAI - Deployment Diagram
Employee / Agent / HR Browser
React Frontend
HTTP/HTTPS (JWT Auth)
Spring Boot Server
Backend API
Schedule Reports / Reminders
Fraud Analysis via REST API
Fraud Alerts
Upload / Retrieve Documents
Send Email/SMS
Cloud / Hosting
Scheduler Service
Fraud Detection Service (Python ML)
CRUD Operations
File Storage / Docs
Email / SMS Service
Fetch data
Database Server
MySQL/PostgreSQL

InsurAI - ER Diagram