



**Topic** : Unified System for Insurance Claim Management

**Group no** : ITP25\_B4\_96

**Campus** : Malabe

Reg No	Name	Email	Contact No
IT23834774	NATH I M N	it23834774@my.sliit.lk	+94 77 429 0347
IT23836440	FERNANDO PULLE N S	it23836440@my.sliit.lk	+94 70 311 4390
IT23830332	PATHIRANA P U O R	it23830332@my.sliit.lk	+94 71 679 2331
IT23725010	PERERA B I V	it23725010@my.sliit.lk	+94 70 134 6360
IT23828766	SENARATHNA P G R M	it23828766@my.sliit.lk	+94 71 777 6610

#### Contents

<b>Background .....</b>	<b>3</b>
<b>Problem and motivation .....</b>	<b>4</b>
<b>Problem statement .....</b>	<b>4</b>
<b>Current problem and Current process .....</b>	<b>4</b>
<b>Motivation.....</b>	<b>6</b>
<b>Aim and objectives .....</b>	<b>7</b>
<b>Aim (Goal) .....</b>	<b>7</b>
<b>Objectives.....</b>	<b>7</b>
<b>System overview .....</b>	<b>8</b>
<b>Functional Requirements for main stakeholders.....</b>	<b>8</b>
<b>Non-functional requirements .....</b>	<b>9</b>
<b>Technical requirements .....</b>	<b>10</b>
<b>System Diagram .....</b>	<b>11</b>
<b>Main functions.....</b>	<b>12</b>
<b>Additional functions.....</b>	<b>13</b>
<b>Literature review .....</b>	<b>14</b>
<b>Existing Insurance Claim Management Platforms.....</b>	<b>14</b>
i) Available insurance claim management solutions worldwide.....	15
ii) Sri Lankan Insurance Applications.....	16
<b>Summary.....</b>	<b>17</b>
<b>Methodology .....</b>	<b>18</b>
<b>Requirements Engineering Methods.....</b>	<b>18</b>
<b>Design Methods .....</b>	<b>19</b>
<b>Evaluation Method.....</b>	<b>26</b>
<b>References .....</b>	<b>26</b>
<b>Appendix.....</b>	<b>29</b>

## **Background**

We are developing our application, Lumiere: A Unified System for Insurance Claim Management, as per request of our client, a senior business analyst for the Janashakthi Group of Companies<sup>[1]</sup>, a prominent Sri Lankan conglomerate with business interests spanning insurance, finance, real estate, and investment. Some of the companies within the group are, Janashakthi Insurance PLC<sup>[2]</sup>, Janashakthi Finance PLC<sup>[3]</sup>, First Capital Limited<sup>[4]</sup> and Janashakthi Corporate Services Limited<sup>[5]</sup>.

Within the organization, employees are often covered under group insurance policies for life and medical coverage, and in some cases, individual vehicle insurance. These policies are maintained through partnerships with multiple third party insurance providers. Currently, companies within the Janashakthi Group handle most of their insurance claim processing manually, relying on physical document submissions to the HR departments.

Since this is a very time consuming and confusing task, they are seeking a web based solution to standardize the operations occurring within the company more systematically.

We are developing this system based on our client's requirements, with a focus on streamlining the insurance claim process, enabling both employees and HR staff to efficiently manage claims, communicate, and carry out related tasks with improved accuracy and convenience.

**Assignment 1 – Project Proposal****Problem and motivation****Problem statement**

Insurance claim processing within the Janashakthi Group is currently handled through a manual, document-based system and this is often a confusing and grueling task with delays and other such issues, especially due to having to communicate with multiple external insurance providers.

**Current problem and Current process**

When an employee faces an incident that requires an insurance claim, such as a medical emergency, some terminal illness or vehicle damage, they must initiate the claiming process by contacting the HR department of their respective company.

**1. Initial Claim Request**

The employee visits the HR department and fills in the required forms, including bio data, dependent information and details about the incident. Employee also hands over the relevant physical documents such as medical bills, police reports, photos etc. to the HR department. A HR officer verifies the employee's eligibility for the claim based on the assigned insurance policy.

**2. Verification by HR**

HR department cross checks the claim details with internal employee details and their assigned insurance policies. If there's any information missing, the HR officer informs the employee to resubmit the claim request or revise the documents.

**Assignment 1 – Project Proposal****3. Forwarding to Insurance Provider**

Once the claim request is verified, the HR officer forwards the claim request to the relevant insurance provider(s) associated with the employee's insurance policies. This communication is done via email, and it requires HR officers scanning documents manually.

**4. Insurance Provider Processing**

The claims team of the insurance provider reviews the claim, validates the submitted documents, and decided on approval or rejection of the claim request. They also determine the claim amount if the claim request is approved. Their final decision is then sent back to the HR department for confirmation.

**5. Employee Notification**

Once a decision is made, the HR department notifies the employee of the outcome. The payment process is handled independently by the insurance company and a notification is sent to the employee.

**6. Record Maintenance**

All claim details, approvals, rejections, and communications are maintained manually in company files and spreadsheets. There is no centralized system to track claims, view policy usage, or analyze patterns over time.

This current procedure is prone to delays, document misplacement, and inefficiencies, especially when dealing with multiple insurance providers and a high volume of employee claims. This is the primary motivation for developing a Unifies Insurance Claim Management System to streamline and automate the entire workflow.

**Assignment 1 – Project Proposal****Motivation**

Implementing a Unified Insurance Claim Management System will bring significant advantages to both the HR department and employees within the Janashakthi Group. The new system will reduce manual workload, eliminate paperwork related delays, and ensure faster and more transparent claim processing.

User	Benefits
Site Admin	Centralized control over HR accounts and insurance provider data
HR Officer	Organized claim tracking, document handling, and insurer coordination
Regular Employee	Can easily claim life insurance and track status
Executive Employee	Can also claim vehicle insurance and upload documents
Insurance Agent	Faster, digital access to employee claims; easy approval/rejection

**General Benefits**

- Reduced processing time.
- Better management of documents.
- Claim status transparency.
- Centralized system for multiple insurers.
- Ease of use.

**Assignment 1 – Project Proposal****Aim and objectives****Aim (Goal)**

To develop a unified digital platform that effectively manages employee insurance claims streamlining submission, verification, approval, documentation, and communication ensuring fast, transparent, and error reduced claim processing for all corporate stakeholders.

**Objectives**

Step-by-step objectives that will help us achieve this aim:

Step	What we will do
Understand user needs	Interview the HR and survey both the employees and the insurance agents.
Design the system	Design comprehensive process workflows and mockups of systems.
Develop core modules	Implement user authentication, claim submission, document upload, claim tracking, and messaging.
Test and refine	Conduct end-user acceptance test; gather client feedback from all concerned persons.
Deployment and training	Launch the platform and guide our client on how to use the system effectively.

By following the above steps, our system will minimize claim processing time, enhance transparency, and establish a centralized and efficient workflow among all employees and stakeholders involved in the insurance claim process.

## System overview

### Functional Requirements for main stakeholders

Stakeholder	Functional Requirement
Employees	Register and securely log in to the system
	Upload identity verification documents (NIC/Passport)
	View own insurance policies and remaining claim limits
	Submit insurance claims with supporting documents
	Track the status of submitted claims
	Communicate with HR via built-in messaging
	Receive notifications (e.g., claim status updates, document requests)
HR Department	Secure login for HR users
	Review and validate submitted claims
	Forward valid claims to assigned insurance agents
	View employee insurance history and ID verification status
	Respond to employee messages
	Receive alerts when new claims are submitted or pending
Insurance Agents	Secure login for agents
	Access claims forwarded by HR
	Review claim details and uploaded documents
	Approve, reject, or request additional documents for a claim
	Enter final claim decision and reimbursement amount
	Add internal notes visible to HR
Admin	Verify and activate HR and agent accounts
	Manage system settings (policy types, coverage categories, claim limits)
	Monitor logs and analytics for claims processing
	Configure chatbot knowledge base and notification templates
All Users	Enable secure internal messaging between stakeholders
	Implement chatbot support for FAQs and claim guidance
	Handle secure document uploads linked to claim IDs



### Assignment 1 – Project Proposal

	Generate and deliver in-app, email, or SMS notifications
	Track individual and group policy usage (claims vs. total limit)

### Non-functional requirements

Stakeholder	Non-Functional Requirement
Employees	Usability: Interface should be intuitive for claim submission, status tracking, and document upload
	Security: Personal and claim data must be encrypted and protected from unauthorized access
	Performance: The system should allow smooth upload of documents and load dashboards quickly
	Accessibility: System should be accessible via both desktop and mobile devices
	Reliability: Users must be able to access the system without unexpected downtimes
HR Department	Usability: HR dashboard should enable quick filtering, sorting, and viewing of employee claims
	Security: Only authorized HR personnel can view/edit employee data
	Performance: System should handle large numbers of concurrent claim reviews without lag
	Auditability: All claim interactions must be logged for internal auditing
Insurance Agents	Usability: Clear interface for reviewing claims, viewing attached documents, and entering decisions
	Security: Access to only forwarded claims; ability to add notes securely
	Performance: Capable of processing multiple claims in parallel efficiently
	Availability: System should be accessible at all times for real-time decision processing
Admin	Security: Role-based access control for all user types (employees, HR, agents)
	Performance: Tools should allow for real-time monitoring and management
	Maintainability: Easy deployment of feature updates and patches
	Scalability: Ability to support increasing number of users and claim volume
System-Wide	Localization: Interface should support multiple languages if needed

**Assignment 1 – Project Proposal**

	Backup & Recovery: Regular backups of data and fast recovery in case of failure
	Integration: Should integrate seamlessly with external systems (payment gateways, identity verification, etc.)
	Chatbot: Fast and context-aware responses; ability to be updated with new queries
	Notification System: Should deliver alerts instantly with retry mechanisms in case of failure

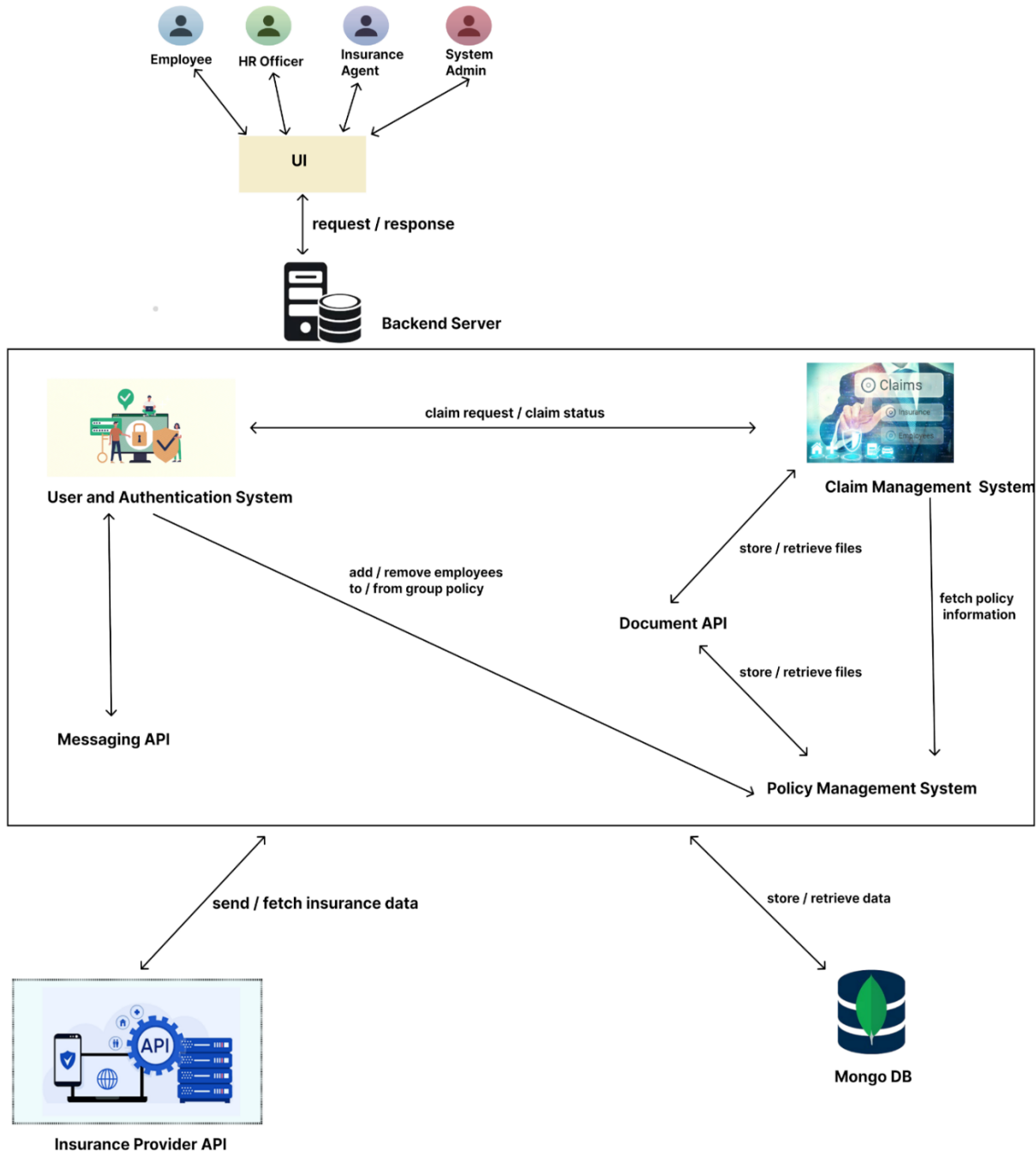
**Technical requirements**

Front end - react.js (Tailwind CSS and bootstrap, material UI for UI)

Backend - node.js (express.js)

Data base - MongoDB

## System Diagram



**Assignment 1 – Project Proposal****Main functions****1. User Management**

The module for user management provides secure and role-based access for all the system users such as site administrators, employees, HR officers, and insurance officers. It provides account creation, login, profile management, and permission management to ensure that every user uses the system based on his or her duties. The system imposes authentication and authorization best practices, ensuring data privacy and operational integrity.

**2. Policy Management**

This module supports management of group and individual life and car insurance policies. It offers policy creation, renewal, cancellation, and categorization. The users can view, edit, or print policy details while administrators and officers can have control over the complete policy life cycle, view active coverage, and manage policyholder records easily.

**3. Claims Management**

The claims management module makes the entire insurance claims process easier - starting from claim initiation by employees to verification by HR and sending, and ultimate assessment and settlement by insurance officers. It has document upload facility, status tracking, auto-routing, and audit trails, ensuring transparency and real-time insurance claim processing.

**4. Document Management API and Frontend**

This module provides a central API and user interface for policy, claim, and user document management. All users can easily upload, classify, search, and retrieve files.

It stores and accesses sensitive documents securely and organizes them, making the administrative functions easier along with regulatory compliance.

**Assignment 1 – Project Proposal****5. Messaging API**

The messaging module supports formalized communication between various stakeholders such as staff, HR administrators, and insurance officers. It enables direct messaging, group chat, and contextual conversation related to claims or policies, which promotes collaboration and decision-making within the system.

**Additional functions****6. AI Chatbot**

An integrated AI chatbot assists individuals by providing answers to common questions, guiding them through claim and policy processes, and delivering quick navigation support. It uses natural language processing to provide 24/7 support, reduce staff dependency on standard tasks, and provide overall user experience improvement.

**7. Notification System**

The notification system informs users with minute-by-minute notifications of significant activities such as approval of claims, renewal of policies, document uploads, and user messages.

It accommodates email as well as in-app notifications to ensure users are informed of all activity and deadlines of significance within the system.

**Assignment 1 – Project Proposal****Literature review**

In the insurance industry, several mobile applications are available for customers to submit individual insurance claims, and most insurance companies in Sri Lanka offer such platforms. However, as of now, there is no solution specifically designed to manage insurance claims from the perspective of a company's HR department, particularly for handling claims made by employees under group insurance policies offered by their employer as an additional benefit.

Large-scale organizations that partner with multiple insurance providers to offer such benefits often face difficulties in managing claims internally. While some companies have requested insurance providers to deliver better systems that support HR teams in processing claims - especially under group policies - no such solution has yet been successfully implemented. Our client, in particular, emphasized the need for a centralized platform to reduce delays, improve accessibility, and simplify the overall process. They also believe such a solution would be in high demand among corporates with large workforces.

**Existing Insurance Claim Management Platforms**

We researched global and domestic claim management platforms. Global platforms like Guidewire<sup>[6]</sup> and FINEOS ClaimVantage<sup>[7]</sup> possess strong functionality but are complex, costly, and geared towards being used by insurance carriers more than corporate HR.

Local Sri Lankan solutions are vendor-specific - developed for one insurer like SLIC, Union Assurance, etc. These packages do not support multi-insurers and therefore turn out to be unsuitable for the firms that deal with claims under multiple insurers. Our solution meets this need by offering a centralized, multi-provider platform that is designed for internal organizational use.

**Assignment 1 – Project Proposal****i) Available insurance claim management solutions worldwide****Guidewire P&C Insurance Solutions**

Guidewire provides Property and Casualty (P&C) insurance products utilized by top global insurance companies such as Allianz Insurance<sup>[8]</sup> and Tokio Marine. One of its most high-profile products, Guidewire ClaimCenter, delivers comprehensive end-to-end automation of the claims process and is particularly suitable for business organizations with a high volume of claims processing. Its strongest points are its excellent scalability, faultless interfacing with other fundamental insurance systems like underwriting, policy, and billing, and an extensive range of analytics and reporting capabilities. It also supports automated alerts to inform stakeholders regarding the status of claims.

Nevertheless, despite its advantages, Guidewire solutions come with a number of shortcomings when used internally in a corporate context. The application is expensive, with paid licensing and considerable investment in installation and training. It is also technically demanding and difficult to implement and maintain, especially for companies with poor IT backup. Most importantly, it is designed essentially for insurance companies rather than corporate or internal HR department use. For non-technical staff, it will be difficult to learn, and therefore not ideal for firms seeking a straightforward and effective way to manage internal insurance claims. In summary, while potent for insurers, Guidewire is too costly and complex for corporates who simply need to automate internal claim handling.

**FINEOS ClaimVantage**

FINEOS, the ClaimVantage acquirer, offers cloud-based software tailor-made for insurance payers and third-party administrators (TPAs). The software is particularly effective when working with life, disability, and absence claims, and supports the automation of workflow and tracking of cases, making it a suitable tool for large insurance operations. Its cloud-native architecture also provides global accessibility from any geographical location, providing the flexibility to support remote work arrangements.

**Assignment 1 – Project Proposal**

While FINEOS ClaimVantage does have its strengths, it is not ideally suited for direct application by corporate HR departments. The system is best designed for external insurance carriers, and there are no integrated customization features for internal corporate processes. Those HR-specific features addressing internal employee database integration or multi-insurer group policy management might be missing or might require some degree of extensive customization. Thus, even though FINEOS is a good software for the insurance business, it is too insurance-oriented and not flexible enough to suit generic corporate claims procedures.

**ii) Sri Lankan Insurance Applications**

Most insurance apps in Sri Lanka belong to a single insurer. While they enable submission of claims and access to policies, they are non-interoperable and do not accept claims from multiple providers. They force corporates to manually process claims on different platforms, leading to inefficiency.

Functionality	HDFC Life App	SLIC App	SLIC B-Connect	Softlogic LifeUP	Lumiere (Proposed)
Policy management	✗	✓	✓	✓	✓
Claim Handling	✓	✓ View only	✗	✓	✓ (Initiation , Tracking)
Notification System	✓	✓	✓	✓	✓
Chatbot Support	✗	✗	✗	✗	✓
Messaging Window (Chat)	✗	✗	✗	✓	✓
HR-Focused Workflow	✗	✗	✗	✗	✓
Multi-provider Claim Support	✗	✗	✗	✗	✓



**Assignment 1 – Project Proposal****Summary**

Although many insurance companies now offer online platforms for individual claim management, these systems are often too technical or fragmented for employees to handle independently. As a result, most employees forward their claims to the HR department manually. This creates a significant burden for HR officers, especially in large organizations with over 10,000 employees and multiple insurance providers covering different types of insurance policies. This highlights the urgent need for a unified and streamlined insurance claim management system tailored to HR operations - one that Lumiere aims to fulfill.

## **Methodology**

### **Requirements Engineering Methods**

#### Methods Used:

Stakeholder Interviews & Meetings - To gather core system requirements directly from client representatives (HR officers, insurance agents, IT coordinators, business analysts).

Observation of Insurance Provider Systems - We studied how existing insurance claim portals function, specially the data flow and process flow in them.

#### Alternatives Considered:

Formal Requirements Specification Documents - These involve writing exhaustive software requirement specification (SRS) documents using IEEE standards.

Questionnaires and Surveys - These can help reach a larger group of users but often lack depth and context, especially when dealing with non-technical users.

#### Justification for Chosen Methods:

The combination of interviews, meetings, and observing similar systems ensures a comprehensive, real-world understanding of both user needs and existing industry standards. It encourages direct communication, allows for follow-up clarifications, and is more agile-friendly compared to rigid formal documentation.



*Figure 1 - client interview at the client's office*



*Figure 2 - Virtual client meeting via Zoom to discuss project scope*

## **Design Methods**

### **Methods Used:**

Figma - For prototypes and UI/UX design.

Canva - For presentation visuals.

Draw.io - For use case diagrams, ER diagrams, system flowcharts.

### **Alternatives Considered:**

Motiff - Modern design tool for prototyping.

MS PowerPoint - A very sophisticated presentation tool.

**Assignment 1 – Project Proposal**

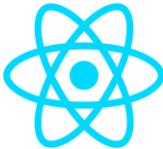




**Justification for Chosen Methods:**

Figma was chosen mainly for familiarity, its ease of use and excellent third party integration. Canva was chosen over MS PowerPoint also for its simplicity and ease of use while being able to deliver highly visually appealing presentation slides.





**Development Tools & Technologies**

**Methods Used:**

MERN Stack - For the development of Lumiere, we have selected the MERN stack, a modern JavaScript-based web development stack consisting of MongoDB, Express.js, React, and Node.js, enabling full-stack development using a single language.

	React - A fast, component-based JavaScript library for building responsive and modular user interfaces.
	Node.js - A server-side runtime environment that enables the execution of JavaScript outside the browser for scalable backend development.
	Express.js - A fast and minimalist web framework for Node.js, used to build efficient APIs and web applications.
	MongoDB with Atlas - A flexible NoSQL cloud database service that stores data as JSON-like documents, offering scalability and real-time access.
	Visual Studio Code - A modern, lightweight code editor with robust features for debugging, version control, and extension support.

### Assignment 1 – Project Proposal

	Postman - A platform for backend route development, testing, and documentation with an intuitive interface.
	Git & GitHub - Git is a distributed version control system, and GitHub is a cloud-based platform for managing and collaborating on code repositories.
	MS Azure - A comprehensive cloud computing platform by Microsoft for deploying, managing, and scaling applications and services.
	ClickUp - A productivity platform that helps teams manage tasks, projects, documents, and workflows in a customizable and collaborative environment.

#### Alternatives Considered:

Spring Boot Stack - A Java-based backend framework that simplifies enterprise application development.

LAMP Stack - A classic web development stack using PHP as the server-side language.

#### Justification for Chosen Methods:

We chose the MERN stack (MongoDB, Express.js, React, Node.js) for its ability to support full-stack development using a single language - JavaScript - across both frontend and backend. This simplifies development, improves team productivity, and allows for faster prototyping. React enables a dynamic and responsive user interface, while MongoDB offers flexible schema design ideal for handling varied insurance claim data. Compared to Spring Boot and PHP stacks, MERN is lighter, easier to learn, and more aligned with modern web development practices, making it the most suitable choice for our project.

**Assignment 1 – Project Proposal****Testing Methods****Methods Used:**

Unit Testing: To test individual components such as APIs and database models to ensure they function correctly in isolation.

Integration Testing: To test the interaction between modules such as user authentication, document upload, and claim processing.

System Testing: To validate the complete workflow of the application and ensure all integrated modules work seamlessly.

User Acceptance Testing: To confirm the system meets real user expectations by allowing actual HR staff, employees, and insurance agents to test features and provide feedback.

**Alternatives Considered:**

Automated Testing Tools: Testing tools like Selenium and Cypress were considered but not implemented due to time and resource constraints.

**Justification for Chosen Methods:**

The selected manual testing methods provide comprehensive coverage for a mid-scale system like ours. Unit and integration testing ensure that each component and its interactions are reliable. System testing verifies the entire flow, while UAT ensures the system is practical and user-friendly. Although automated tools were considered, the manual approach was more feasible within the project timeline.

#### Project Management & Collaboration

##### Methods Used:

**GitHub:** Used for version control, team collaboration on code, issue tracking, and managing branches for each feature.

**Agile Methodology:** The project follows a 12-week Agile sprint plan, allowing iterative development and continuous feedback.

**ClickUp:** Used for sprint planning, task assignment, progress tracking, and overall team coordination.

Sprint	Goal	Tasks	Deliverables
Sprint 1 (Weeks 1-2)	System foundation - design and setup	<ul style="list-style-type: none"> <li>• Design the database schema for all major collections</li> <li>• Set up MongoDB hosting (MongoDB Atlas)</li> <li>• Initialize Git repository and create feature branches</li> <li>• Design base UI/UX</li> <li>• Set up Node.js backend project structure with basic User Management API</li> </ul>	<ul style="list-style-type: none"> <li>• Initial schema design document</li> <li>• Git repository with separate branches</li> <li>• UI/UX prototype for core UI components (Figma)</li> <li>• Running backend with basic structure</li> <li>• Authenticated user flow working via API (bcrypt for password hashing)</li> </ul>
Sprint 2 (Weeks 3-4)	Core backend development - Claims and Policies APIs	<ul style="list-style-type: none"> <li>• Implement Claims API and Policies API (CRUD + validations)</li> <li>• Extend User Management API to support roles and authorization</li> <li>• Build React components for login and static pages</li> </ul>	<ul style="list-style-type: none"> <li>• Functional backend APIs for Claims and Policies</li> <li>• Component-based layout and routing setup in React</li> <li>• Postman collection for existing routes</li> </ul>
Sprint 3 (Weeks 5-6)	Support systems - Documents and Messaging + Feedback	<ul style="list-style-type: none"> <li>• Implement Document Upload API (file storage with metadata)</li> <li>• Implement Messaging API</li> <li>• Integrate document logic into Claims flow</li> <li>• Develop core functional React components</li> </ul>	<ul style="list-style-type: none"> <li>• Document and Messaging APIs ready and tested</li> <li>• Integrated backend and frontend logic</li> <li>• React components for submitting</li> </ul>

### Assignment 1 – Project Proposal

		<ul style="list-style-type: none"> <li>• Collect client feedback on the current implementation</li> </ul>	claims and messaging • Mid-sprint client feedback report
Sprint 4 (Weeks 7-8)	AI and Notifications modules	<ul style="list-style-type: none"> <li>• Build LLM-powered Chatbot API (using gemini API)</li> <li>• Implement Notifications API (email or in-app alerts or both)</li> <li>• Develop React components for chatbot interaction and alerts</li> </ul>	<ul style="list-style-type: none"> <li>• Working chatbot module with sample queries</li> <li>• Notification triggers on claim updates/messages</li> <li>• Chat and notification components integrated in UI</li> </ul>
Sprint 5 (Weeks 9-10)	System integration and refinement	<ul style="list-style-type: none"> <li>• Integrate all frontend and backend modules end-to-end</li> <li>• Secure backend routes with proper role checks</li> <li>• Add validation and clear error feedback on all forms</li> <li>• Improve UI styling and flow consistency</li> </ul>	<ul style="list-style-type: none"> <li>• Fully functional and integrated system</li> <li>• Role-based access control working correctly</li> <li>• Consistent and user-friendly UI</li> </ul>
Sprint 6 (Weeks 11-12)	Final feedback, testing, and deployment	<ul style="list-style-type: none"> <li>• Collect final feedback from client and refine system</li> <li>• Conduct full system testing (Unit, Integration, System)</li> <li>• Fix bugs, polish UX, and finalize features</li> <li>• Deploy the app to a cloud host (Azure)</li> </ul>	<ul style="list-style-type: none"> <li>• Finalized and tested app</li> <li>• Client-approved final build</li> <li>• Live deployed version with access credentials</li> </ul>

#### Alternatives Considered:

Jira was considered for task management and sprint planning.

A centralized version controlling system or a manually managed cloud drive was considered for version control and code hosting.

#### Justification for Chosen Methods:

ClickUp was chosen for its simplicity, ease of use, and all-in-one capability to manage tasks, sprints, and team communication.



**Assignment 1 – Project Proposal**

GitHub, as a distributed version control system, offers greater flexibility, offline access, and collaboration capabilities compared to centralized VCS tools.

**Deployment & Integration****Methods Used:**

We are using Microsoft Azure for deploying and hosting the backend and frontend components, and MongoDB Atlas for cloud-hosted database services to ensure secure, scalable, and accessible data storage.

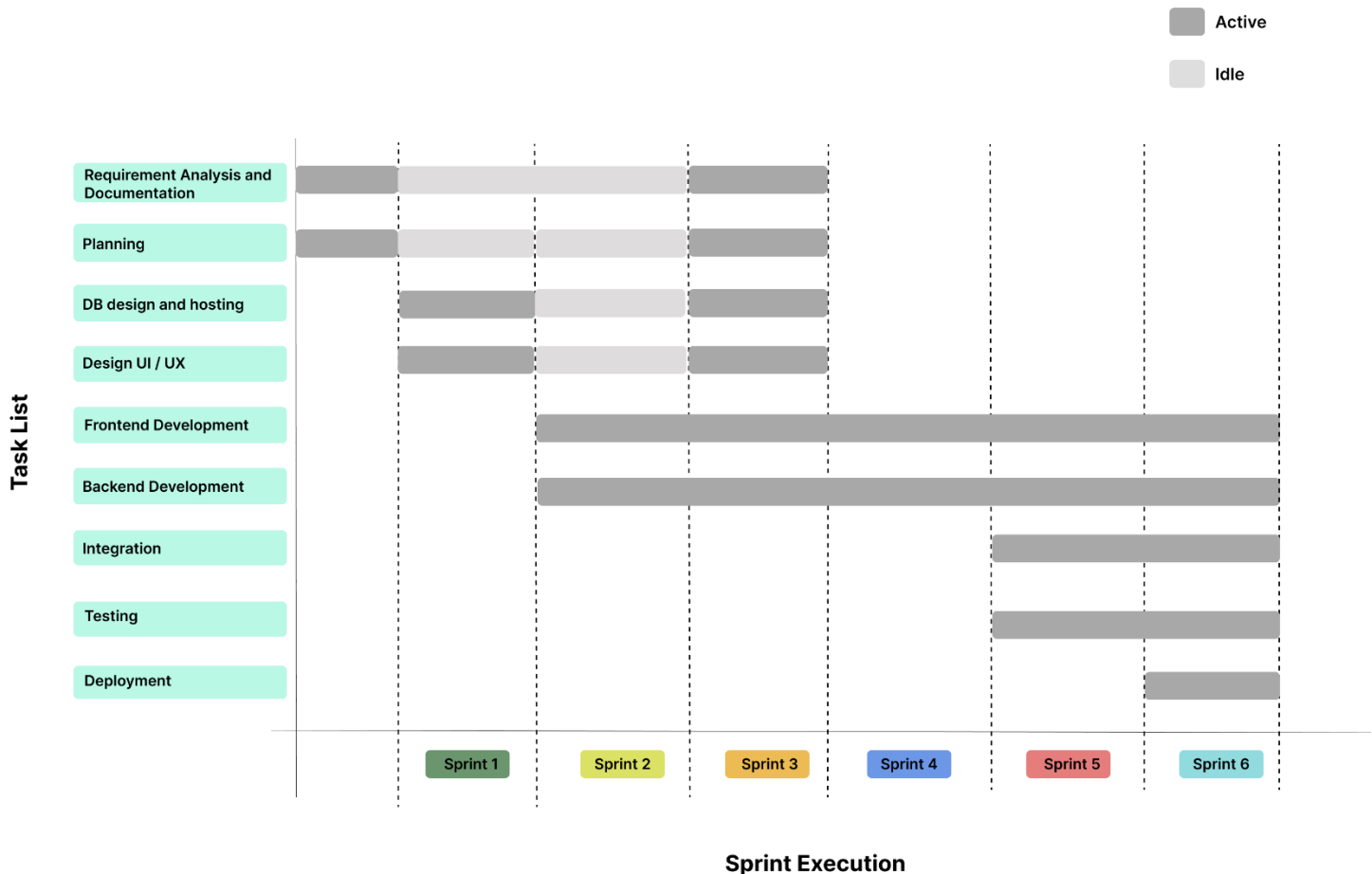
**Alternatives Considered:**

Alternatives considered include Netlify and Vercel for frontend hosting with MongoDB Atlas. We also considered Google Cloud Platform and Amazon Web Services for full stack deployment.

**Justification for Chosen Methods:**

As a privilege of being a SLIIT student, some MS tools are offered to us free of charge. Therefore, we chose Microsoft Azure as our server-side code hosting platform. We decided to use MongoDB Atlas as our database hosting platform because of its ease of use and it is free without requiring credit card information.

## Evaluation Method



A comprehensive evaluation process is essential to validate the effectiveness of our insurance claim management system and ensure it fulfills the expectations of Janashakthi Group and its users. This evaluation will adopt a multi-layered approach, focusing on the following key aspects:

- **Functionality:** Verification that all major system features work as intended - including user registration, group policy handling, claim submission, document upload, messaging, and claim review workflows for HR officers and insurance agents.

**Assignment 1 – Project Proposal**

- **Performance:** Assessment of the system's responsiveness, scalability, and stability, especially under simultaneous usage by employees, HR personnel, and agents.
- **Usability:** Evaluation of the user interface to ensure it is intuitive, accessible, and efficient for all roles, from non-technical employees to administrative staff.
- **Security:** In-depth testing of authentication mechanisms, role-based access control, secure file uploads, and protection against vulnerabilities like unauthorized access or data leaks.

The evaluation will include systematic testing methodologies, such as:

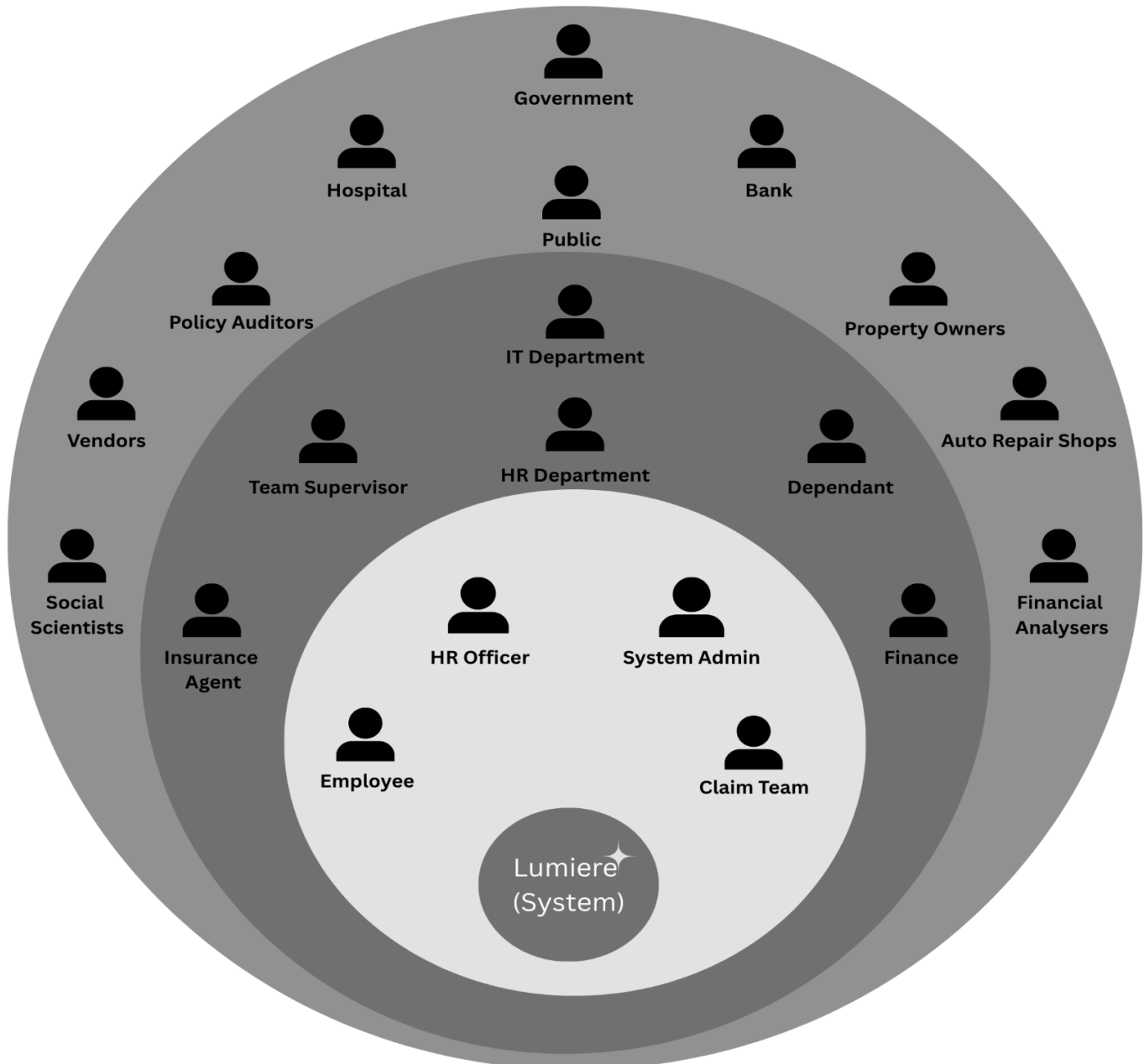
- **Unit Testing:** Ensuring that backend modules like user management, claims processing, and messaging operate correctly in isolation.
- **Integration Testing:** Verifying seamless interaction between the document storage, claims API, and external systems (e.g., insurance provider databases).
- **User Acceptance Testing (UAT):** Involving real users - employees, HR officers, and insurance agents - to test the system in realistic workflows and provide critical feedback for refinement.

Through this structured evaluation, the team aims to identify any gaps or inconsistencies early, allowing iterative improvements that result in a reliable, secure, and user-friendly platform for insurance claim management.

## **References**

- [1]. “Home,” *Janashakthi Group*. <https://www.jxg.lk/>
  
- [2]. “Home - Janashakthi Insurance PLC,” *www.janashakthi.com*. <https://www.janashakthi.com/>
  
- [3]. “Home,” *Janashakthi Finance*, Jul. 28, 2025. <https://www.janashakthifinance.lk/>
  
- [4]. “Home - FirstCap Limited,” *FirstCap Limited*, Aug. 2025. <https://firstcapltd.com/>
  
- [5]. *Jcsl.lk*, 2025. <https://jcsl.lk/> (accessed Aug. 06, 2025).
  
- [6]. Guidewire, “Guidewire,” *Guidewire*, 2019. <https://www.guidewire.com/>
  
- [7]. *Majesco.com*, 2025. <https://www.majesco.com/core-software-insurance-solutions/claimvantage/>
  
- [8]. “Make a Claim | Allianz Australia,” *AU\_site*, 2025. <https://www.allianz.com.au/claims.html>

## Appendixes



*Figure 3 - Onion diagram*

**Assignment 1 – Project Proposal**

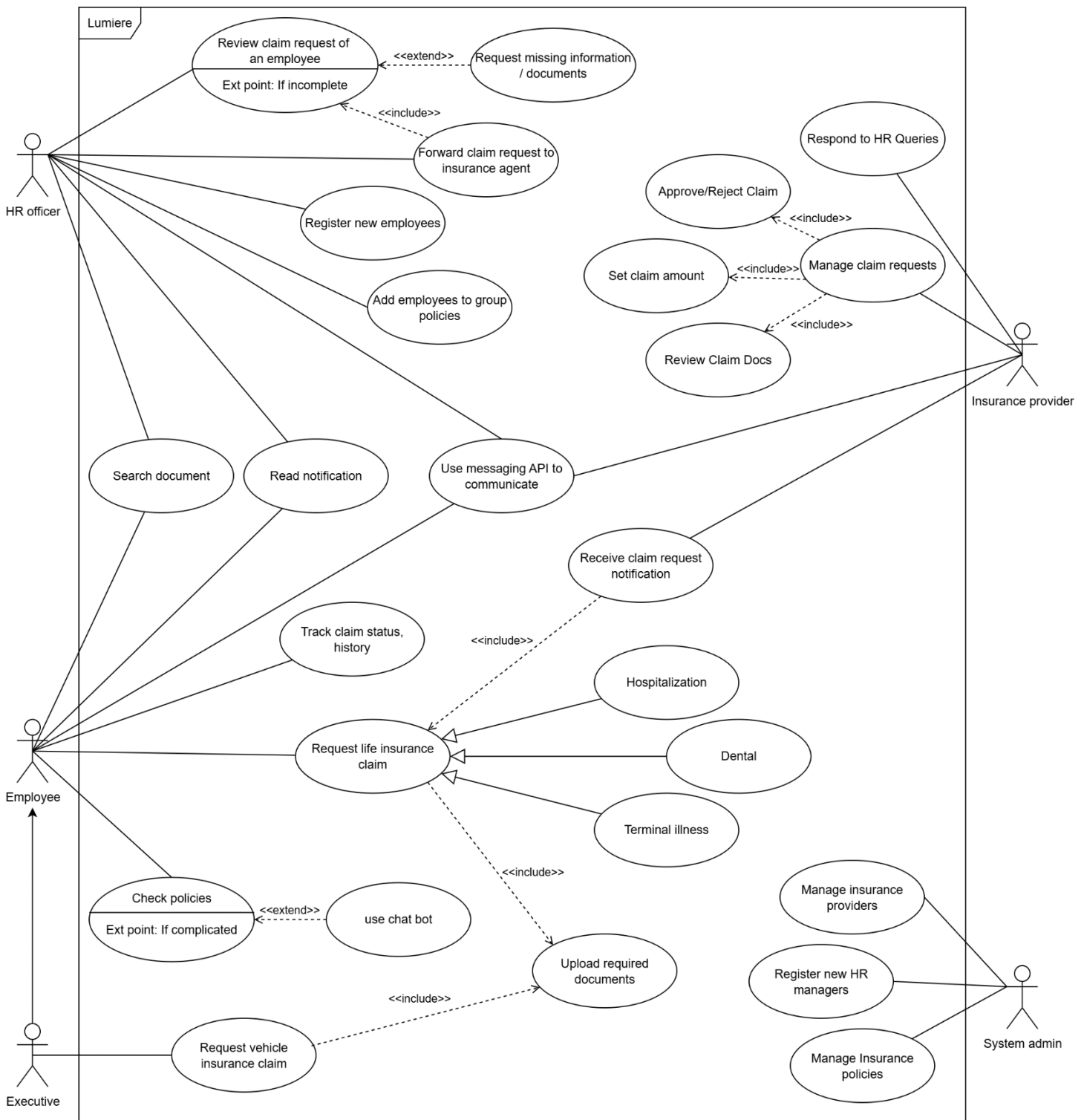


Figure 4 - Usecase diagram

Assignment 1 – Project Proposal

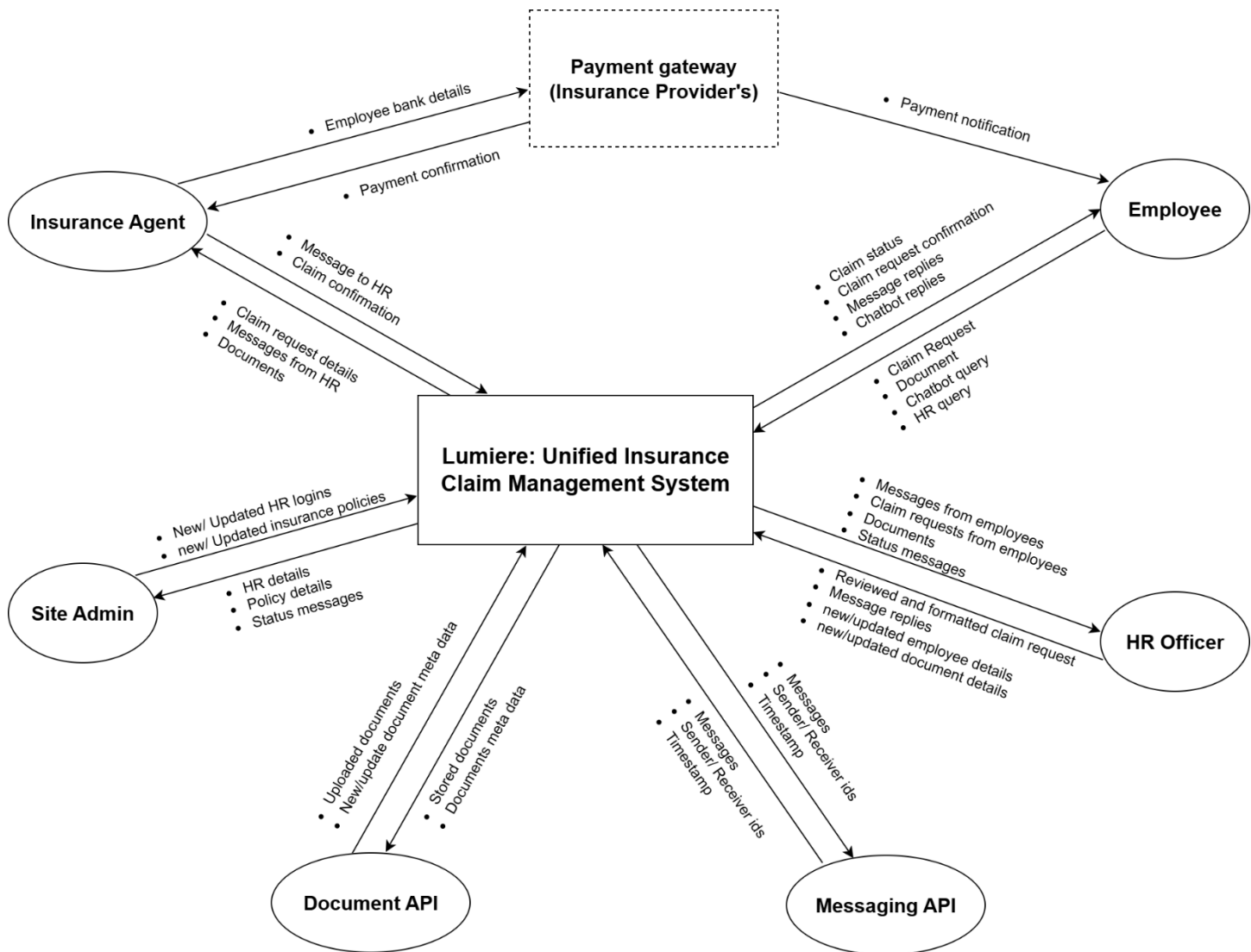


Figure 5 - Data flow diagram

Assignment 1 – Project Proposal

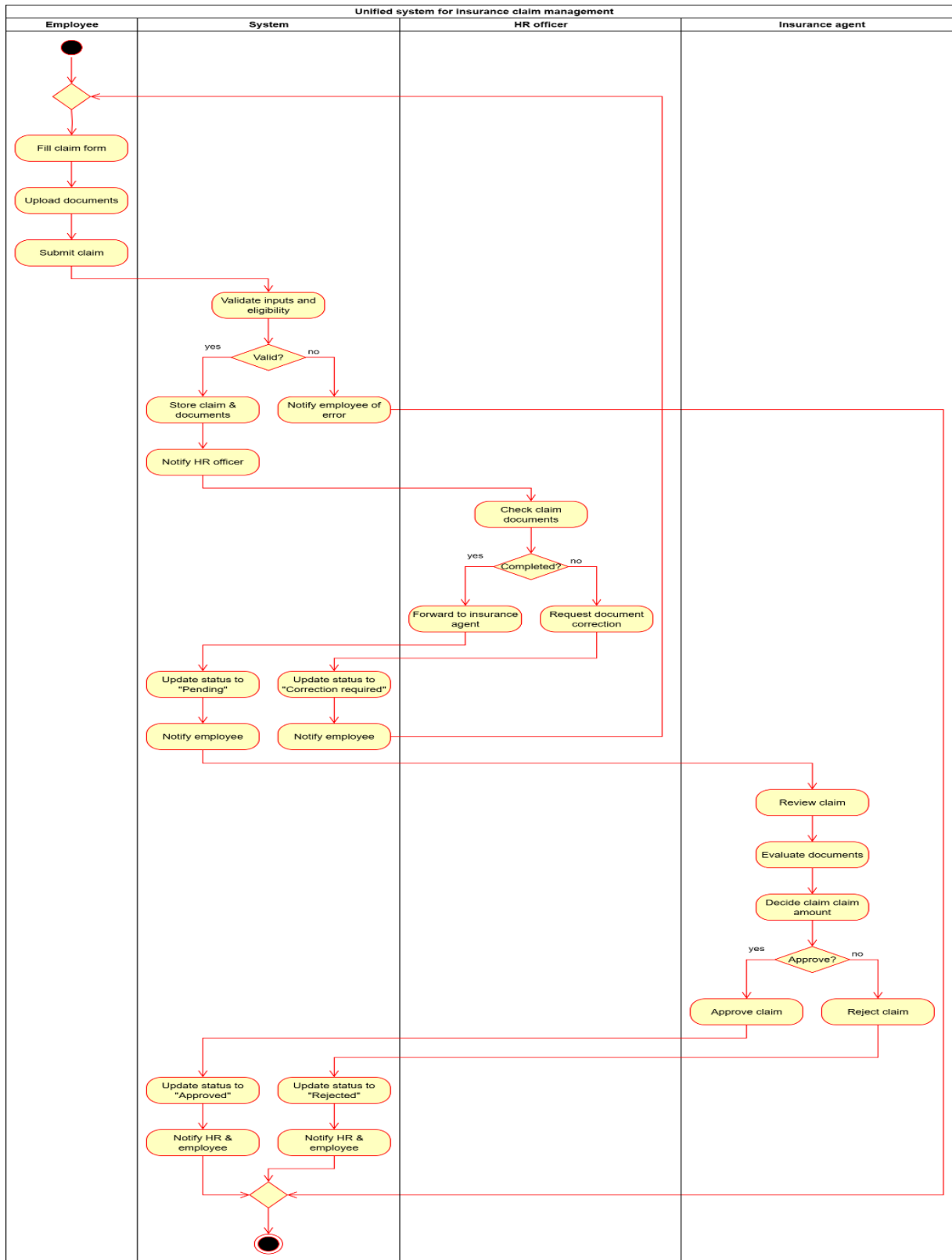


Figure 6 - System activity diagram



**Assignment 1 – Project Proposal**

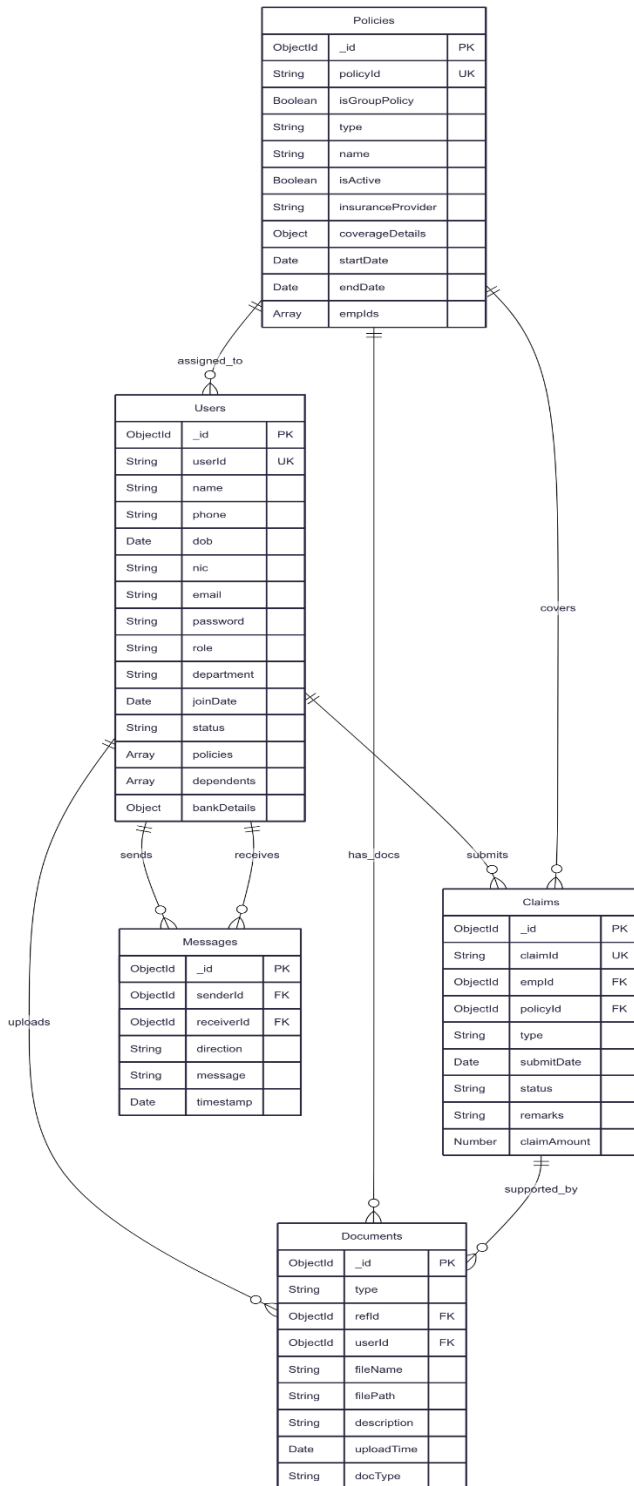


Figure 7 - Mermaid chart for DB schema