Hajj Care

Prepared By :

Ebtihal Alomari

Rafa Alamer

Shahad

Norah Alofie

Maryam Aloraini

Maysan Alsallum

Tala Albishri

Shatha Alharbi

Presented To :

Prof. Fahad Alturise

TABLE OF CONTENTS

01

Software Requirements Specification (SRS)

1 Introduction 04
2 System Overview 05
3 Requirements Engineering 07
4 Functional Requirements 09
5 Non - Functional Requirements 11
6 External Interface Requirements 12
7 UML - Use Case Diagram 13
8 Ethics and Responsibilities 14

02

Software Design Document (SDD)

1 Software Architecture Design 16
2 Process Model 21
3 Data Design 23
  3.1 Data Flow Diagrams (DFD) 23
4 Component Design 25
  4.1 Class Diagram 25
  4.2 State Diagram 26
  4.3 Activity Diagram 27
  4.4 Sequence Diagrams 28
5 Contributions Table 32

03

Project Meetings Notes

1 First meeting 34
2 Second Meeting 35
3 Third Meeting 36
4 Fourth Meeting 37
5 Fifth Meeting 38
6 Sixth Meeting 39
7 Seventh Meeting 40
8 Eighth Meeting 41

Software Requirements Specification (SRS)

1 Introduction

1.1 Purpose

The purpose of this project is to develop a smart health monitoring system to support the safety of pilgrims during the Hajj season. The system aims to help medical teams detect and respond to emergency health conditions in real-time by using smart bands and a mobile application. The app will serve three types of users: Pilgrims and Medical Teams and Admins.

1.2 Product Scope

This system will consist of two main components:

- Smart Bands:

Worn by pilgrims to monitor vital signs such as heart rate, body temperature and blood oxygen level. These bands continuously send data to the system.

- Mobile Application:

A user-friendly app that offers different interfaces based on user roles:

- Pilgrims:
  - View their ID, name, vital signs, and medical history.
- Medical Teams:
  - Receive and view emergency alerts.
  - View detailed case information.
  - Mark emergency cases as resolved.
- Admins:
  - Add and manage medical team accounts.
  - Monitor emergency detections.
  - Generate health and system reports.

The system helps improve response times, save lives, and maintain safety throughout the Hajj journey.

1.3 References

- Saudi Ministry of Health - Hajj Health Guidelines
- Ministry of Hajj and Umrah
- Nusuk Platform
- IEEE Software Requirements Specification (SRS) standard
- Research papers on smart health monitoring systems and IoT in healthcare

2 System Overview

Hajj Care is wearable health monitoring system to support the safety and well-being of pilgrims during Hajj. The idea is based on using smart wristbands that track important health signs like heart rate, body temperature, and oxygen levels.
If the system notices anything unusual, it automatically sends an alert to the nearest medical team. The alert includes the pilgrim's name, ID, health data, and their exact GPS location, so help can arrive quickly.
There's also a mobile app with different interfaces depending on who's using it, so that everyone knows their role and what they're responsible for.

2.1 Stakeholders

Our system involves three main types of users:

- **Pilgrims:** Wear the wristbands and use the app to view their health status and medical history.
- **Medical Teams:** Receive alerts, view emergency cases with pilgrim health details, and handle medical responses.
- **Admins:** Manage the whole platform, the medical teams, and make reports.

2.2 System Environment

The system uses wireless technologies such as Bluetooth, LTE, and GPS to transfer data in real time. All health information is securely stored and processed through a cloud-server. The mobile application offers different interfaces to each user to ensure smooth communication and quick response during emergencies.
Medical teams receive real-time alerts whenever unnormal signs are detected, including the pilgrim's location and health details, so they can act quickly. System admins can intervene if any issues arise to maintain smooth operation. The application is designed to minimize delays and ensure the safety of pilgrims at all times.

2.3 System Features

- **Real-time monitoring of vital signs**

The wearable band keeps tracks heart rate, body temperature, and blood oxygen level for each pilgrim.

- **Automatic emergency detection and alert system**

If any vital sign crosses a dangerous threshold, the system immediately sends an alert to the nearest medical team, including the pilgrim's data and location.

- **GPS-based location tracking**

The system uses GPS to send the exact location along with the emergency alert so medical teams can respond fast.

- **Role-based user interfaces**
  - Pilgrims can view their ID, name, health readings, and medical history.
  - Medical teams can view, manage, and mark emergency cases as resolved.
  - Admins can add medical teams, track system-wide emergencies, and generate reports.
- **Admin screen for monitoring and reporting**

Admins can oversee system activity, view health trends, and generate reports to evaluate emergency cases and team performance.

2.4 System Limitations

- **Battery limitations of wearable devices**

Continuous monitoring may drain the battery quickly, especially during long Hajj days, requiring optimized power consumption.

- **Dependence on stable wireless connectivity**

The system requires Bluetooth, LTE, and GPS to function properly, which may be affected in crowded or remote areas.

- **Data privacy and security requirements**

The system must ensure secure transmission and storage of personal health data, complying with health and data protection standards.

3 Requirements Engineering

3.1 Requirements Gathering Methods

- **Observation:**

Observing the behavior and movement of pilgrims during Hajj to understand their physical needs, environmental challenges, and emergency scenarios.

- **Interviews:**

Conducted with medical personnel, emergency responders, and technical support teams to gather insights on realistic health risks, emergency response processes, and technical limitations.

- **Questionnaires:**

Distributed to potential users (pilgrims and volunteers) to collect feedback on system expectations, preferred features, and ease of use.

- **Domain Analysis:**

Studying existing IoT-based wearable health monitoring solutions and analyzing similar systems used in large-scale crowd events to identify industry standards and best practices.

3.2 Requirements Analysis Methods

- **Use Case Modeling:**

Used to define and visualize the interactions between the system and different user types (pilgrims, medical teams, and admins).

- **MoSCoW Method:**

MoSCoW method is used to prioritize the system features. We divided the requirements into four categories: Must Have, Should Have, Could Have, and Won't Have.

For example, real-time health monitoring and emergency alerts were considered Must Have features because they are critical for the safety of pilgrims.

- **Feasibility Analysis:**

Analyzed the technical feasibility of deploying the system in the Hajj environment, considering power availability, internet connectivity, user diversity, and crowd density.

3.3 Requirements Validation Methods

- **Prototyping:**

Creating wireframes or clickable prototypes of the mobile app interfaces to gather user and expert feedback before full implementation.

- **Review Sessions:**

Organizing structured walkthroughs and review meetings with domain experts, healthcare professionals, and academic supervisors to validate requirements.

- **Requirements Traceability Matrix:**

Mapping all functional and non-functional requirements to their corresponding design elements and test cases to ensure complete validation and verification.

These methods helped ensure that all requirements were clear, testable, and aligned with user needs.

4 Functional Requirements

4.1 User Registration

- The system shall allow all user types Pilgrims, Medical Teams, and Admins to register by entering a valid system-provided ID.
- Upon registration, users shall be directed to their respective role-specific interfaces automatically.
- For pilgrims, the system shall link the registration to their assigned smart band using the device's unique ID.

4.2 Smart Band Connectivity

- The system shall establish a secure connection with the smart band assigned to each pilgrim.
- The system shall retrieve real-time health metrics including:
  - Heart rate
  - Body temperature
  - Blood oxygen level

4.3 Real-Time Health Monitoring

- The system shall receive updated health data from each smart band every few seconds.
- All collected health data shall be securely stored in a cloud-based database.

4.4 Emergency Detection

- The system shall automatically analyze incoming health data in real-time.
- Emergency conditions shall be identified when:
  - Body temperature is abnormally high
  - Blood oxygen is critically low
  - Heart rate is irregular or exceeds safe limits

4.5 Emergency Alert Notifications

- When an emergency is detected, the system shall send a real-time alert to the nearest available medical team.
- The alert shall include:
  - Pilgrim's ID, name and age
  - Type of health issue detected
  - Exact GPS location
  - Current health readings

4.6 Medical Team Interface Features

The medical team interface shall:

- Display a list of active emergency cases.
- Show full case details including vital signs.
- Provide a live map with directions to the pilgrim's location.
- Allow the medical staff to update the case status to "Solved".

4.7 Medical Team Availability Management

- Medical team members shall be able to set their status as "Available" or "Busy".
- The system shall route alerts based on team availability and proximity to the emergency.

4.8 Administrative Controls

Admin users shall be able to:

- Register new medical teams and view their status.
- Assign specific zones or operational areas to teams.
- Monitor system-wide activity and emergency flow in real time.

4.9 Pilgrim Interface Features

The pilgrim's mobile interface shall allow them to:

- View their registered ID and name.
- Monitor current health signs from the smart band.
- Access their medical history (if available).

4.10 Reporting and Analytics

The system shall generate scheduled and on-demand reports that include:

- Number and type of emergency alerts.
- Average response time per case.
- Resolution status of all cases.
- Admins shall use these reports to assess performance and improve service quality.

5 Non-Functional Requirements

5.1 Performance (Fast Response)

- The system shall display real-time health data and emergency alerts instantly, without noticeable delay.
- This is essential for ensuring quick reaction times from the medical teams during critical situations.

5.2 Reliability

- The system shall maintain a 99.9% uptime during the Hajj period to ensure continuous health monitoring and alert delivery.
- System components (server, network, sensors) must be designed to withstand high loads and avoid failure.

5.3 Security

- All health data and user information shall be encrypted during transmission and storage to protect patient privacy.
- Only authorized roles (medical teams and administrators) shall have access to sensitive information such as medical history and emergency cases.
- Pilgrims shall not access data of others under any circumstance.

5.4 Usability

- The system interfaces shall be intuitive and user-friendly.
- Medical teams should be able to respond to alerts quickly and without confusion, even under pressure.
- Admin users should be able to manage teams, view reports, and assign zones easily through a well-organized dashboard.

5.5 Scalability

- The system shall be capable of supporting thousands of concurrent users and devices (smart bands) without affecting performance.
- The cloud infrastructure must scale automatically based on load.

5.6 Availability

- The system shall be accessible and operational 24/7 throughout the Hajj season.
- Emergency alerts and health monitoring must remain active at all times.

5.7 Maintainability and Updatability

- The system architecture shall support future updates, including:
- Bug fixes
- Feature enhancements
- Compatibility with new devices or sensors
- Maintenance should not disrupt the active service, especially during Hajj.

6 External Interface Requirements

6.1 Mobile Application Interface

The smart band must connect to a dedicated mobile application through Bluetooth or an internet connection. This app will display the user's real-time health data and notify medical personnel in case of emergency conditions.

6.2 Medical Emergency System Communication

When a critical health reading is detected (e.g., abnormal temperature, heart rate, or oxygen levels), the system must automatically send an alert, along with the user's information and current GPS location, to the nearest medical response unit.

6.3 GPS System Interface

The smart band must interface with GPS services to continuously track the user's location. This data must be used to accurately guide emergency responders to the pilgrim's location if needed.

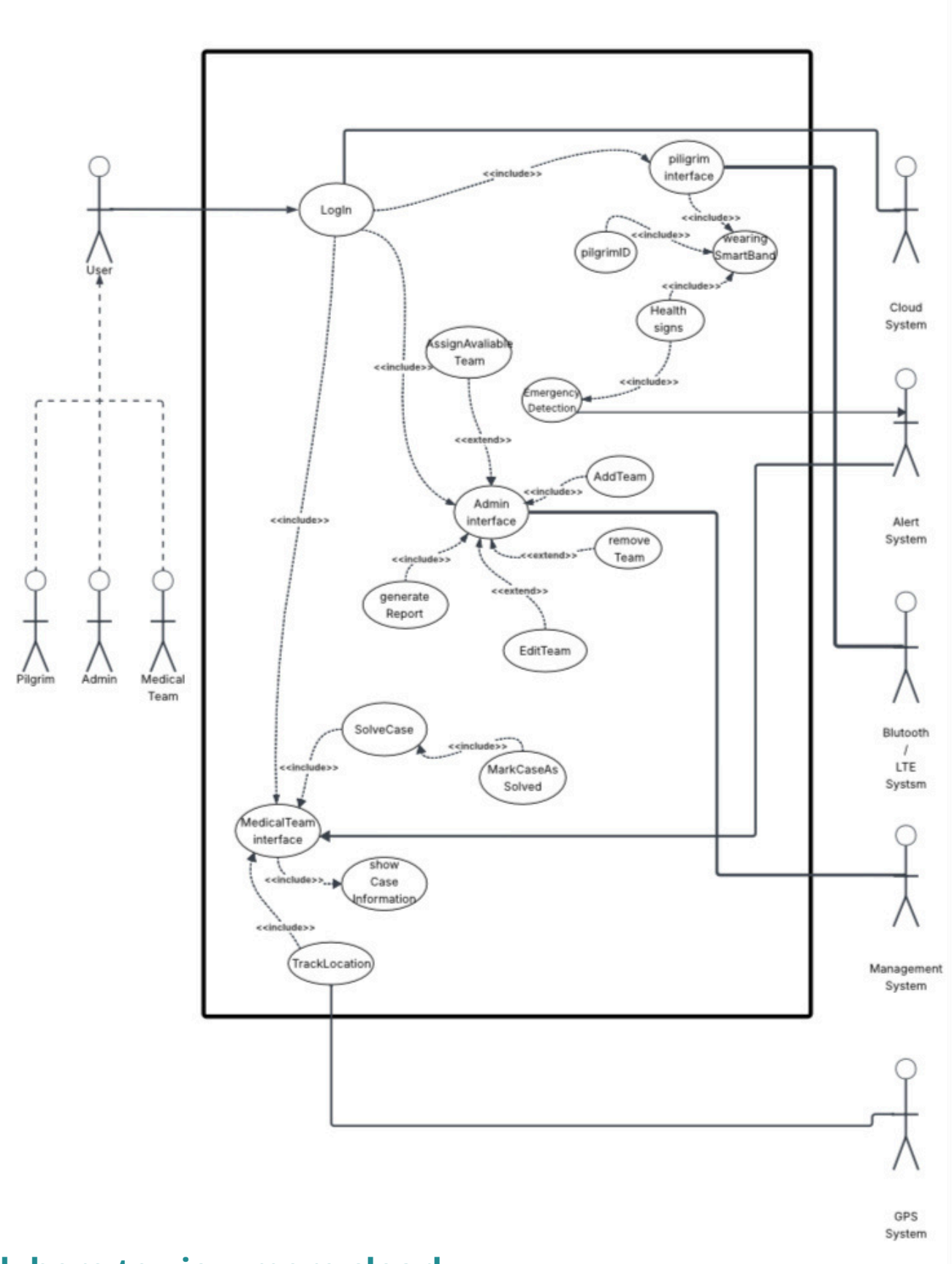
6.4 Cloud Storage Interface

The system must connect to a secure cloud-based database to store user health records, alert history, and sensor data for future reference or analysis.

6.5 Health Sensor Interface

The smart band must integrate with embedded health sensors (e.g., heart rate monitor, thermometer, pulse oximeter) using internal communication protocols (such as I2C or SPI) to gather vital signs and transmit them to the processing unit.

7 UML: Use Case Diagram of Hajj Care System



[Click here to view more clearly]

8 Software Engineering Ethics and Responsibilities:

8.1 Public Safety

We need to make sure the system keeps users safe and protects their health.
Example *:*
The smart bracelet checks heart rate and temperature and oxygen.
If something is wrong it sends an alert fast so the medical team can help quickly.

8.2 Product Quality

The system should work well and be simple to use even in busy places.
Example:
The app gives real-time health monitoring.
If something is wrong, it sends alerts in seconds.

8.3 Honesty and Clarity

We must explain clearly how the system works and respect users' rights.
Example:
Before using the bracelet, pilgrims are told that it will check their health.
They give permission first, so they know and agree with it.

8.4 Professional Judgment

We must make good and careful choices based on what we know.
Example:
We chose only the important features that help save lives, like health alerts and GPS
We did not add hard or confusing things for users
We always think about keeping users safe and making it easy for them.

Software Design Document (SDD)

1 Software Architecture Design

1.1 Requirement 1 : Smart Band Connectivity

Description:

The system shall connect to a smart band worn by the pilgrim. It will retrieve real-time health metrics such as heart rate, body temperature, and blood oxygen level.

Input:

- Raw health signals from sensors embedded in the smart band.

Output:

- Transmitted health data displayed in the medical app and stored in the cloud.

Pre-conditions:

- The smart band must be powered on and paired with the app.
- A stable Bluetooth or LTE connection must be available.

Post-conditions:

- The user's health data is updated in the system.
- Data is stored and available for further analysis or alerts.

Operational Considerations:

Efficient Bluetooth/LTE usage and optimized polling intervals ensure minimal battery and bandwidth usage.

1.2 Requirement 2 : Real-Time Health Monitoring

Description:

The system shall monitor vital health data in real time and maintain updated readings every few seconds.

Input:

- Streaming sensor data (heart rate, temperature, oxygen level).

Output:

- Real-time visualization in the app and logs stored in the cloud database.

Pre-conditions:

- Sensor data must be transmitted without interruption.
- Internet or Bluetooth connectivity must be active.

Post-conditions:

- The application displays current health status.
- Alerts may be triggered based on threshold values**.**

Operational Considerations:

Data handling is optimized to minimize overhead. Monitoring is resilient against brief connectivity losses through local buffering and reconnection mechanisms.

1.3 Requirement 3 : Emergency Detection

Description:

The system shall analyze incoming health data continuously and detect emergency conditions such as high temperature, low oxygen, or abnormal heart rate.

Input:

- Live data from health sensors.

Output:

- Emergency status flag, triggering notification logic.

Pre-conditions:

  - Threshold rules must be predefined.
  - Sufficient data must be received to calculate abnormal conditions.

Post-conditions:

  - Emergency alerts are generated and queued.
  - System status reflects critical condition.

Operational Considerations:

Smoothing filters applied to reduce noise and false alerts.

1.4 Requirement 4 : Emergency Alert Notifications

Description:

Upon detecting a health emergency, the system shall send an instant alert to the nearest available medical team.

Input:

- Emergency flag
- user profile
- GPS coordinates.

Output:

- Alert notification with ID, issue type, and location.

Pre-conditions:

- Medical teams must be registered and available.
- Device must have access to GPS services.

Post-conditions:

- Medical staff receives the alert.
- The case is logged in the alert history.

Side Effects:

- Network latency may delay alert delivery.
- Possible privacy risk if GPS data is not encrypted.

Operational Note:

Uses MQTT protocol with message delivery acknowledgement.

1.5 Requirement 5: Medical Team Application Features

Description:

The app shall allow medical teams to view alerts, locate patients, and mark cases as resolved.

Input:

Incoming alerts and case information.

Output:

Map view with directions, case status update.

Pre-conditions:

- Medical team must be logged in and online.
- GPS services must be active.

Post-conditions:

- Case status updated as “Solved” in the system.
- Historical log updated.
- **Side Effects:**

Potential UI overload if multiple alerts arrive simultaneously.

1.6 Requirement 6 : Administrative Controls

Description:

System administrators shall manage medical teams, including adding, editing, or removing team profiles and assigning coverage zones.

Input:

Admin actions via dashboard (e.g., add team, assign region).

Output:

Updated system configuration for team assignments.

Pre-conditions:

- Admin must be authenticated.
- Network access must be available.

Post-conditions:

- Medical team data updated in real-time.
- Dashboard reflects changes.

Side Effects:

- Misconfiguration can affect alert routing accuracy.

1.7 Requirement 7: Medical Team Availability Management

Description:

Medical staff shall update their availability status (Available/Busy), which helps in intelligent alert routing.

Input:

Team status toggle input from the app.

Output:

Updated availability list for routing engine.

Pre-conditions:

- Medical staff must be signed into the system.
- Device must sync to cloud.

Post-conditions:

- Alert engine routes to available teams only.
- Admins can view live status.

Side Effects:

Inaccurate availability data may delay response.

1.8 Requirement 8: Reporting and Analytics

Description:

The system shall generate both real-time and scheduled reports about alerts, response times, and resolutions.

Input:

Historical data from cloud database.

Output:

PDF or dashboard report for admin review.

Pre-conditions:

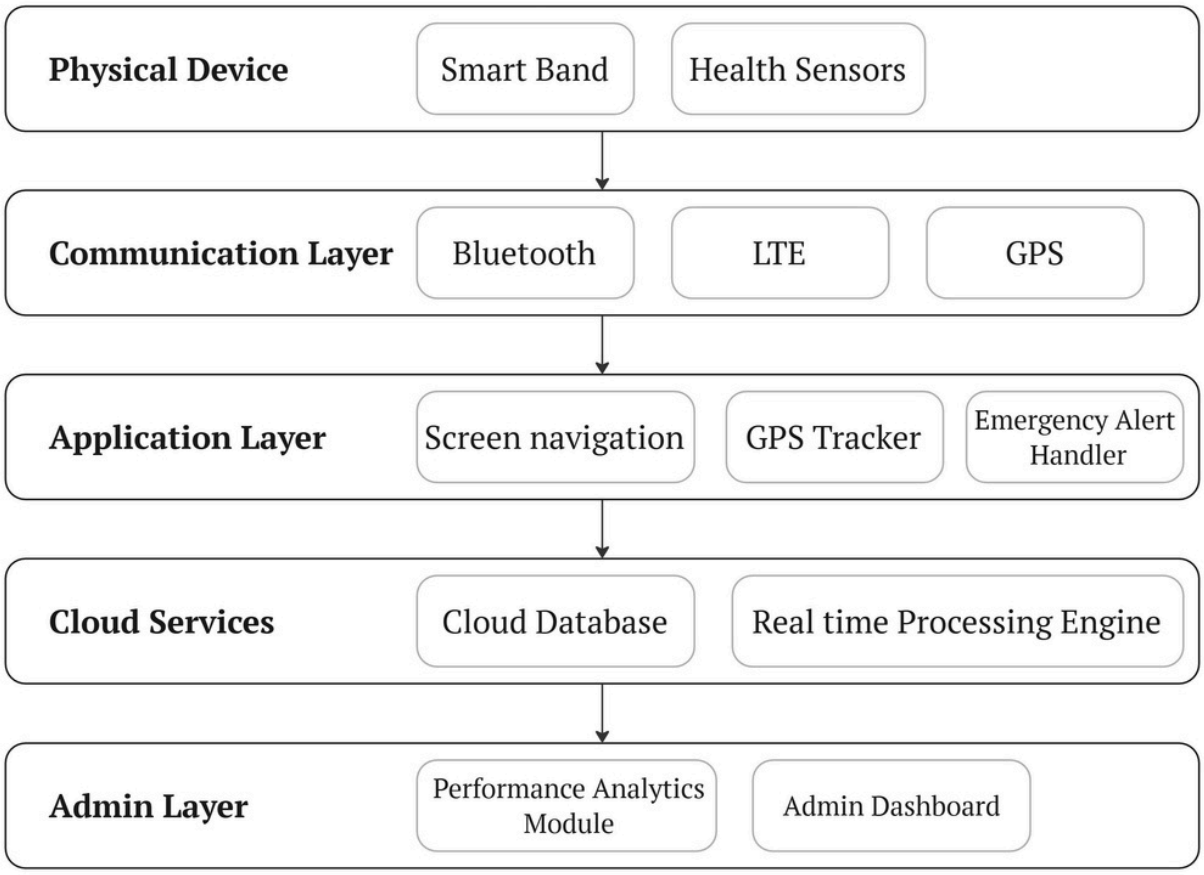
- Data must be logged consistently.
- Admin must initiate report generation.

Post-conditions:

- Reports are stored or downloaded.
- Used in evaluation and improvement cycles.
- **Side Effects:**

Report generation may temporarily slow down the dashboard.

1.9 Layered architecture:



2 Process model

Our project Hajj Care is a smart wearable system designed specifically for hajj pilgrims.

The system consists of a wearable bracelet designed to perform the following tasks:

- It measures the vital signs of the pilgrim such as blood pressure, pulse, body temperature, and more.
- Sends alerts to medical teams automatically when a change in normal vital signs is detected.
- Shares the pilgrim's location via GPS with the nearest medical unit.
- The band is connected to the official Nusuk mobile application, which handles data visualization, tracking, and coordination with healthcare staff.

We have selected the **Agile process model**, using the Scrum framework as our development approach. This choice is based entirely on the nature and structure of the Hajj care project:
The system involves multiple interconnected components (band, mobile app, GPS, emergency alerts, medical response.

Scrum enables us to build and test each part in separate, manageable sprints, such as:

- One sprint for monitoring vital signs
- Another sprint works on the alert mechanism for medical teams.
- A sprint for tracking GPS location and relaying it to emergencies.
- Another sprint to connect the team with the Nusuk application.

This method encourages us to work collaboratively and provides an organized way to perform tasks and track progress.

The development will be divided into several sprints. Each sprint will focus on building one key functionality:

Sprints	Main task
sprint 1	Sensor data collection and threshold settings (fever, high BP)
sprint 2	Emergency alert system and connection to medical response unit
sprint 3	GPS location transmission and mapping integration
sprint 4	Mobile app interface and dashboard
sprint 5	Testing, integration with Nusuk, and final refinement

Agile Scrum is the ideal model for our project, Because it supports:

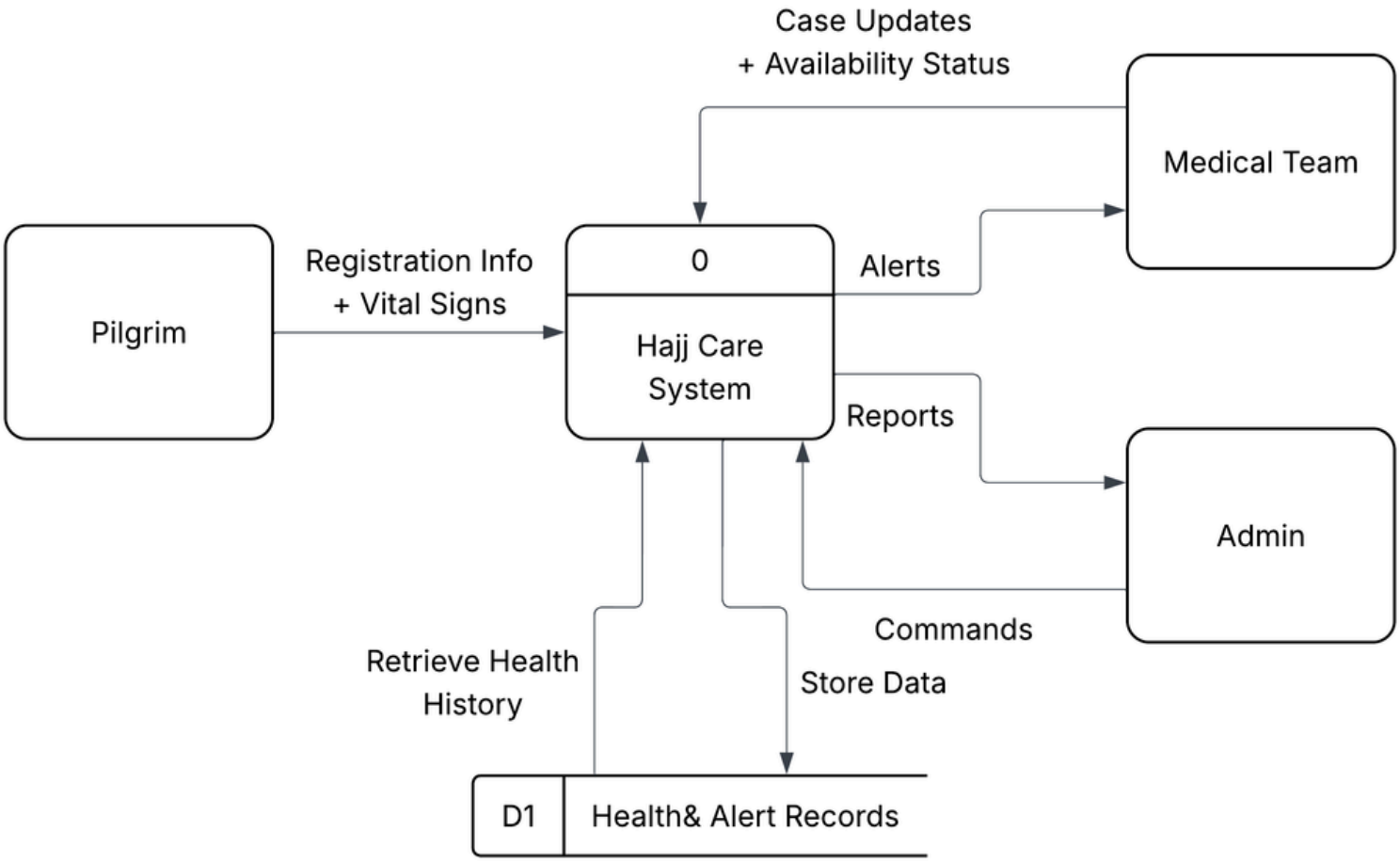
- Enhanced teamwork and progress tracking.
- Independent testing of each part before final integration.
- Clear structure for delivering functional modules.
- flexibility in building and modifying the system during the development stages and after its completion.

By applying Scrum to the pilgrimage care project, we aim to ensure that the system is reliable, scalable, and ready to assist pilgrims in critical health situations.

3 Data Design

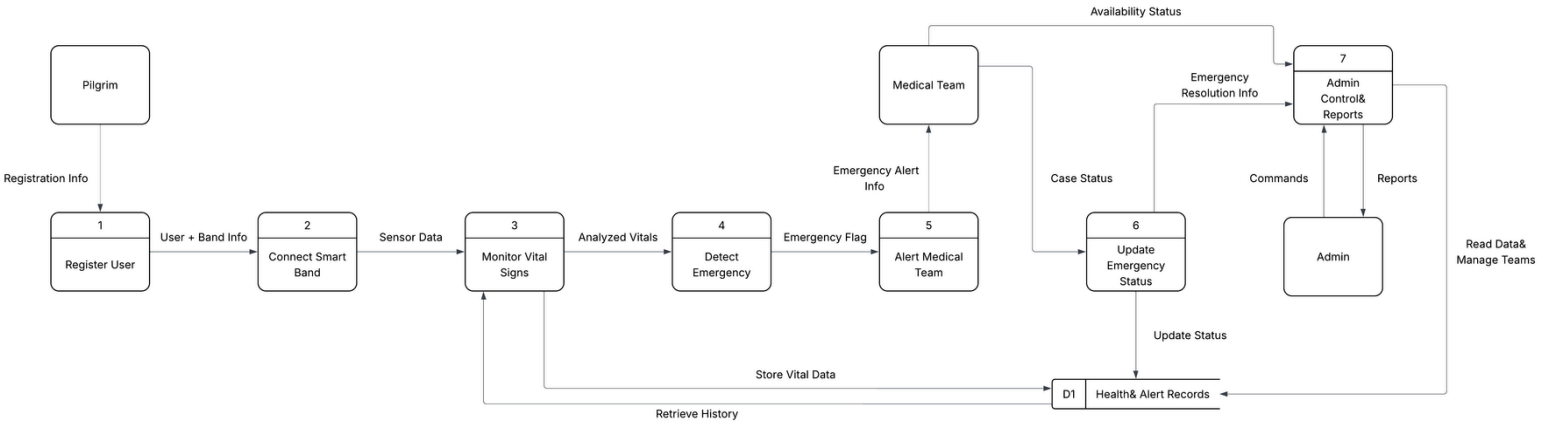
3.1 Data Flow Diagrams (DFD)

3.1.1 DFD Level 0



[Click here to view more clearly]

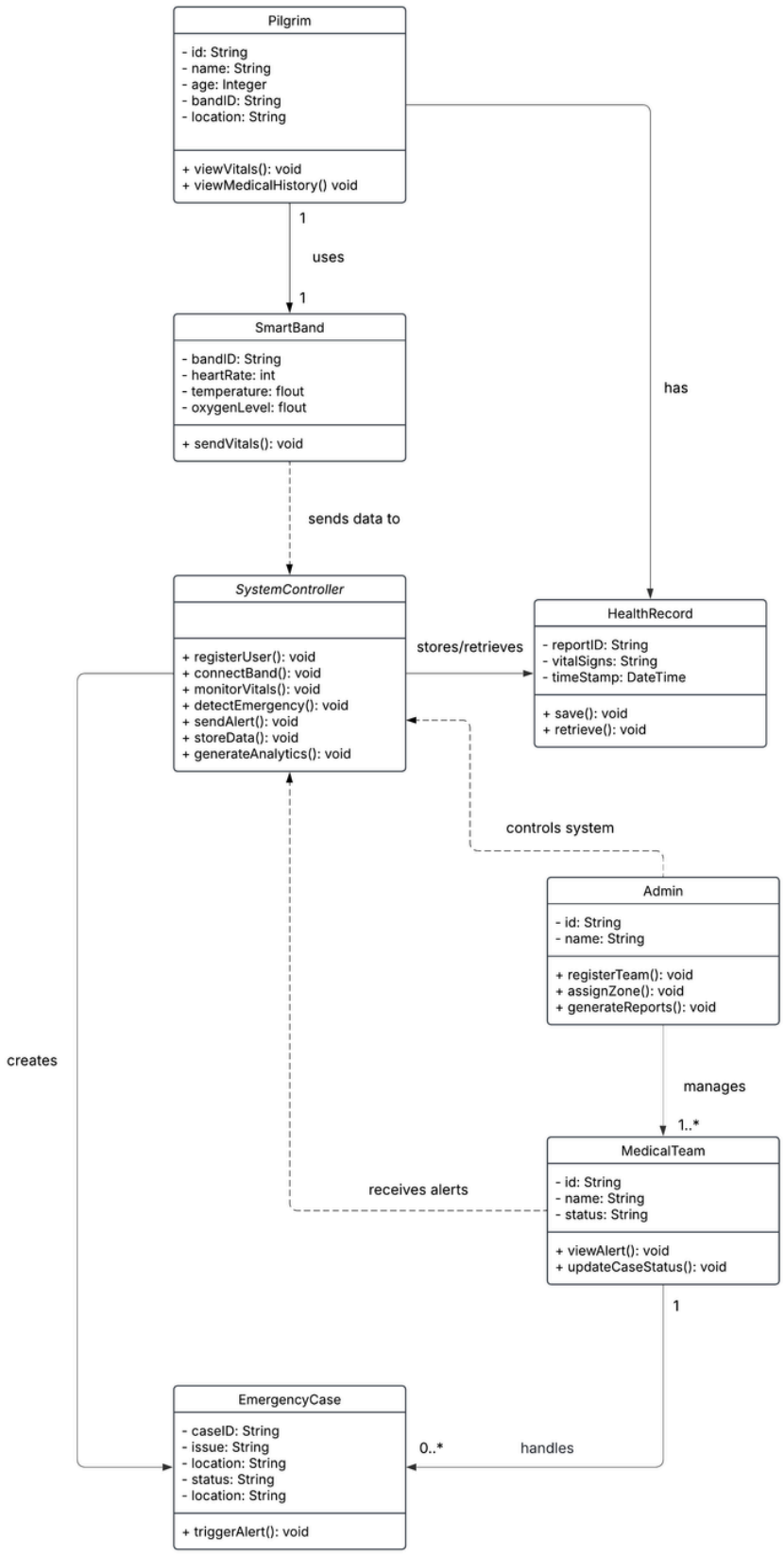
3.1.2 DFD Level 1



[Click here to view more clearly]

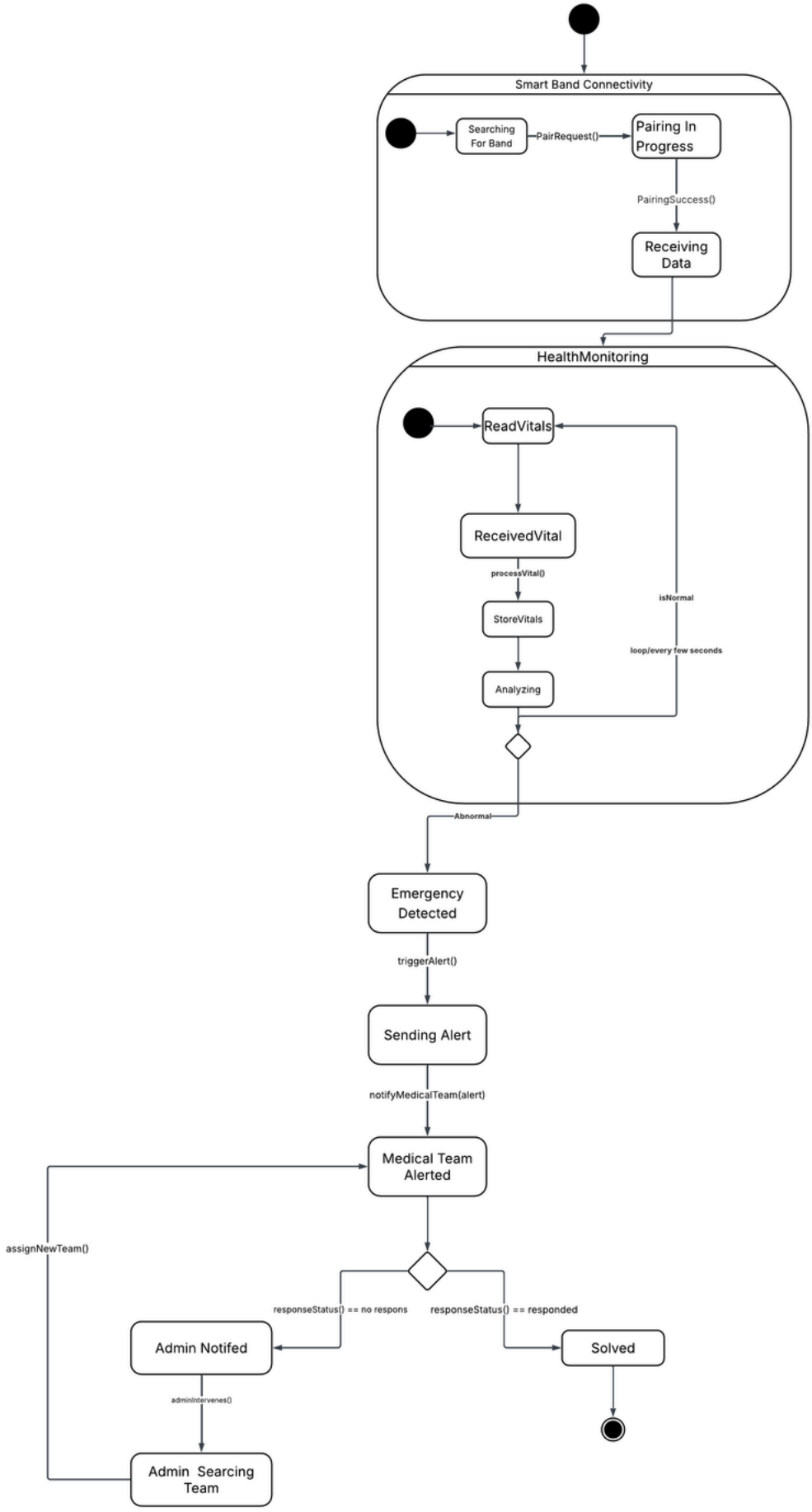
4 Component Design

4.1 Class Diagram



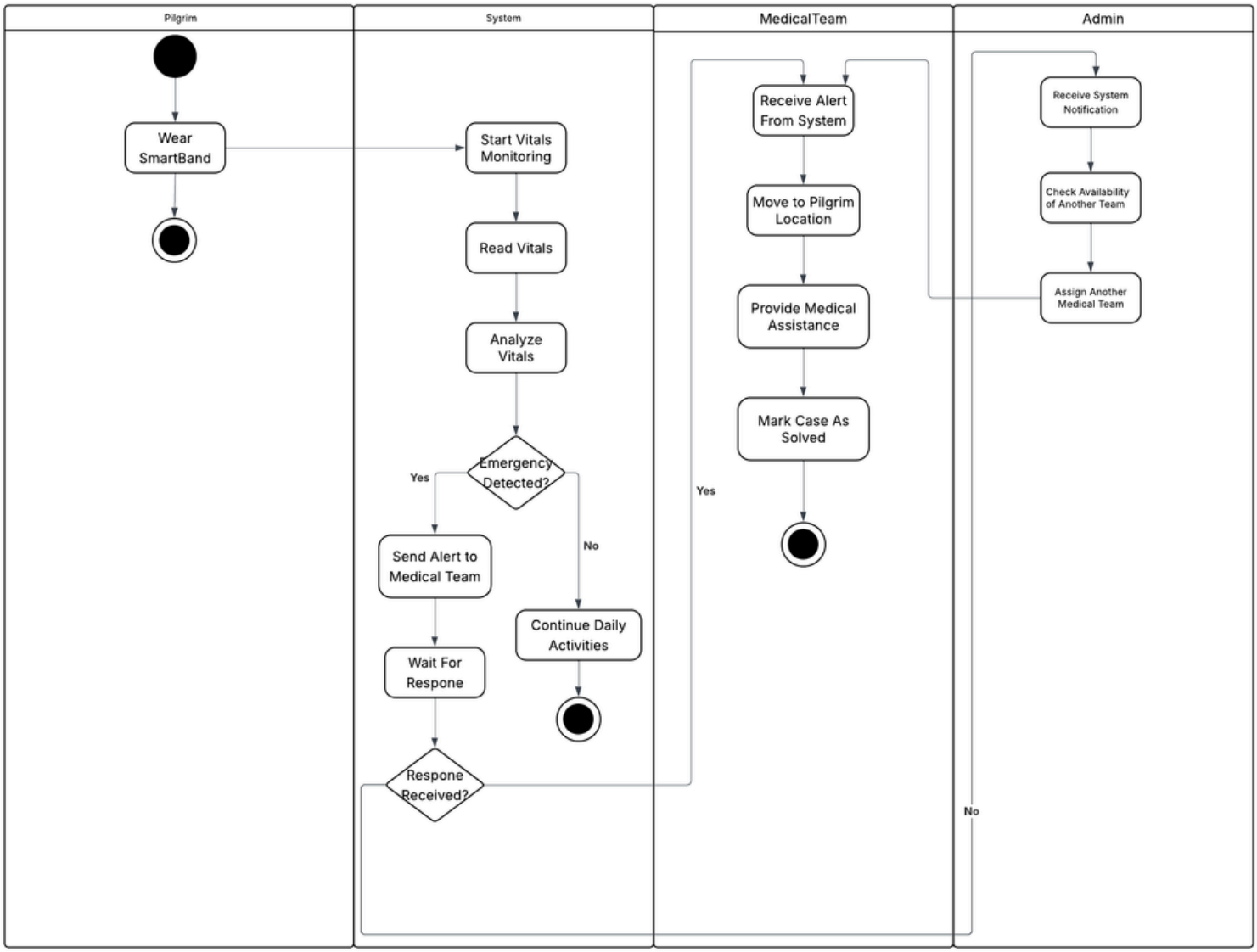
[Click here to view more clearly]

4.2 State Diagram



[Click here to view more clearly]

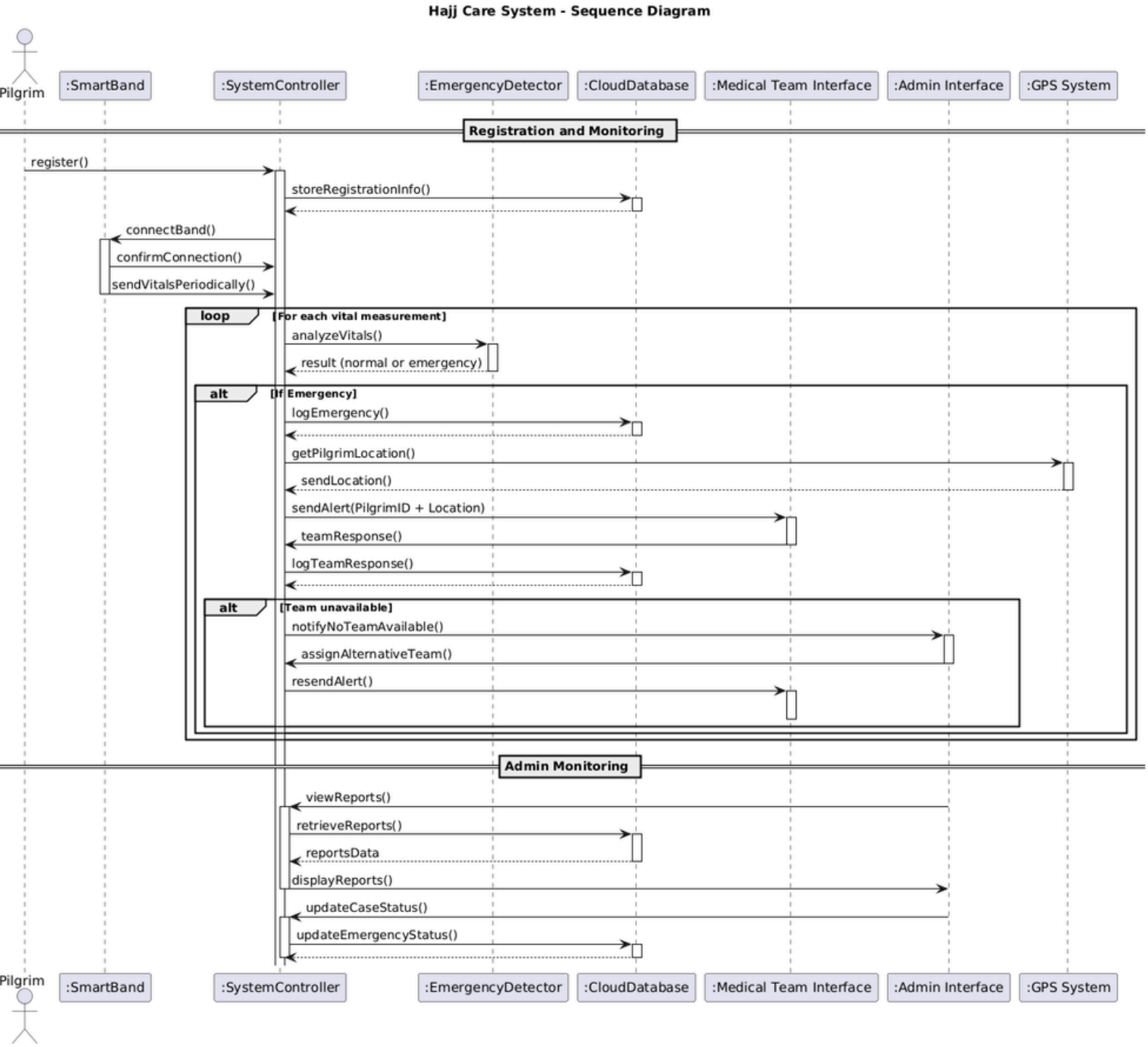
4.3 Activity Diagram



[Click here to view more clearly]

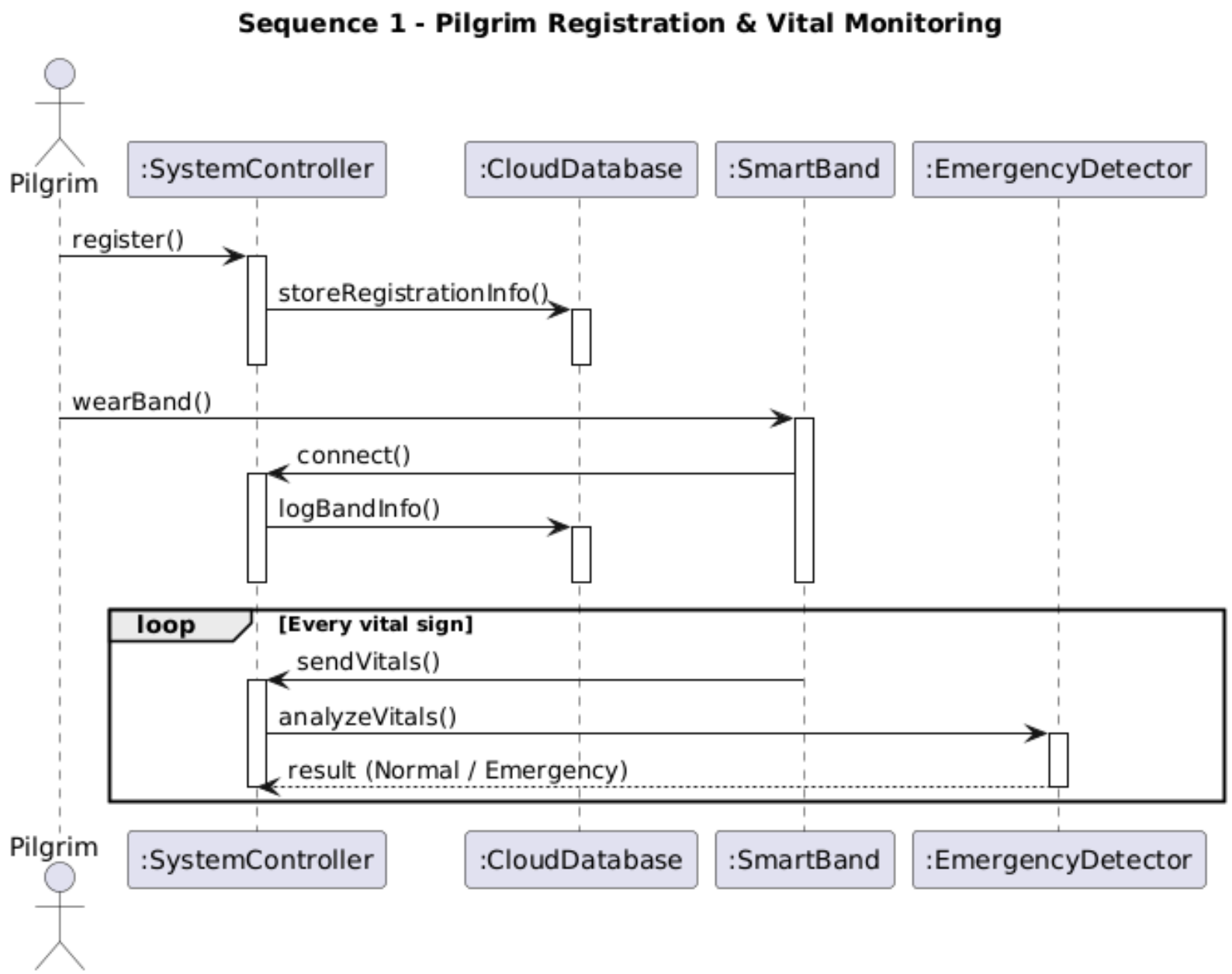
4.4 Sequence Diagrams

4.4.1 General Sequence Diagram



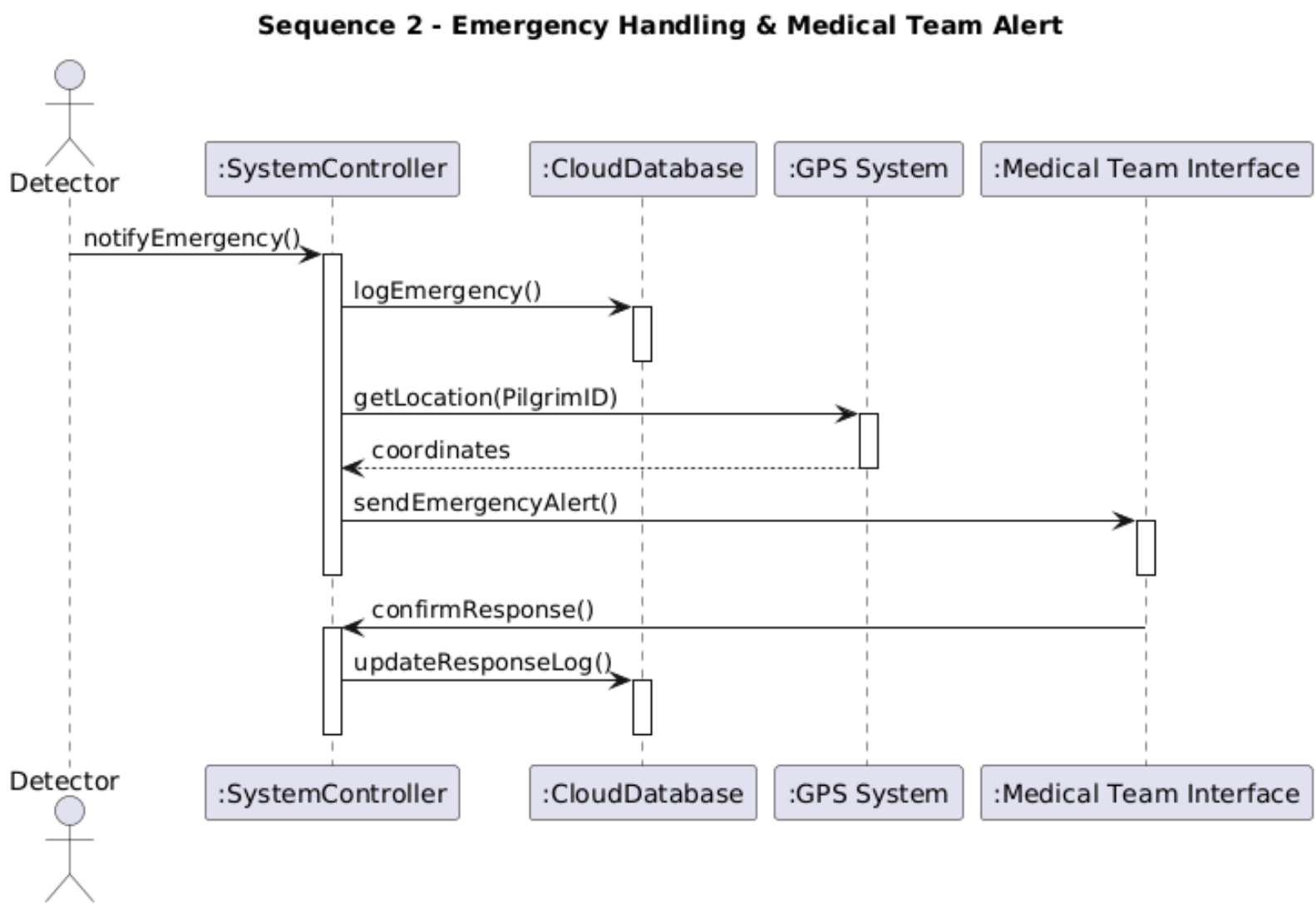
[Click here to view more clearly]

4.4.2 Registration & Vital Signs Monitoring Sequence Diagram



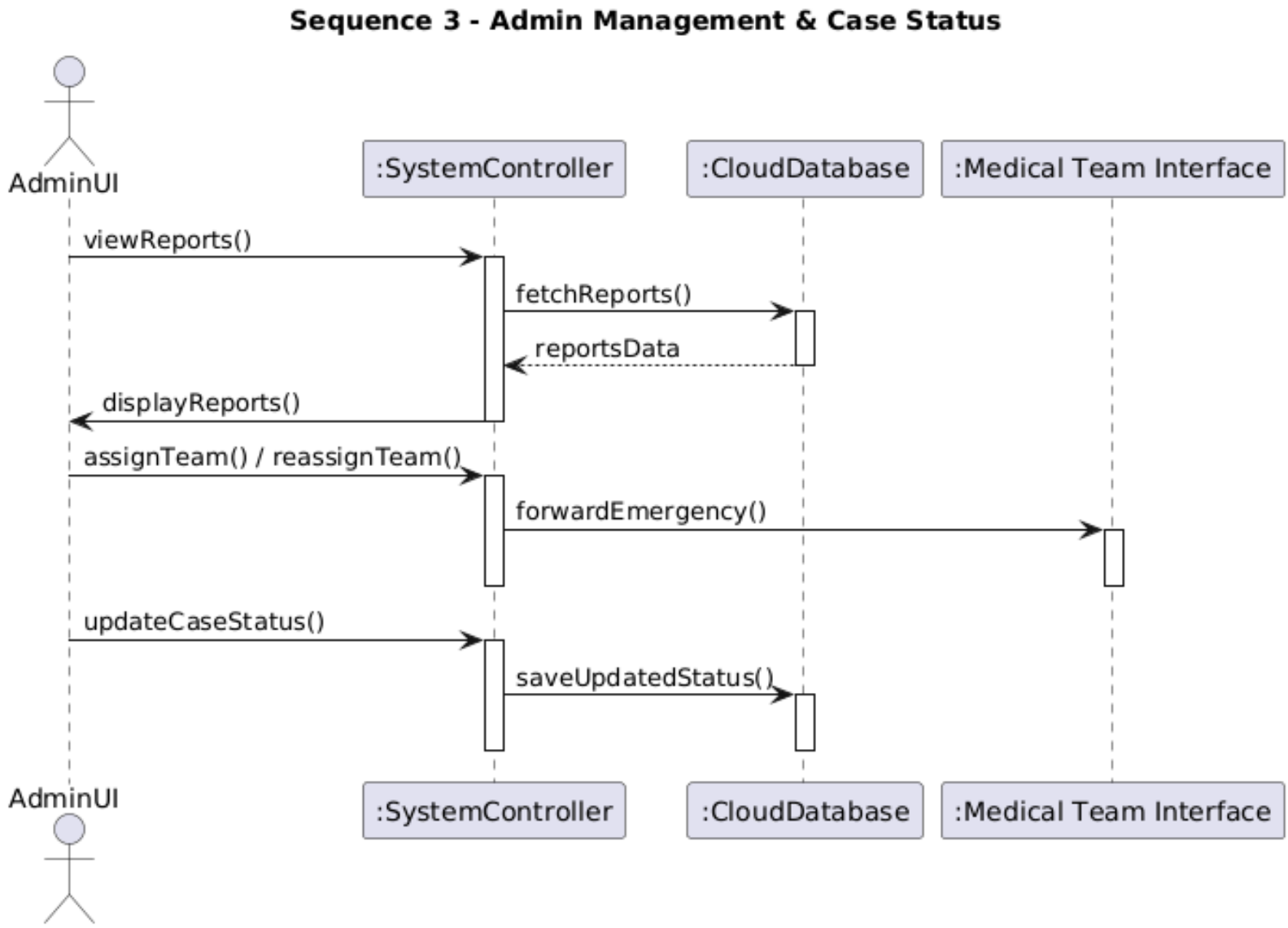
[Click here to view more clearly]

4.4.3 Emergency Sequence Diagram



Click here to view more clearly

4.4.4 Managment Sequence Diagram



[Click here to view more clearly]

Contributions

Member	Contribution
Ebtihal	1-Addressing functional & non-functional requirements 2-Project Leadership
Rafa	1-UML diagrams (Activity, Sequence, State)
Shahad	1-DFD diagrams (Level 0,Level 1) 2-Class diagram
Norah	1-Requirements Engineering and system overview 2-use case diagram
Maryam	1-Addressing ethics and responsibilities
Maysan	1-Process model
Tala	1-Addressing external interface requirements
Shatha	1-Architecture design

Project Meetings Notes

Hajj Care

MEETING NOTE

First Meeting

Details

Date: 22 / 6 / 2025

Time: 8:30AM - 10:30AM (2 hours)

Location: Group chat

Topic: Project Kickoff Meeting

Attendees

Meeting Leader: Shahad

Notetaker: Shahad

Attendees: All 8 members

Absence: -

Agenda

1. Team Introductions
    - Brief introduction of each team member.
    - Sharing backgrounds and relevant skills.
2. Project Overview
    - Understanding the project's objective and expectations.
3. Initial Idea Discussion
    - Brainstorming possible project ideas.
    - Shortlisting ideas for further exploration.
4. Deadlines and Timeline
    - Planning next team meeting.

Action Points:

1. Team members introduced themselves and shared backgrounds.
2. Got a general understanding of the project scope and requirements.
3. Discussed several possible ideas for the project.
4. Agreed on the final project idea to proceed with.
5. Set the date of the next meeting (after 4 days).

Hajj Care

MEETING NOTE

Second Meeting

Details

Date: 26 / 6 / 2025

Time: 11AM - 12PM (1 hour)

Location: Group chat

Topic: Task Distribution & Planning

Attendees

Meeting Leader: Rafa

Notetaker: Rafa

Attendees: All 8 members

Absence: -

Agenda

1. Detailed Project Planning
    - Breakdown of the project into tasks/sections.
    - Clarifying project goals and deliverables.
2. Task Assignment
    - Distributing responsibilities among team members.
    - Confirming who will handle which part.
3. Deadlines and Timeline
    - Setting task deadlines.
    - Planning next team meeting.

Action Points:

1. Reviewed the project topic in more detail.
2. Broke down the project into clear tasks/sections.
3. Assigned responsibilities to each team member.
4. Set internal deadlines and created a basic timeline.
5. Set the date of the next meeting (after 9 days).

Hajj Care

MEETING NOTE

Third Meeting

Details

Date: 5 / 7 / 2025

Time: 10AM - 11AM (1 hour)

Location: Group chat

Topic: SRS Progress Check & Preparation for Full Review

Attendees

Meeting Leader: Ebtihal

Notetaker: Ebtihal

Attendees: All 8 members

Absence: -

Agenda

1. Progress Updates
   - Each member shares what they have started working on.
   - Overview of completed and in-progress tasks.
2. SRS Draft Review
   - Discuss current status of the Software Requirements Specification (SRS) Part.
   - Identify which sections have been completed and which are still pending.
3. Deadlines and Timeline
   -Finalize what needs to be done before next meeting.
   - Planning next team meeting.

Action Points:

1. Team members shared updates on their individual progress.
2. Reviewed the current draft of the SRS document.
3. Identified completed sections and areas needing more work.
4. Clarified several points related to the SRS structure and content.
5. Decided to meet again tomorrow for a full and final SRS review.

Hajj Care

MEETING NOTE

Fourth Meeting

Details

Date: 6 / 7 / 2025

Time: 10AM - 12PM (2 hours)

Location: Zoom Meeting

Topic: Full SRS Review & Team Alignment

Attendees

Meeting Leader: Maryam

Notetaker: Maryam

Attendees: All 8 members

Absence: -

Agenda

1. Full SRS Review
   - Go through each section of the Software Requirements Specification (SRS).
   - Explain the content to ensure team-wide understanding.
2. Team Alignment
   - Confirm that all members understand the project direction and tasks.
   - Ensure everyone is on the same page before moving forward.
3. Break Planning
   - Plan for a one-week break due to exams.

Action Points:

1. Completed a full team review of the SRS document.
2. Each member explained or asked about different parts to ensure full understanding.
3. Confirmed that the entire team is aligned on the project scope and requirements.
4. Decided to pause work for one week to focus on exams.
5. Next meeting to be scheduled after the exam period ends.

Hajj Care

MEETING NOTE

Fifth Meeting

Details

Date: 19 / 7 / 2025

Time: 10AM - 12PM (2 hours)

Location: Group chat

Topic: Kickoff for SDD & Diagrams Planning

Attendees

Meeting Leader: Norah

Notetaker: Norah

Attendees: All 8 members

Absence: -

Agenda

1. Start of SDD Phase
   - Brief introduction to the Software Design Document (SDD).
2. Diagram Planning
   - Identify the types of diagrams required (DFD, Class, Sequence, etc.).
3. Tools and Collaboration
   - Agree on tools to use for creating diagrams.
   - Ensure shared access and consistent formatting.
4. Deadlines and Timeline
   - Set target to complete diagrams and related sections within 6 days.

Action Points:

1. Officially started work on the SDD phase.
2. Discussed and listed the key diagrams required.
3. Assigned initial diagram tasks to team members.
4. Agree to use *Lucidchart* for creating diagrams.
5. Agreed to complete diagrams within the next 6 days.
6. Set the date of the next meeting (after 6 days).

Hajj Care

MEETING NOTE

Sixth Meeting

Details

Date: 25 / 7 / 2025

Time: 5PM - 6:30PM (1.5 hours)

Location: Zoom Meeting

Topic: Diagrams Review & Preparation for Final SDD Check

Attendees

Meeting Leader: Maysan

Notetaker: Maysan

Attendees: All 8 members

Absence: -

Agenda

1. Diagrams Review
    - Go through all submitted diagrams related to the SDD.
    - Discuss accuracy, clarity, and completeness.
2. Feedback and Corrections
    - Identify areas that need changes or improvements.
3. Deadlines and Timeline
    - Set date and expectations for full SDD document review.

Action Points:

1. Reviewed all current diagrams in the SDD.
2. Gave feedback and decided on several changes and improvements.
3. Agreed to finalize all diagrams and remaining sections within 2 days.
4. Set the date of the next meeting (after 2 days).

Hajj Care

MEETING NOTE

Seventh Meeting

Details

Date: 27 / 7 / 2025

Time: 5PM - 6PM (1 hours)

Location: Zoom Meeting

Topic: Final SDD Review

Attendees

Meeting Leader: Tala

Notetaker: Tala

Attendees: All 8 members

Absence: -

Agenda

1. Final Diagram Review
    - Go through all diagrams one last time.
    - Ensure clarity, accuracy, and consistency across all visuals.
2. SDD Final Check
    - Review full Software Design Document (SDD) content.
3. Team Understanding
    - Ensure all team members fully understand the design and structure.
4. Timelines
    - Plan for the final team meeting before submission.

Action Points:

1. Completed a full review of all SDD diagrams.
2. Ensured everyone clearly understood the structure and purpose of each diagram.
3. Finalized and confirmed the content of the SDD.
4. Set the date for the final team meeting for the next day to review the entire project.

Hajj Care

MEETING NOTE

Eighth Meeting

Details

Date: 28 / 7 / 2025

Time: 5PM - 7PM (2 hours)

Location: Zoom Meeting

Topic: Final Project Review & Presentation Preparation

Attendees

Meeting Leader: Shatha

Notetaker: Shatha

Attendees: All 8 members

Absence: -

Agenda

1. Full Project Review
    - Go through all parts of the project: documents, diagrams, and any deliverables.
    - Verify completeness and consistency.
2. Presentation Preparation
    - Draft and finalize the presentation slides.
    - Assign speaking roles and rehearse key points.
    - Ensure visual and content clarity.
2. Final Formatting & Polish
    - Ensure all files are properly formatted and well-organized.
3. Submission Readiness
    - Confirm that everything is ready for submission.
4. Closing Discussion
    - Reflect briefly on teamwork and the project journey.
    - Celebrate completion!

Action Points:

1. Reviewed the entire project.
2. Ensured everything was complete, accurate, and properly formatted.
3. Prepared and rehearsed the presentation.
4. Confirmed submission readiness.
5. Decided to proceed with submitting the project.