SafetyNet Alerts Project Overview

By: Faiza Ahmed

Emergency Information Dispatch System

SAFETYNET ALERTS

Spring Boot, Maven, JUnit, JaCoCo

Project Purpose

Why SafetyNet Alerts?

- Provide emergency responders with **critical information quickly**

- Access residents' details like **phone, age, medications, and allergies**

- Notify residents of emergencies like **fires, storms, and floods**

Technology Stack

Backend: Spring Boot (MVC architecture)

Build Tool: Maven

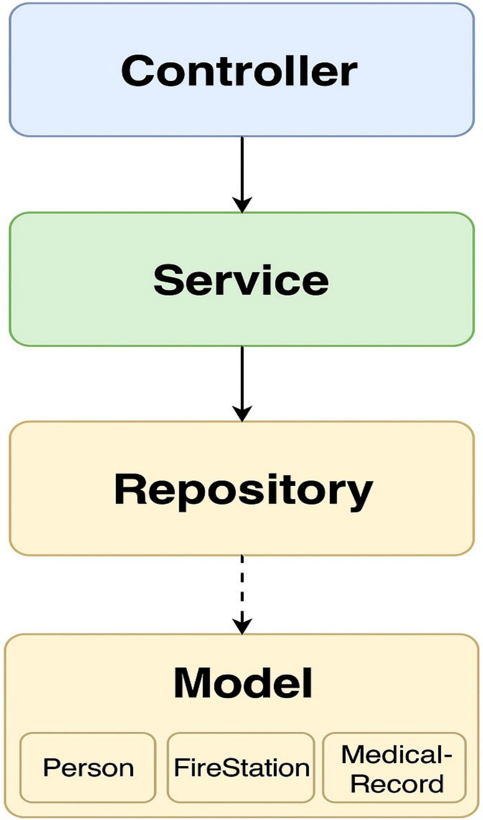
Testing: JUnit 5

Code Coverage: JaCoCo

Logging: Log4j

API Format: REST + JSON

High-Level Architecture



Domain Model

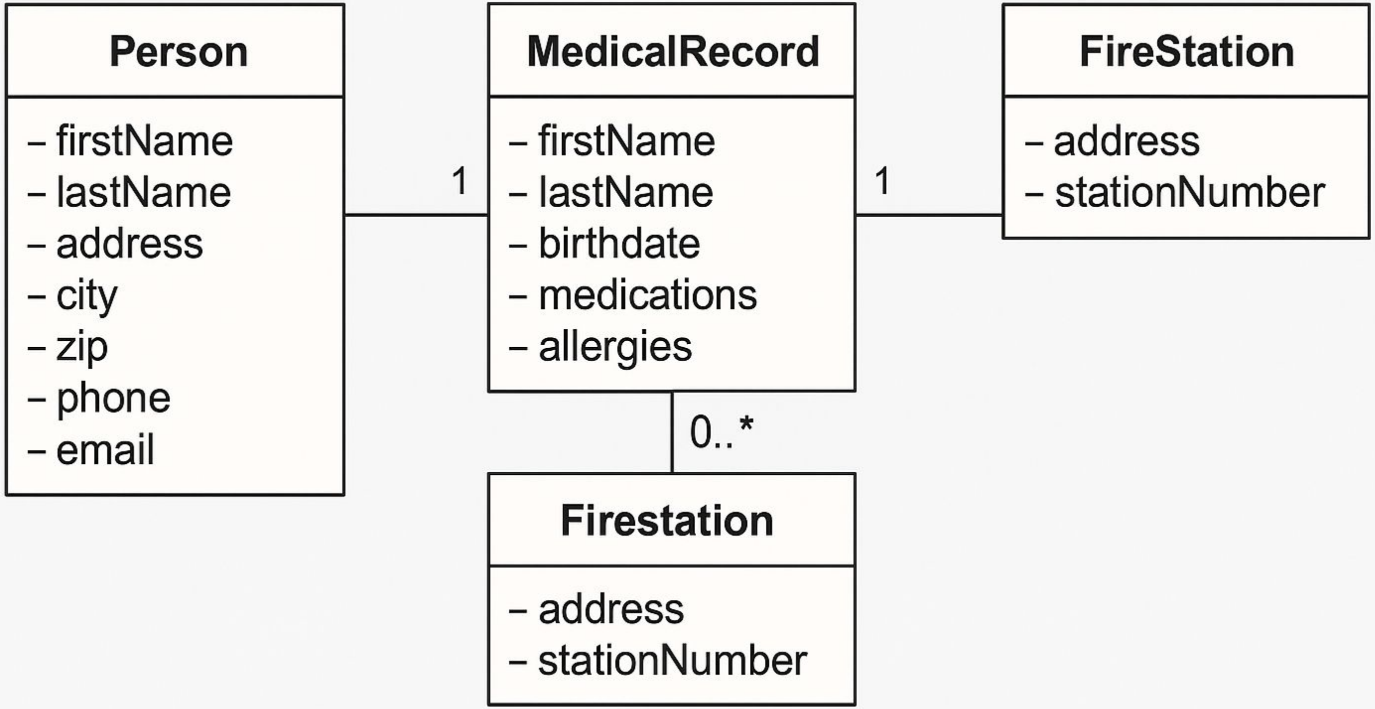
Entities:

- **Person**: firstName, lastName, address, city, zip, phone, email
- **MedicalRecord**: firstName, lastName, birthdate, medications, allergies
- **FireStation**: address, station number

Relationships:

- Person ↔ MedicalRecord: 1:1 (by name)
- Address ↔ FireStation: 1:1
- Address ↔ Persons: 1:N

Domain Model



Person

- firstName
- lastName
- address
- city
- zip
- phone
- email

MedicalRecord

- firstName
- lastName
- birthdate
- medications
- allergies

FireStation

- address
- stationNumber

Firestation

- address
- stationNumber

REST Endpoints and URLs

Application Related:

http://localhost:8080/actuator/metrics

It is used to show metrics information for the current application.

http://localhost:8080/actuator/health

It is used to show application health information.

http://localhost:8080/actuator/info

It is used to display arbitrary application info.

Person Endpoints

http://localhost:8080/person → Use Post/Put/Delete HTTP request on this url to add, modify or delete a person and it's related information.

http://localhost:8080/communityEmail?city=city → Use Get HTTP request on this url to request all the emails from the persons living in the city used as parameter

http://localhost:8080/personInfo?firstName=firstName&lastName=lastName → Use Get HTTP request on this url to request name, address, age, email and medical record of the person requested in parameter

http://localhost:8080/childAlert?address=address → Use Get HTTP request on this url to request the list of the children living at an address (containing their firstname, lastname and age) plus the list of other people living at this address.

http://localhost:8080/allPersons → List all persons information.

FireStation Endpoints

http://localhost:8080/firestation/address → Use Post/Put/Delete HTTP request on this url to add, modify or delete a Firestation and it's related information.

http://localhost:8080/fire?address=address → Use Get HTTP request on this url to request name, phone number, age and medical record of people living at the address plus the firestation number.

http://localhost:8080/firestation?stationNumber=station_number → URL should return a list of people serviced by the corresponding fire station.

http://localhost:8080/flood/stations?stations=list of station_numbers → Return a list of all the households in each fire station's jurisdiction. List needs to group people by household address, include name, phone number, and age of each person, and any medications (with dosages) and allergies beside each person's name.

http://localhost:8080/allStations → List all fire stations information.

MedicalRecord Endpoints

http://localhost:8080/medicalRecord

Use Post/Put/Delete HTTP request on this url to add, modify or delete a medicalRecord and it's related information.

Logging and Reporting

📄 Structured Logging

Enables traceability across services and better debugging with consistent, timestamped log entries.

📊 JaCoCo Code Coverage

Measures unit test coverage, highlighting untested code blocks and supporting quality assurance.

✅ Surefire Report

Summarizes unit test results (pass/fail/skip), helping identify failing test cases quickly.