

# **Infirmary Management System**

## **Authors:**

- Anutej Poddaturi [NUID: 002922161]
- Kshama Aditi Lethakula [NUID: 002776998]
- Nikhil Reddy Polepally [NUID: 002776998]

## **Reviewer:**

- Professor Kal Bugrara.

## **Problem Statement:**

The main issue with the existing system is that because all the systems are not connected, it will be challenging to process and handle requests when the system becomes overloaded. It is crucial to do tasks promptly and correctly in the modern day and in society. The request queue becomes more prominent due to this ineffective method of request resolution, making it hard to clear. The suggested approach ensures that each job has adequate access to the necessary actions to complete the work at hand correctly and precisely. This solution ensures that the User is constantly updated with the status of his record in the system.

## **Proposed Solution:**

Infirmary Management System is an organized application designed and programmed to deal with day-to-day operations and management of services that patients need access to in cases of emergency or injury. The program can manage inpatients, outpatients' records, data about lab tests, the status of the patient, and billings in the pharmacy. The project is about a hospital where patients can be registered through the reception login process and go through a doctor's consultation, take lab tests as prescribed by the doctor, request blood from blood banks, and take medicine from the dispensary as prescribed by the doctor. In cases of emergency, an ambulance can also register a patient from the

accident location. Finally, the patient can view their bill and choose to pay themselves or through insurance, which requires verification from the insurance enterprise and a related verification from a police officer from the police enterprise. Finally, the patient can be discharged after receiving approval from outpatient administration.

## **Entities:**

### **• Enterprise**

- 1) Hospital
- 2) Blood Bank
- 3) Insurance
- 4) Police

### **• Organization**

- 1) Dispensary Organization
- 2) Lab Organization
- 3) Ambulance Organization
- 4) Hospital Organization
- 5) Blood Bank Organization
- 6) Insurer Organization
- 7) Police Department Organization

### **• Roles**

- 1) Doctor
- 2) Dispensary Administrator
- 3) Lab Manager
- 4) Ambulance Admin
- 5) OutPatient Department Admin
- 6) Blood Bank System Admin Role
- 7) Insurance Manager
- 8) Police Officer

## Steps to regenerate this project:

- Git clone the project by running the below command in the terminal at the location you want to clone locally: `` git clone <https://github.com/apoddaturi/Final-Project-AED.git>``
- Open the project in Eclipse or Netbeans.
- Add all the dependencies to libraries, right-click on the project from the project file manager, then click clean and build.
- Run it.

## Functional Description:

1. As a user, one should be able to login with their credentials to perform various operations as per the role.
2. Once the user login if he signs in and his role is:
  - a. **OutPatient Department Admin:** He/She would be able to create, update and delete a patient. Call an ambulance for an emergency, discharge a patient, bill the patient and look at the past patient's record.
  - b. **Doctor Admin:** He would be able to add, update and delete doctors, and he would be able to assign a doctor to a patient.
  - c. **Doctor:** The person could order a diagnostic test, medicines from a dispensary, and blood packs for emergencies. He would also assess results from the above entities and discharge the patient.
  - d. **Dispensary Administrator:** Check the medication in a table and update or delete the medications.
  - e. **Lab Manager:** Would be able to add delete and update tests, their costs and their description.
  - f. **Ambulance Admin:** Admin would be able to add new ambulances, see current patients, and set the driver free once the task is complete.
  - g. **Blood Bank System Admin Role:** The User would process the request raised by the doctor so that the patient can be discharged
  - h. **Insurance Manager:** View the patients' current insurance claim and check older claims by the patient; if he needs a verification, he can send for verification and await their response

- i. **Police Officer:** The User, if he is a police officer, would be able to complete verification so that the insurance manager can approve the claim for the raised patient.
3. The User can log out at any point in time and return to the login page.

### **User Interface:**

1. The Whole application is made sure to be built keeping in mind various types of users that would be using the application.
2. The Application has a very simple UI.

### **Goals and Milestones:**

1. At the end of the project, we need to have a working application that encompasses all the roles mentioned earlier and assists each person with the end-to-end operations of their role.
2. The users should be able to receive emails from the Infirmary Management System that would give the User track of the expenses and keep them updated with the creation and discharge status of the patient.

## UML::



## Hierarchy:

