NewHat Triage Application

Design Document

Version 1.0

11/30/2020

Table of Contents

1. Introduction

1.1 Purpose of the Application

A hospital's Emergency Department should be prepared in the event of a catastrophic event resulting in mass casualties. This means there should be a Triage process in place to effectively manage ED workers to ensure patients get the care needed in a timely manner. The ability to manage such an event requires processes that include:

- Identify doctors and nurses who are on-call
- Patient evaluation and assessment
- Patient rating to prioritize patients who need immediate care, patients with little survival chances and patients who may need to be transported to external trauma center, and
- Patient progress tracking

The purpose of this application is to help hospital staff so that triage patients can get timely attention before they enter the hospital.

-Triage is the process of classifying patients according to the severity of their injuries.

-Triage system determines how quickly they need care to avoid patients' deaths.

-Many hospitals use triage when medical-care system is overloaded and there are people who need care.

-Hospitals use a 5-level system, considering current resources and the status of the patient (ESI).

-Various codes exist in FHIR to display these levels

2. General Overview and Design Guidelines/Approach

This section describes the principles and strategies to be used as guidelines when designing and implementing the system.

2.1 General Overview

Our application will facilitate the following tasks:

- ✔ Gathering patient information
- ✔ Assigning a triage severity level to a patient
- ✔ Monitoring patient progress
- ✔ Tracking staff and managing capacity
- ✔ Pairing injuries with appropriate staff
- ✔ Updating staff assignment

2.2 Assumptions/Constraints/Risks

2.2.1 Assumptions

Following are the assumptions made:

- Hospitals staff are registered on the FHIR database.
- Patients exist on FHIR database.

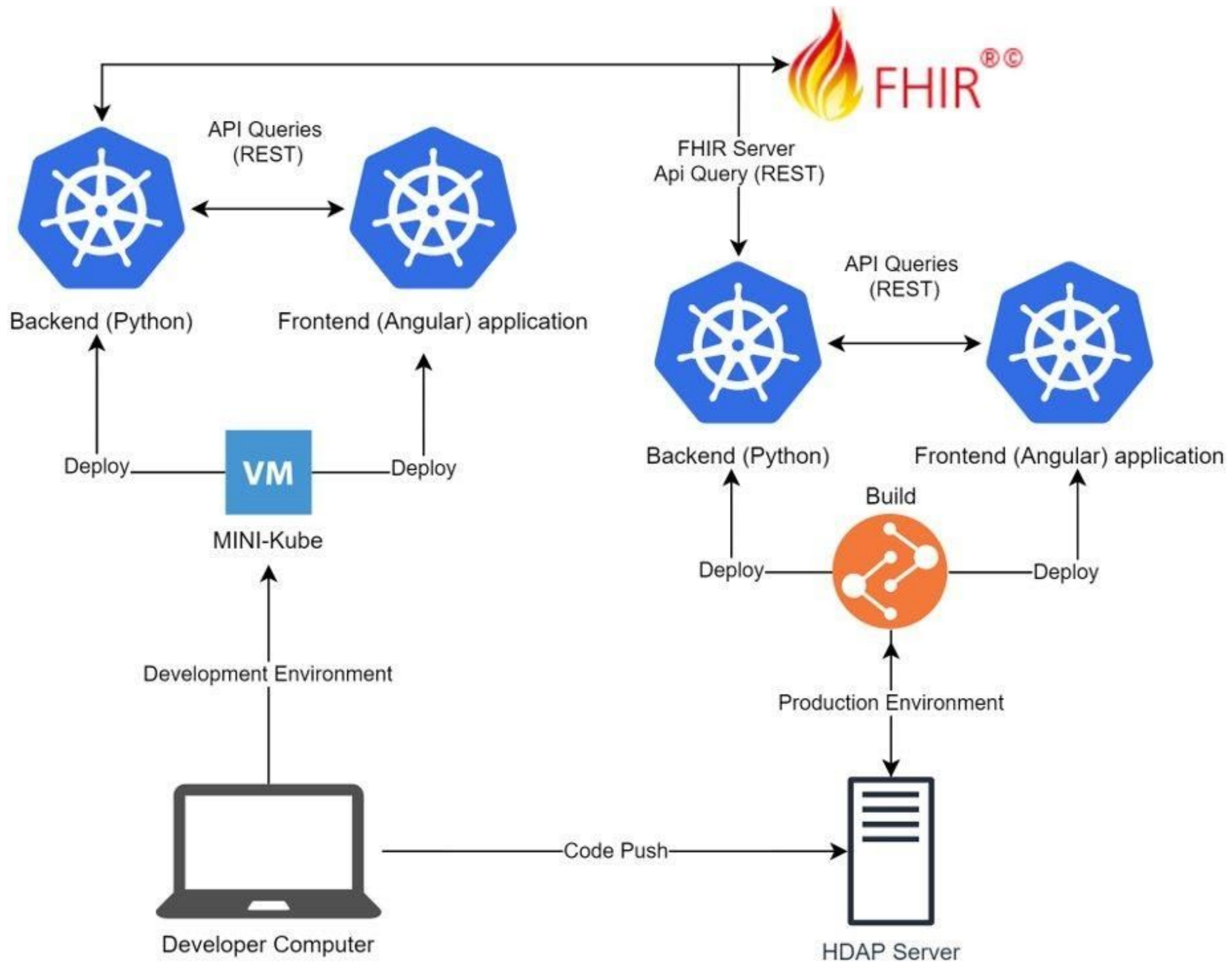
2.2.2 Constraints

- FHIR database is up and running.
- HDAP server is up and running.

3. Software Architecture and Architecture Design

This section outlines the software architecture design of the system.

3.1 Software Architecture



Angular Front-End Container:

- Angular 10 (Javascript / Typescript) will be the language we use to write our front-end.

Python / Flask Back-End Container

- Python will act as the backend, communicating with all other components and translating database / FHIR calls to the front-end.

- We will use Flask to create the API calls needed for the front-end

- For FHIR processing, we will use the SMART on FHIR framework for Python: https://github.com/smart-on-fhir/client-py

PostgreSQL Database Container

- The database (PostgreSQL) will store physician availability information and anything which can't be stored via the HDAP FHIR server.

4. System Design

4.1 Business Requirements

Below are some high level business requirements:

- Application should be able to search for patients on FHIR and add the patient reference to the new system when found.
- Hospital staff can set ESI code and physicians, once the patient is added to the system.
- Staff can add body injury and system injury for a patient.
- Staff should be able to update patients status until he/she is discharged from the hospital.
- Staff should be able to all patients on one page.
- Application should color code each patient according to their urgency level.
- Application should keep patient history.

4.2 Deployment Design

4.2.1Containers

Three containers are created for deployment into a single cluster on Kubernetes. This includes once container for the frontend, one for the backend, and one for the database.

4.2.1.1 Frontend

The frontend service is based on the Node version 12 docker container. The port 4200 is exposed to allow for visibility of the frontend. All files from the app/frontend directory are copied onto the container to /app. Upon start of the container, the command "ng serve --host 0.0.0.0 --disable-host-check --prod=true --live-reload=false" is run.

4.2.1.2 Backend

The backend container runs our flask application for the backend. The application provides an API for the frontend to access information for display and an API for pushing data to both the FHIR server and Postgres database. The container is based on the Ubuntu version 20.10 with python3-pip, libpq-dev, and flask installed. Port 5000 is exposed for the running API. All files from the app/backend directory are copied onto the container to /home/edts-flask. Upon start of the container, the command "python app.py" is run.

4.2.1.3 Database

This container runs the Postgres database for extra data not able to be stored in on the FHIR server. The port 5432 is exposed for database access. The TriageDB.sql file is copied to /docker-entrypoint-initdb.d.

4.2.2Deployment

Deployment was done on the GATech HDAP Server. Builds were automatically performed by Drone which then pushed to Rancher for deployment with Kubernetes.

4.2.2.1 Drone Build

The build consists of 5 steps. It is triggered on a push to the deploy branch within the git repository.

4.2.2.1.1 deploy_application_frontend

This step builds and pushes the frontend image to the HDAP Docker registry. Details are as follows:

repo: gt-build.hdap.gatech.edu/emergency-department-triage-system

dockerfile: app/front-end/Dockerfile

4.2.2.1.2 deploy_application_backend

This step builds and pushes the backend image to the HDAP Docker registry. Details are as follows:

repo: gt-build.hdap.gatech.edu/emergency-department-triage-system-backend

dockerfile: app/back-end/Dockerfile

4.2.2.1.3 edts-db

This step builds and pushes the database image to the HDAP Docker registry. Details are as follows:

repo: gt-build.hdap.gatech.edu/emergency-department-triage-system-backend

dockerfile: app/back-end/Dockerfile

4.2.2.1.4 get_chart_builder

This step pulls down the template for deployment to HDAP K8S.

4.2.2.1.5 copy_namespace

This step sets the namespace for deployment to
ns-emergency-department-triage-system-2.

4.2.2.1.6 deploy_to_k8s

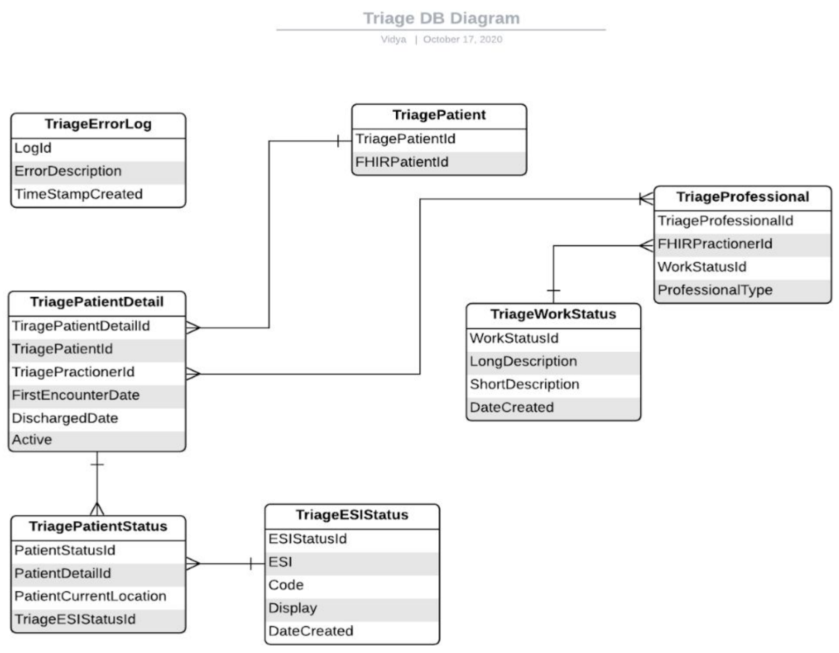
This step pushes the build to HDAP K8S.

4.2.2.2 Rancher Configuration

No modifications need to be made on Rancher once deployed by Docker. The cluster is published in the namespace ns-emergency-department-triage-system-2. It can be used to access the individual containers and access their command lines.

4.3 Database Design

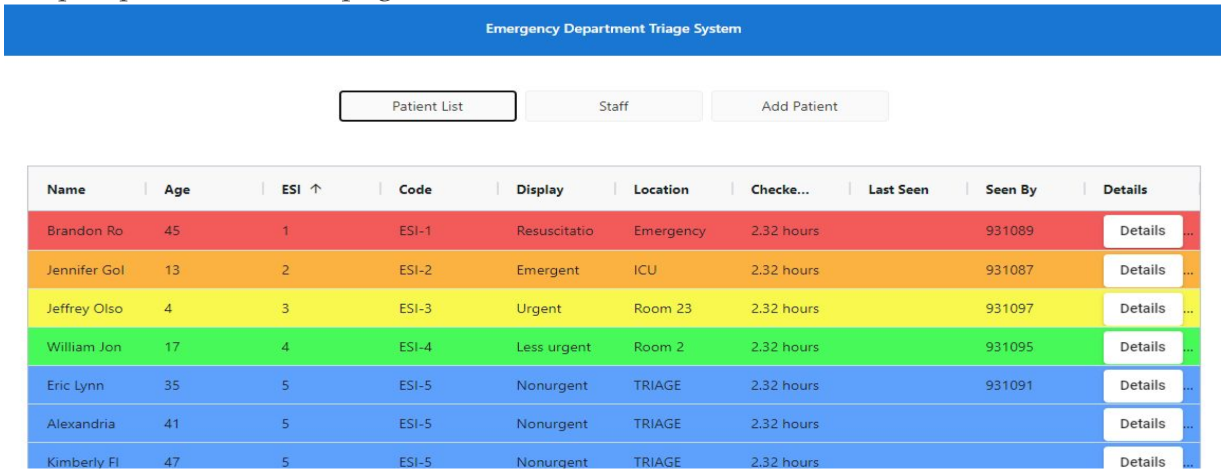
Below is the database design diagram:



4.4 User Interface Design

4.4.1 Patient Priority List

Patient Priority List user interface is fetching the patients information from the backend and displaying on the user interface in a table format. Also, a ‘Details’ button is added to the open patients details page.



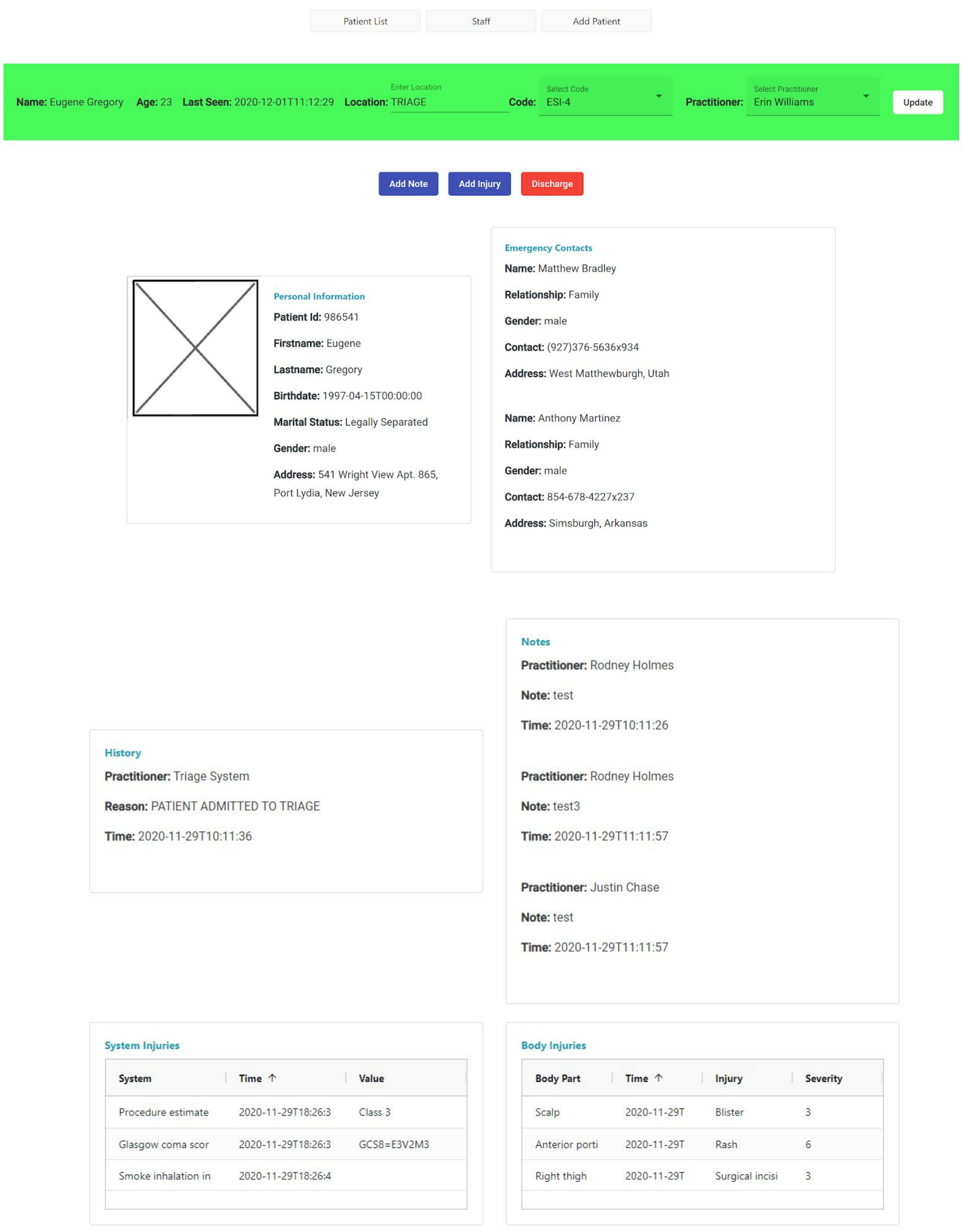
Emergency Department Triage System

Name	Age	ESI ↑	Code	Display	Location	Checke...	Last Seen	Seen By	Details
Brandon Ro	45	1	ESI-1	Resuscitatio	Emergency	2.32 hours		931089	Details
Jennifer Gol	13	2	ESI-2	Emergent	ICU	2.32 hours		931087	Details
Jeffrey Olso	4	3	ESI-3	Urgent	Room 23	2.32 hours		931097	Details
William Jon	17	4	ESI-4	Less urgent	Room 2	2.32 hours		931095	Details
Eric Lynn	35	5	ESI-5	Nonurgent	TRIAGE	2.32 hours		931091	Details
Alexandria	41	5	ESI-5	Nonurgent	TRIAGE	2.32 hours			Details
Kimberly Fl	47	5	ESI-5	Nonurgent	TRIAGE	2.32 hours			Details

Figure 1: Patient Priority List

4.4.2 Patient Details UI

Patient Details user interface is fetching patients data from the backend using patient id. It is showing patient details with emergency contacts history, notes, system injuries and body injuries.



Name: Eugene Gregory **Age:** 23 **Last Seen:** 2020-12-01T11:12:29 **Location:** TRIAGE **Code:** ESI-4 **Practitioner:** Erin Williams

Personal Information

Patient Id: 986541

Firstname: Eugene

Lastname: Gregory

Birthdate: 1997-04-15T00:00:00

Marital Status: Legally Separated

Gender: male

Address: 541 Wright View Apt. 865, Port Lydia, New Jersey

Emergency Contacts

Name: Matthew Bradley

Relationship: Family

Gender: male

Contact: (927)376-5636x934

Address: West Matthewburgh, Utah

Name: Anthony Martinez

Relationship: Family

Gender: male

Contact: 854-678-4227x237

Address: Simsburgh, Arkansas

History

Practitioner: Triage System

Reason: PATIENT ADMITTED TO TRIAGE

Time: 2020-11-29T10:11:36

Notes

Practitioner: Rodney Holmes

Note: test

Time: 2020-11-29T10:11:26

Practitioner: Rodney Holmes

Note: test3

Time: 2020-11-29T11:11:57

Practitioner: Justin Chase

Note: test

Time: 2020-11-29T11:11:57

System Injuries

System	Time ↑	Value
Procedure estimate	2020-11-29T18:26:3	Class 3
Glasgow coma scor	2020-11-29T18:26:3	GCS8=E3V2M3
Smoke inhalation in	2020-11-29T18:26:4	

Body Injuries

Body Part	Time ↑	Injury	Severity
Scalp	2020-11-29T	Blister	3
Anterior porti	2020-11-29T	Rash	6
Right thigh	2020-11-29T	Surgical incisi	3

Figure 2: Patient Details UI

4.4.3 Add Patient UI

The add Patient user interface is developed and connected with the backend. It is saving a patient into the FHIR and triage system database.

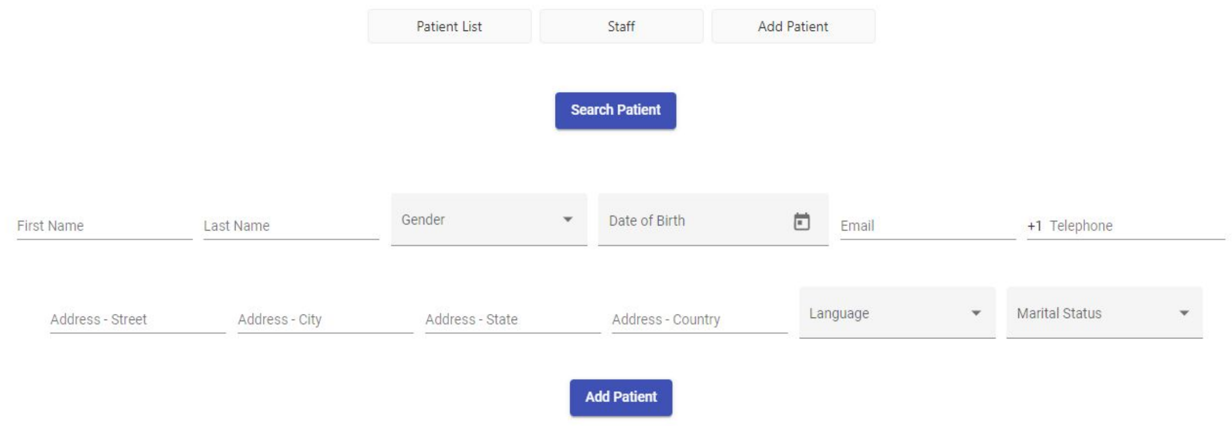
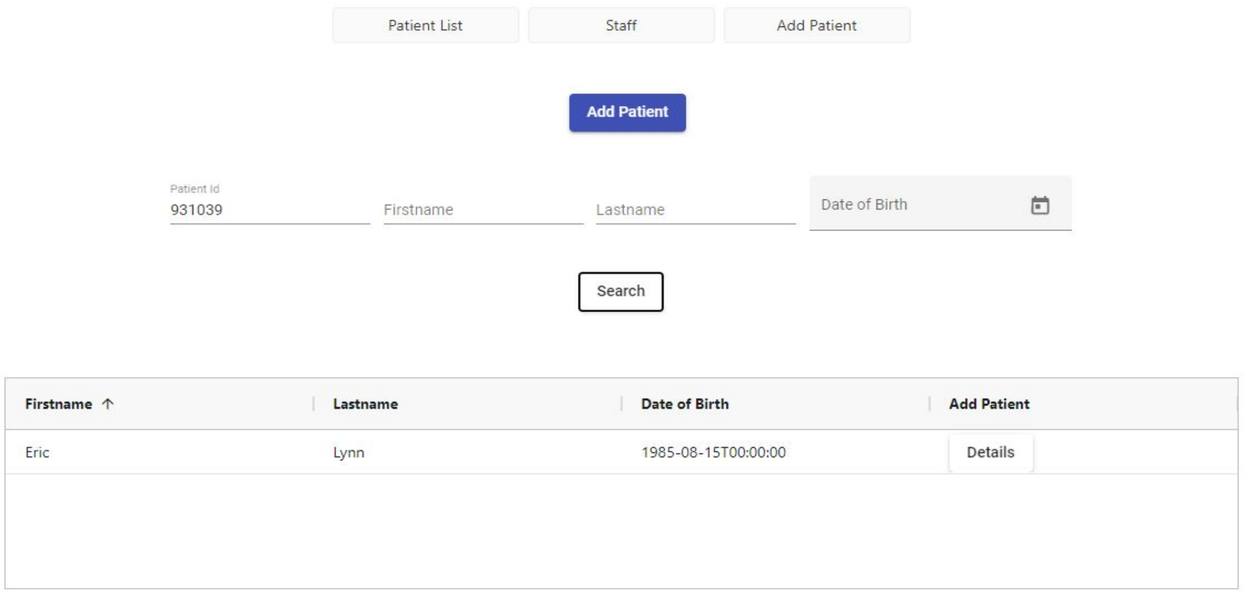


Figure 3: Add Patient UI

4.4.4 Search Patient UI

Search Patient user interface is developed and connected with the backend. It is searching a patient and showing the result in a table format. Users can click on the details button to open the patient details page.



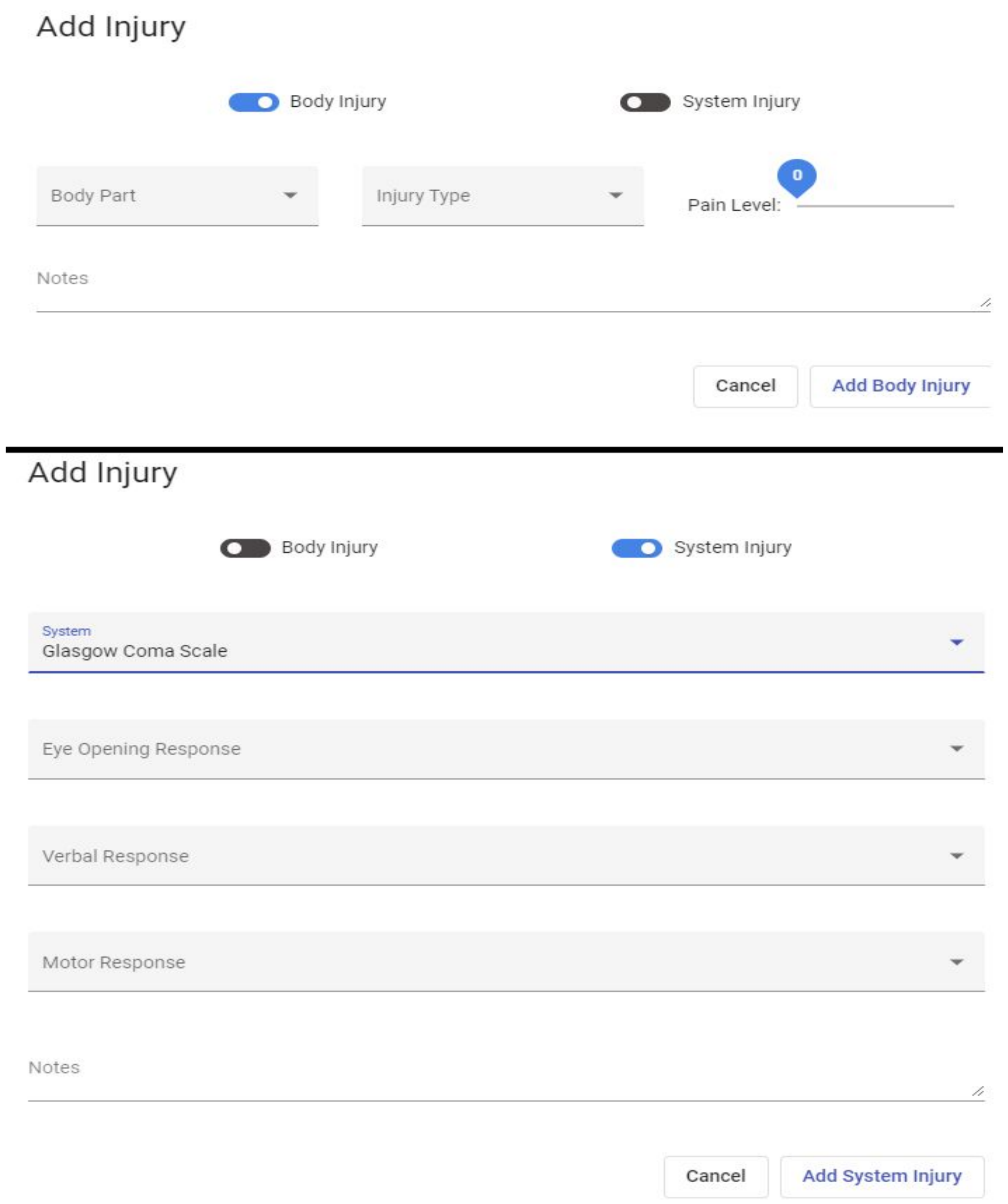
Patient Id: 931039

Firstname ↑	Lastname	Date of Birth	Add Patient
Eric	Lynn	1985-08-15T00:00:00	Details

Figure 4: Search Patient UI

4.4.5 Add Injuries

Add body and system injury user interface is developed and connected with the backend. It’ll save an injury into FHIR observation.



System
Glasgow Coma Scale

Figure 5: Add Injury

4.4.6 Add Note

Add note user interface is developed and connected with the backend. It is saving a note with practitioner id into the triage system database.

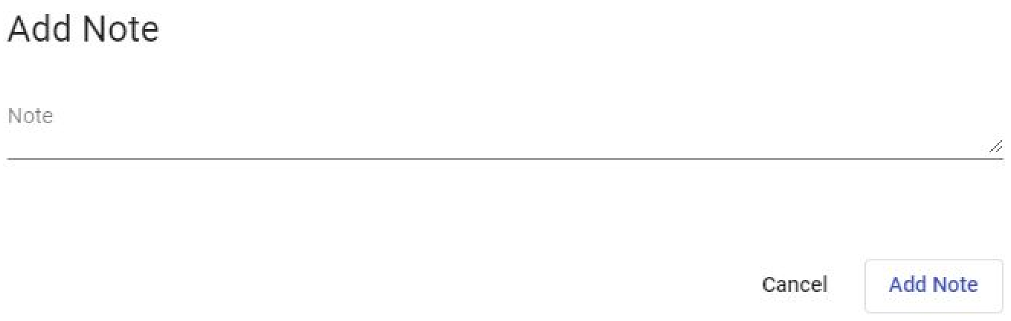
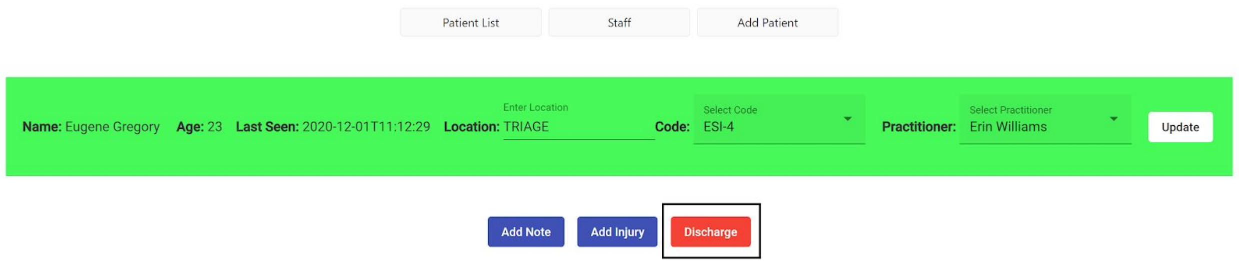


Figure 6: Add Note

4.4.7 Discharge Patient

Discharge patient button is developed and connected with the backend. Users can use this button to discharge a patient.



Name: Eugene Gregory **Age:** 23 **Last Seen:** 2020-12-01T11:12:29 **Location:** Enter Location TRIAGE **Code:** Select Code ESI-4 **Practitioner:** Select Practitioner Erin Williams

Figure 7: Discharge Patient