National COVID Management System
Software Requirements Specification

Version 1.0

Revision History

Date	Version	Description	Author
18/08/21	1.0	This is the first version of the requirements specification document	K.A.S.H. Kumarasinghe

Table of Contents

Introduction **6**

Purpose 6

Scope 6

Definitions, Acronyms, and Abbreviations 7

References 7

Overview 8

Overall Description **8**

product perspective 8

product functions 8

user characteristics 9

Design and Implementation Constraints 10

assumptions and dependencies 11

Specific Requirements **11**

Functionality 11

Functional requirements for patients 11

View serial number and the hospital details 11

view serial number and queue number 11

Functional requirements for hospital staff 11

View patient information and serial number 11

Functional requirements for doctors 12

Update the severity level of the patients 12

Functional requirements for chief doctors (directors) 12

Discharge the patients in the hospital 12

Add doctors to the system 12

Add hospital staff to the system 12

Functional requirements for authorities of MoH 12

View patient statistics 12

View hospital and bed statistic 12
Get notified of the need for a new hospital 12
Add new hospitals 13
Edit chief doctors; details 13
Functional requirements for citizens 13
Register to the system as a patient with geolocation coordinates of the home 13
View patient statistics 13
Usability 13
Training time 13
Language 14
User Interface 14
User Experience 14
Reliability 14
Availability 14
Accuracy 14
Performance and Security 14
Response time 14
Resource utilization 15
Security 15
Supportability 15
Naming convention 15
Class libraries 15
Design Constraints 15
Software Language 15
Class Libraries 15
Online User Documentation and Help System Requirements 16
Purchased Components 16
Interfaces 16

User Interfaces 16

Hardware Interfaces 23

Software Interfaces 23

Communications Interfaces 24

Database Requirements 24

Licensing, Legal, Copyright, and Other Notices 24

Applicable Standards 24

Supporting Information 24

Appendices 24

Software Requirements Specification

1. Introduction

1.1 Purpose

A Software Requirements Specification (SRS) is a complete description of the behaviour of the system to be developed. The purpose of this document is to present a detailed description of the “National COVID Management System (NCMS) ”. It will describe the system's function and features, as well as the system's interfaces, what the system will do, the constraints that it must work under, and how the system will respond to external stimuli.

It includes a set of use cases that describe all the interactions the users will have with the software. Use cases are also known as functional requirements. In addition to the use cases, the SRS also contains non-functional (or supplementary) requirements. Nonfunctional requirements are requirements that impose constraints on the design or implementation (such as performance engineering requirements, quality standards, or design constraints)

1.2 Scope

The document covers the hardware and software aspects of the National COVID Management System (NCMS) which is necessary for development. This system is built to support and improve the process of bed booking. This system will be more effective when compared with current manual systems.

The system will include six main categories

1. patients
2. hospital staffs
3. doctors
4. Chief doctors - directors of the hospitals
5. authorities of the country (Ministry of Health)
6. Citizens

This system will be a centralized system. Thus it can be used to analyze statistical

data. The main goal of the team is to manage the COVID-19 situation efficiently using modern technology.

1.3 Definitions, Acronyms, and Abbreviations

SRS	Software Requirements Specification
NCMS	National COVID Management System
MoH	Ministry of Health
COVID-19	CoronaVirus Disease 2019 caused by SARS-CoV-2
REST	REpresentational State Transfer
API	Application Programming Interface
HTTP	HyperText Transfer Protocol

1.4 References

[1] Docs.microsoft.com. 2021. User Interface Principles - Win32 apps. [online] Available at: <https://docs.microsoft.com/en-us/windows/win32/appuistart/-user-interface-principles#spacing-and-positioning> [Accessed 18 April 2021].

[2] Computerhope.com. 2021. What is Camelcase?. [online] Available at: <https://www.computerhope.com/jargon/c/camelcase.htm> [Accessed 18 April 2021].

[3] Adobe. 2021. Adobe XD | Fast & Powerful UI/UX Design & Collaboration Tool. [online] Available at: <https://www.adobe.com/products/xd.html> [Accessed 18 April 2021].

1.5 Overview

The SRS provides a detailed description of the requirements associated with the development of NCMS. Usability, functionality, performance, reliability, design constraints, interfaces, and identification of the association with the system's users will all be classified in this document.

2. Overall Description

2.1 product perspective

The system which is specified in this SRS is a COVID Management System. The main users in this specified system include patients, hospital staff, doctors, chief doctors (directors), citizens and authorities of the country like MoH. s. In this project, a web-based system will be developed where the patient can register in NCMS with the geolocation coordinates of the home if they show COVID19 symptoms and get allocated to a bed in the nearest hospital. This system will search for hospitals with available beds in the whole country, and allocate patients to beds of the nearest hospital to the patient with a unique serial number. This serial number with the hospital information is then sent back to the patient while it would be visible to the hospital staff. If a bed is not available, the system will put the patient into a queue with a serial number, and the patient will be notified with the serial number and the queue number.

2.2 product functions

- **Patients**
  - View serial number with the hospital information
  - View queue details
- **Hospital staff**
  - View patients serial numbers
- **Doctors**
  - Assign severity level of the patients
  - view patient statistics

- **Chief doctors (directors)**
    - view patient statistics
    - Discharge the patients
    - Add doctors to the system
    - Add hospital staff to the system
    - View statistics of the hospital
- **Authorities of MoH**
    - view patient statistics
    - view hospitals and bed statistics
    - view notification of need of the new hospital
    - Add hospitals to the system
    - Edit chief doctors of the hospitals
- **Citizen**
    - View patient statistics
    - Register in the system

2.3 user characteristics

- **Patients**

  Since the goal of the system is to manage the COVID-19 spreading situation patients are the most important user. If a citizen shows COVID19 symptoms, they might immediately register in NCMS with the geolocation coordinates of the home and the system will offer bed details of the nearest hospital as discussed in the above part. Since patients are the most critical user class in this system, significant attention should be given to fulfil the requirements expected by them and the system should also guarantee that their data is secured.

- **Hospital staff**

  Hospital staff can view serial numbers with patient information. Although the hospital staff user class is not as critical as the patient's user class, some attention should be given to make the system efficient.

- **Doctors**

Doctors are also an important user class in the system and have higher access privilege than the hospital staff. They should have access to view patient information and update the severity level of the patient. Even Though the doctor's user class is not as critical as the patient's user class, special attention should be given to make the system efficient.

- **Chief doctors (directors)**

  Chief doctors (directors) have higher access privileges than doctors. They have access privileges of the doctors and also, they can discharge the patients from the hospitals. Even though the Chief doctors (directors) user class is not as critical as the patient's user class, special attention should be given to make the system efficient.

- **Authorities of MoH**

  Authorities of MoH are the users who manage the system and they have the highest access privileges. They are the ones in charge of the entire system. This user class has a higher level of technical expertise than the previous user classes.

- **Citizens**

  Citizens are the user class who doesn't require to register in the system. They can view the statistical data of the patients. If they have symptoms of COVID-19 they can register in the system as a patient.

2.4 Design and Implementation Constraints

The system will need an Email API and the system needs to be hosted. For hosting space and the Email API will be used freely available at the start of the project and a considerable cost will be incurred when the project needs to be scaled up.

2.5 assumptions and dependencies

The latest versions of the third-party APIs, libraries and tools that will be used in this system. Also, JUnit will be used for automated testing and GIT for version control. It is assumed that these tools and technology that is being used will

be reliable with minor bug fixes.

3. Specific Requirements

3.1 Functionality

This system will have six types of users namely patients, hospital staff, doctors, chief doctors (directors), authorities of MoH and citizens. This section of the requirements specification document is organized by the user, explaining in detail each functional requirement of the system associated with that user.

3.1.1 Functional requirements for patients

3.1.1.1 View serial number and the hospital details

If the beds are available in the nearest hospital they will receive a serial number and the information of the hospital. The patients should be able to view their serial number and the hospital information, bed number that was given by the NCMS.

3.1.1.2 view serial number and queue number

If the beds are not available in the hospital the patient will receive a queue number and serial number. The patient should be able to view his serial number and queue number.

3.1.2 Functional requirements for hospital staff

3.1.2.1 View patient information and serial number

Hospital staff should be able to view information and the serial numbers of the

patient of their hospital.

3.1.3 Functional requirements for doctors

3.1.3.1 Update the severity level of the patients

Doctors should be able to update the severity level of the patient to the system upon the patient's arrival. The possible severity level is low, moderate, and critical.

3.1.4 Functional requirements for chief doctors (directors)

3.1.4.1 Discharge the patients in the hospital

Chief doctors, the directors of the hospital should be able to discharge the patients from their hospital.

3.1.4.2 Add doctors to the system

Chief doctors, the directors of the hospital should be able to add doctors of their hospital to the system.

3.1.4.3 Add hospital staff to the system

Chief doctors, the directors of the hospital should be able to add the staff of their hospital to the system.

3.1.5 Functional requirements for authorities of MoH

3.1.5.1 View patient statistics

MoH authorities should be able to view statistical data of the patients. They should be able to view daily patient statistics at the country level, district level and hospital level. Also should be able to view the overall status until now. This allows them to understand the current COVID-19 situation of the country.

3.1.5.2 View hospital and bed statistic

MoH authorities should be able to view hospital and bed statistics. They should be able to view daily patient statistics at the country level, district level and hospital level. Also should be able to view the overall status until now. This allows them to understand the current situation of the hospitals in the country.

3.1.5.3 Get notified of the need for a new hospital

MoH authorities should be able to get notified of a new hospital with district

information if the queue of patients exceeds 4 allocations. The district will select from the majority of patients in the queue and if the number of districts is equal, then the district will be where the first patient in the queue belongs.

3.1.5.4 Add new hospitals

MoH authorities should be able to add new hospital details to the system. When adding a hospital MoH authorities will need to add hospital name, location, chief doctor's details and contact details.

3.1.5.5 Edit chief doctors; details

MoH authorities should be able to edit chief doctors' details to the system. Even though the chief doctor added in when adding the hospital details when the change of the chief doctor system should be updated with the new details.

3.1.6 Functional requirements for citizens

3.1.6.1 Register to the system as a patient with geolocation coordinates of the home

Citizens should be able to register in NCMS with the geolocation coordinates of the home if they have COVID-19 symptoms.

3.1.6.2 View patient statistics

Citizens should be able to view statistical data of the patients. They should be able to view daily patient statistics at the country level, district level and hospital level. Also should be able to view the overall status until now. This allows them to understand the current situation of the country.

3.2 Usability

3.2.1 Training time

Specific training is not required to understand the system. But on average it will take half an hour to completely understand and use all the functions of the system for beginners. People who are familiar with web-based systems need less than 15 minutes to master the system.

3.2.2 Language

Hence English is the main international language and the system will be developed in English.

3.2.3 User Interface

The user interface will follow the Microsoft user interface principles[1]. For the controls, proper colour codes and simplified languages will be used so that users can easily understand the interface.

3.2.4 User Experience

To standardize the GUI and enhance usability, Microsoft's User Interface Principles will be used.

3.3 Reliability

3.3.1 Availability

Since the NCMS is a system in the medical domain it needs to be updated with real-time data.

3.3.2 Accuracy

Since the NCMS will operate within the medical domain the accuracy is important, hence the data will be updated in real-time.

3.4 Performance and Security

3.4.1 Response time

Data transfer time:

To send data from website to server

Average: 5seconds Maximum: 10seconds

However, network speed could be a bottleneck, affecting response times and

loading times. Hardware has a minor impact.

3.4.2 Resource utilization

Data will not be loaded from the server unless requested by the user. For example, until user request hospitals and bed statistics, corresponding data will not be received from the server

3.4.3 Security

The NCMS will contain sensitive details of the patients. Privacy of the consumer data should be assured. Any information related to patients should not be used or shared without users consent.

3.5 Supportability

3.5.1 Naming convention

The NCMS will be implemented using camelCase[2] variable declaration at the coding level.

3.5.2 Class libraries

- Spring Framework
- Spring JPA
- AngularJs
- HTTP libraries

3.6 Design Constraints

3.6.1 Software Language

The web application will be developed using AngularJs. The backend will be a REST API developed by Spring Boot.

3.6.2 Class Libraries

- Spring Framework
- Spring JPA
- AngularJs
- HTTP libraries

3.7 Online User Documentation and Help System Requirements

Online user documents will be available for the Authorities of MoH, hospital staff, doctors and chief doctors.

3.8 Purchased Components

Email API is in the free trial during the development phase but it will be migrated to paid versions when the number of users is increased. There will be an associated hosting cost. The central database that will be used across the web app, and the web application itself should be hosted to a cloud server to be accessed by users via the internet.

3.9 Interfaces

3.9.1 User Interfaces

The NCMS will be a web application and this will be used by all the user roles of the system. In this application even though a citizen does not have an account registered in the system they will be able to view the statistics of the patients. The user can view the daily update at country level, district level, hospital level by choosing the level drop-down list. Also can view overall statistics by selecting the radio button. (figure 1)

If a citizen has a symptom of COVID-19 he/she can register the system as a patient. (figure 2).

After registration, the user will get a unique serial number and the hospital details. That will display under the ‘Info’ section. (figure 3)

The MoH authority is also able to view patient statistics, the user interface will be a bit similar to the citizen interface. (figure 1)

The MoH authority is able to view hospital and bed statistics. The user will be able to

select a hospital from the hospital dropdown. (figure 4)

After the queue exceeds 4 patient MoH will get notified from the system with the details of the new hospital. (figure 5).

After building the hospital the Moh authority should be able to add the new hospital details to the system. When adding the hospital details the user should enter the chief doctor details of the hospital. (figure 6)

The MoH authority is able to edit the chief doctors' details and add new MoH authority users to the system. The chief doctor is able to add doctors and the staff of the hospital. The interfaces will be a form like in figure 6.

The chief doctor is able to view hospital statistics. (figure 7)

The chief doctor is able to discharge the patient from the hospital. (figure 8)

The doctors are able to update the severity level of the patient. (figure 9)

The chief doctor, doctor and the hospital staff are able to view patient details of their hospital. The UI will be a bit similar to webserverfigure 8.

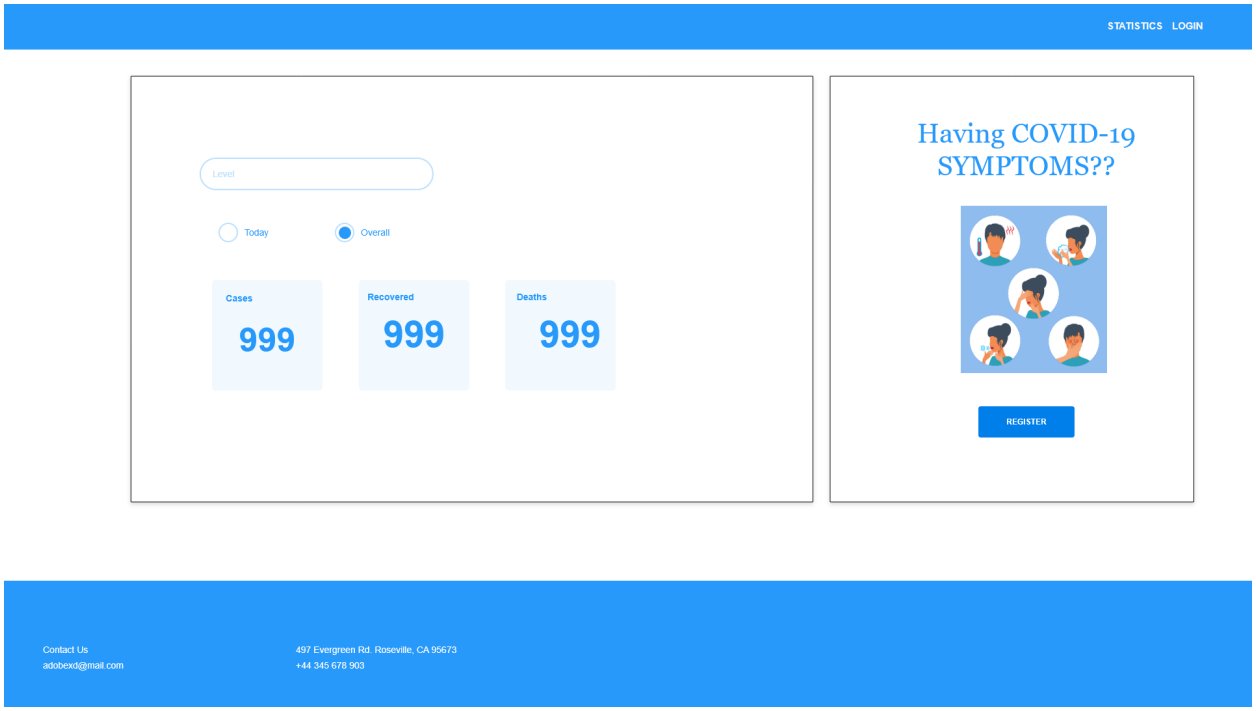

figure 1 - Patient Statistic

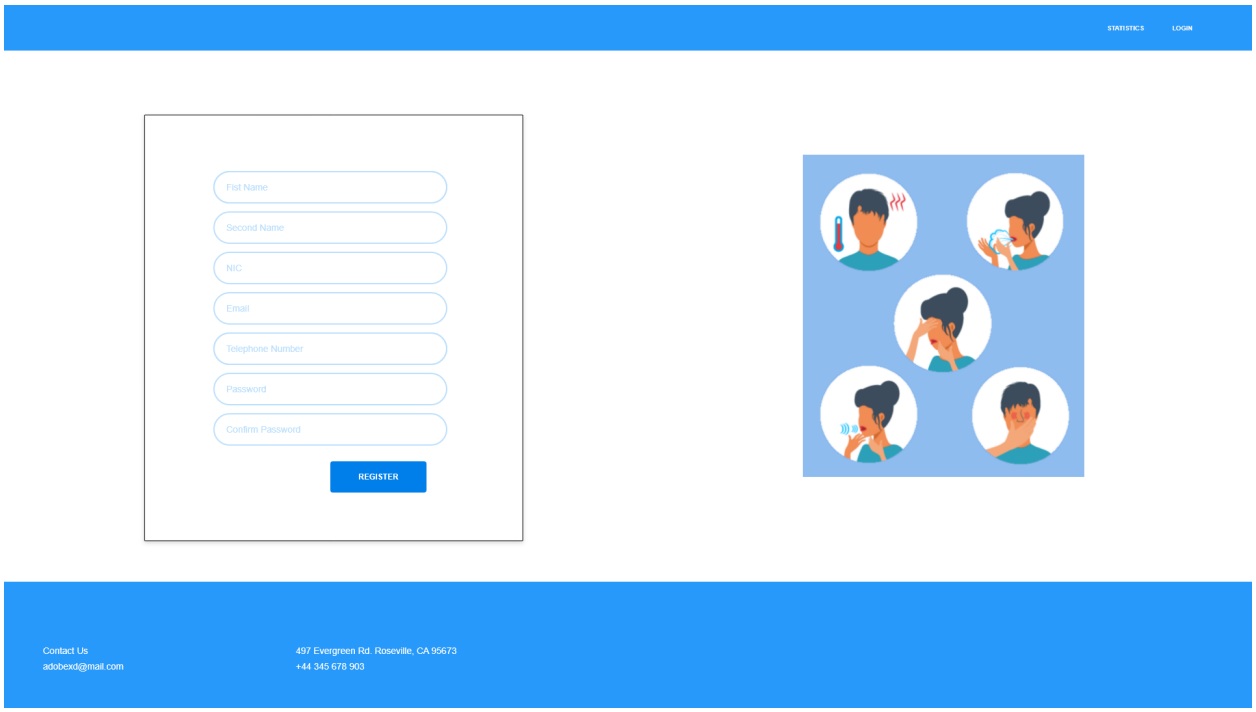

figure 2 - Register Form

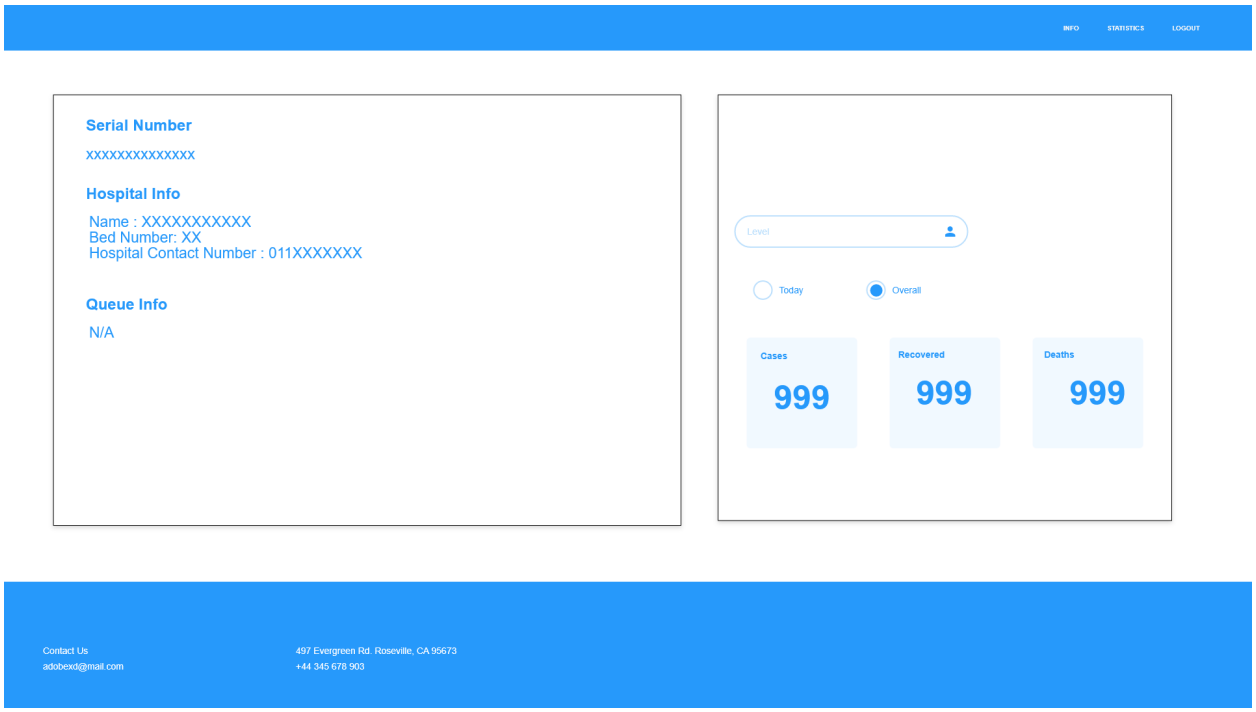

figure 3 - Information of patient

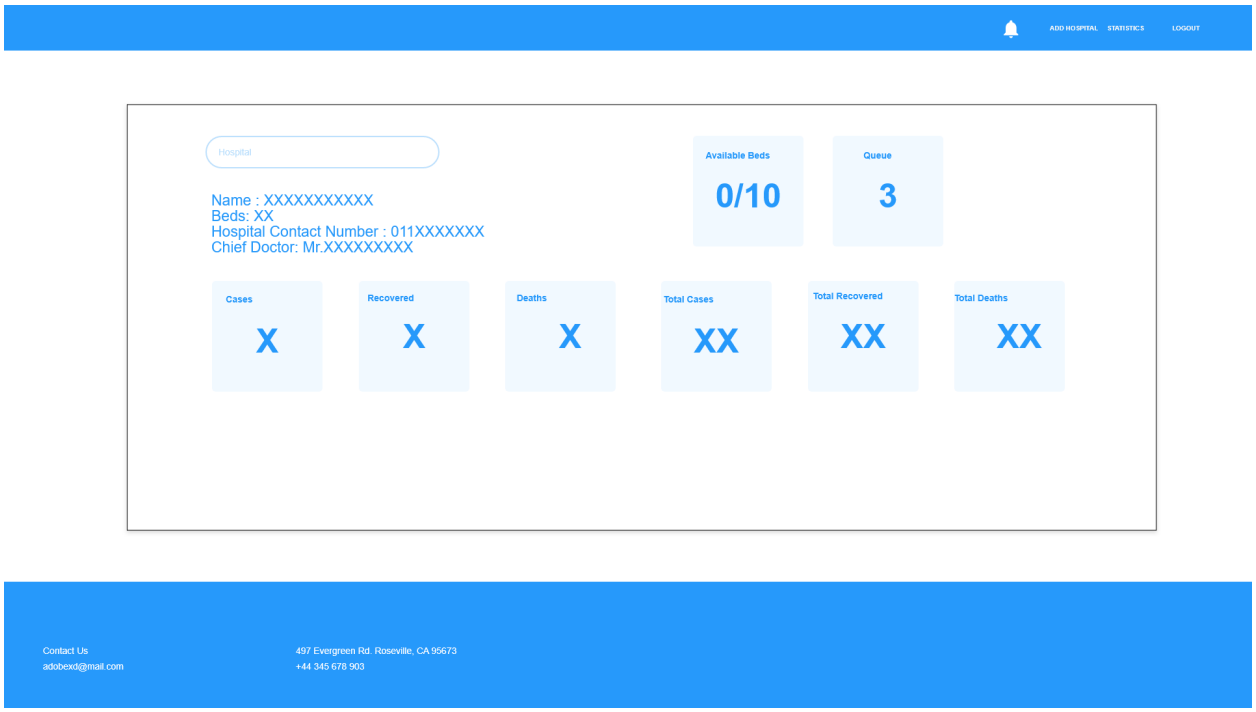

figure 4 - Hospital and bed statistic

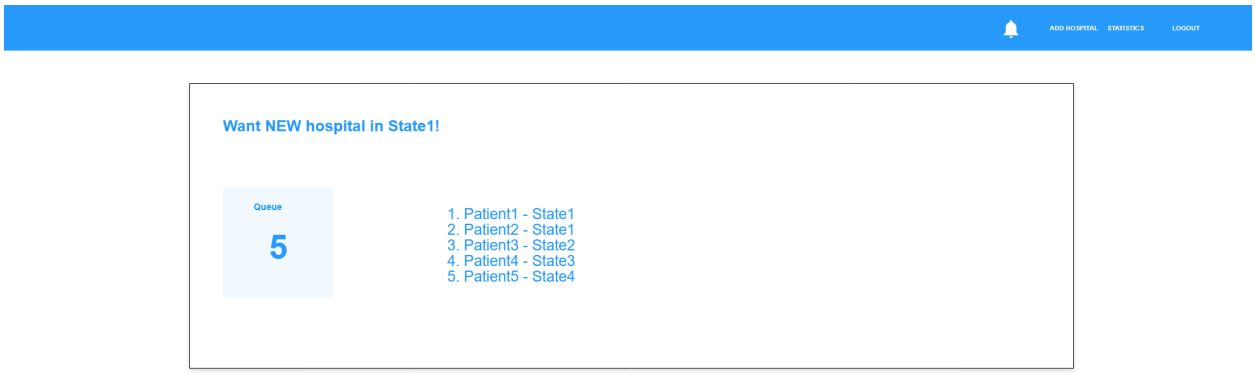

figure 5 - Notification

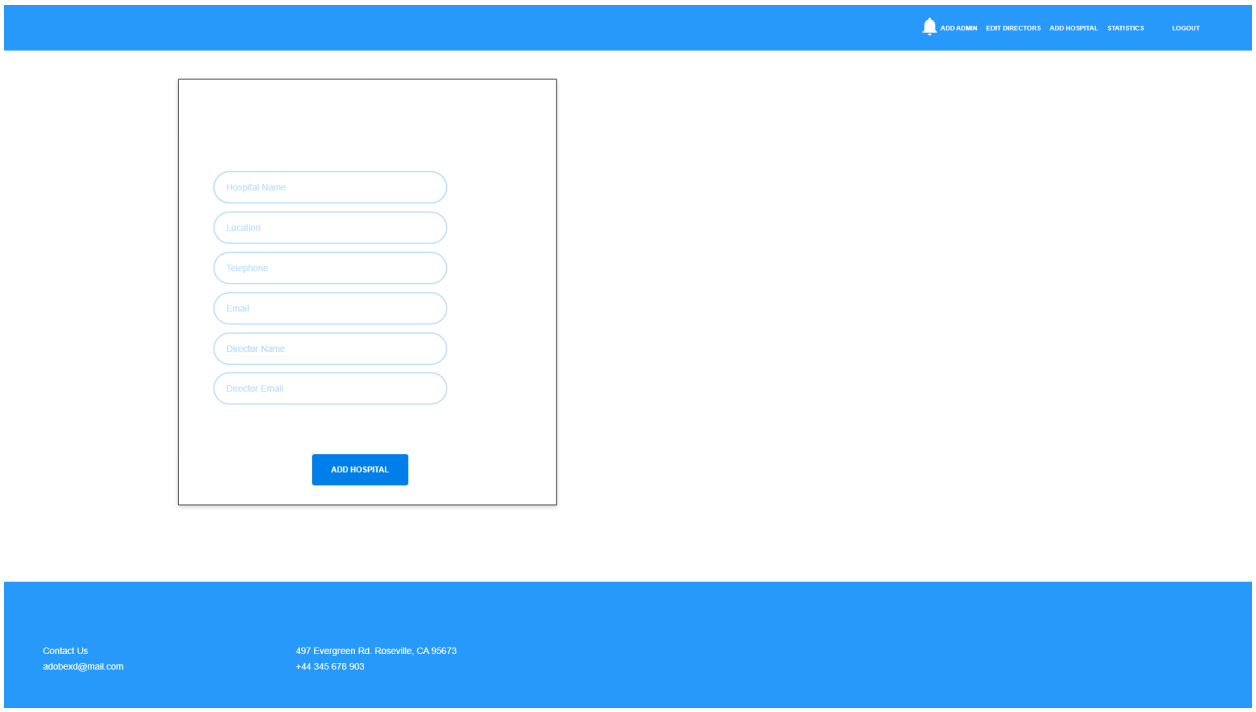

figure 6 - Add a hospital

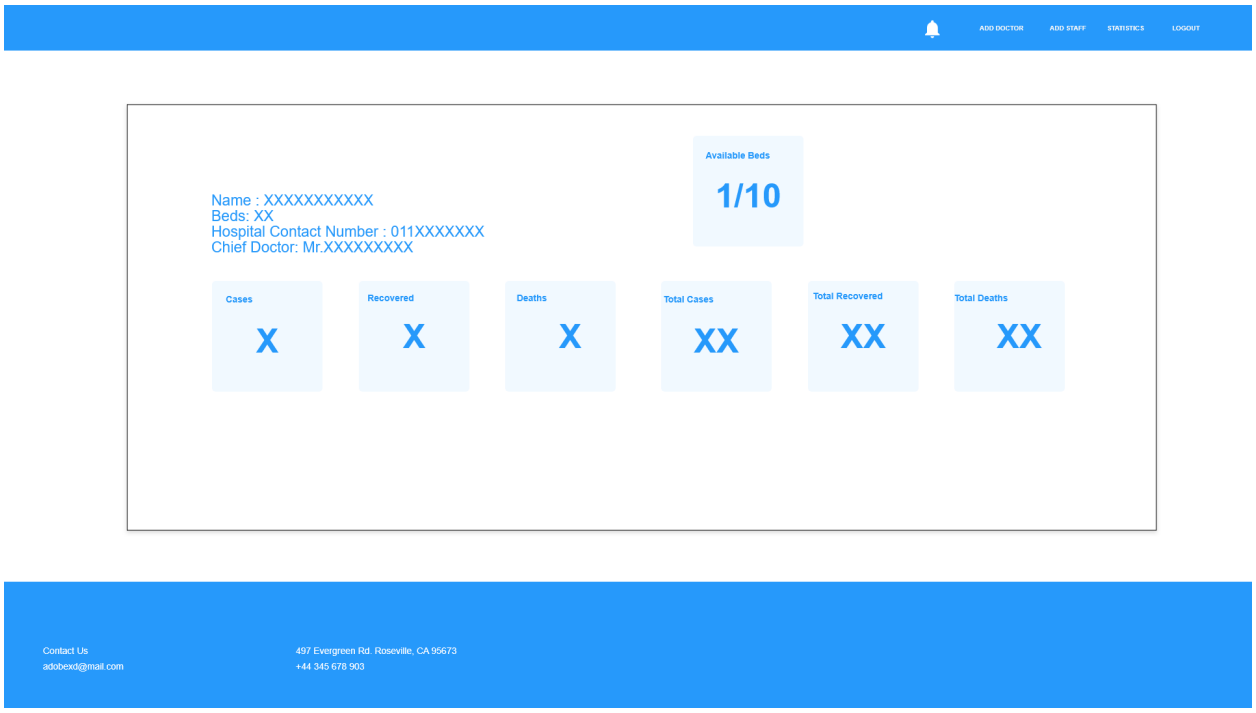

figure 7 - hospital statistic

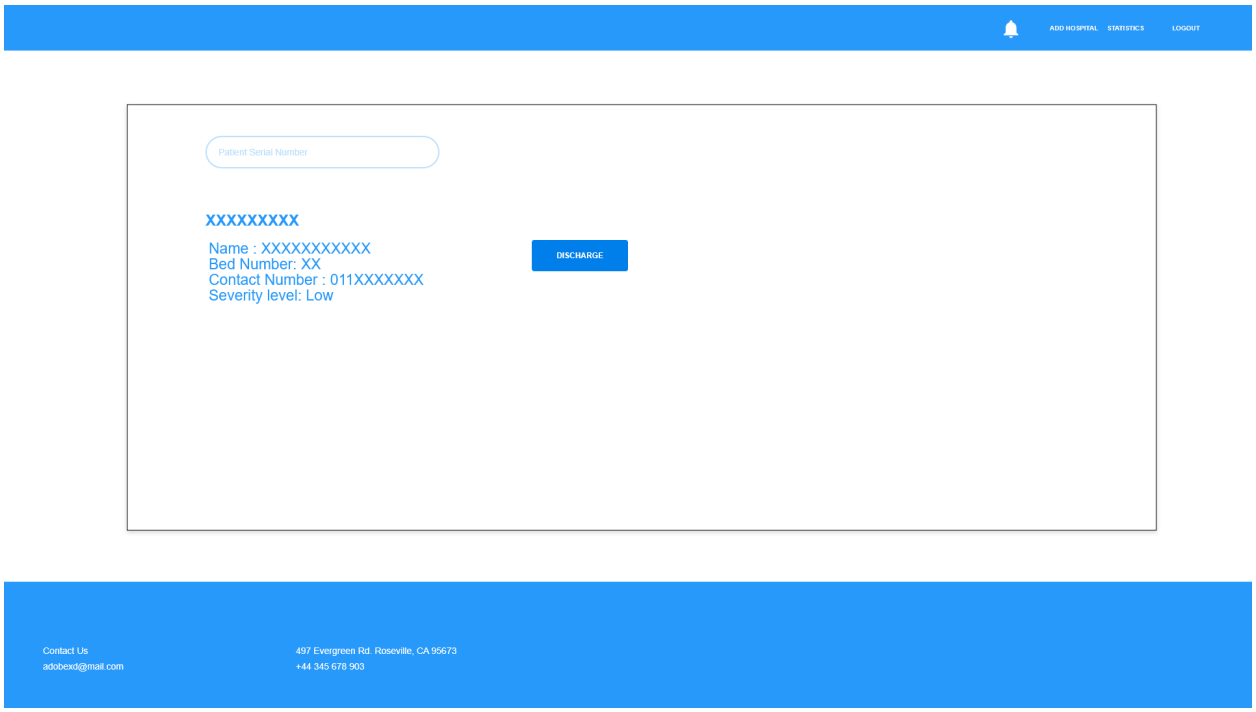

figure 8 - discharge

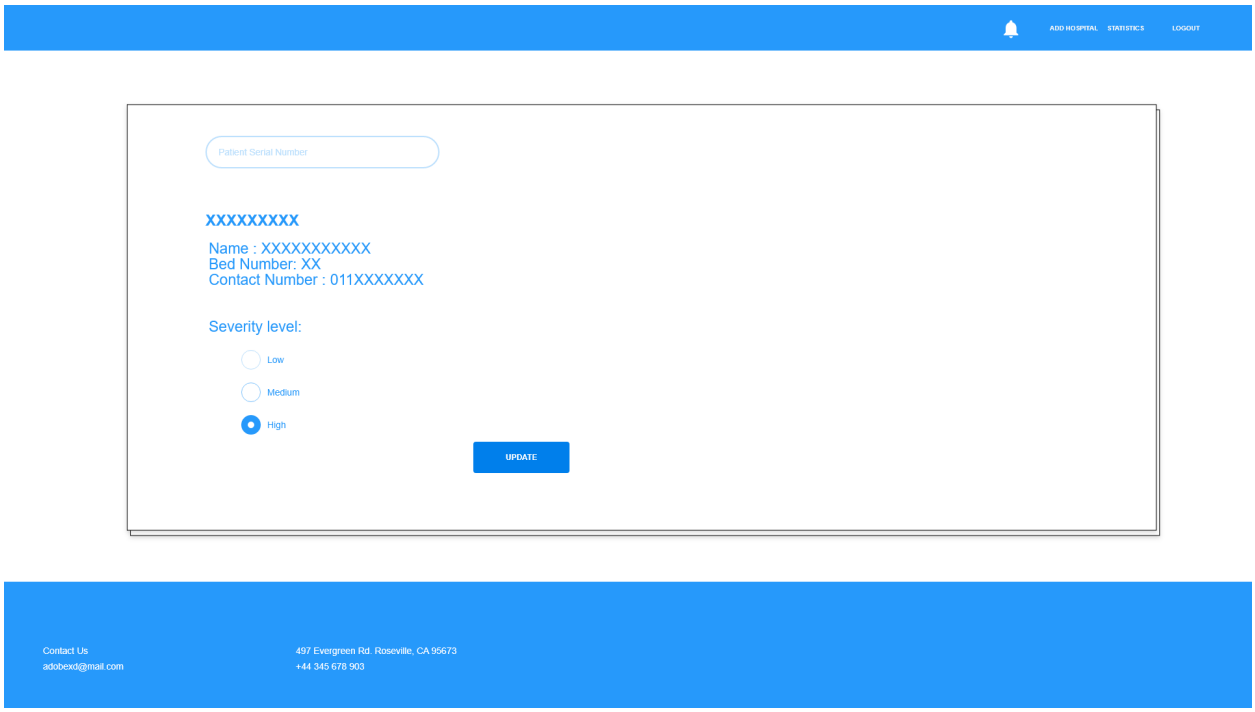

figure 9 - severity level

3.9.2 Hardware Interfaces

The web application does not have any designated hardware so that it does not have any direct hardware interfaces. The connection of the hardware to the database server and the backend service is managed by the underlying OS on the web server.

3.9.3 Software Interfaces

The web application will communicate with the REST API, the backend of the system. The authentication system backend will communicate with the Sendgrid email API. The communication with the MySQL database and the REST API consists of operations concerning read, write and modify data.

3.9.4 Communications Interfaces

The communication of the web applications is done by the underlying OS. The web application sends HTTP requests to the REST API and REST API sends HTTP responses to the web application.

3.10 Database Requirements

MySQL will be used as the database.

3.11 Licensing, Legal, Copyright, and Other Notices

At this point, there is nothing to license

3.12 Applicable Standards

No applicable standards at this point.

4. Supporting Information

4.1 Appendices

- Software architecture document