A PROJECT REPORT ON

"COVID-SHIELD"

Submitted to the Mangalore University in partial fulfillment of the requirement for the award of the Degree of

BACHELOR OF COMPUTER APPLICATION

Submitted by

AMAN S D'SOUZA

REGISTER NUMBER: 186231903

And

SANDEEP BHANDARI

REGISTER NUMBER: 186232002

UNDER THE VALUABLE GUIDANCE OF

Mrs. BHAT S. SNEHA GANESH

Project Guide

SHRI DHARMASTHALA MANJUNATESHWARA

COLLEGE OF BUSINESS MANAGEMENT

MANGALURU – 575003

2020 – 2021

Shri Dharmasthala Manjunatheshwara College of Business Management,
Mangaluru - 575003
(Affiliated to Mangalore University)

CERTIFICATE

This is to certify that the project report entitles **"COVID-SHIELD"** is an authenticated record of the project work carried out by **AMAN S D'SOUZA** bearing Register No.**186231903** **and** **SANDEEP BHANDARI** bearing Register No.**186232002** in partial fulfillment of the requirement for the award of **Bachelor's Degree in Computer Application** of Mangalore University under the guidance and supervision during the year 2020 - 2021.

Forwarded to Principal for Approval

Place : Mangaluru
Date : 27 September 2021

(Mrs. Bhat S. Sneha Ganesh)
Project Guide

Approved and Forwarded to Mangalore University

Place : Mangaluru
Date : 27 September 2021

(Mrs. Aruna P. Kamath)
Principal

Signature of the Examiners :

1…… **2……**

DECLARATION

We hereby declare that the project report titled as **"COVID-SHIELD"** has been prepared by us during the year 2020 – 2021 under the valuable guidance and supervision of **Mrs. Bhat S. Sneha Ganesh**, Assistant Professor and project guide, SDM College of Business Management, Mangaluru, in partial fulfillment of the requirement for the award of degree in Bachelor of Computer Application from Mangalore University for the academic year 2020 - 2021.

We also declare that this project is the result of our own effort and has not been submitted to any other University for the award of any degree or diploma.

……..	…….
AMAN S D'SOUZA	**SANDEEP BHANDARI**
Reg. No. : 186231903	**Reg. No. : 186232002**

Place : Mangaluru
Date : 27 September 2021

ACKNOWLEDGEMENT

Whatever we are able to put forward in terms of our work, is only due to few people from whom we have learnt. Thus it is a great pleasure in mentioning their names.

Working on the live project was very interesting and really enhanced our knowledge.

We would like to express our sincere gratitude to our beloved principal **Mrs. Aruna P. Kamath** without whose permission we would not be able to do our project and for taking keen interest for the students of BCA in providing useful guidelines and giving all the necessary facilities.

We extend thanks to our guide **Mrs. Bhat S. Sneha Ganesh**, Assistant Professor of Computer Application Department for her able guidance and constant encouragement, which helped us in successfully completing our project.

Finally, we extend thanks to our parents and friends who were directly or indirectly involved in the completion of our project.

…….. ……..….

AMAN S D'SOUZA **SANDEEP BHANDARI**

Reg. No. : 186231903 **Reg. No. : 186232002**

Place : Mangaluru
Date : 27 September 2021

CONTENTS

COVID – SHIELD (Modern Mankind)

"Synopsis"

1.1 Project Title

- Covid – Shield

1.2 Introduction

- This application uses ORS (Online Registration System) framework to link various hospitals across the country from Aadhaar card number and Mobile number based online registration and appointment system, where registration and appointment system through HMIS (Hospital Management Information System) has been digitalized.

1.3 Objective

- The main objective or the purpose of this web application is to reach the maximum number of patients who is suffering from the virus namely, Covid-19 who is in need of vaccine and beds in hospital across the country through the ORS.
- Patients who have recovered from Covid-19 are eligible to donate plasma and can help the patients recover soon from this virus.
- Large problems are sub-divided into smaller once to make them understandable and easy for finding solutions and in this project all the task are been sub-divided and categorized.

1.4 Hardware and Software Requirements

1.4.1 Hardware Requirements

- Processor : Intel(R) 2.10GHz or any higher version.
- Installed memory (RAM) : 4 GB or more.
- Hard Disk : 100 GB or above.

1.4.2 Software Requirements

- Operating System : Windows 7 or above.
- Text Editor : Notepad++, Sublime Text 3.
- Server : XAMP.
- Browser : Google Chrome, Mozilla Firefox or any other.

- Microsoft SQL Server .

1.5 Languages Used

- Front End: HTML, CSS, JavaScript.
- Back End: PHP.
- Database: MySQL.

1.6 Operating System Used

- Windows 10 Operating System.

1.7 Project Category

- Web Based Relational Database Management System.

1.8 Modules

1.8.1 Modules list

User Side (Patient):

- Register and Login.
- Home Page: Home, About, Services, Feedback.
- About: What we do, Our Mission, Our Goal.
- Services: Live Tracker, Covid Test, Vaccination, Bed Booking, Plasma Donation and Notification.
- Payment.
- Feedback
- Logout.

Vendor Side (Hospital):

- Register and Login.
- Dashboard.
- Requests: Test Request, Vaccine Request and Bed Request.
- Plasma Donator.
- Updates: Test Results, Vaccine Status, Bed Status and Test Info.
- Logout.

Admin Side:

- Login.
- Dashboard.
- Registered Hospitals.
- Hospital Approval.
- Registered Users.
- Plasma Donors.
- View Feedback.
- Logout.

1.8.2 Modules description

User Side (Patient):

- Register & Login
  - User (Patient) can register and login with user details.

- Home Page: Home, About, Services, Feedback
  - Home: User can view information related to Covid-19.
  - About: User can understand the mission and goal of the website.
  - Services: User get the facility like Covid Test, Vaccination, Bed Booking and Plasma Donation.
  - Feedback: User can response / comment about the website.

- Services: Live Tracker, Covid Test, Vaccination, Bed Booking, Plasma Donation and Notification
  - Live Trackers: User can view the actual status of Covid-19 in the world.
  - Covid Test: User can search and request to the hospital for the Covid test and after the approval, patient can take the test from the hospital.
  - Vaccination: User can search the request to the hospital for vaccination and after the approval, patient can take the vaccination from the hospital.
  - Bed Booking: User will book the bed in the hospital.
  - Plasma Donation: User can search and request to the hospital for the plasma donation.
  - Notification: User can view the Covid-19 test result report, vaccination taken details and details about plasma donors.

- Payment
  - User can pay their bill amount through debit and credit cards.

- Logout
  - User can logout from the website.

Vendor Side (Hospital):

- Register & Login
  - Vendor (Hospital) can register and login with vendor details.
- Dashboard
  - Vendor can view the Covid status of their particular hospital.
- Requests: Test Request, Vaccine Request and Bed Request
  - Test Request: Vendor can view request done by user for Covid test.
  - Vaccine Request: Vendor can view request done by user for vaccination.
  - Bed Request: Vendor can view request done by user for bed.

- Plasma Donator
  - Vendor can view the details of the user who are interested to donate the plasma.

- Updates: Test Results, Vaccine Status, Bed Status and Test Info
  - Test Results: Vendor will update report whether it is Positive / Negative.
  - Vaccine Status: Vendor will update the availability of vaccine in the hospital.
  - Bed Status: Vendor will update the availability of bed in the hospital.
  - Test Info: Vendor will update the price and duration of test.

- Logout
  - Vendor will logout from the website.

Admin Side:

- Login
  - Admin will login.
- Dashboard
  - Admin can view the Covid status of all the hospital.

- Registered Hospitals
  - Admin can view the number of hospitals registered and can delete them also.

- Hospital Approval
    - Admin can Approve / Reject the hospital approval.

- Registered Users
    - Admin can view the user details.

- Plasma Donors
    - Admin can view the plasma donor's details.

- View Feedback
    - Admin can view the feedback gave by the user.

- Logout
    - Admin will logout.

1.9 Assumptions

- User should access Covid-Shield application only with desktop or laptop device.

1.10 Scope

- This application is useful if the community adopts.
- This application can be used by n number of Hospitals.
- This application is user friendly and reduces the time and human errors.
- It can be implemented to handle pandemic and health related issues in future

1.11 Conclusion

- This project satisfies all the requirements and also capable to provide facilities to the users in this critical situation.
- This web-based application helps to reduces the waiting time of the users.
- Irrespective of areas(city/village) this application helps every needy people.

"Software Requirement Analysis and Specification"

2.1 Introduction

- The SRS document describes the external behaviour of the software.
- The work of a software developer is to study the system to be developed and specify the user requirements before going to the design phase.
- This document will let us know how this system behaves and responds to it.

2.2 Purpose

- Software Requirements Specification (SRS) describes how the software should work and requirements or the functions of the software.
- This is the first step in the implementation of any project or the software.
- This document contains the requirements of the client which is discussed and put into a format and referred in the future implementation of the project.
- This document is the main reference for the development of the software and for the testing of the project.

2.3 Scope

- This Software Requirements Specification describes the requirement of the system.
- This is meant for the developer to use and will be validating the final delivered system, any changes made to the requirement in the future will have to go through the formal change approval process.

2.4 Definition, Acronyms, and Abbreviations

SRS	Software Requirements Specifications
DFD	Data Flow Diagram
CFD	Context Flow Diagram
SQL	Structured Query Language
RAM	Random Access Memory
PHP	PHP Hypertext Pre-processor
CSS	Cascading Style Sheet
DBMS	Database Management System
TCP/IP	Transmission Control Protocol/ Internet Protocol
GUI	Graphical User Interface

2.4.1 References

- www.w3schools.com
- www.fontawesome.com

2.5 Overall Description

2.5.1 Product perspective

- This application will be connected to various hospitals across the country, where registration and appointment has been digitalized.
- The users of the system are categorized as admin and user and vendor.
- Application provides good and easy graphical user interface to both new as well as experienced users of the computers.
- Through the application, irrespective of area patient will get appointment in the respective hospital.
- This project mainly focuses on the solution regarding the pandemic situation.

2.5.2 Product Function

- The project "covid-shield" is developed to provide the help for the mankind who is suffering from this pandemic.
- All the available hospitals are authorized and are filtered.
- The system has kept updated on current situation and availability of the vaccines, hospitals and beds.
- It has 3 level of Actors.

User Side (Patient):

- Register and Login.
- Home Page: Home, About, Services, Feedback.
- About: What we do, Our Mission, Our Goal.
- Services: Live Tracker, Covid Test, Vaccination, Bed Booking, Plasma Donation and Notification.
- Payment.
- Feedback
- Logout.

Vendor Side (Hospital):

- Register and Login.
- Dashboard.
- Requests: Test Request, Vaccine Request and Bed Request.
- Plasma Donator.

- Updates: Test Results, Vaccine Status, Bed Status and Test Info.
- Logout.

Admin Side:

- Login.
- Dashboard.
- Registered Hospitals.
- Hospital Approval.
- Registered Users.
- Plasma Donors.
- View Feedback.
- Logout.

2.5.3 User Characteristics

- The system is designed to provide easy to use simple system.
- The user must have a basic knowledge about the computer and web application.
- The user need not be a technical person i.e., not much programing knowledge is required.
- The user should have the knowledge about internet.
- The user should be able to understand the GUI (Graphical User Interface).

2.5.4 General Constraints

- It requires all the mandatory fields to be filled with the exact and valid information.

2.5.5 Assumption and Dependencies

- The system is completely dependent on internet connection.
- The information provided by the user should be genuine.

2.6 Specific Requirements

- This section describes all the details and the system developer needs to know for designing and developing this system.
- These requirements can be organized by the modes of operation user class object, features, and functional hierarchies.

2.6.1 External Interface Requirements

- It specifies all the interface of the system to the people, hardware and other system.

2.6.1.1 User Interfaces

- The user can access the site through a web browser.
- Home page which has links to other pages.
- The start validation is provided to the login form.
- The permission to use the system is provided on successful validation.

2.6.1.2 Hardware Interfaces

- Any device which has a web browser installed in it.

2.6.1.3 Software Interfaces

- In software Interfaces, the system requirements which is required for the project has to run on the system is specified.
    - Any web browser

2.6.1.4 Communication Interfaces

- TCP/IP

2.6.2 Functional Requirements

- Functional Requirement specifies which output should be produced for the given inputs.
- All inputs are entered according to the data type and no blanks are allowed for mandatory fields.
- Invalid inputs are not allowed in the system and it prompts to re-enter the data.
- Appropriate error messages will be displayed.

User Side (Patient):

- Register & Login
    - User (Patient) can register and login with user details.

- Home Page: Home, About, Services, Feedback
    - Home: User can view information related to Covid-19.
    - About: User can understand the mission and goal of the website.
    - Services: User get the facility like Covid Test, Vaccination, Bed Booking and Plasma Donation.
    - Feedback: User can response / comment about the website.

- Services: Live Tracker, Covid Test, Vaccination, Bed Booking, Plasma Donation and Notification
    - Live Trackers: User can view the actual status of Covid-19 in the world.

Covid – Shield

- Covid Test: User can search and request to the hospital for the Covid test and after the approval, patient can take the test from the hospital.
- Vaccination: User can search the request to the hospital for vaccination and after the approval, patient can take the vaccination from the hospital.
- Bed Booking: User will book the bed in the hospital.
- Plasma Donation: User can search and request to the hospital for the plasma donation.
- Notification: User can view the Covid-19 test result report, vaccination taken details and details about plasma donors.

- Payment
  - User can pay their bill amount through debit and credit cards.
- Logout
  - User can logout from the website.

Vendor Side (Hospital):

- Register & Login
  - Vendor (Hospital) can register and login with vendor details.
- Dashboard
  - Vendor can view the Covid status of their particular hospital.
- Requests: Test Request, Vaccine Request and Bed Request
  - Test Request: Vendor can view request done by user for Covid test.
  - Vaccine Request: Vendor can view request done by user for vaccination.
  - Bed Request: Vendor can view request done by user for bed.
- Plasma Donator
  - Vendor can view the details of the user who are interested to donate the plasma.
- Updates: Test Results, Vaccine Status, Bed Status and Test Info
  - Test Results: Vendor will update report whether it is Positive / Negative.
  - Vaccine Status: Vendor will update the availability of vaccine in the hospital.
  - Bed Status: Vendor will update the availability of bed in the hospital.
  - Test Info: Vendor will update the price and duration of test.

- Logout
  - Vendor will logout from the website.

Admin Side:

- Login
  - Admin will login.
- Dashboard
  - Admin can view the Covid status of all the hospital.
- Registered Hospitals
  - Admin can view the number of hospitals registered and can delete them also.
- Hospital Approval
  - Admin can Approve / Reject the hospital approval.
- Registered Users
  - Admin can view the user details.
- Plasma Donors
  - Admin can view the plasma donor's details.
- View Feedback
  - Admin can view the feedback gave by the user.
- Logout
  - Admin will logout.

2.6.3 Non-Functional Requirements

2.6.3.1 Performance requirements

- The performance of overall system should be faster and error free with built in error checking and correction facilities.
- In order to run this application, we require:
  - An Internet with minimum 56 kbps bandwidth.
  - To access this page, we require IE6 or any higher version browser.

2.6.3.2 Design Constraints

- The application shall have a relational database.
- Email field to be entered with the text in email format itself.
- Should specify the information for all the mandatory fields.
- The application will display error messages to the user when an error is detected.

2.6.3.3 Security Requirements

- The solidness of the application is maintained by the use of login system and Signup system with certain set of constraints.

2.6.3.4 System attributes

- This section of Covid-Shield application SRS describes the applications attributes and its properties.

Reliability:

- The data will be maintained in a centralized server.
- The changes in the data will be only done by the authorized users.
- The reliability of data is well-maintained, but the inputs should be valid.

Maintainability:

- The SRS document can be referred for validation during maintenance stage.
- The developers take up the responsibility of system maintenance.

Usability:

- The user interface of the application is very user friendly that the users can use the system without any confusion, but the user should have basic knowledge about the computer and internet.

Portability:

- The application can be used in any system at any time but system should have a strong and active internet connection.

"System Design"

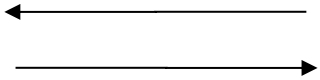
3.1 Introduction

- The software analysis and system design mainly deal with the software development activity.
- The Covid-Shield design document explains how exactly this web-based application will work.
- It includes Use case models, DFDs, ER Diagrams, and the class diagrams.

3.2 Data Flow Diagram

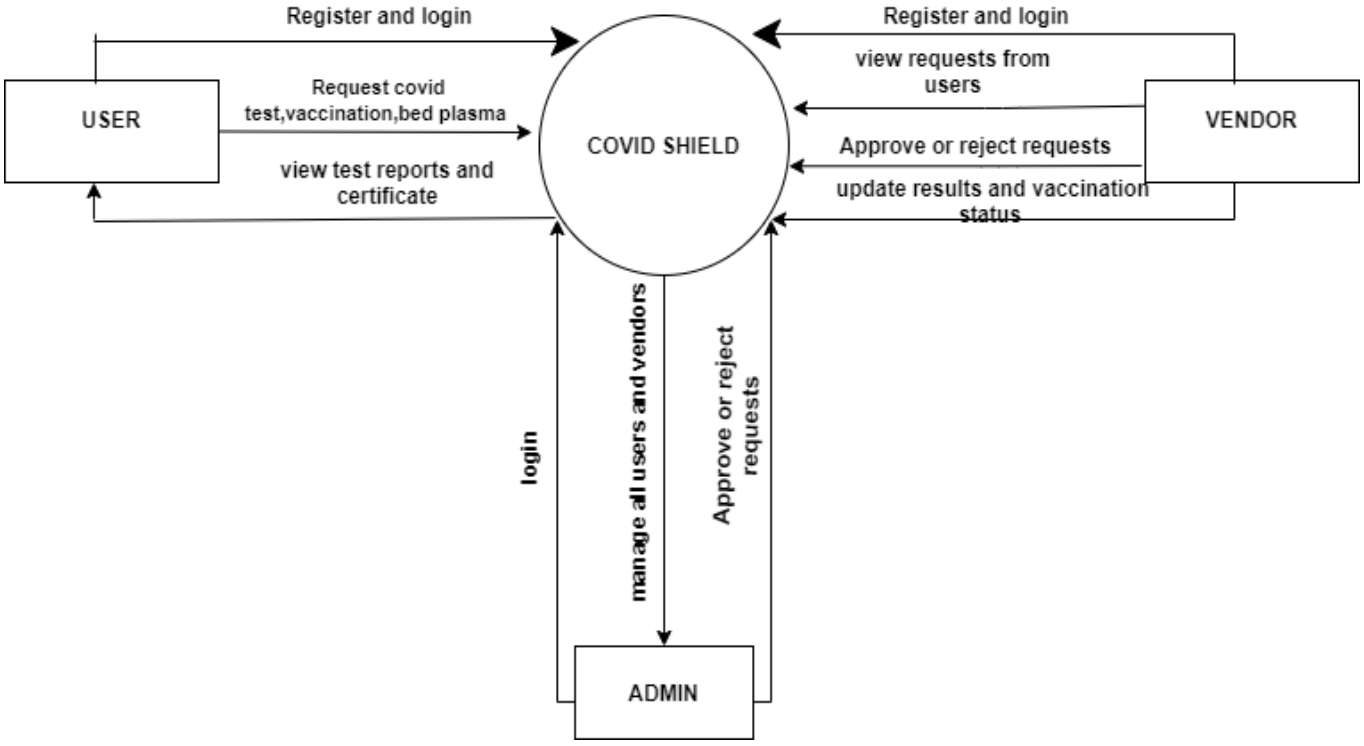
- A DFD shows the flow of data through system.
- It views a system as a function that transforms the input into desired output.
- Data flow diagram is drawn to pictorially represent the data stored, processed, entities involved in the system and information flow.
- DFDs are useful in understanding a system and can be efficiently used during problem analysis.

Basic Notations in DFD:

Notation	Components	Descriptions
	Process	A circle represents a process or transform that is applied to data or control and changes it in some way.
	External Entity	A rectangle is used to represent an external entity that is system element or another system that produces information by the software or receives the information produced by the software.
	Data Flows	An arrow represents one or more data items or data objects.
	Data Store	Data stores are repositories of data in the system. They are sometime referred to as files.
	Data flows	Data flows are pipelines through which packets of information flow. Label the arrows with the name of the data that moves through it.

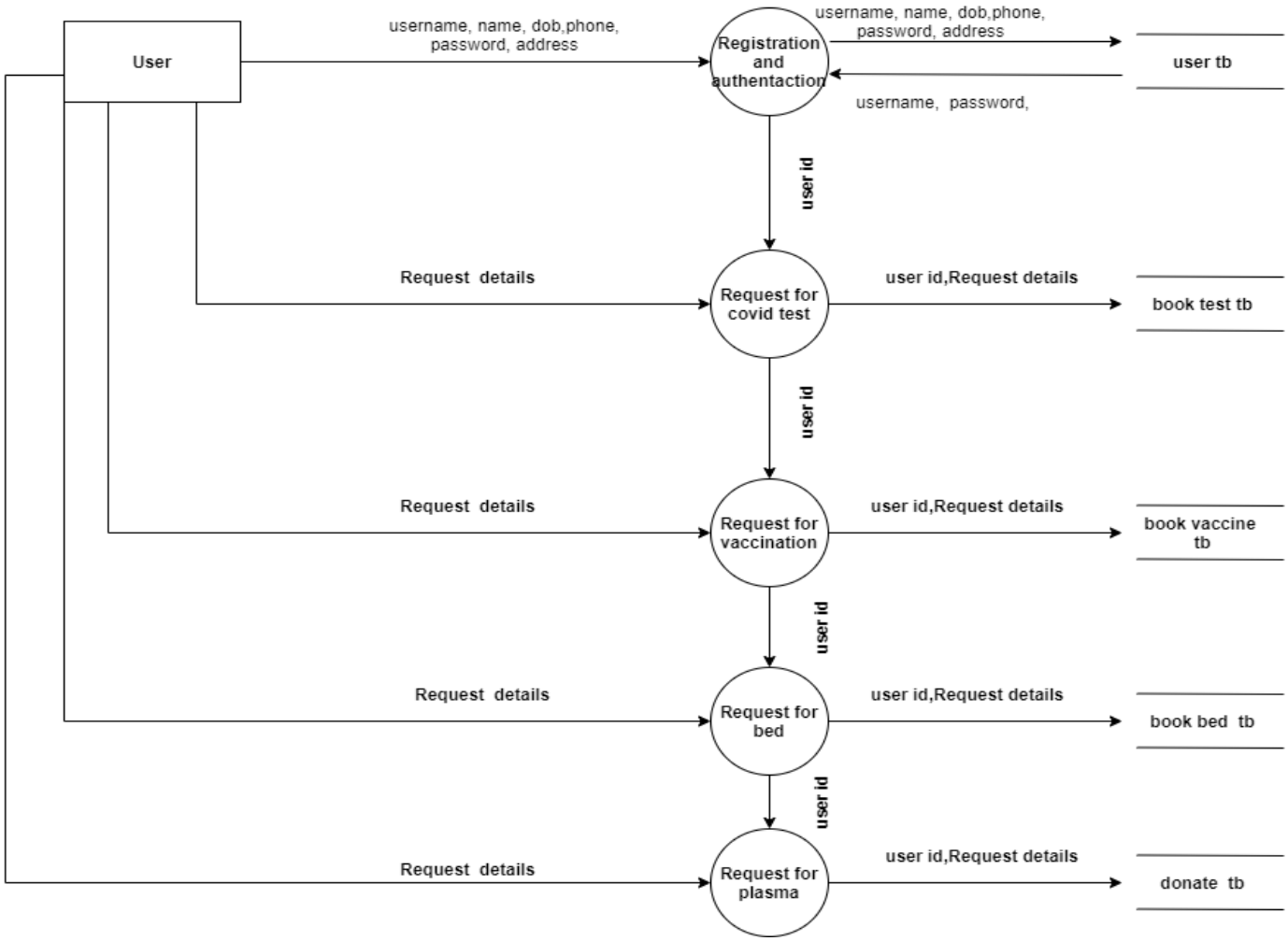
3.3 Context Flow Diagram (CFD)

- Context flow diagram is a top-level data flow diagram.
- It only contains one process node that generalize the function of the entire system in relationship to external entities.
- In context diagram the entire system is treated as a single process and all its inputs, outputs, sinks and sources are identified and shown.

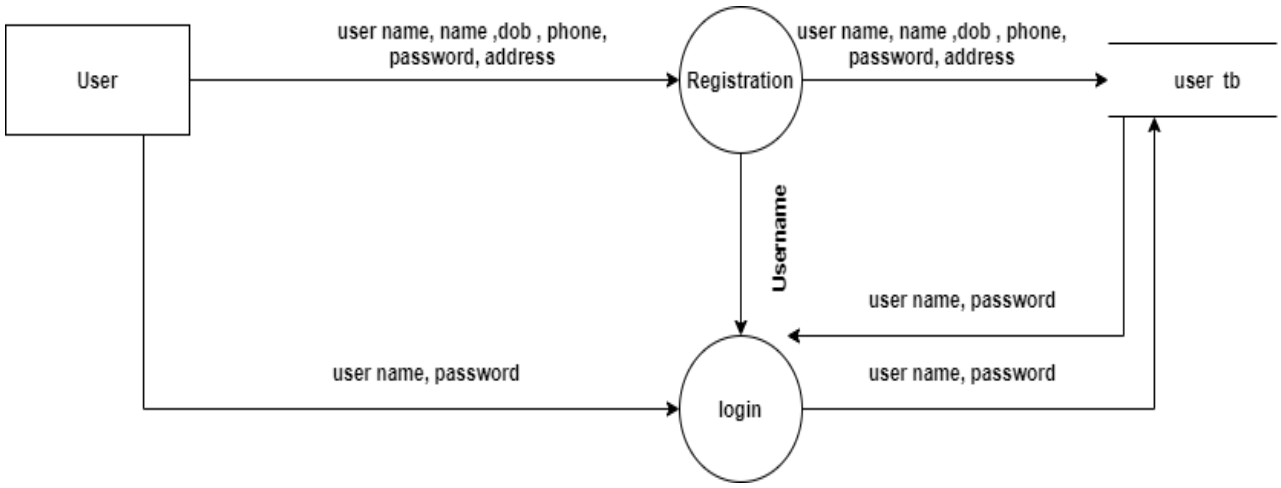


Data Flow Diagram (DFD)

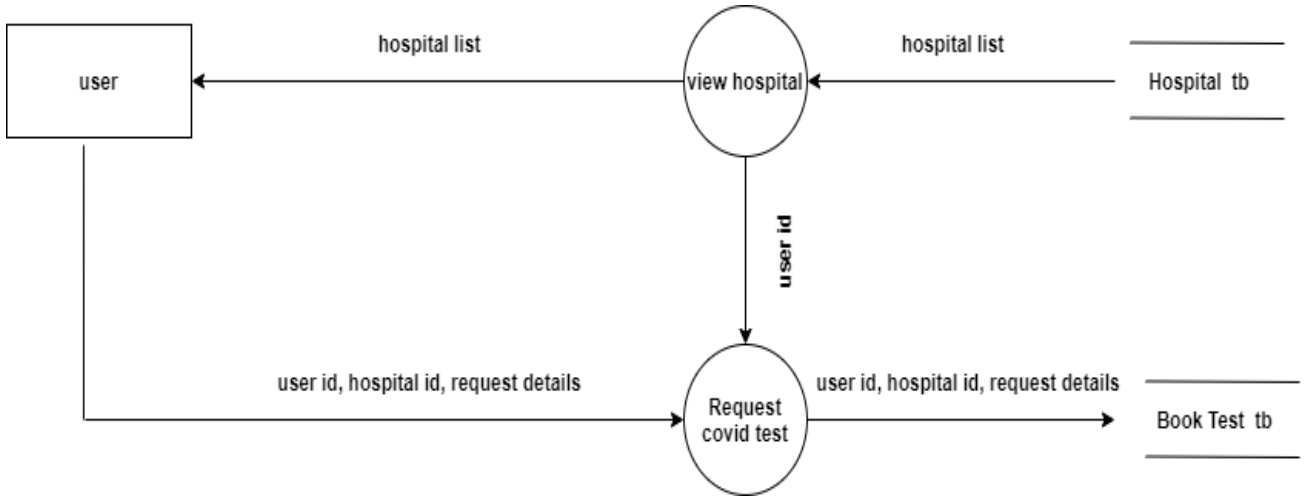
3.3.1 Level-1 DFD for User



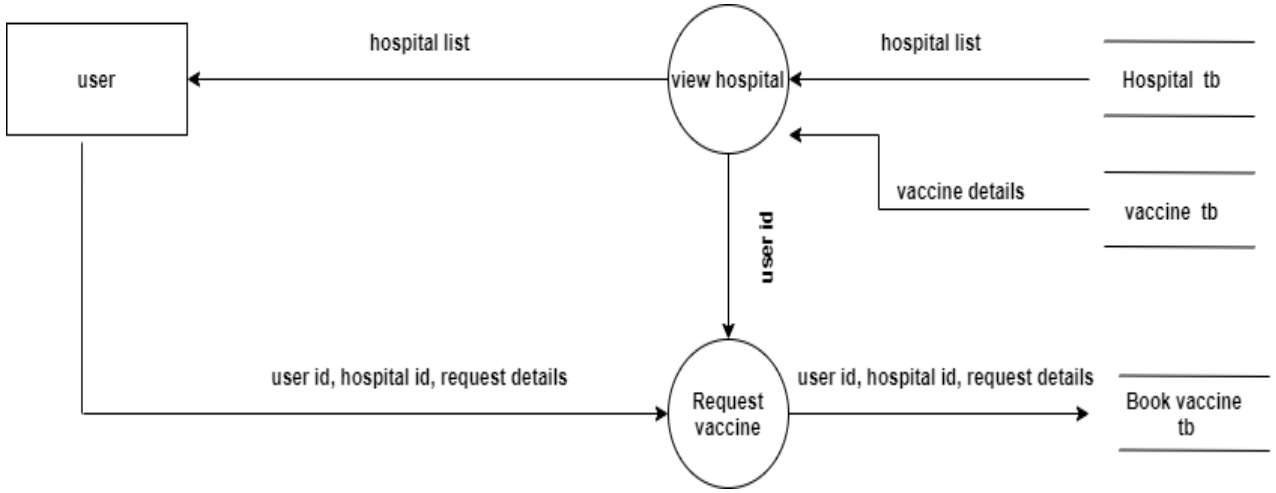
3.3.2 Level-2 DFD for User (Authentication and Login)



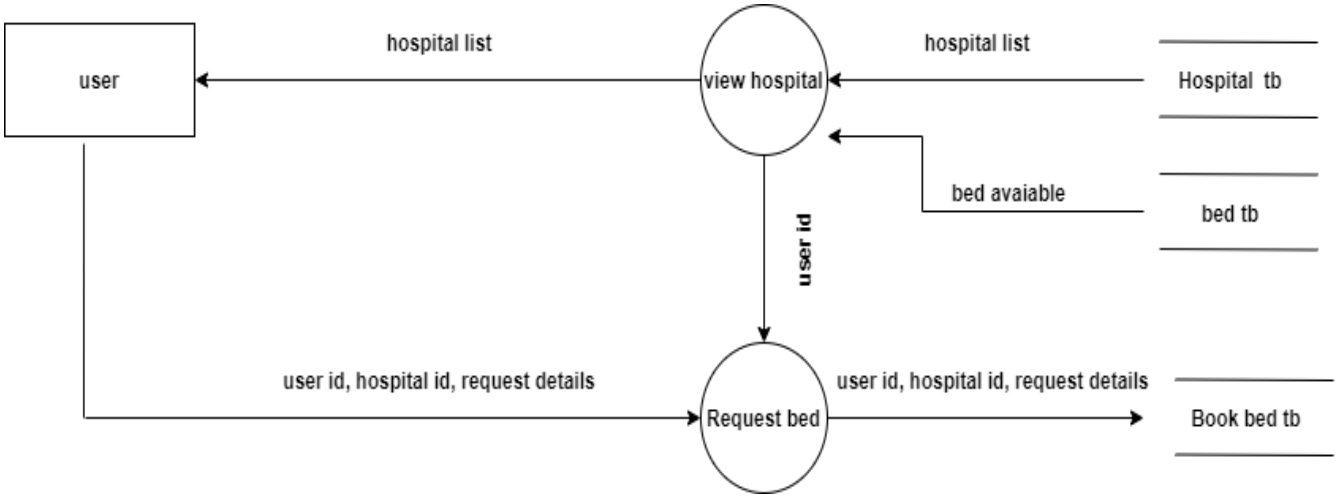
3.3.3 Level-2 DFD for User (Request Covid Test)



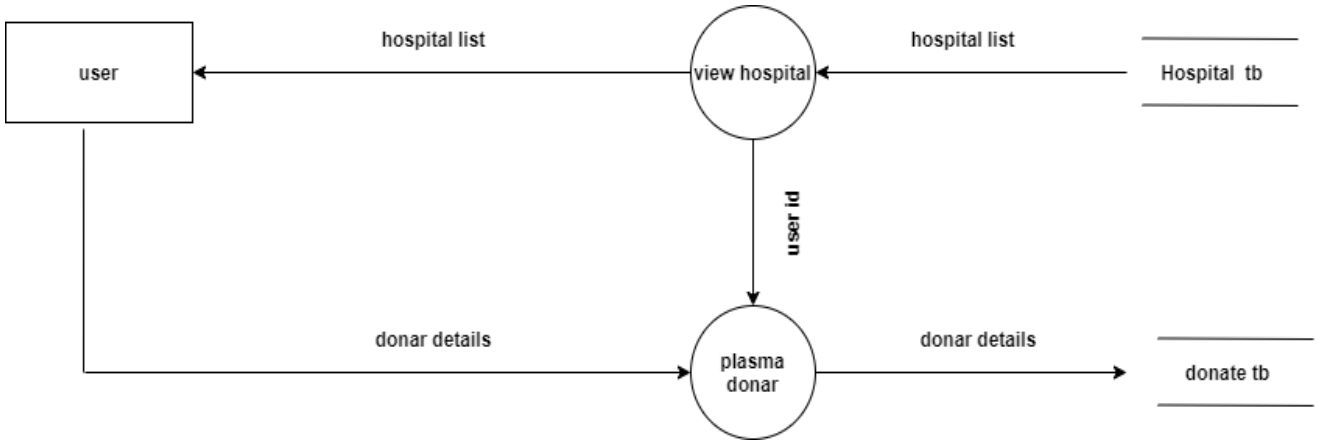
3.3.4 Level-2 DFD for User (Request Vaccine)



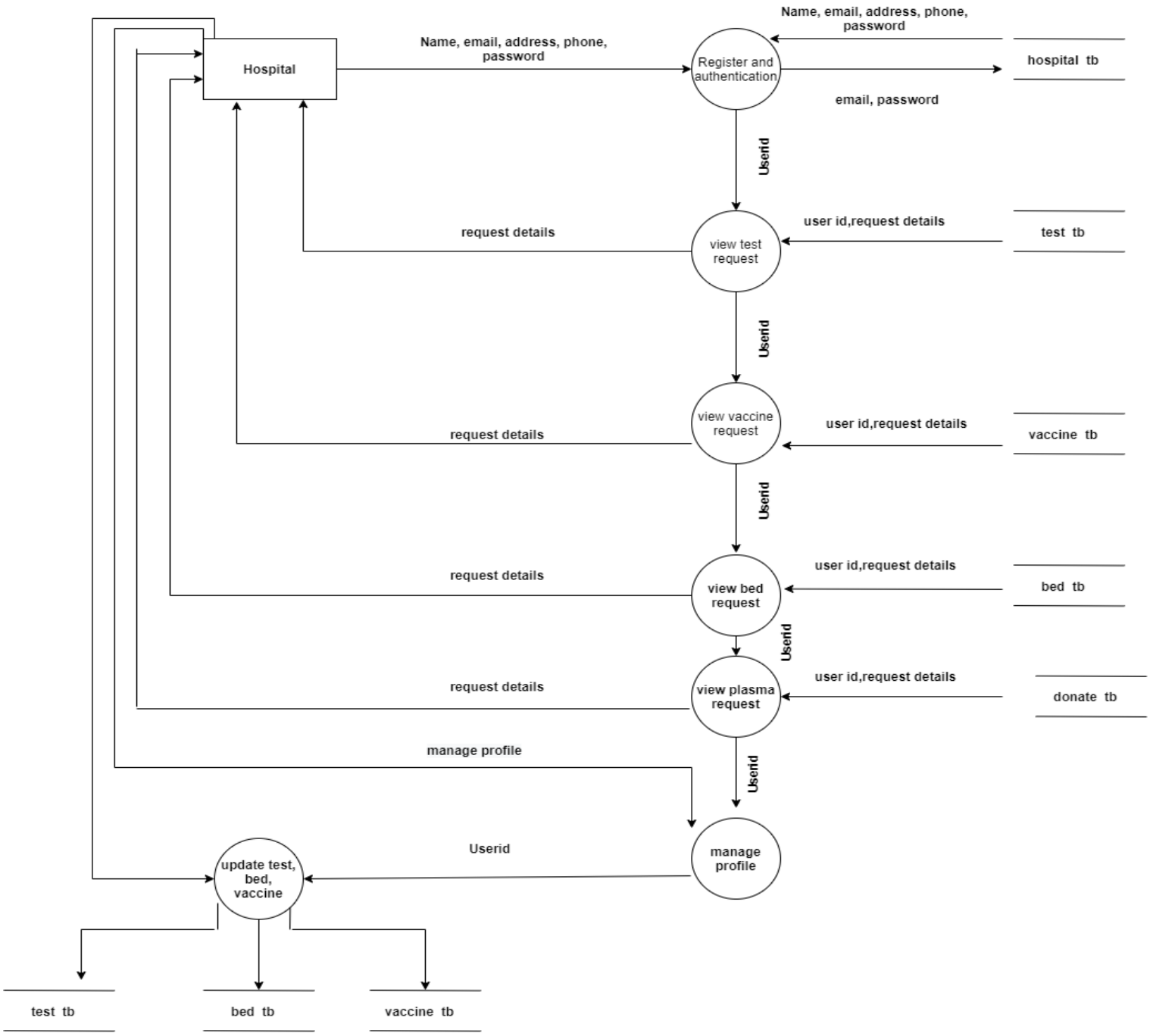
3.3.5 Level-2 DFD for User (Request Bed)



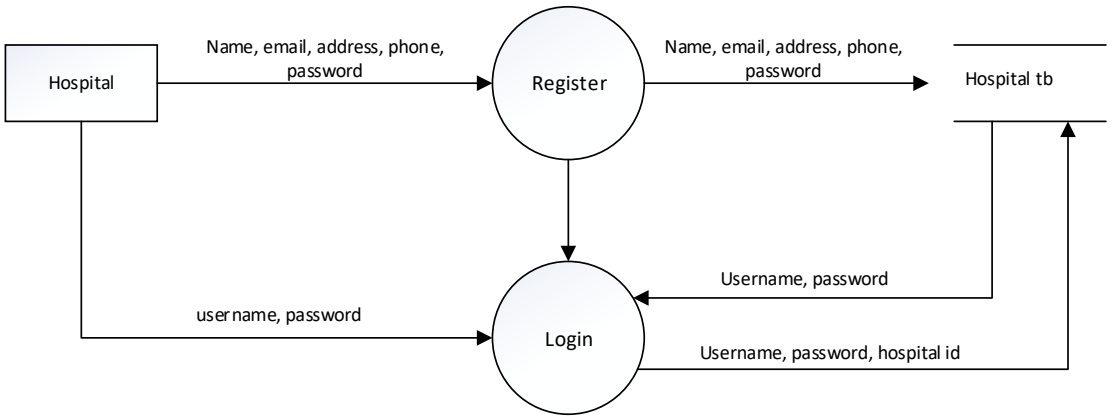
3.3.6 Level-2 DFD for User (Plasma Donation)



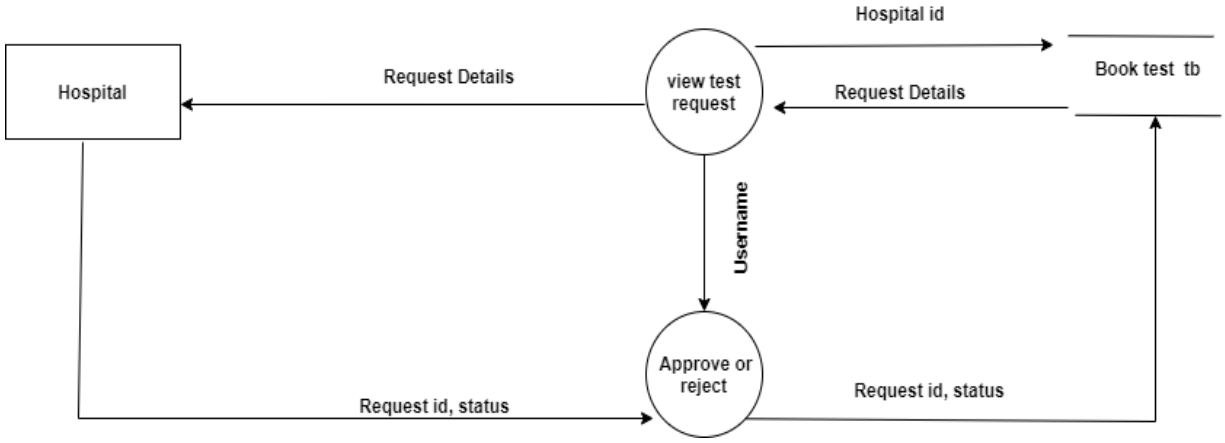
3.3.7 Level-1 DFD for Vendor



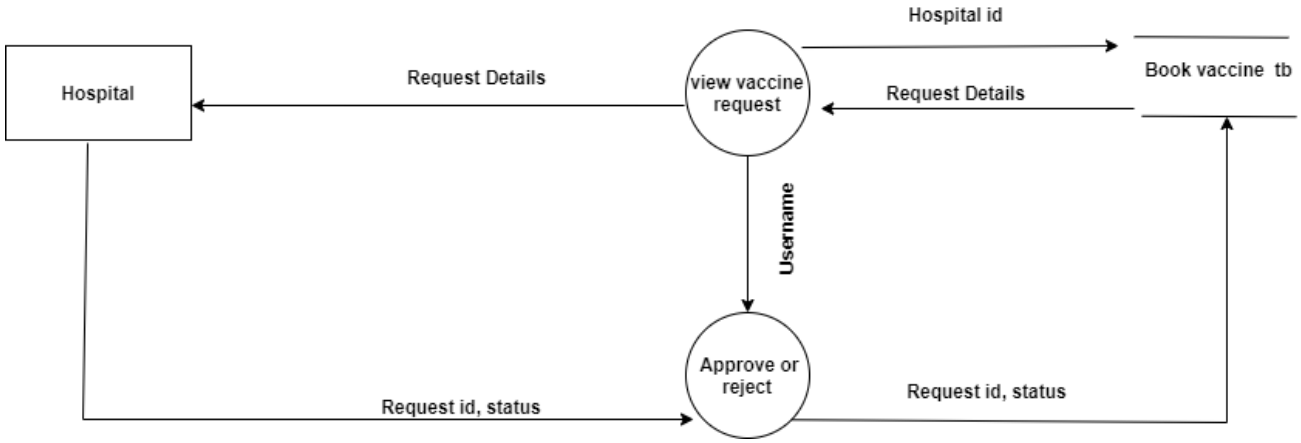
3.3.8 Level-2 DFD for Vendor (Authentication and Login)



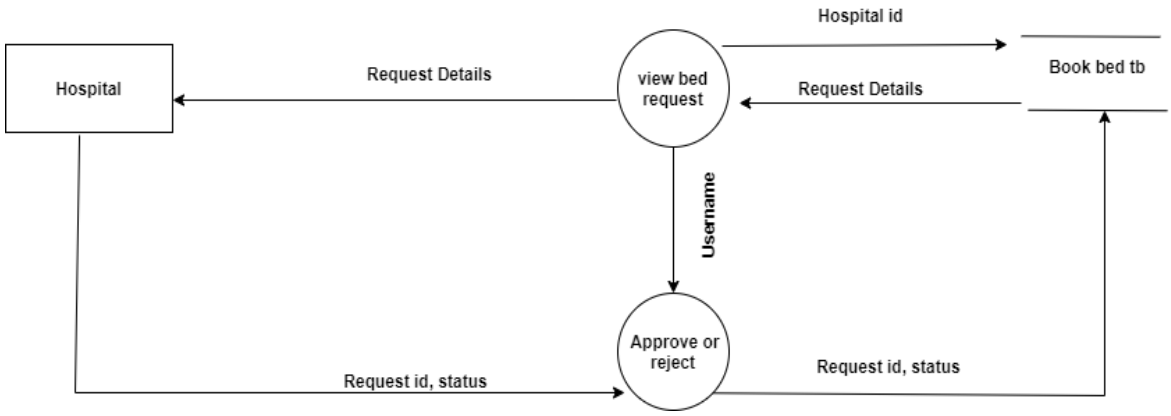
3.3.9 Level-2 DFD for Vendor (View Test Request)



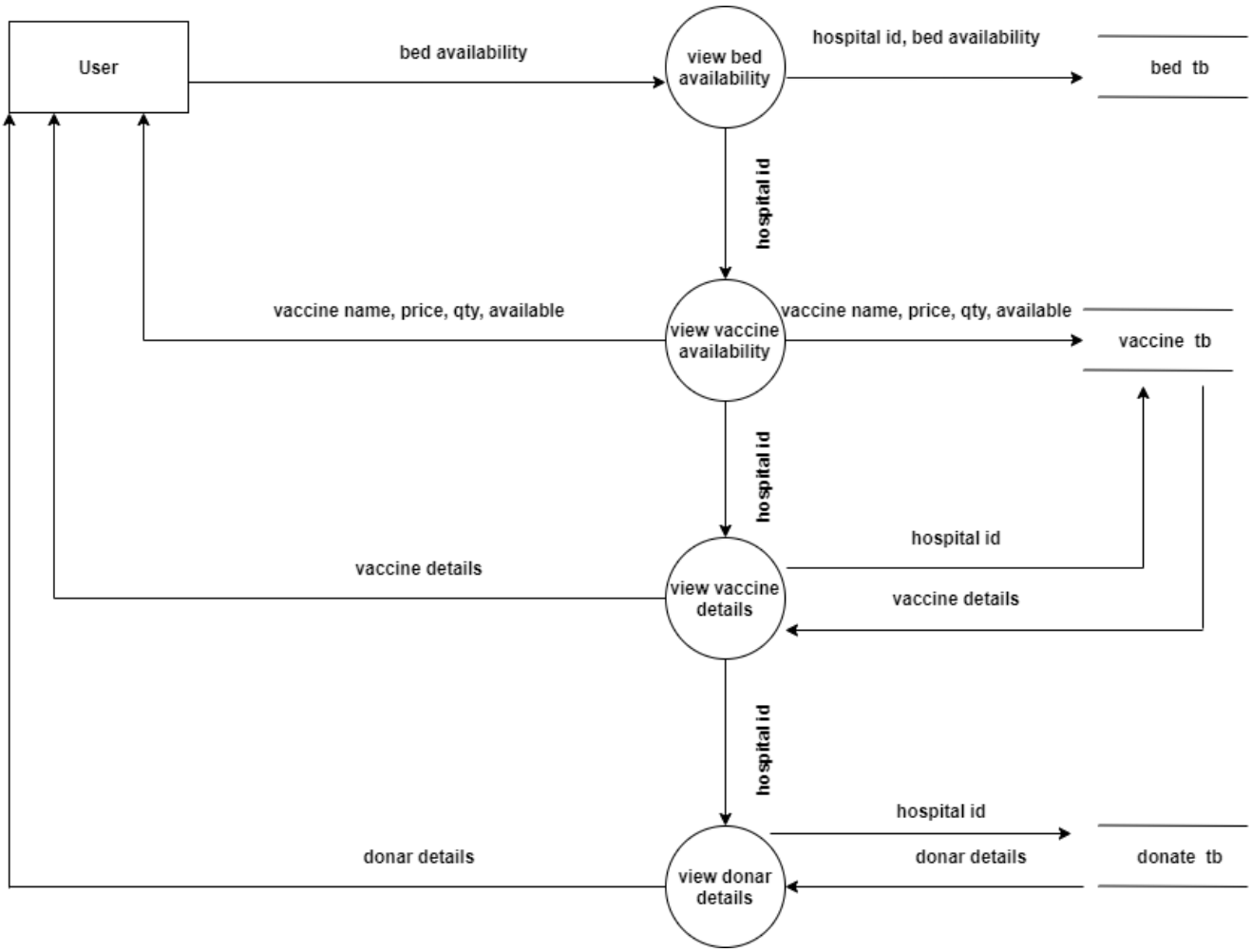
3.3.10 Level-2 DFD for Vendor (View Vaccine Request)



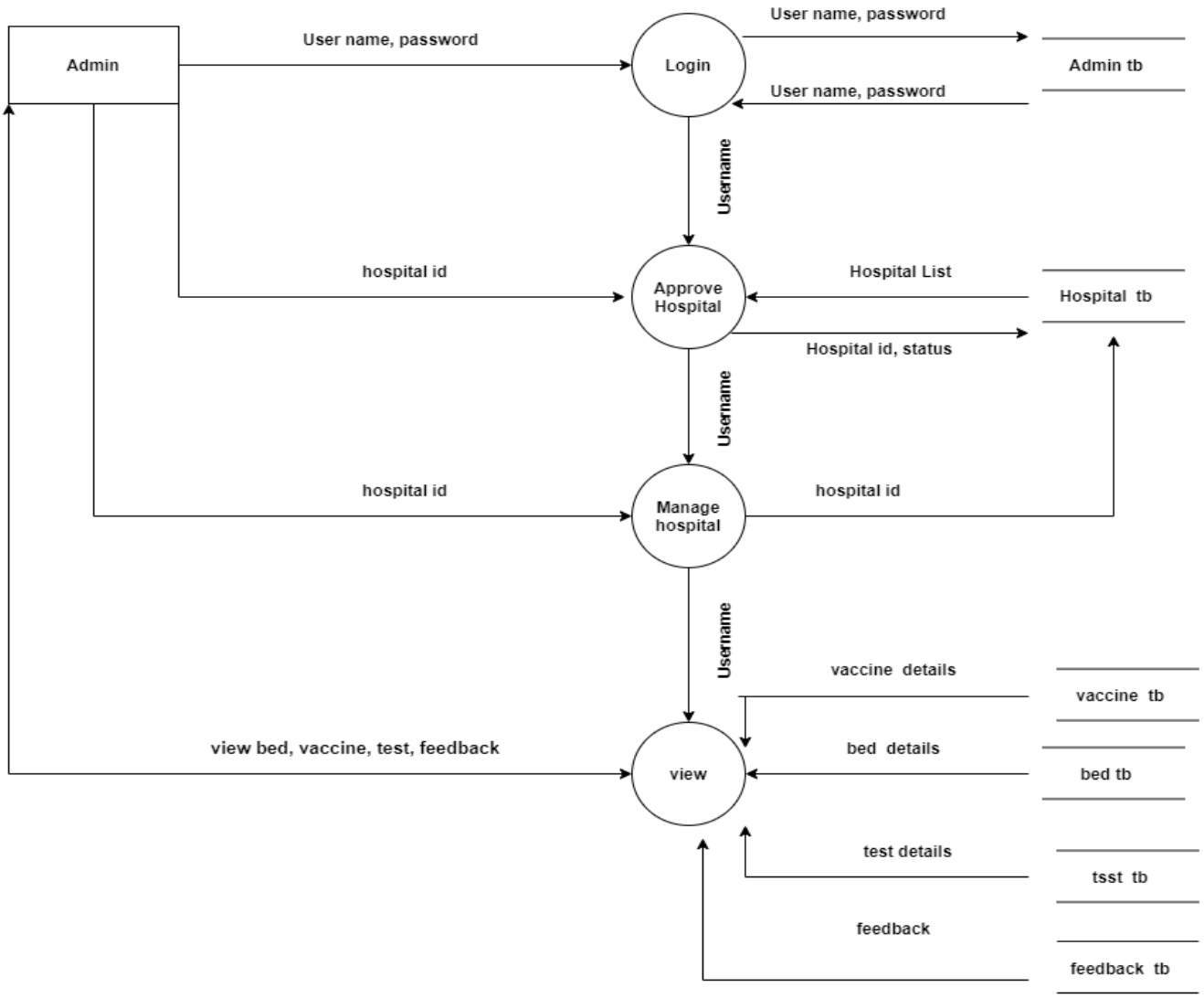
3.3.11 Level-2 DFD for Vendor (View Bed Request)



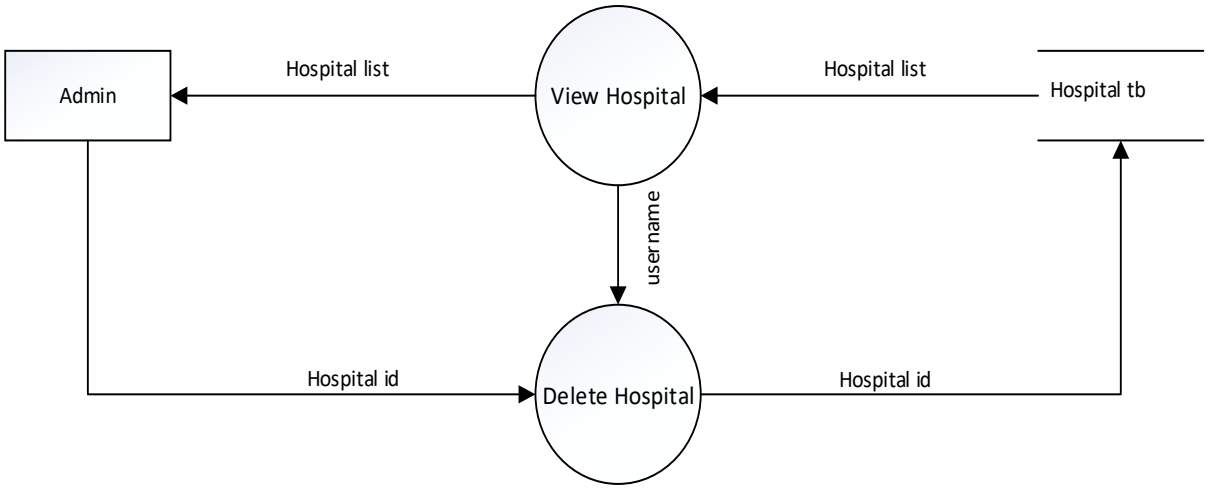
3.3.12 Level-2 DFD for Vendor (Update Results)



3.3.13 Level-1 DFD for Admin



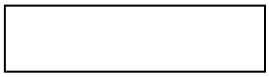
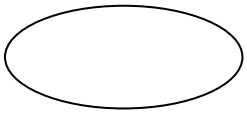
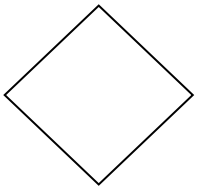
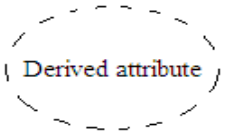
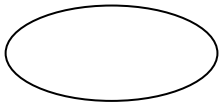
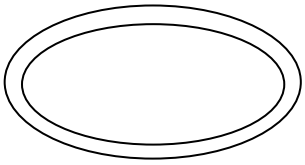
3.3.14 Level-2 DFD for Admin (Manage Hospital)



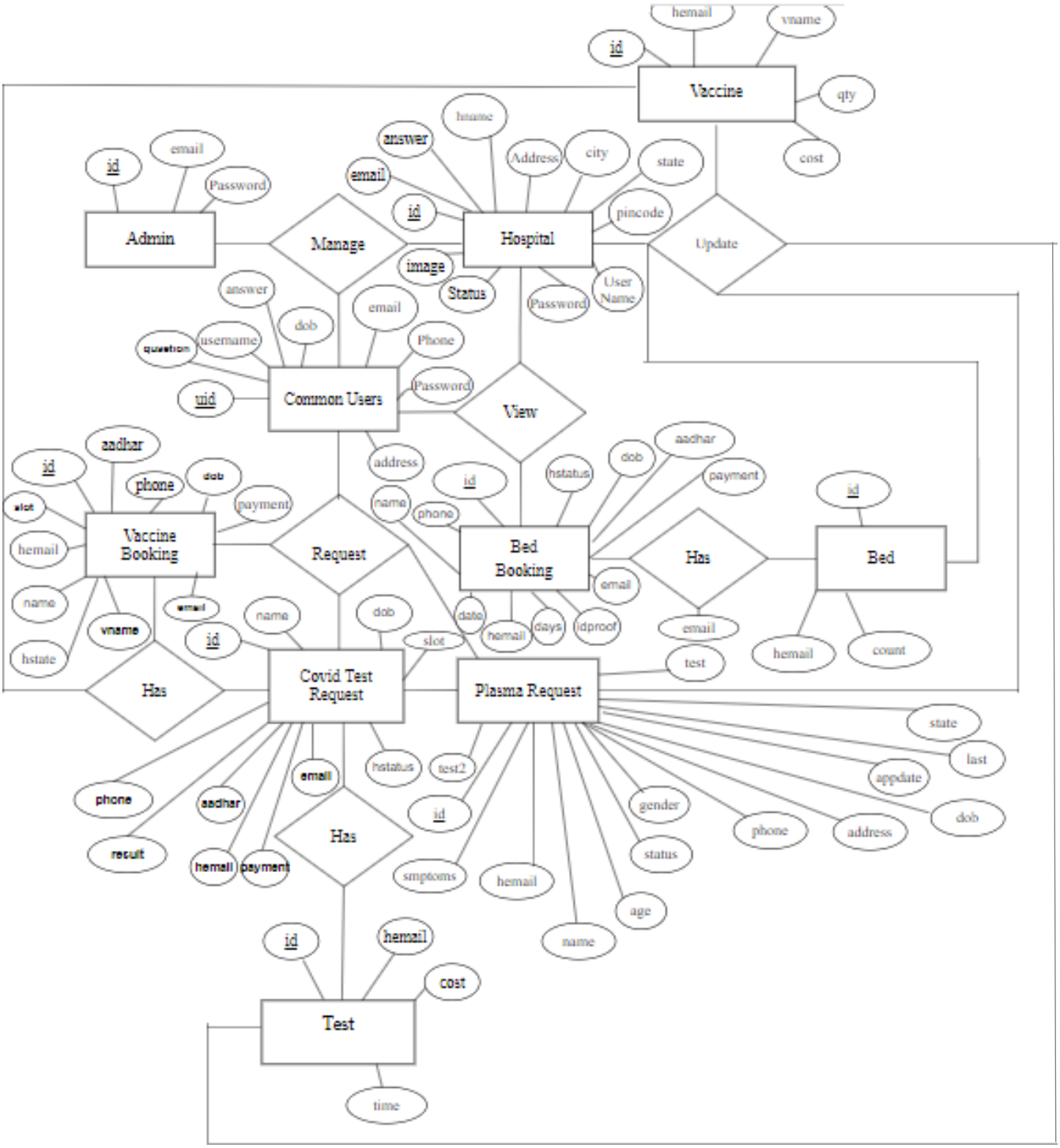
3.4 Entity Relationship Diagram:

- Entity relationship model is a data model for describing a database and it may be defined as a data structure that is recognized by being capable of independent existence entities are named as singular nouns and are shown in rectangles in ER diagram.

ER-Diagram Symbols:

Name	Notation	Description
Entity		It may be an object with the physical existence or conceptual existence. It is represented by a Rectangle.
Attribute		The properties of the entity can be a attribute. It is represented by a Ellipse.
Relationship		Whenever an attribute of one entity refers to another entity, some relationship exists. It is represented by a Diamond.
Link		Lines link attributes to entity sets and entity sets to relation.
Derived Attribute		Dashed ellipse denotes derived attributes.
Key Attribute		An entity type usually has an attribute whose values are distinct for each individual entry in the entity set. It is represented by a Underlined word in ellipse.
Multi valued Attribute		Attributes that have different numbers of values for a particular attribute. It is represented by a Double ellipse represents multi-valued attributes.
Cardinality Ratio	1) 1:1 2) 1:M 3) M:1 4) M:M	It specifies the maximum number of relationships instances that an entity can participate in. There are four cardinality ratios.

3.4.1 ER Diagram



"Database Design"

4.1 Introduction

- Database can be extremely important tools for managing large amount of data.
- A database is a collection of related data, which can be of any size and complexity and an also provide important security.
- By using the concept of Database, we can easily store and retrieve the data.
- The major purpose of a database is to provide the information, which utilizes it with the information that the system needs according to its own requirements.
- The major purpose of a database is to provide the information, which utilizes it with the information's that the system needs according to its own requirements.
- A database administrator is responsible for the maintenance, permission and privileges of database.
- A database consists of a number of interrelated tables each table has a number of records which are used to represents real world objects.
- Each record has a number of fields which are data items used to specify a characteristic of the record.
- A table is a collection of related data held in structure format with in a database.
- It consists columns and row.
- A table has specified number of columns, but can have any number of rows.
- Each row is identified by one or more values appearing in a particular column subset.
- The columns subset which uniquely identifies a row is called the primary key, so by using primary key we can delete or update specified field's values of the row.

4.2 Normalized Table

Table Name: admin

Description:

Column Name	Data Type	Constraints	Descriptions
id	Int(11)	Primary Key	Admin id
email	Varchar(50)	Not Null	Email of the admin
password	Varchar(50)	Not Null	Password of the admin

Table Name: hospital

Description:

Column Name	Data Type	Constraints	Descriptions
hid	Int(11)	Primary Key	Hospital id
hname	Varchar(100)	Not Null	Name of the hospital
address	Varchar(100)	Not Null	Address of the hospital
phone	Varchar(50)	Not Null	Contact number of the hospital
question	Varchar(50)	Not Null	Question
answer	Varchar(50)	Not Null	Answer
username	Varchar(50)	Not Null	Name of the user
password	Varchar(50)	Not Null	Password
status	Varchar(50)	Not Null	Current status
email	Varchar(50)	Not Null	Email of the hospital
city	Varchar(50)	Not Null	City
state	Varchar(50)	Not Null	State
image	Varchar(50)	Not Null	Image of the hospital
pincode	Int(30)	Not Null	Pin code

Table Name: user

Description:

Column Name	Data Type	Constraints	Descriptions
uid	Int(11)	Primary Key	User id
username	Varchar(50)	Not Null	Name of the user
dob	Date	Not Null	Date of birth of the user
phone	Varchar(50)	Not Null	Contact number of the user
email	Varchar(50)	Not Null	Email of the user
Password	Varchar(50)	Not Null	Password
question	Varchar(50)	Not Null	Question
answer	Varchar(50)	Not Null	Answer

Table Name: bed
Description:

Column Name	Data Type	Constraints	Descriptions
id	Int(11)	Primary Key	Bed id
hemail	Varchar(50)	Not Null	Hospital email

count	Int(11)	Not Null	Total number of the bed

Table Name: vaccine

Description:

Column Name	Data Type	Constraints	Descriptions
id	Int(11)	Primary Key	Vaccine id
hemail	Varchar(50)	Not Null	Hospital email
uname	Varchar(100)	Not Null	Name of the user
qty	Int(20)	Not Null	Quantity
cost	Int(20)	Not Null	Price of the vaccine

Table Name: bookvaccine

Description:

Column Name	Data Type	Constraints	Descriptions
id	Int(11)	Primary Key	Vaccine id
name	Varchar(50)	Not Null	Name of the user
aadhar	Varchar(50)	Not Null	Aadhar number of the user
phone	Varchar(50)	Not Null	Phone number of the user
dob	Date	Not Null	Date of birth of the user
hemail	Varchar(50)	Not Null	Email of the hospital
vname	Varchar(50)	Not Null	Name of the vaccine
slot	date	Not Null	Slot available
email	Varchar(50)	Not Null	Email of the user
payment	Varchar(50)	Not Null	Payment
hstatus	Varchar(50)	Not Null	Hospital status

Table Name: test

Description:

Column Name	Data Type	Constraints	Descriptions
id	Int(11)	Primary Key	Test id
hemail	Varchar(50)	Not Null	Email of the hospital
cost	Int(30)	Not Null	Price of the test
time	Varchar(50)	Not Null	Timing of the test

Table Name: bookbed

Description:

Column Name	Data Type	Constraints	Descriptions
id	Int(11)	Primary Key	Bed id
name	Varchar(50)	Not Null	Name of the user
aadhar	Varchar(50)	Not Null	Aadhar number of the user
phone	Varchar(50)	Not Null	Phone number of the user
dob	Date	Not Null	Date of birth of the user
idproof	Varchar(100)	Not Null	Id proof of the user
days	Int(30)	Not Null	Number of days
date	Date	Not Null	Date
hemail	Varchar(50)	Not Null	Email of the hospital
email	Varchar(50)	Not Null	Email of the user
payment	Varchar(50)	Not Null	Payment
hstatus	Varchar(50)	Not Null	Hospital status

Table Name: booktest

Description:

Column Name	Data Type	Constraints	Descriptions
id	Int(11)	Primary Key	Id
name	Varchar(50)	Not Null	Name of the user
aadhar	Varchar(50)	Not Null	Aadhar number of the user
phone	Varchar(50)	Not Null	Phone number of the user
dob	Date	Not Null	Date of birth of the user
slot	Date	Not Null	Slot available
uemail	Varchar(50)	Not Null	Email of the user
hemail	Varchar(50)	Not Null	Email of the hospital
payment	Varchar(50)	Not Null	Payment
hstatus	Varchar(50)	Not Null	Hospital current status
result	Varchar(50)	Not Null	Result of the test

Covid – Shield

Table Name: donate

Description:

Column Name	Data Type	Constraints	Descriptions
id	Int(11)	Primary Key	Id
name	Varchar(50)	Not Null	Name of the donor
dob	Date	Not Null	Date
age	Varchar(50)	Not Null	Age of the donor
gender	Varchar(50)	Not Null	Gender of the donor
phone	Varchar(50)	Not Null	Phone number of the donor
state	Varchar(50)	Not Null	State of the donor
address	Varchar(50)	Not Null	Address of the donor
blood	Varchar(50)	Not Null	Blood group of the donor
test	Varchar(50)	Not Null	Diagnosis test
smptoms	Varchar(50)	Not Null	Symptoms
test2	Varchar(50)	Not Null	Covid-19 test
last	Date	Not Null	Last recorded covid positive result
email	Varchar(50)	Not Null	Email of the donor
hemail	Varchar(50)	Not Null	Email of the hospital
status	Varchar(50)	Not Null	Current status
appdate	Varchar(50)	Not Null	Appointment date

Table Name: feedback

Description:

Column Name	Data Type	Constraints	Descriptions
id1	Int(11)	Primary Key	Feedback id
subject	Varchar(50)	Not Null	Subject
message	Varchar(100)	Not Null	Message
email	Varchar(50)	Not Null	Email of the user

"Detailed Design"

5.1 Introduction:

- Detailed design is the second level of the design process.
- Detailed design starts after the system design phase and the system has been certified through the review.
- The goal of this phase is to develop the internal logic of each of the modules identified during the system design.
- In the system design, the focus is on identifying the modules, whereas during detailed design the focus is on designing the logic for the modules.
- In other words, in system design attention is on what components are needed, while in the detailed design emphasis is on is on the issue of how the various components are implemented in the software.
- During detailed design, we specify how the module in the system interacts with each other and the internal logic of each of the modules specified during system design is decided.
- Detailed design essentially expands the system design and database design to contain a more detailed description of the processing logic and data structures so that the design is sufficiently complete for coding.
- The design process for software system has two levels.
- At the first level focus is on deciding which modules are needed for the system, the specification of these modules and how the modules should be interconnected.
- This is called system design or top-level design, in the specification of the module can be satisfied is decided.
- This design level is often called detailed design or logic design.
- Because the detailed design is extension of system design, system design controls the major structural characteristics of the system.

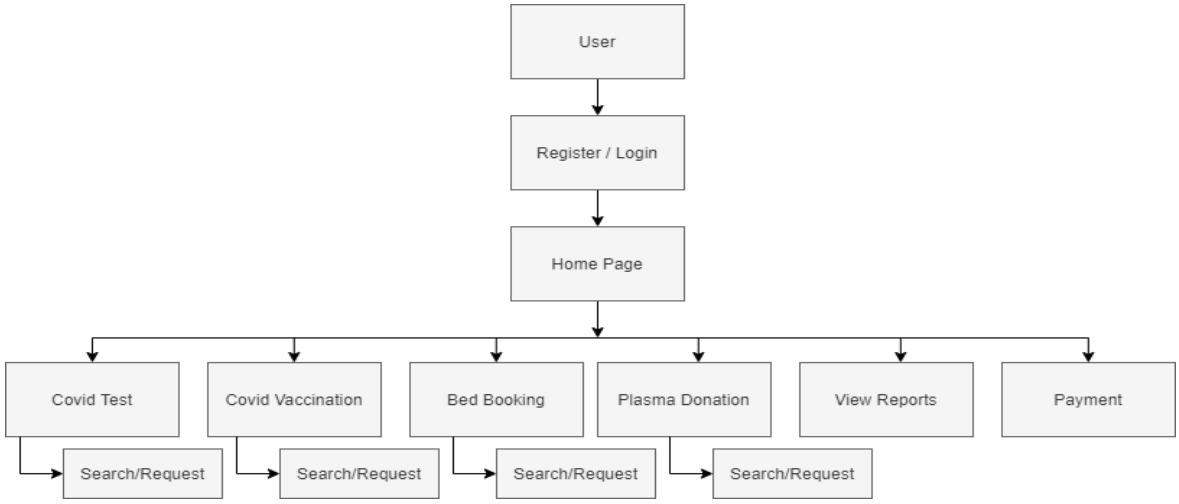
5.2 Applicable Documents:

- Synopsis document for "**Covid-Shield".**
- Software Requirement Specification (SRS) for "**Covid-Shield".**
- Database Design document for "**Covid-Shield".**

5.3 Structure of the software program:

- Functional component 1: User Module
- Functional component 2: Vendor Module
- Functional component 3: Admin Module

5.3.1 Structure chart for User:



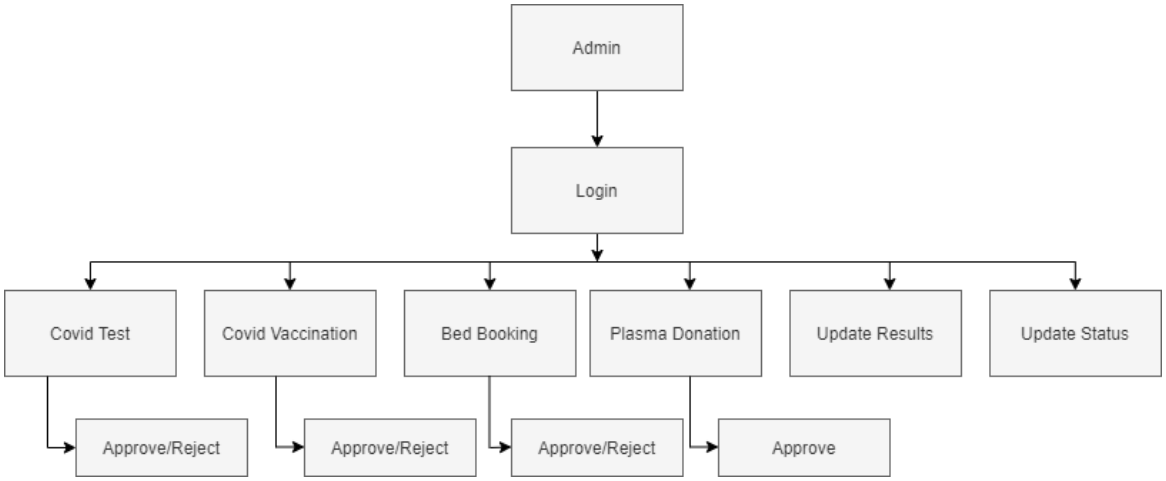
Modules of User:

- Register and Login
- Search and Request Hospitals for Covid-19 Test
- Search and Request Hospitals for Covid-19 Vaccination
- Search and Request Hospitals for Bed booking
- Search and Request Hospitals for Plasma Donation

Identification module description:

- **Register and Login**
  - **Input:** username, dob, phone, email, question, answer, password.
  - **Output:** User can register and login into the application.
- **Search and Request Hospitals for Covid-19 Test**
  - **Input:** name, aadhar, phone, dob, slot, uemail, hemail.
  - **Output:** User can search and request to the hospital for the Covid-19 test.
- **Search and Request Hospitals for Covid-19 Vaccination**
  - **Input:** name, aadhar, phone, dob, hemail, vname, slot, email.
  - **Output:** User can search and request to the hospital for the Covid-19 vaccination.
- **Search and Request Hospitals for Bed booking**
  - **Input:** name, phone, aadhar, dob, email, idproof, days, date, hemail.
  - **Output:** User can search and request to the hospital for the bed booking.
- **Search and Request Hospitals for Plasma Donation**
  - **Input:** name, dob, age, gender, phone, state, address, blood, test, smptoms, test2, last, email, hemail.
  - **Output:** User can search and request to the hospital for the plasma donation.

5.3.2 Structure chart for Vendor:



Modules of Vendor :

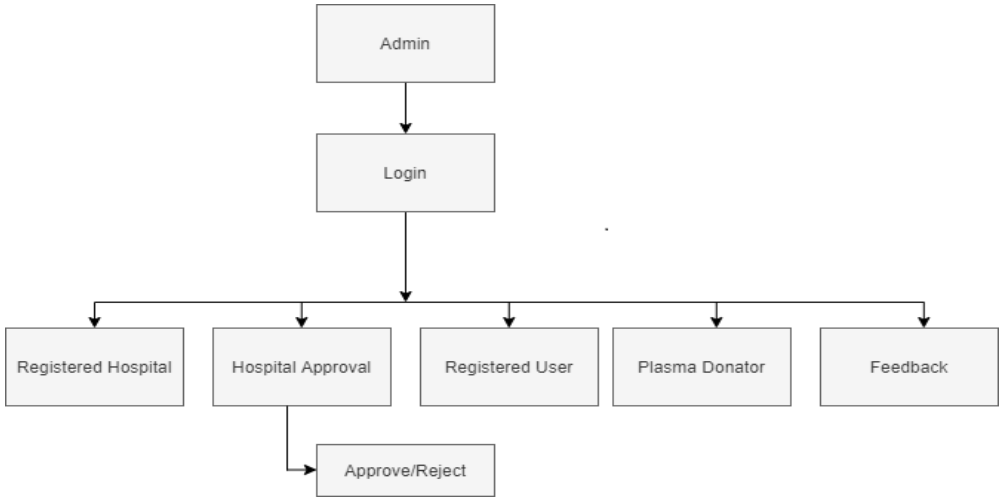
- Register and Login
- Request from patient (Covid-19 Test)
- Request from patient (Covid-19 Vaccination)
- Request from patient (Bed Booking)
- Request from patient (Plasma Donation)
- Update Covid-19 test result.
- Update Vaccination Status.

Identification module description:

- **Register and Login**
  - **Input:** hname, question, answer, email, phone, address, city, state, pincode, username, password.
  - **Output:** Vendor can register and login into the application.
- **Request from patient (Covid-19 Test)**
  - **Input:** hemail, cost, time.
  - **Output:** Vendor can view request and will approve / reject for test from the user side.
- **Request from patient (Covid-19 Vaccination)**
  - **Input:** hemail, vname, qty, cost.
  - **Output:** Vendor can view request and will approve / reject for vaccination from the user side.
- **Request from patient (Bed Booking)**
  - **Input:** id, hemail, count.
  - **Output:** Vendor can view request and will approve / reject for bed booking from the user side.
- **Request from patient (Plasma Donation)**
  - **Input:** name, dob, age, gender, phone, state, address, blood, test, smptoms, test2, last.
  - **Output:** Vendor can view request and will have to approve for plasma donation from the user side.

- **Update Covid-19 test result**
  - **Input:** hemail, cost, time.
  - **Output:** Vendor will be update the patient's Covid-19 results.
- **Update Vaccination Status**
  - **Input:** hemail, vname, qty, cost.
  - **Output:** Vendor will be update the patient's Covid-19 vaccination status.

5.3.3 Structure chart for Admin:



Modules of Admin :

- Login.

Identification module description:

- **Login**
  - **Input:** email, password.
  - **Output:** Admin can login and manage the hospitals.

"Coding and Implementation"

6.1 Introduction

- All the software engineering steps that have been discussed so far are directed towards a final objective of translating the modules into the software form that could be understood by the computer.
- Coding is an important aspect in the system development and proper care has been taken to ensure that the application is efficient and suits requirements.
- Coding is viewed as a natural consequence of design.

6.2 Development Guideline

The coding is done with the following characteristics:

- Ease of design to code translation.
- Code efficiency.
- Memory efficiency.
- Response time.
- Maintainability.
- Security.
- Simple and easy to understand code.
- Consistent logic.

6.3 Coding

- The coding standards are necessary for a system because, people other than programmers can easily understand the orderly convention of the system.
- Standards facilitate easy maintenance of the system, as someone else other than the actual developers, can later modify or upgrade system.
- The naming convention used here is similar to that of window classes.

6.4 Code Review and walk-through

6.4.1 Code Review

- The goal of code review is to translate the design of the system into code in the given programming language.
- The code is written in java and xml is used to improve layout rendering.
- Well written code reduces the test and maintenance effort since the testing and maintenance costs of software are much higher than the coding costs, the goal of coding should be to reduce this effort.

6.4.2Walk-through

- The purpose of a systematic walk through is to evaluate a software project.
- A walk-through may be held for the purpose of educating an audience regarding a software product.
- The major objectives are to:
  - Find anomalies.
  - Improve Software products.
  - Consider alternate implementation.
  - Evaluate conformance to standards and specification.
- Other important objectives of walk-through include exchange of technique, style variation and training of the participants.
- The guideline is given to the user to use the application and inputs are entered a standard format.

6.5 Module Coding

User:

Config File:

```
<?php
$host = "localhost";    /* Host name */
$user = "root";         /* User */
$password = "";         /* Password */
$dbname = "covidshield";   /* Database name */
// Create connection
$con = mysqli_connect($host, $user, $password,$dbname);
// Check connection
if (!$con) {
die("Connection failed: " . mysqli_connect_error());
}
```

Cbed:

```
<?php
if(isset($_POST['save']))
{
error_reporting(1);
include("config.php");
$name=$_POST['name'];
```

```
$aadhar=$_POST['aadhar'];
$phone=$_POST['phone'];
$dob=$_POST['dob'];
$days=$_POST['days'];
$date=$_POST['date'];
$fname = $_FILES["image"]["name"];
$filename=$name.$fname;
$tempname = $_FILES["image"]["tmp_name"];
$folder = "uploads/".$filename;
if (move_uploaded_file($tempname, $folder))
{
$sql1 = "INSERT INTO `bookbed`(`name`, `phone`, `aadhar`, `dob`, `email`, `idproof`, `days`, `date`,
`hemail`) VALUES
('".$name."','".$phone."','".$aadhar."','".$dob."','".$_SESSION['user']."','".$filename."','".$days."','".$date."
','".$hemail."')";
if(mysqli_query($con,$sql1))
{
echo "<script>
alert('Info Saved');
</script>";
echo "<script> location.href='services.php'; </script>";
}
else
{
echo "<script>
alert('Update Failed');
</script>";
echo "<script> location.href='services.php'; </script>";
}
}
else
{
echo "<script>
alert('Upload Failed');
</script>";
}
}
```

```
?>
```

Ctest:

```
<?php
if(isset($_POST['save']))
{
error_reporting(1);
include("config.php");
$aadhar=$_POST['aadhar'];
$sql = "select * from booktest where aadhar='$aadhar'";
$result = mysqli_query($con,$sql);
$count=mysqlI_num_rows($result);
if($count>0)
{
echo "<script>
alert('There is an existing account associated with this aadhar number.');
</script>";
echo "<script> location.href='services.php'; </script>";
}
else
{
$name=$_POST['name'];
$aadhar=$_POST['aadhar'];
$phone=$_POST['phone'];
$dob=$_POST['dob'];
$slot=$_POST['slot'];
$sql1 = "INSERT INTO `booktest`(`name`, `aadhar`, `phone`, `dob`, `slot`, `uemail`, `hemail`) VALUES
('".$name."','".$aadhar."','".$phone."','".$dob."','".$slot."','".$_SESSION['user']."','".$hemail."')";
if(mysqli_query($con,$sql1))
{
echo "<script>
alert('Info Saved');
</script>";
echo "<script> location.href='services.php'; </script>";
}
else
```

```
{
echo "<script>
alert('Update Failed');
</script>";
echo "<script> location.href='services.php'; </script>";
}
}
}
?>
```

Feedback:

```
<?php
if(isset($_POST['send']))
{
$sub=$_POST['sub'];
$msg=$_POST['msg'];
$mail=$_SESSION['user'];
include("config.php");
$sql1 = "INSERT INTO `feedback`(`subject`, `message`, `email`) VALUES
('".$sub."','".$msg."','".$mail."')";
if(mysqli_query($con,$sql1))
{
echo "<script>
alert('Feedback sent');
</script>";
echo "<script> location.href='feedback.php'; </script>";
}
}
?>
```

Forgot:

```
<?php
if(isset($_POST['submit']))
{
 include('config.php');
 $email=$_POST['email'];
```

```
 $question=$_POST['question'];
 $answer=$_POST['answer'];
 $password=$_POST['password'];
$sql="update user set password='".$password."' where email='".$email."' and question='".$question."'
and answer='".$answer."'";
 mysqli_query($con, $sql);
 $count=mysqli_affected_rows($con);
 echo $count;
 if($count>0)
{
echo "<script>
alert('Password has been reset');
</script>";
echo "<script> location.href='index.php'; </script>";
}
else
{
echo "<script>
alert('invalid information');
</script>";
echo "<script> location.href='index.php'; </script>";
}
}
?>
```

Plasma:

```
<?php
include 'config.php';
$sql1 = "select * from hospital";
$result1 = mysqli_query($con,$sql1);
$num1=mysqlI_num_rows($result1);
$sl=0;
if($num1 > 0)
{
while($row1 = mysqli_fetch_array($result1))
{
```

```
$name=$row1[1];
$address=$row1[6];
$city=$row1[7];
$state=$row1[8];
$pin=$row1[9];
$phone=$row1['phone'];
$hemail=$row1[4];
$image=$row1['image'];
?>
<div class="col-lg-3 col-md-6 d-flex mb-sm-4 ftco-animate">
<div class="staff">
<div class="img mb-4" style="background-image: url(../Hospital/uploads/<?php echo $image;
?>);"></div>
<div class="info text-center">
<h3><?php echo $name; ?></h3>
<div class="text">
<p>Address: <?php echo $address; echo ', '; echo $city; echo', '; echo $state; echo', '; echo $pin; ?></p>
<p>Phone:<?php echo $phone; ?></p>
<a href="plasma.php?hid=<?php echo $hemail; ?>" class="py-2 d-block">Donate Here</a>
</div>
</div>
</div>
</div>
<?php }}?>
```

Index:

```
<?php
if(isset($_POST['register']))
{
error_reporting(1);
include("config.php");
$email=$_POST['email'];
$sql = "select * from register where email='$email'";
$result = mysqli_query($con,$sql);
$count=mysqlI_num_rows($result);
if($count>0)
```

```
{
echo "<script>
alert('There is an existing account associated with this email.');
</script>";
echo "<script> location.href='index.php'; </script>";
}
else
{
 $username=$_POST['username'];
$dob=$_POST['dob'];
$phone=$_POST['phone'];
$email=$_POST['email'];
$question=$_POST['question'];
$answer=$_POST['answer'];
$password=$_POST['password'];
$cpassword=$_POST['cpassword'];
if($password==$cpassword)
{
$query= "INSERT INTO `user`(`username`, `dob`, `phone`, `email`, `password`, `question`,      `answer`)
VALUES
('".$username."','".$dob."','".$phone."','".$email."','".$password."','".$question."','".$answer."')";
mysqli_query($con,$query) or die(mysqli_error($con));
echo "<script>
alert('Registeration Completed, Please Login.');
</script>";
echo "<script> location.href='index.php'; </script>";
}
else
{
 echo "<script>
 alert('Passwords dont match');
 </script>";
 echo "<script> location.href='register.php'; </script>";
 }
 }
 }
 ?>
```

Payment

```
<?php
 if(isset($_POST['submit']))
 {
 include("config.php");
 if($type="test")
 {
  $query= "update booktest set payment='paid' where `id`='".$id."'";
  mysqli_query($con,$query) or die(mysqli_error($con));
  echo "<script>
  alert('Payment Successful');
  </script>";
 echo "<script> location.href='home.php'; </script>";
 }
 if($type="bed")
{
 $query= "update bookbed set payment='paid' where `id`='".$id."'";
 mysqli_query($con,$query) or die(mysqli_error($con));
 echo "<script>
 alert('Payment Successful');
</script>";
echo "<script> location.href='home.php'; </script>";
}
if($type="vaccine")
 {
 $query= "update bookvaccine set payment='paid' where `id`='".$id."'";
 mysqli_query($con,$query) or die(mysqli_error($con));
 echo "<script>
 alert('Payment Successful');
 </script>";
 echo "<script> location.href='home.php'; </script>";
 }
 echo "<script>
 alert('Payment Module in Development');
 </script>";
 echo "<script> location.href='not.php'; </script>";
```

```
}
?>
```

Plasma Donation:

```
<?php
if(isset($_POST['save']))
{
error_reporting(1);
include("config.php");
$name=$_POST['name'];
$dob=$_POST['dob'];
$age=$_POST['age'];
$gender=$_POST['gender'];
$phone=$_POST['phone'];
$state=$_POST['state'];
$ad=$_POST['ad'];
$bg=$_POST['bg'];
$test=$_POST['test'];
$symp=$_POST['symp'];
$rep=$_POST['rep'];
$last=$_POST['last'];
$sql1 = "INSERT INTO `donate`(`name`, `dob`, `age`, `gender`, `phone`, `state`, `address`, `blood`,
`test`, `smptoms`, `test2`, `last`, `email`, `hemail`) VALUES
('".$name."','".$dob."','".$age."','".$gender."','".$phone."','".$state."','".$ad."','".$bg."','".$test."','".$symp."
','".$rep."','".$last."','".$_SESSION['user']."','".$hemail."')";
if(mysqli_query($con,$sql1))
{
echo "<script>
alert('Info Saved');
</script>";
echo "<script> location.href='services.php'; </script>";
}
else
{
echo "<script>
alert('Update Failed');
</script>";
echo "<script> location.href='hplasma.php'; </script>";
```

```
}
}
?>
```

Vaccine:

```
<?php
include("config.php");
$sql = "select hname,email from hospital";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
if($count>0)
{
while($row = mysqli_fetch_array($result))
{
$hname=$row[0];
$hmail=$row[1];
?>
<option value="<?php echo $hmail; ?>"><?php echo $hname; ?></option>
<?php }}?>
 <?php
include("config.php");
$sql = "select hospital.hname, vaccine.vname, vaccine.qty, vaccine.cost from hospital inner join
vaccine on vaccine.hemail=hospital.email";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
if($count>0)
{
while($row = mysqli_fetch_array($result))
{
 $hname=$row[0];
 $vname=$row[1];
 $qty=$row[2];
 $cost=$row[3];
  ?>
 <tr>
 <td><?php echo $hname ?></td>
```

```
 <td><?php echo $vname ?></td>
 <td><?php echo $qty ?></td>
 <td><?php echo $cost ?></td>
 </tr>
<?php }}?>
<?php
if(isset($_POST['save']))
{
error_reporting(1);
include("config.php");
$aadhar=$_POST['aadhar'];

  $sql = "select * from bookvaccine where aadhar='$aadhar'";
  $result = mysqli_query($con,$sql);
  $count=mysqlI_num_rows($result);

  if($count>0)
  {
   echo "<script>
   alert('There is an existing account associated with this aadhar number.');
  </script>";
   echo "<script> location.href='services.php'; </script>";
  }
$name=$_POST['name'];
$aadhar=$_POST['aadhar'];
$phone=$_POST['phone'];
$dob=$_POST['dob'];
$hospital=$_POST['hospital'];
$vaccine=$_POST['vaccine'];
$slot=$_POST['slot'];
$sql2 = "select * from vaccine where hemail='".$hospital."' and vname='".$vaccine."' and qty>0";
$result2 = mysqli_query($con,$sql2);
$count2 =mysqlI_num_rows($result2);
if($count2>0)
{
$sql0 = "select * from bookvaccine where aadhar='".$aadhar."'";
$result0 = mysqli_query($con,$sql0);
```

```
$count0 =mysqlI_num_rows($result0);
if($count0>0)
{
echo "<script>
alert('The User with this Aadhar No has already booked a Vaccine');
</script>";
}
else
{
$sql1 = "INSERT INTO `bookvaccine`(`name`, `aadhar`, `phone`, `dob`, `hemail`, `vname`,
`slot`,`email`) VALUES
('".$name."','".$aadhar."','".$phone."','".$dob."','".$hospital."','".$vaccine."','".$slot."','".$_SESSION['us
er']."')";
if(mysqli_query($con,$sql1))
{
$sql3 = "Update vaccine set qty=qty-1 where hemail='".$hospital."' and vname='".$vaccine."'";
if (mysqli_query($con,$sql3))
{
echo "<script>
alert('Info Saved');
</script>";
echo "<script> location.href='services.php'; </script>";
}
}
else
{
echo "<script>
alert('Update Failed');
</script>";
echo "<script> location.href='vaccine.php'; </script>";
}
}
}
else
{
echo "<script>
alert('Vaccine not availale $vaccine $hospital');
```

```
</script>";
echo "<script> location.href='vaccine.php'; </script>";
}
}
?>
```

Notification:

```
<?php
include("config.php");
$sql = "select booktest.*, hospital.hname, test.cost from booktest inner join hospital on
hospital.email=booktest.hemail inner join test on hospital.email=test.hemail where
booktest.uemail='".$_SESSION['user']."'";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);

if($count>0)
{
while($row = mysqli_fetch_array($result))
{
$id=$row[0];
$name=$row[1];
$aadhar=$row[2];
$phone=$row[3];
$dob=$row[4];
$slot=$row[5];
$email=$row[6];
$hemail=$row[7];
$payment=$row[8];
$report=$row[10];
$hname=$row['hname'];
$cost=$row['cost'];
?>
<tr>
<td><?php echo $hname; ?></td>
<td><?php echo $name; ?></td>
<td><?php echo $aadhar; ?></td>
<td><?php echo $phone; ?></td>
<td><?php echo $dob; ?></td>
```

```
<td><?php echo $slot; ?></td>
<td><?php echo $email; ?></td>
<td><?php echo $cost; ?></td>
<td><?php echo $payment; ?></td>
<td><?php echo $report; ?></td>
<?php
if($report=='Positive')
{ ?>
<td><center> <a href="images/pos.png" download="Positive">Download
Certificate</a></center></td>
<?php
}
else if($report=='Negative')
{ ?>
<td> <center><a href="images/neg.png" download="Negative">Download
Certificate</a></center></td>
<?php
}
?>
<td><?php if($payment=='pending'){?><a href="payment.php?id=<?php echo $id;
?>&type=test&cost=<?php echo $cost ?>"><input type="submit" name="save" value="Pay" class="btn
btn-primary py-3 px-5"></a><?php }?></td>
</tr>
<?php }}?>
</table>
```

Hospital:

Bed:

```
<?php
include("config.php");
$sql1 = "select bookbed.*, hospital.hname from bookbed inner join hospital on
hospital.email=bookbed.hemail where bookbed.hemail='".$_SESSION['huser']."'";
$result1 = mysqli_query($con,$sql1);
$count1 =mysqlI_num_rows($result1);
if($count1>0)
{
```

```
$sl=0;
while($row1 = mysqli_fetch_array($result1))
{
$sl+=1;
$id=$row1[0];
$name=$row1[1];
$phone=$row1[2];
$aadhar=$row1[3];
$dob=$row1[4];
$email=$row1[5];
$img=$row1[6];
$days=$row1[7];
$date=$row1[8];
$hemail=$row1[9];
$payment=$row1[10];
$status=$row1[11];
 ?>
 <tr>
 <td><?php echo $sl; ?></td>
 <td><?php echo $name; ?></td>
 <td><?php echo $phone; ?></td>
 <td><?php echo $aadhar; ?></td>
 <td><?php echo $dob; ?></td>
 <td><?php echo $email; ?></td>
 <td><img src="../user/uploads/<?php echo $img; ?>" style="height:80px;"></td>
<td><?php echo $days; ?></td>
<td><?php echo $date; ?></td>
<td><?php echo $payment; ?></td>
<td><?php echo $status; ?></td>
<td> <a href="bed.php?uid=<?php echo $id; ?>"><input type="submit" value="Confirm"
style="background-color: blue;color: white;border-color: white"></a></td>
</tr>
<?php }}?>
</table>
<?php
if(isset($_GET['uid']))
{
```

```
$uid=$_GET['uid'];
include("config.php");
$sql1 = "update bookbed set hstatus='Confirmed' where id='".$uid."'";
$result1 = mysqli_query($con,$sql1);
echo "<script> location.href='bed.php'; </script>";
}
 ?>
```

Bed Update:

```
<?php
include("config.php");
$sql = "select count from bed where hemail='".$_SESSION['huser']."'";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
if($count>0)
{
while($row = mysqli_fetch_array($result))
{
$bedc=$row[0];
}
}
else
{
$bedc='';
}
?>
<?php
if(isset($_POST['update']))
{
error_reporting(1);
include("config.php");
$bedcount=$_POST['count'];
$sql = "select count from bed where hemail='".$_SESSION['huser']."'";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
if($count>0)
```

```
{
$sql2 = "update bed set count='".$bedcount."' where hemail='".$_SESSION['huser']."'";
if(mysqli_query($con,$sql2))
{
echo "<script>
alert('Info Updated');
</script>";
echo "<script> location.href='bedupdate.php'; </script>";
}
}
else
{
$sql1 = "insert into bed(hemail, count) values('".$_SESSION['huser']."','$bedcount')";
if(mysqli_query($con,$sql1))
{
echo "<script>
alert('Info Updated');
</script>";
echo "<script> location.href='bedupdate.php'; </script>";
}
else
{
echo "<script>
alert('Update Failed');
</script>";
echo "<script> location.href='bedupdate.php'; </script>";
}
}
}
?>
```

Index:

```
<?php
if(isset($_POST['login']))
{
error_reporting(1);
include("config.php");
```

```
$hospital=$_POST['username'];
$email=$_POST['email'];
$password=$_POST['password'];
$sql = "select status from hospital where email='$email' and password='$password' and
hname='$hospital'";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
$approve="approve";
$reject="reject";
if($count>0)
{
while($row = mysqli_fetch_array($result))
{
$status=$row[0];
}
if($status==$approve)
{
$_SESSION['huser']=$email;
echo "<script>
alert('Login Successful');
</script>";
echo "<script> location.href='home.php'; </script>";
}
else if($status==$reject)
{
echo "<script>
alert('Your Registration is Rejected');
</script>";
echo "<script> location.href='index.php'; </script>";
}
else
{
echo "<script>
alert('Your Approval is Pending, Please Wait $status');
</script>";
echo "<script> location.href='index.php'; </script>";
}
```

```
}
else
{
echo "<script>
alert('Invalid Email or Password');
</script>";
echo "<script> location.href='index.php'; </script>";
}
}
?>
```

Plasma:

```
<?php
include("config.php");
$sql4 = "select * from donate where hemail='".$_SESSION['huser']."'";
$result4 = mysqli_query($con,$sql4);
$count4 =mysqlI_num_rows($result4);
if($count4>0)
{
while($row4 = mysqli_fetch_array($result4))
{
$id=$row4[0];
$name=$row4[1];
$dob=$row4[2];
$age=$row4[3];
$gender=$row4[4];
$phone=$row4[5];
$state=$row4[6];
$address=$row4[7];
$blood=$row4[8];
$test=$row4[9];
$symp=$row4[10];
$rep=$row4[11];
$last=$row4[12];
$email=$row4[13];
$hemail=$row4[14];
$status=$row4[15];
```

```
$appdate=$row4[16];
?>
 <tr>
 <td><?php echo $name; ?></td>
 <td><?php echo $dob; ?></td>
 <td><?php echo $age; ?></td>
 <td><?php echo $gender; ?></td>
 <td><?php echo $email; ?></td>
 <td><?php echo $phone; ?></td>
 <td><?php echo $state; ?></td>
 <td><?php echo $address; ?></td>
 <td><?php echo $blood; ?></td>
 <td><?php echo $test; ?></td>
 <td><?php echo $symp; ?></td>
 <td><?php echo $rep; ?></td>
 <td><?php echo $last; ?></td>
 <td><?php echo $status; ?></td>
 <td><?php echo $appdate; ?></td>
  <td><?php if($status=='pending'){ ?><a href="plasmadet.php?uid=<?php echo $id; ?>"><input
  type="submit" value="Accept/Reject" style="background-color: blue;color: white;border-color:
  white"></a><?php } ?></td>
 </tr>
<?php }} ?>
```

Plasmadet:

```
<?php session_start();
$uid=$_GET['uid'];
?>
<?php
include 'config.php';
$sql1 = "select name from donate where id='".$uid."'";
$result1 = mysqli_query($con,$sql1);
$num1=mysqlI_num_rows($result1);
$sl=0;
if($num1 > 0)
{
```

```
while($row1 = mysqli_fetch_array($result1))
{
$name=$row1[0];
}}?>
<?php echo $name; ?></td>
<?php
if(isset($_POST['save']))
{
$status=$_POST['status'];
if($status=='Approved')
{
$apdate=$_POST['appdate'];
}
else
{
$apdate='';
}
include("config.php");
$sql1 = "update donate set status='$status', appdate='$apdate' where id='".$uid."'";
$result1 = mysqli_query($con,$sql1);
echo "<script> location.href='plasma.php'; </script>";
}
?>
```

Register:

```
<?php
if(isset($_POST['register']))
{
error_reporting(1);
include("config.php");
$email=$_POST['email'];
$sql = "select * from register where email='$email'";
$result = mysqli_query($con,$sql);
$count=mysqlI_num_rows($result);
if($count>0)
{
```

```
echo "<script>
alert('There is an existing account associated with this email.');
</script>";
echo "<script> location.href='index.php'; </script>";
}
else
{
$hospital=$_POST['hospital'];
$question=$_POST['question'];
$answer=$_POST['answer'];
$email=$_POST['email'];
$phone=$_POST['phone'];
$address=$_POST['address'];
$city=$_POST['city'];
$state=$_POST['state'];
$pin=$_POST['pin'];
$username=$_POST['username'];
$password=$_POST['password'];
$cpassword=$_POST['cpassword'];
$fname = $_FILES["image"]["name"];
$tempname = $_FILES["image"]["tmp_name"];
$folder = "uploads/".$fname;
if($password==$cpassword)
{
if (move_uploaded_file($tempname, $folder))  {
$query= "INSERT INTO `hospital`(`hname`,`question`, `answer`, `email`, `phone`, `address`, `city`,
`state`, `pincode`, `username`, `password`, `image`) VALUES
('".$hospital."','".$question."','".$answer."','".$email."','".$phone."','".$address."','".$city."','".$state."','".$pi
n."','".$username."','".$password."','".$fname."')";
mysqli_query($con,$query) or die(mysqli_error($con));
echo "<script>
alert('Registeration Completed, Please Login.');
</script>";
echo "<script> location.href='index.php'; </script>";
}
else
{
```

```
echo "<script>
alert('Upload small sized image');
</script>";
echo "<script> location.href='register.php'; </script>";
}
}
else
{
echo "<script>
alert('Passwords dont match');
</script>";
echo "<script> location.href='register.php'; </script>";
}
}
}
?>
```

Test:

```
<?php session_start(); ?>
<?php
include("config.php");
$sql = "select booktest.*, hospital.hname from booktest inner join hospital on
hospital.email=booktest.hemail where booktest.hemail='".$_SESSION['huser']."'";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
if($count>0)
{
$sl=0;
while($row = mysqli_fetch_array($result))
{
$id=$row[0];
$name=$row[1];
$aadhar=$row[2];
$phone=$row[3];
$dob=$row[4];
$slot=$row[5];
$email=$row[6];
```

```
$hemail=$row[7];
$payment=$row[8];
$status=$row[9];
$hname=$row['hname'];
$sl+=1;
?>
<tr>
<td><?php echo $sl; ?></td>
<td><?php echo $name; ?></td>
<td><?php echo $aadhar; ?></td>
<td><?php echo $phone; ?></td>
<td><?php echo $dob; ?></td>
<td><?php echo $slot; ?></td>
<td><?php echo $email; ?></td>
<td><?php echo $payment; ?></td>
<td><?php echo $status; ?></td>
<td> <a href="test.php?uid=<?php echo $id; ?>"><input type="submit" style="background-color:
blue;color: white;border-color: white"></a></td>
</tr>
<?php }}?>
<?php
 if(isset($_GET['uid']))
{
 $uid=$_GET['uid'];
include("config.php");
$sql1 = "update booktest set hstatus='Completed' where id='".$uid."'";
$result1 = mysqli_query($con,$sql1);
echo "<script> location.href='test.php'; </script>";
 }
 ?>
```

Test Info:

```
<?php session_start(); ?>
<?php
include("config.php");
$sql = "select * from test where hemail='".$_SESSION['huser']."'";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
if($count>0)
{
while($row = mysqli_fetch_array($result))
{
$cost=$row[2];
$time=$row[3];
}
}
else
{
$cost='';
$time='';
}
?>
<?php
if(isset($_POST['update']))
{
error_reporting(1);
include("config.php");
$costs=$_POST['cost'];
$times=$_POST['time'];
if($count<=0)
{
$sql1 = "insert into test(hemail, cost, time) values('".$_SESSION['huser']."','$costs','$times')";
if(mysqli_query($con,$sql1))
{
echo "<script>
alert('Info Updated');
</script>";
```

```
echo "<script> location.href='testinfo.php'; </script>";
}
else
{
echo "<script>
alert('Update Failed');
</script>";
echo "<script> location.href='testinfo.php'; </script>";
}
}
else
{
$sql2 = "update test set cost='$costs', time='$times' where hemail='".$_SESSION['huser']."'";
if(mysqli_query($con,$sql2))
{
echo "<script>
alert('Info Updated');
</script>";
echo "<script> location.href='testinfo.php'; </script>";
}
}
}
?>
```

Test Update:

```
<?php session_start();
?>
<?php
include("config.php");
$sql = "select booktest.*, hospital.hname from booktest inner join hospital on
hospital.email=booktest.hemail where booktest.hemail='".$_SESSION['huser']."' and
booktest.hstatus='Completed'";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
if($count>0)
{
```

```
$sl=0;
while($row = mysqli_fetch_array($result))
{
$id=$row[0];
$name=$row[1];
$aadhar=$row[2];
$phone=$row[3];
$dob=$row[4];
$slot=$row[5];
$email=$row[6];
$hemail=$row[7];
$payment=$row[8];
$status=$row[9];
$results=$row[10];
$hname=$row['hname'];
$sl+=1;
?>
<tr>
<td><?php echo $sl; ?></td>
<td><?php echo $name; ?></td>
<td><?php echo $aadhar; ?></td>
<td><?php echo $phone; ?></td>
<td><?php echo $dob; ?></td>
<td><?php echo $slot; ?></td>
<td><?php echo $email; ?></td>
<?php if($results=='No Result')
{
?>
<?php
}
else
{
?>
<td><?php echo $results; ?></td>
<?php }?>
<?php
if(isset($_POST['save']))
```

```
{
$result=$_POST['report'];
include("config.php");
$uid=$_GET['uid'];
$sql1 = "update booktest set result='$result' where id='".$uid."'";
$result1 = mysqli_query($con,$sql1);
echo "<script> location.href='testupdate.php'; </script>";
}
?>
</tr>
<?php }}?>
```

Vaccine:

```
<?php session_start(); ?>
<?php
include("config.php");
$sql1 = "select bookvaccine.*, hospital.hname from bookvaccine inner join hospital on
hospital.email=bookvaccine.hemail where bookvaccine.hemail='".$_SESSION['huser']."'";
$result1 = mysqli_query($con,$sql1);
$count1 =mysqlI_num_rows($result1);
if($count1>0)
{
$sl=0;
while($row1 = mysqli_fetch_array($result1))
{
$sl+=1;
$id=$row1[0];
$name=$row1[1];
$aadhar=$row1[2];
$phone=$row1[3];
$dob=$row1[4];
$email=$row1[5];
$vaccine=$row1[6];
$date=$row1[7];
$payment=$row1[9];
$status=$row1[10];
```

```
?>
<tr>
<td><?php echo $sl; ?></td>
<td><?php echo $vaccine; ?></td>
<td><?php echo $name; ?></td>
<td><?php echo $aadhar; ?></td>
<td><?php echo $phone; ?></td>
<td><?php echo $dob; ?></td>
<td><?php echo $email; ?></td>
<td><?php echo $date; ?></td>
<td><?php echo $payment; ?></td>
<td><?php echo $status; ?></td>
<td> <a href="vaccine.php?uid=<?php echo $id; ?>"><input type="submit" style="background-color:
blue;color: white;border-color: white"></a></td>
</tr>
<?php }}?>
</table>
<?php
if(isset($_GET['uid']))
{
$uid=$_GET['uid'];
include("config.php");
$sql1 = "update bookvaccine set hstatus='Completed' where id='".$uid."'";
$result1 = mysqli_query($con,$sql1);
echo "<script> location.href='vaccine.php'; </script>";
 }
 ?>
```

Vaccine Update:

```
<?php session_start(); ?>
<?php
if(isset($_POST['update']))
{
error_reporting(1);
include("config.php");
$sql = "select * from vaccine where hemail='".$_SESSION['huser']."'";
```

```
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
if($count>0)
{
while($row = mysqli_fetch_array($result))
{
$vname=$row[2];
$qty=$row[3];
$cost=$row[4];
}
}
else
{
$vname='';
}
$vaccine=$_POST['vaccine'];
$quantity=$_POST['quantity'];
$price=$_POST['price'];
if($vname!=$vaccine)
{
$sql1 = "insert into vaccine(hemail, vname, qty, cost)
values('".$_SESSION['huser']."','$vaccine','$quantity','$price')";
if(mysqli_query($con,$sql1))
{
echo "<script>
alert('Info Updated');
</script>";
echo "<script> location.href='vaccineupdate.php'; </script>";
}
else
{
echo "<script>
alert('Update Failed');
</script>";
echo "<script> location.href='vaccineupdate.php'; </script>";
}
}
```

```
else
{
$sql2 = "update vaccine set cost='".$price."', qty='".$quantity."' where hemail='".$_SESSION['huser']."'
and vname='$vaccine'";
if(mysqli_query($con,$sql2))
{
echo "<script>
alert('Info Updated');
</script>";
echo "<script> location.href='vaccineupdate.php'; </script>";
}
}
}
?>
<?php
include("config.php");
$sql = "select * from vaccine where hemail='".$_SESSION['huser']."'";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
if($count>0)
{
while($row = mysqli_fetch_array($result))
{
$vname=$row[2];
$qty=$row[3];
$cost=$row[4];
?>
<tr>
<td><?php echo $vname ?></td>
<td><?php echo $qty ?></td>
<td><?php echo $cost ?></td>
</tr>
<?php
}
}
 ?>
```

Admin:

Feedback:

```
<?php
include("config.php");
$sql4 = "select * from feedback";
$result4 = mysqli_query($con,$sql4);
$count4 =mysqlI_num_rows($result4);
if($count4>0)
{
$sl=0;
while($row4 = mysqli_fetch_array($result4))
{
$sl+=1;
$id=$row4[0];
$sub=$row4[1];
$msg=$row4[2];
$email=$row4[3];
?>
<tr>
<td><?php echo $sl; ?></td>
<td><?php echo $sub; ?></td>
<td><?php echo $msg; ?></td>
<td><?php echo $email; ?></td>
</tr>
<?php }}?>
```

Hospital Approval:

```
<?php
include("config.php");
$sql = "select hid, hname, phone, address, city, state, pincode, status from hospital";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
if($count>0)
{
$sl=0;
```

```
while($row = mysqli_fetch_array($result))
{
$sl+=1;
$id=$row[0];
$name=$row[1];
$phone=$row[2];
$address=$row[3];
$city=$row[4];
$state=$row[5];
$pin=$row[6];
$status=$row[7];
?>
<tr>
<th scope="row"><?php echo $sl; ?></th>
<td><?php echo $name; ?></td>
<td><?php echo $phone; ?></td>
<td><?php echo $address; echo ', '; echo $city; echo ', '; echo $state; echo ', '; echo $pin; ?></td>
<td><?php echo $status; ?></td>
<td>
<a href="?id=<?php echo $id; ?>&status=approve"><button type="button" class="btn btn-light btn-block">Approve</button></a>
</td>
<td>
<a href="?id=<?php echo $id; ?>&status=reject"><button type="button" class="btn btn-light btn-block">Reject</button></a>
</td>
</tr>
<?php }}?>
<?php
if(isset($_GET['id']))
{
$hid=$_GET['id'];
$status=$_GET['status'];
include("config.php");
$sql1 = "update hospital set status='".$status."' where hid='".$hid."'";
$result1 = mysqli_query($con,$sql1);
echo "<script>
```

```
alert('updated successfully');
</script>";
echo "<script> location.href='HospitalApproval.php'; </script>";
}
 ?>
```

Index:

```
<?php
if(isset($_POST['login']))
{
error_reporting(1);
include("config.php");
$email=$_POST['email'];
$password=$_POST['password'];
$sql = "select * from admin where email='$email' and password='$password'";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
if($count>0)
{
$_SESSION['admin']=$email;
echo "<script>
alert('Login Successful');
</script>";
echo "<script> location.href='dasboard.php'; </script>";
}
else
{
echo "<script>
alert('Invalid Email or Password');
</script>";
echo "<script> location.href='index.php'; </script>";
}
}
?>
```

Plasma:

```
<?php
    include("config.php");
    $sql4 = "select * from donate";
    $result4 = mysqli_query($con,$sql4);
    $count4 =mysqlI_num_rows($result4);
    if($count4>0)
    {
    while($row4 = mysqli_fetch_array($result4))
    {
    $id=$row4[0];
    $name=$row4[1];
    $dob=$row4[2];
    $age=$row4[3];
    $gender=$row4[4];
    $phone=$row4[5];
    $state=$row4[6];
    $address=$row4[7];
    $blood=$row4[8];
    $test=$row4[9];
    $symp=$row4[10];
    $rep=$row4[11];
    $last=$row4[12];
    $email=$row4[13];
    $hemail=$row4[14];
    $status=$row4[15];
    $appdate=$row4[16];
    ?>
    <tr>
     <td><?php echo $name; ?></td>
     <td><?php echo $dob; ?></td>
    <td><?php echo $age; ?></td>
    <td><?php echo $gender; ?></td>
    <td><?php echo $email; ?></td>
    <td><?php echo $phone; ?></td>
    <td><?php echo $state; ?></td>
    <td><?php echo $address; ?></td>
```

```
<td><?php echo $blood; ?></td>
<td><?php echo $test; ?></td>
<td><?php echo $symp; ?></td>
<td><?php echo $rep; ?></td>
<td><?php echo $last; ?></td>
<td><?php echo $status; ?></td>
<td><?php echo $appdate; ?></td>
</tr>
<?php }}?>
```

Registered Hospital:

```
<?php
include("config.php");
$sql = "select hid, hname, phone, address, city, state, pincode from hospital";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
if($count>0)
{
$sl=0;
while($row = mysqli_fetch_array($result))
{
$sl+=1;
$id=$row[0];
$name=$row[1];
$phone=$row[2];
$address=$row[3];
$city=$row[4];
$state=$row[5];
$pin=$row[6];
?>
<tr>
<th scope="row"><?php echo $sl; ?></th>
<td><?php echo $name; ?></td>
<td><?php echo $phone; ?></td>
<td><?php echo $address; echo ', '; echo $city; echo ', '; echo $state; echo ', '; echo $pin; ?></td>
```

```
<td><a href="RegisteredHospital.php?id=<?php echo $id; ?>"><button type="button" class="btn btn-
light btn-block">Delete</button></a></td>
</tr>
<?php }}?>
<?php
if(isset($_GET['id']))
{
$hid=$_GET['id'];
include("config.php");
$sql1 = "delete from hospital where hid='".$id."'";
$result1 = mysqli_query($con,$sql1);
echo "<script>
alert('hospital Deleted');
</script>";
echo "<script> location.href='RegisteredHospital.php'; </script>";
}
?>
```

Registered User:

```
<?php
include("config.php");
$sql = "select * from user";
$result = mysqli_query($con,$sql);
$count =mysqlI_num_rows($result);
if($count>0)
{
$sl=0;
while($row = mysqli_fetch_array($result))
{
$sl+=1;
$id=$row[0];
$name=$row[1];
$dob=$row[2];
$phone=$row[3];
$email=$row[4];
?>
```

```
<tr>
<th scope="row"><?php echo $sl; ?></th>
<td><?php echo $name; ?></td>
<td><?php echo $phone; ?></td>
<td><?php echo $email; ?></td>
<td><?php echo $dob; ?></td>
</tr>
<?php }}?>
```

Sql Database

```
-- phpMyAdmin SQL Dump
-- version 4.8.5
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Generation Time: Sep 27, 2021 at 07:42 PM
-- Server version: 10.1.39-MariaDB
-- PHP Version: 7.3.5
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET AUTOCOMMIT = 0;
START TRANSACTION;
SET time_zone = "+00:00";
/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;
--
-- Database: `covidshield`
--
-------------------------------------------------------
--
-- Table structure for table `admin`
--
CREATE TABLE `admin` (
 `id` int(11) NOT NULL,
 `email` varchar(50) NOT NULL,
```

```
  `password` varchar(50) NOT NULL
  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
  --
  -- Dumping data for table `admin`
  --
  INSERT INTO `admin` (`id`, `email`, `password`) VALUES
  (1, 'admin@gmail.com', 'Admin123');
  ---------------------------------------------------------
  --
  -- Table structure for table `bed`
  --
  CREATE TABLE `bed` (
    `id` int(11) NOT NULL,
    `hemail` varchar(50) NOT NULL,
    `count` int(11) NOT NULL
  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
  --
  -- Dumping data for table `bed`
  --
  INSERT INTO `bed` (`id`, `hemail`, `count`) VALUES
  (1, 'kmc@gmail.com', 100);
  ---------------------------------------------------------
  --
  -- Table structure for table `bookbed`
  --
  CREATE TABLE `bookbed` (
    `id` int(11) NOT NULL,
    `name` varchar(50) NOT NULL,
    `phone` varchar(30) NOT NULL,
    `aadhar` varchar(50) NOT NULL,
    `dob` date NOT NULL,
    `email` varchar(50) NOT NULL,
    `idproof` varchar(100) NOT NULL,
    `days` int(30) NOT NULL,
    `date` date NOT NULL,
    `hemail` varchar(50) NOT NULL,
    `payment` varchar(50) NOT NULL DEFAULT 'pending',
```

```
  `hstatus` varchar(50) NOT NULL DEFAULT 'pending'
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `bookbed`
--
INSERT INTO `bookbed` (`id`, `name`, `phone`, `aadhar`, `dob`, `email`, `idproof`, `days`, `date`,
`hemail`, `payment`, `hstatus`) VALUES
(1, 'Sebastian Dsilva', '09448726975', '1243 1234 1234', '2021-09-17', 'tom@gmail.com', 'Sebastian
Dsilvabg.png', 2, '2021-09-17', 'kmc@gmail.com', 'paid', 'Confirmed'),
(2, 'Jack', '0985 465 7758', '1243 1234 1234', '2021-09-10', 'tom@gmail.com', 'Jackbg2.png', 1, '2021-09-
17', 'kmc@gmail.com', 'pending', 'pending');
--------------------------------------------------------
--
-- Table structure for table `booktest`
--
CREATE TABLE `booktest` (
  `id` int(11) NOT NULL,
  `name` varchar(50) NOT NULL,
  `aadhar` int(50) NOT NULL,
  `phone` varchar(50) NOT NULL,
  `dob` date NOT NULL,
  `slot` date NOT NULL,
  `uemail` varchar(50) NOT NULL,
  `hemail` varchar(50) NOT NULL,
  `payment` varchar(50) NOT NULL DEFAULT 'pending',
  `hstatus` varchar(50) NOT NULL DEFAULT 'pending',
  `result` varchar(50) NOT NULL DEFAULT 'No Result'
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `booktest`
--
INSERT INTO `booktest` (`id`, `name`, `aadhar`, `phone`, `dob`, `slot`, `uemail`, `hemail`, `payment`,
`hstatus`, `result`) VALUES
(1, 'charlie', 1243, '6758847654', '2021-09-03', '2021-09-24', 'tom@gmail.com', 'kmc@gmail.com', 'paid',
'Completed', 'Negative'),
(2, 'aman', 1243, '9789898730', '1999-06-24', '2021-09-28', 'tom@gmail.com', 'kmc@gmail.com',
'pending', 'pending', 'No Result');
```

```
--------------------------------------------------------
--
-- Table structure for table `bookvaccine`
--
CREATE TABLE `bookvaccine` (
  `id` int(11) NOT NULL,
  `name` varchar(50) NOT NULL,
  `aadhar` varchar(50) NOT NULL,
  `phone` varchar(30) NOT NULL,
  `dob` date NOT NULL,
  `hemail` varchar(50) NOT NULL,
  `vname` varchar(50) NOT NULL,
  `slot` date NOT NULL,
  `email` varchar(50) NOT NULL,
  `payment` varchar(50) NOT NULL DEFAULT 'pending',
  `hstatus` varchar(50) NOT NULL DEFAULT 'pending'
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `bookvaccine`
--
INSERT INTO `bookvaccine` (`id`, `name`, `aadhar`, `phone`, `dob`, `hemail`, `vname`, `slot`, `email`,
`payment`, `hstatus`) VALUES
(1, 'charlie', '123467788', '9807383945', '2021-09-11', 'city@gmail.com', 'Covaxin', '2021-09-17',
'tom@gmail.com', 'paid', 'Completed'),
(2, 'harish12', '234567890121', '7019356629', '2021-09-01', 'kmc@gmail.com', 'Covaxin', '2021-09-18',
'tom@gmail.com', 'pending', 'pending');
--------------------------------------------------------
--
-- Table structure for table `donate`
--
CREATE TABLE `donate` (
  `id` int(11) NOT NULL,
  `name` varchar(50) NOT NULL,
  `dob` date NOT NULL,
  `age` int(50) NOT NULL,
  `gender` varchar(50) NOT NULL,
  `phone` varchar(50) NOT NULL,
```

```
  `state` varchar(50) NOT NULL,
  `address` varchar(50) NOT NULL,
  `blood` varchar(50) NOT NULL,
  `test` varchar(50) NOT NULL,
  `smptoms` varchar(50) NOT NULL,
  `test2` varchar(50) NOT NULL,
  `last` date NOT NULL,
  `email` varchar(50) NOT NULL,
  `hemail` varchar(50) NOT NULL,
  `status` varchar(50) NOT NULL DEFAULT 'pending',
  `appdate` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `donate`
--
INSERT INTO `donate` (`id`, `name`, `dob`, `age`, `gender`, `phone`, `state`, `address`, `blood`, `test`,
`smptoms`, `test2`, `last`, `email`, `hemail`, `status`, `appdate`) VALUES
(1, 'Briden Springfield', '2021-09-03', 35, 'Male', '09449048283', 'Karnataka', 'Green Spring Hill', 'AB-',
'yes', 'No', 'Yes', '2021-09-02', 'tom@gmail.com', 'kmc@gmail.com', 'Approved', '2021-09-17');
-- --------------------------------------------------------
--
-- Table structure for table `feedback`
--
CREATE TABLE `feedback` (
  `id1` int(11) NOT NULL,
  `subject` varchar(50) NOT NULL,
  `message` varchar(100) NOT NULL,
  `email` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `feedback`
--
INSERT INTO `feedback` (`id1`, `subject`, `message`, `email`) VALUES
(1, 'just testing', 'This is testing', 'tom@gmail.com');
--------------------------------------------------------
--
-- Table structure for table `hospital`
```

```
--
CREATE TABLE `hospital` (
  `hid` int(11) NOT NULL,
  `hname` varchar(100) NOT NULL,
  `question` varchar(50) NOT NULL,
  `answer` varchar(50) NOT NULL,
  `email` varchar(50) NOT NULL,
  `phone` varchar(50) NOT NULL,
  `address` varchar(100) NOT NULL,
  `city` varchar(50) NOT NULL,
  `state` varchar(50) NOT NULL,
  `pincode` int(30) NOT NULL,
  `username` varchar(50) NOT NULL,
  `password` varchar(50) NOT NULL,
  `status` varchar(50) NOT NULL DEFAULT 'pending',
  `image` varchar(100) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;


--
-- Dumping data for table `hospital`
--
INSERT INTO `hospital` (`hid`, `hname`, `question`, `answer`, `email`, `phone`, `address`, `city`, `state`,
`pincode`, `username`, `password`, `status`, `image`) VALUES
(2, 'KMC', '4', 'poochi', 'kmc@gmail.com', '9485676453', 'Attavar near Dmart', 'Mangalore', 'Karnataka',
575005, 'Jack', 'Kmc12345', 'approve', 'kmc.jpg'),
(3, 'City Hospital', '2', 'mysore', 'city@gmail.com', '9485676453', 'Attavar near Dmart', 'Mangalore',
'Karnataka', 575005, 'Jack', 'City12345', 'approve', 'img.jpg');
-------------------------------------------------------
--
-- Table structure for table `test`
--
CREATE TABLE `test` (
  `id` int(11) NOT NULL,
  `hemail` varchar(50) NOT NULL,
  `cost` int(30) NOT NULL,
  `time` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--
-- Dumping data for table `test`
--
INSERT INTO `test` (`id`, `hemail`, `cost`, `time`) VALUES
(1, 'kmc@gmail.com', 700, '2 hours'),
(4, 'city@gmail.com', 900, '2 hours');
-------------------------------------------------------
--
-- Table structure for table `user`
--
CREATE TABLE `user` (
  `uid` int(11) NOT NULL,
  `username` varchar(50) NOT NULL,
  `dob` date NOT NULL,
  `phone` varchar(30) NOT NULL,
  `email` varchar(50) NOT NULL,
  `question` varchar(50) NOT NULL,
  `answer` varchar(50) NOT NULL,
  `password` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `user`
--
INSERT INTO `user` (`uid`, `username`, `dob`, `phone`, `email`, `question`, `answer`, `password`)
VALUES
(1, 'Tom', '2021-09-01', '7019235541', 'tom@gmail.com', '2', 'uk', 'User1234');
-------------------------------------------------------
--
-- Table structure for table `vaccine`
--
CREATE TABLE `vaccine` (
  `id` int(11) NOT NULL,
  `hemail` varchar(50) NOT NULL,
  `vname` varchar(100) NOT NULL,
  `qty` int(20) NOT NULL,
  `cost` int(20) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--
-- Dumping data for table `vaccine`
--
INSERT INTO `vaccine` (`id`, `hemail`, `vname`, `qty`, `cost`) VALUES
(2, 'kmc@gmail.com', 'CovidShield', 300, 700),
(3, 'kmc@gmail.com', 'Covaxin', 199, 800),
(4, 'kmc@gmail.com', 'Sputnik', 120, 1200),
(5, 'city@gmail.com', 'CovidShield', 200, 700),
(6, 'city@gmail.com', 'Covaxin', 299, 900);
--
-- Indexes for dumped tables
--
--
-- Indexes for table `admin`
--
ALTER TABLE `admin`
  ADD PRIMARY KEY (`id`);
--
-- Indexes for table `bed`
--
ALTER TABLE `bed`
  ADD PRIMARY KEY (`id`);
--
-- Indexes for table `bookbed`
--
ALTER TABLE `bookbed`
  ADD PRIMARY KEY (`id`);
--
-- Indexes for table `booktest`
--
ALTER TABLE `booktest`
  ADD PRIMARY KEY (`id`);
--
-- Indexes for table `bookvaccine`
--
ALTER TABLE `bookvaccine`
  ADD PRIMARY KEY (`id`);
```

```
--
-- Indexes for table `donate`
--
ALTER TABLE `donate`
  ADD PRIMARY KEY (`id`);
--
-- Indexes for table `feedback`
--
ALTER TABLE `feedback`
  ADD PRIMARY KEY (`id1`);
--
-- Indexes for table `hospital`
--
ALTER TABLE `hospital`
  ADD PRIMARY KEY (`hid`);
--
-- Indexes for table `test`
--
ALTER TABLE `test`
  ADD PRIMARY KEY (`id`);
--
-- Indexes for table `user`
--
ALTER TABLE `user`
  ADD PRIMARY KEY (`uid`);
--
-- Indexes for table `vaccine`
--
ALTER TABLE `vaccine`
  ADD PRIMARY KEY (`id`);
--
-- AUTO_INCREMENT for dumped tables
--
--
-- AUTO_INCREMENT for table `admin`
--
ALTER TABLE `admin`
```

```
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
--
-- AUTO_INCREMENT for table `bed`
--
ALTER TABLE `bed`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;
--
-- AUTO_INCREMENT for table `bookbed`
--
ALTER TABLE `bookbed`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;
--
-- AUTO_INCREMENT for table `booktest`
--
ALTER TABLE `booktest`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;
--
-- AUTO_INCREMENT for table `bookvaccine`
--
ALTER TABLE `bookvaccine`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;
--
-- AUTO_INCREMENT for table `donate`
--
ALTER TABLE `donate`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;
--
-- AUTO_INCREMENT for table `feedback`
--
ALTER TABLE `feedback`
  MODIFY `id1` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
--
-- AUTO_INCREMENT for table `hospital`
--
ALTER TABLE `hospital`
  MODIFY `hid` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=9;
--
```

```
-- AUTO_INCREMENT for table `test`
--
ALTER TABLE `test`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=7;
--
-- AUTO_INCREMENT for table `user`
--
ALTER TABLE `user`
  MODIFY `uid` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
--
-- AUTO_INCREMENT for table `vaccine`--
ALTER TABLE `vaccine`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=8;
COMMIT;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

"Testing"

7.1 Introduction

- Testing is the major quality control measure used during software development it is a basic function to detect errors in the software during the requirement analysis and design, of the output of the document that is usually textual and non-executable after the coding phase.
- The computer programs are available and can be executed for testing purpose this implies that testing not only has to uncover errors introduced during the previous phase.
- The goal of testing is done cover requirement design and coding errors in the program system's working according to the specifications.
- It is the phase where we try to break system and will use it. Implementation is a final important phase.
- It involves user training system testing in order to ensure successful running of the proposed system.
- The user tests the system and changes are made according to their needs.
- The testing involves the testing of the developed system using various kinds of data.
- While testing errors are noted and correctness is maintained.

7.2 Testing Criteria

7.2.1 Objective of testing

- Finding defects which may get created by the programmer while developing the software.
- Gaining confidence in and providing information about the level of quality.
- To prevent defects.
- To make sure that the end results meet the business and user requirements.
- To ensure that it satisfies the BRS (Business Requirement Specification) and SRS (System Requirement Specification).

7.3 Testing Methodology

7.3.1 Unit Testing:

- Unit testing focuses efforts on the smallest unit of software design this is known as module testing or white box testing, the modules are tested separately.
- The test is carried out during the programming stage itself.
- In this step, each module is found to be working satisfactorily as regards to the expected output from the module.

7.3.2 Integration Testing:

- In integration testing the different units of the system are integrated together to form the complete system.
- This type of testing checks the system as a whole to ensure that it is doing what it's supposed to do the testing of an integrated system can be carried out top-down, bottom-up or Big-Bang.
- In this type of testing some parts are tested with white box testing and some with black box testing techniques.
- This type of testing plays a very important role in increasing systems productivity.
- We have checked the system by using integration testing techniques.

7.3.3 System Testing:

- Apart from testing the system to validate the functionality of software against the requirements.
- It is also necessary to test the non-functional aspect of the system, some examples of non-functional tools include tests to check the performance data security usability volume load and stress that we have used in a project to test the various module.
- System testing consists of the following steps:
  - Program(s) Testing.
  - String Testing.
  - System Testing.
  - System Documentation.
  - User Acceptance Test.

7.3.4 Field Testing:

- The special type of testing may be very important in some projects.
- Here the system is tested in actual operational surroundings the interfaces with other systems and the real world are checked.
- This type of testing is very rarely used so far and our project is concerned we haven't tested a project using field testing.

7.3.5 Acceptance Testing:

- After the developer has completed all rounds of testing and he is satisfied with the system, then the user takes over and retest system from his point of view to judge whether it is acceptable according to some previously identified criteria.
- This is almost always a tricky situation in the project because of the inherent conflict between the developer and the user in this project.
- It is the job of the developer of the planner to check the system that whether the system fulfils the goals or not.

7.4 Test Cases

User Registration

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If Strong password not given for password	At least one number and one uppercase and lowercase letter, and at least 8 or more characters	As Expected
3	If Phone number contain more than 10 number	Accepts only numeric values with 10 digit.	As Expected
4	Password entered in confirm password field doesn't match password field	Password confirmation doesn't match	As Expected
5	If Invalid email entered	Invalid email or password	As Expected
6	If Invalid date entered	Invalid date	As Expected
7	If Existing email	There is an existing account associated with this email	As Expected
8	Register button pressed when all fields not entered	Please fill out this field.	As Expected

User Login

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If Invalid email or password entered	Invalid email or password	As Expected
3	If Strong password not given for password	At least one number and one uppercase and lowercase letter, and at least 8 or more characters	As Expected
4	Sign in button pressed when all fields not entered	Please fill out this field.	As Expected

Reset Password

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If Invalid email entered	Invalid email or password	As Expected
3	If old password only entered	Password should be different from previous one	As Expected
4	Reset in button pressed when all fields not entered	Please fill out this field.	As Expected

Book Test

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If Phone number contain more than 10 number	Accepts only numeric values with 10 digit.	As Expected
3	If user name is not entered in text	Letters only	As Expected
4	If aadhar number is not entered in digits	Numbers only	As Expected
5	If dob is more than 2003	Minimum 18 years	As Expected
6	If Existing aadhar	There is an existing account associated with this aadhar	As Expected
7	If Invalid date entered	Invalid date	As Expected
8	Confirm button pressed when all fields not entered	Please fill out this field.	As Expected

Book Vaccine

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If Phone number contain more than 10 number	Accepts only numeric values with 10 digit.	As Expected

3	If user name is not entered in text	Letters only	As Expected
4	If aadhar number is not entered in digits	Numbers only	As Expected
5	If dob is more than 2003	Minimum 18 years	As Expected
6	If Existing aadhar	There is an existing account associated with this aadhar	As Expected
7	If Invalid date selected/entered	Invalid date	As Expected
8	If Select item is not selected	Select an item from the list	As Expected
9	If selected hospital does not contain the required vaccine	Vaccine not available	As Expected
10	Confirm button pressed when all fields not entered	Please fill out this field.	As Expected

Book Bed

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If Phone number contain more than 10 number	Accepts only numeric values with 10 digit.	As Expected
3	If user name is not entered in text	Letters only	As Expected
4	If aadhar number is not entered in digits	Numbers only	As Expected
5	If dob is more than 2003	Minimum 18 years	As Expected

6	If Existing aadhar	There is an existing account associated with this aadhar	As Expected
7	If Invalid from date entered	Invalid date	As Expected
8	If number of days is not entered in digits	Numbers only	As Expected
9	If id proof is not selected	No file is chosen	As Expected
10	Confirm button pressed when all fields not entered	Please fill out this field.	As Expected

Donate Plasma

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If Phone number contain more than 10 number	Accepts only numeric values with 10 digit.	As Expected
3	If donar name is not entered in text	Letters only	As Expected
4	If age is not entered in digits	Numbers only	As Expected
5	If dob is more than 2003	Minimum 18 years	As Expected
6	If Existing aadhar	There is an existing account associated with this aadhar	As Expected

7	If last symptoms date is greater than Current date	Invalid date	As Expected
8	If Select item is not selected	Select an item from the list	As Expected
9	If radio button is not selected	Please fill out this field.	As Expected
10	Submit button pressed when all fields not entered	Please fill out this field.	As Expected

Feedback

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	Send message button pressed when all fields not entered	Please fill out this field.	As Expected

Payment

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If card number is not entered in digits	Numbers only	As Expected
3	If card holder name not entered in text	Text only	As Expected
4	If card date expires	Invalid date	As Expected
5	Payment button pressed when all fields not entered	Please fill out this field.	As Expected

Admin Login

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If Invalid email or password entered	Invalid email or password	As Expected
3	If Strong password not given for password	At least one number and one uppercase and lowercase letter, and at least 8 or more characters	As Expected
4	Sign in button pressed when all fields not entered	Please fill out this field.	As Expected

Hospital Registration

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If Strong password not given for password	At least one number and one uppercase and lowercase letter, and at least 8 or more characters	As Expected
3	If Phone number contain more than 10 number	Accepts only numeric values with 10 digit.	As Expected
4	Password entered in confirm password field doesn't match password field	Password confirmation doesn't match	As Expected

5	If Invalid email format entered	Invalid email format	As Expected
6	If pin code entered not in numbers	Numbers only	As Expected
7	Register button pressed when all fields not entered	Please fill out this field.	As Expected

Hospital Login

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If Invalid email or password entered	Invalid email or password	As Expected
3	If Strong password not given for password	At least one number and one uppercase and lowercase letter, and at least 8 or more characters	As Expected
4	Sign in button pressed when all fields not entered	Please fill out this field.	As Expected

Reset Password

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If Invalid email entered	Invalid email or password	As Expected
3	If old password only entered	Password should be different from previous one	As Expected
4	Reset in button pressed when all fields not entered	Please fill out this field.	As Expected

Update Bed Status

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If number of beds entered not in numbers	Numbers only	As Expected

Update Vaccine Status

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If quantity available entered not in numbers	Numbers only	As Expected
3	If cost entered not in numbers	Numbers only	As Expected
4	If Select item is not selected	Select an item from the list	As Expected

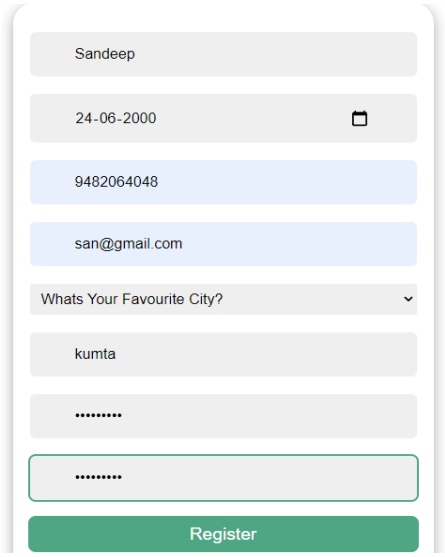
Update Test Info

No	Test Case	Expected output	Output
1	If Required fields are empty	Please fill out this field.	As Expected
2	If cost entered not in numbers	Numbers only	As Expected

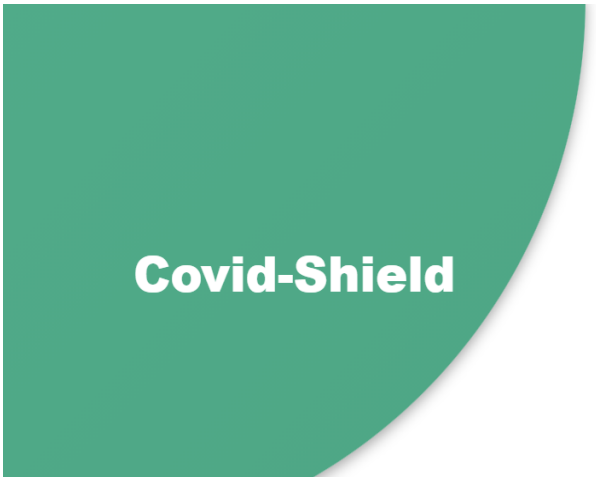
"Screenshots"

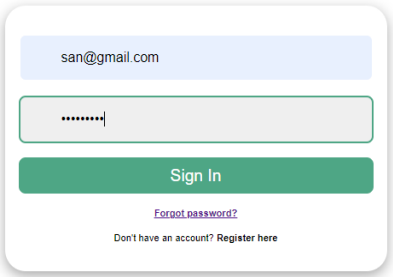
User

User-Register

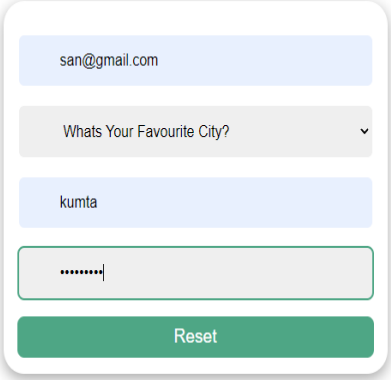


User-Login





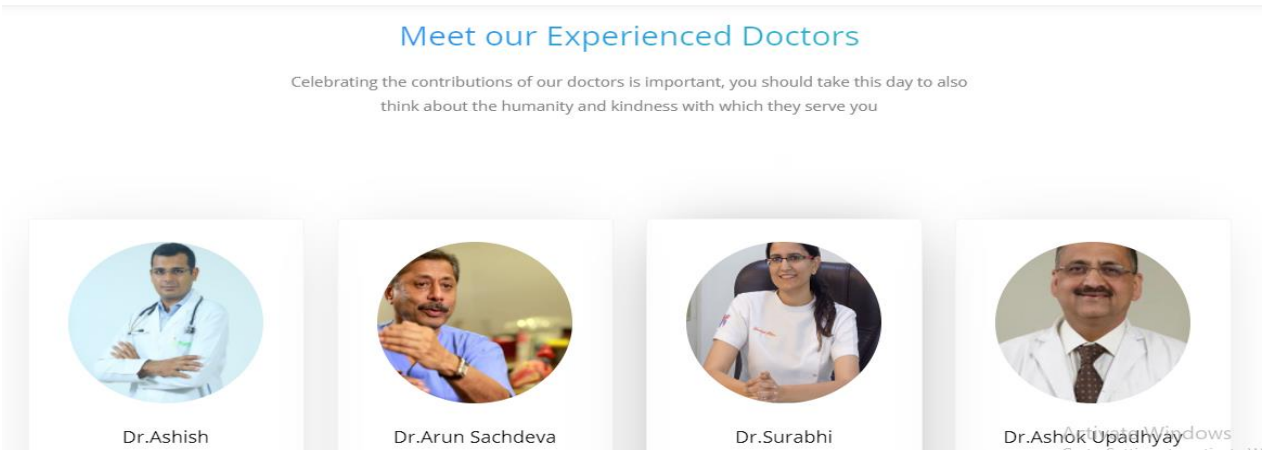
Reset-Password

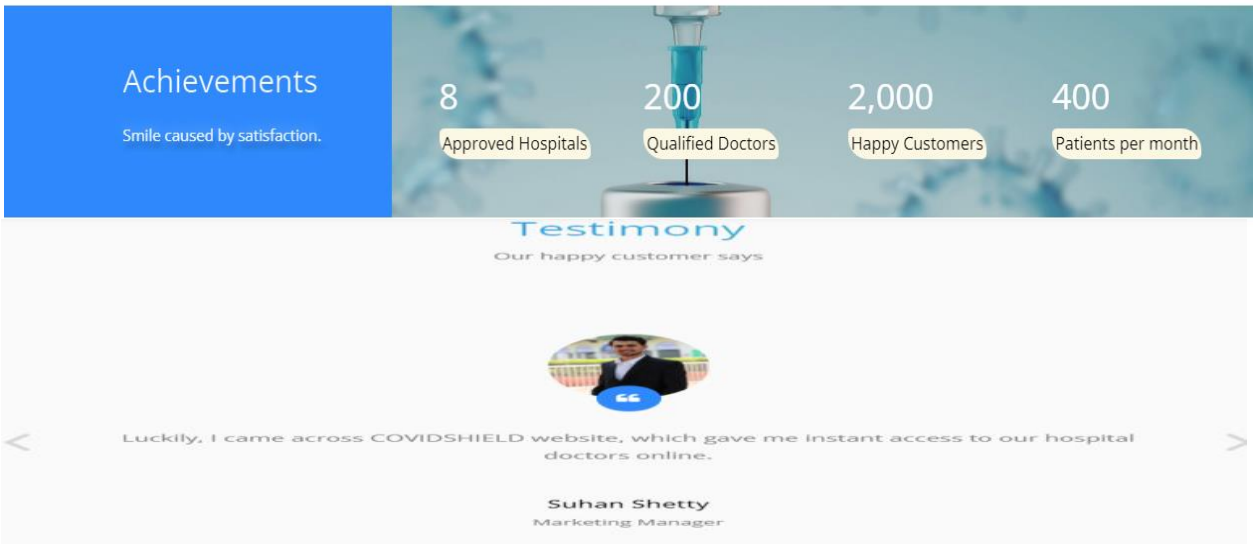


Home-Page

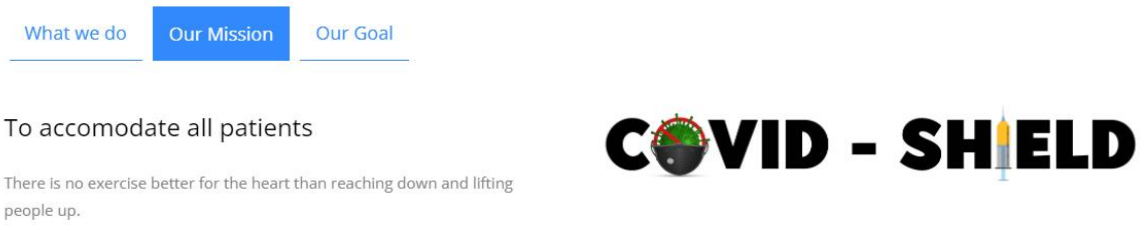

COVIDSHIELD
Home
About
Services
feedback
Logout
Recovery plan for the World
Modern mankind just a click away
COVID-19
VACCINE

CORONAVIRUS
COVID - 19
SYMPTOMS
High Fever
Dry Cough
Sore Throat
Difficulty in Breathing
HOW IT SPREADS
Air by Cough or Sneeze
Personal Contact
Contaminated Objects
Mass Gathering
PREVENTION
Wash your Hands often
Wear a Face Mask
Avoid Contact with Sick People
Always Cover Your Cough or Sneeze

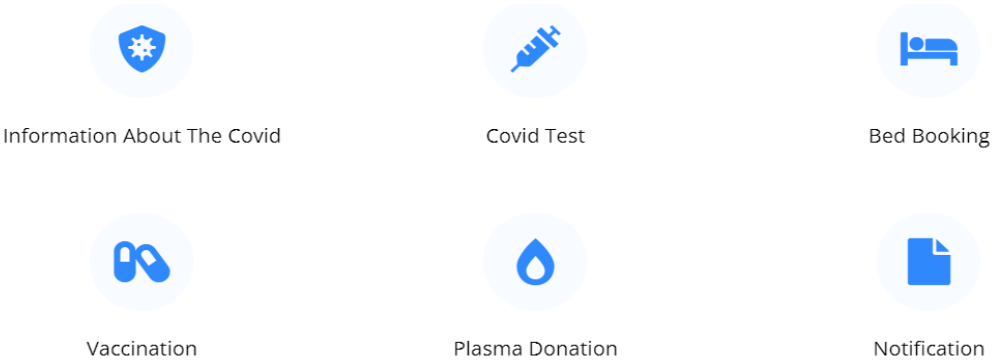

Meet our Experienced Doctors
Celebrating the contributions of our doctors is important, you should take this day to also think about the humanity and kindness with which they serve you
Dr.Ashish
Dr.Arun Sachdeva
Dr.Surabhi
Dr.Ashok Upadhyay


Achievements
Smile caused by satisfaction.
8
Approved Hospitals
200
Qualified Doctors
2,000
Happy Customers
400
Patients per month
Testimony
Our happy customer says
Luckily, I came across COVIDSHIELD website, which gave me instant access to our hospital doctors online.
Suhan Shetty
Marketing Manager

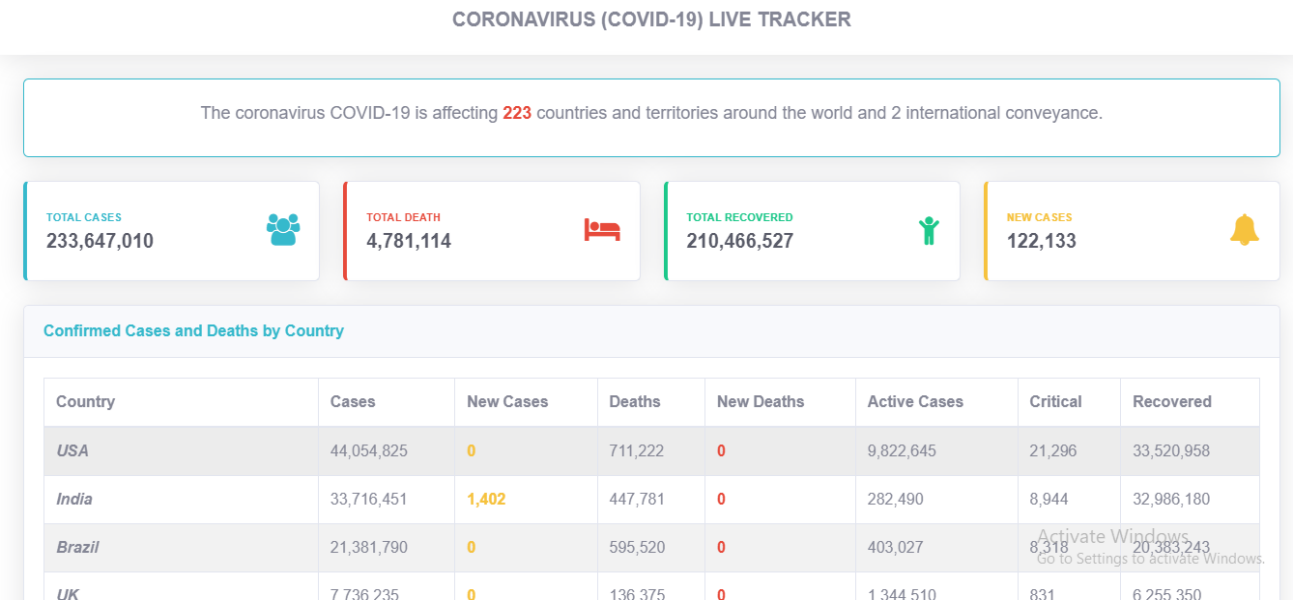
About



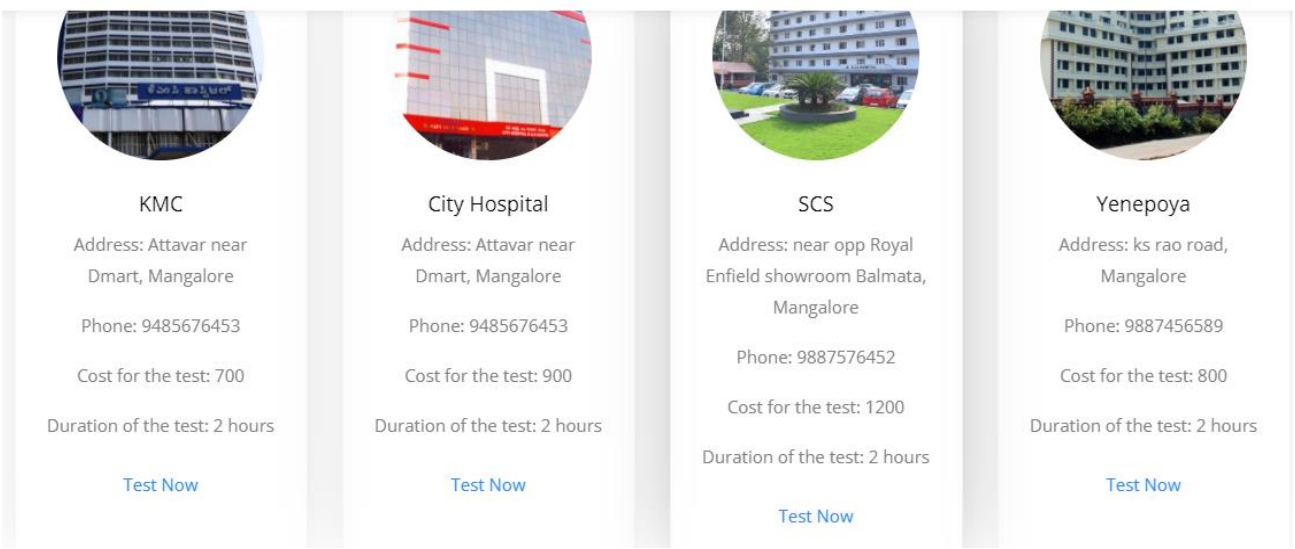
Services

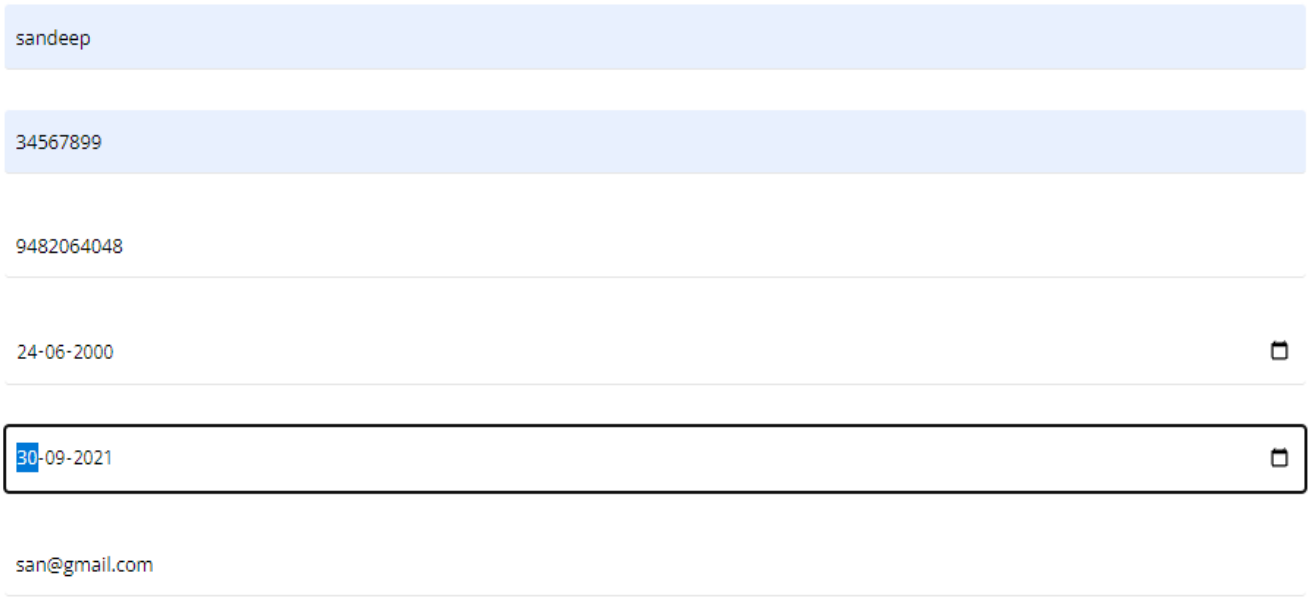


Live Tracker

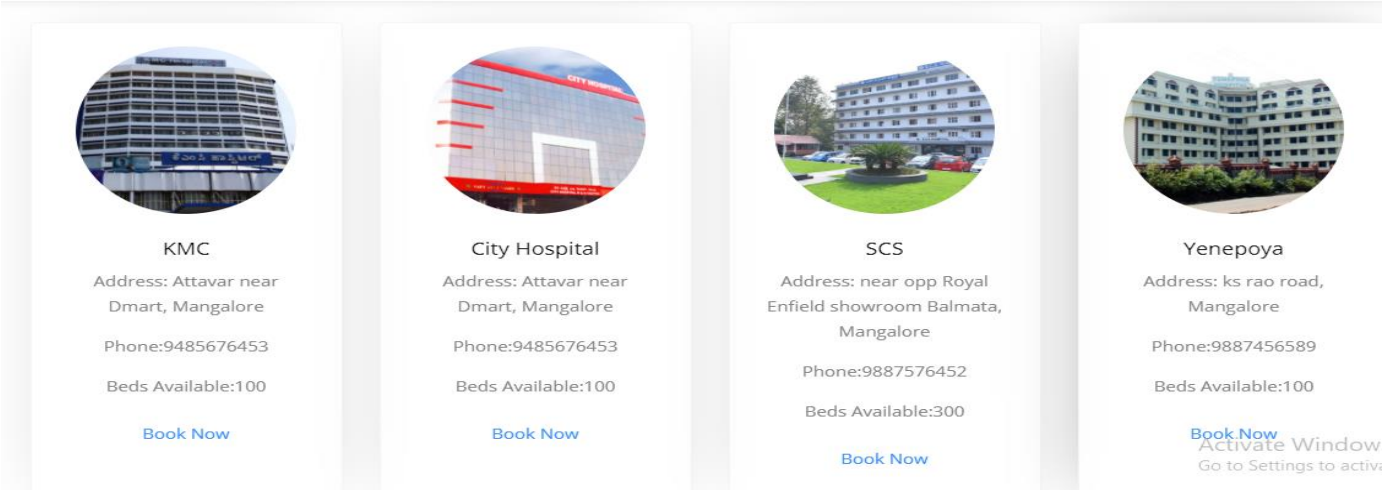


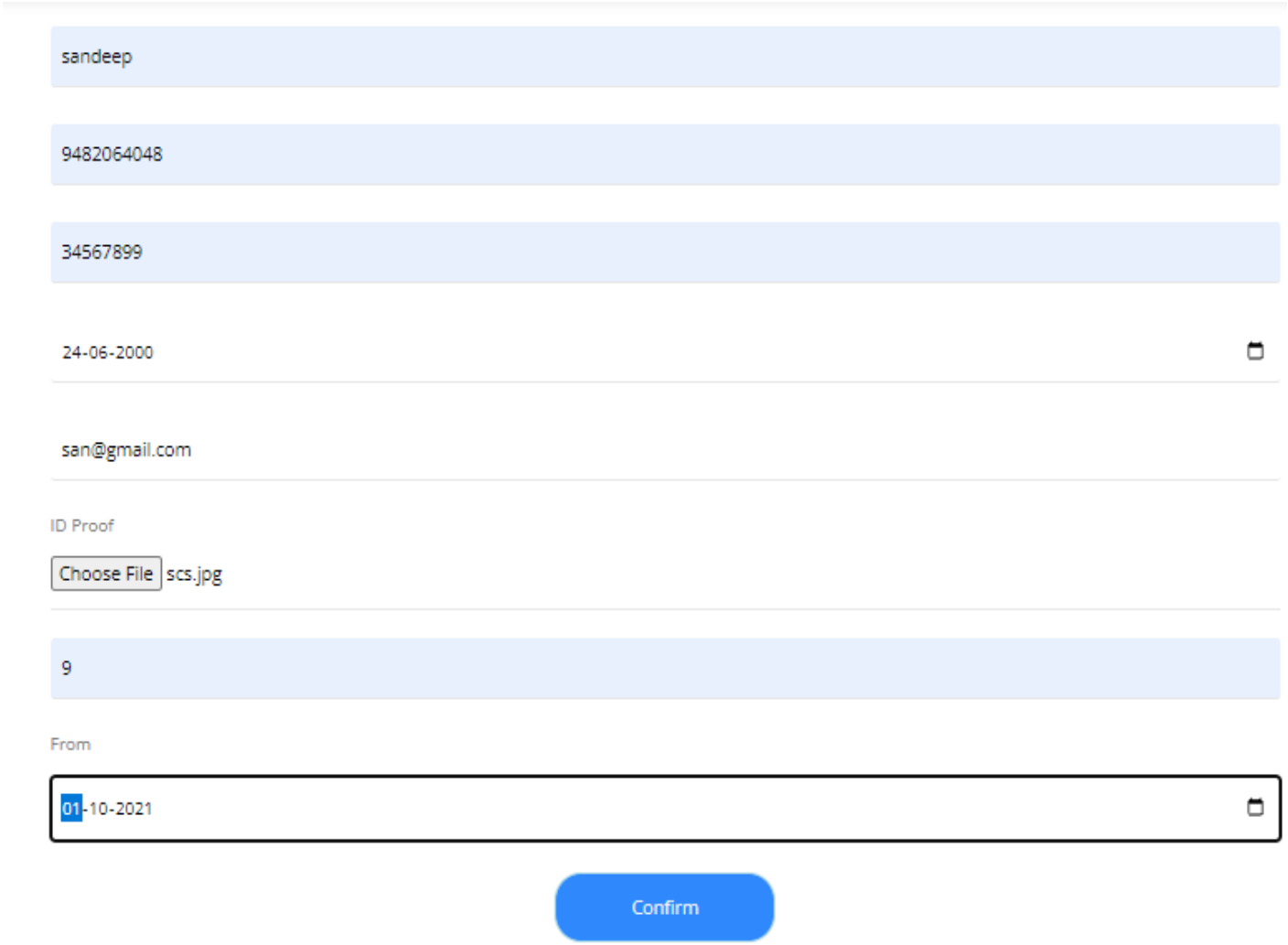
Covid Test





Bed Booking





Vaccination

Covid – Shield

--Select--

Book Slot

Name of the Hospital	Name of the Vaccine	Available/Unavailable	Cost Of The Vacciene
KMC	CovidShield	300	700
KMC	Covaxin	199	800
KMC	Sputnik	120	1200
City Hospital	CovidShield	200	700
City Hospital	Covaxin	299	900
Yenepoya	CovidShield	80	700

Confirm

Plasma Donation

KMC

Address: Attavar near Dmart, Mangalore, Karnataka, 575005

Phone:9485676453

Donate Here

City Hospital

Address: Attavar near Dmart, Mangalore, Karnataka, 575005

Phone:9485676453

Donate Here

SCS

Address: near opp Royal Enfield showroom Balmata, Mangalore, Karnataka, 575003

Phone:9887576452

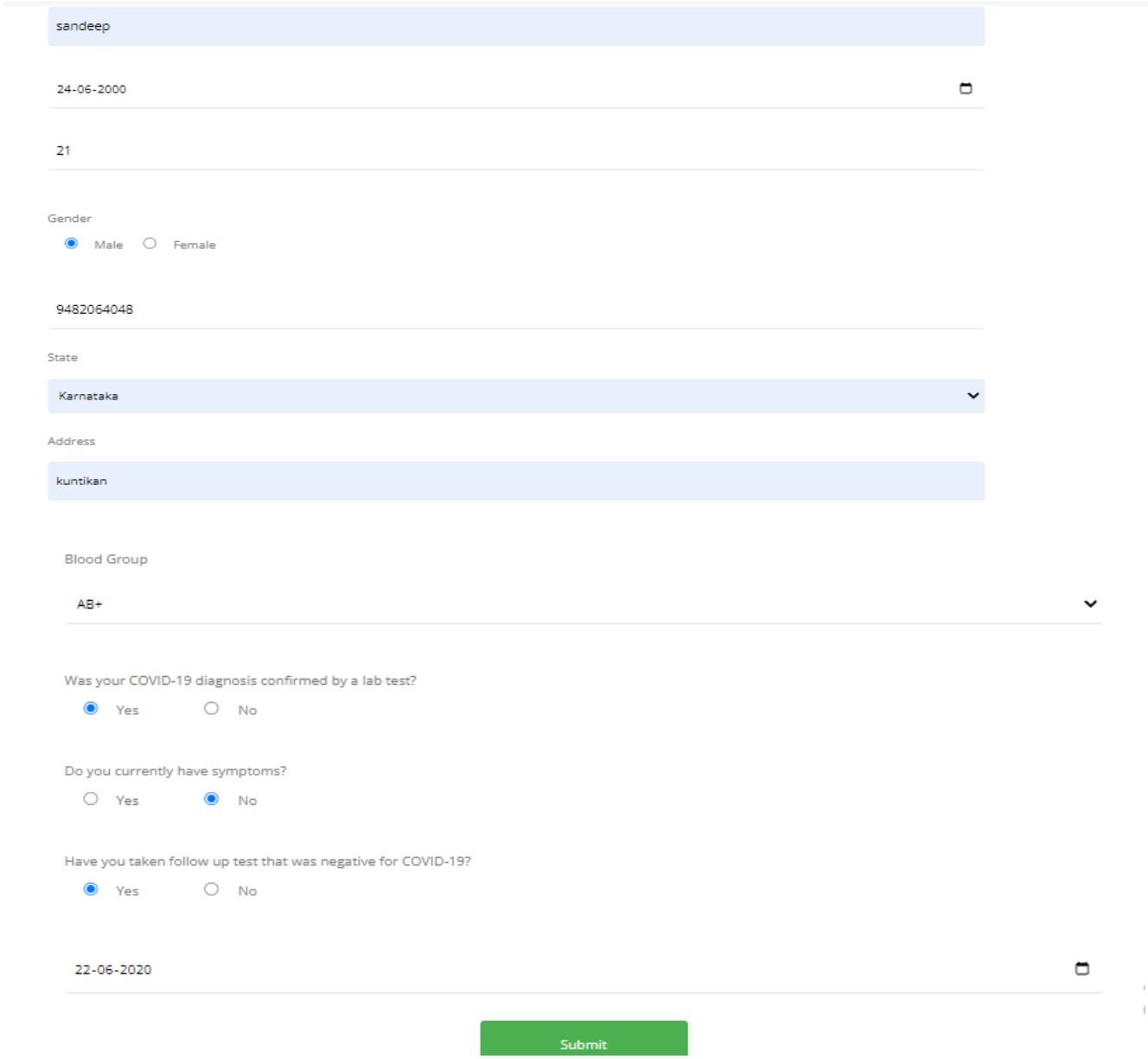
Donate Here

Yenepoya

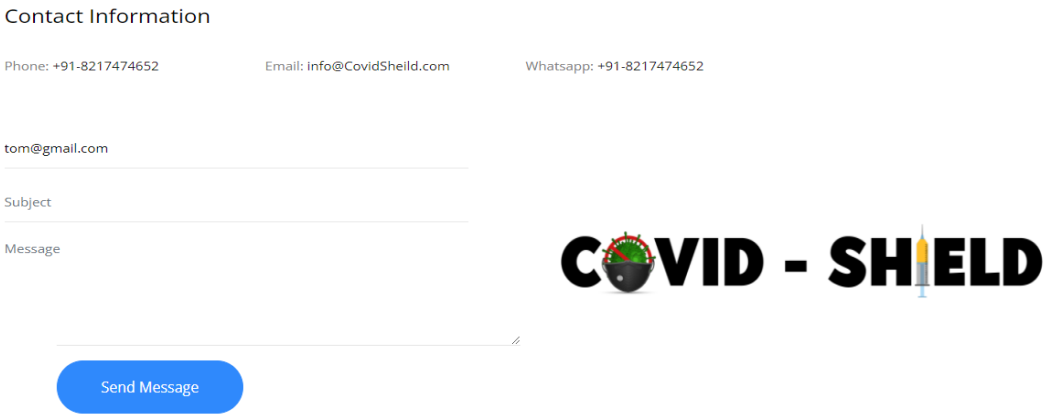
Address: ks rao road, Mangalore, Karnataka, 575002

Phone:9887456589

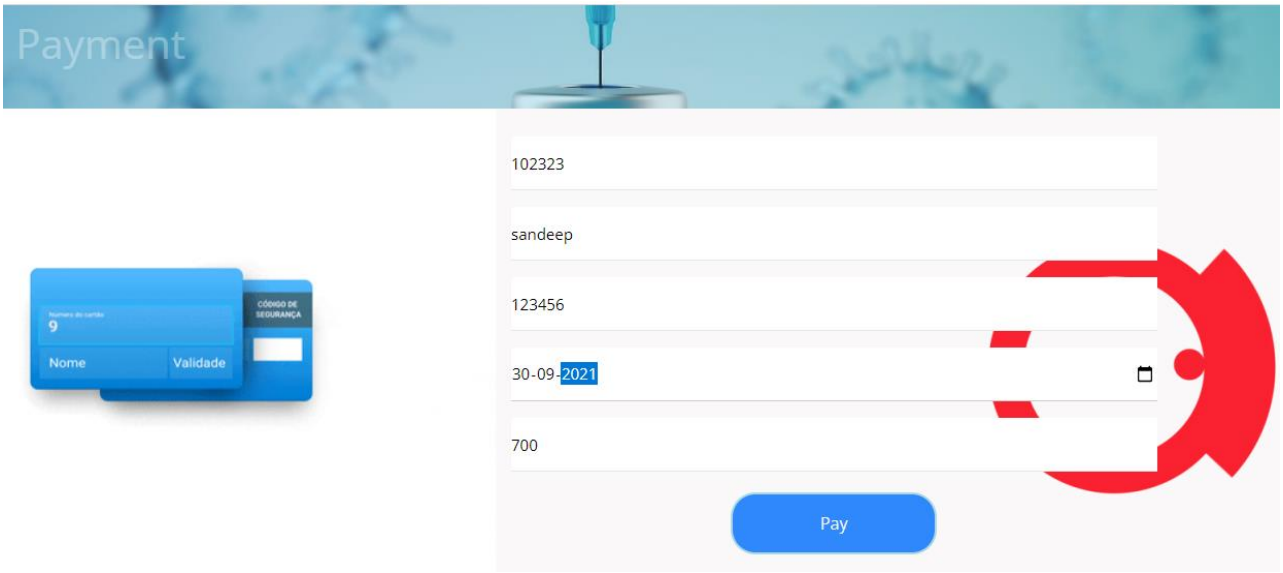
Donate Here


sandeep
24-06-2000
21
Gender
Male
Female
9482064048
State
Karnataka
Address
kuntikan
Blood Group
AB+
Was your COVID-19 diagnosis confirmed by a lab test?
Yes
No
Do you currently have symptoms?
Yes
No
Have you taken follow up test that was negative for COVID-19?
Yes
No
22-06-2020
Submit

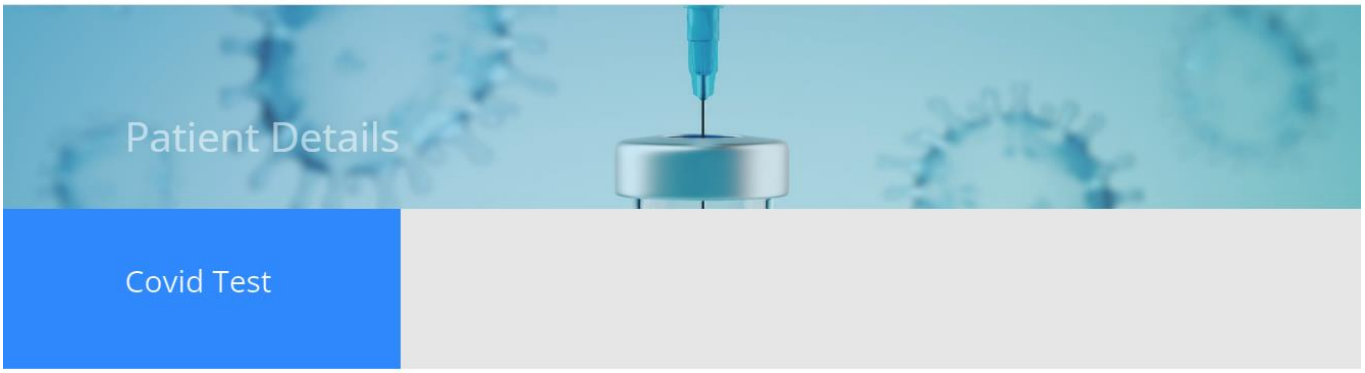
Feedback


Contact Information
Phone: +91-8217474652
Email: info@CovidSheild.com
Whatsapp: +91-8217474652
tom@gmail.com
Subject
Message
Send Message
COVID - SHIELD

Payment



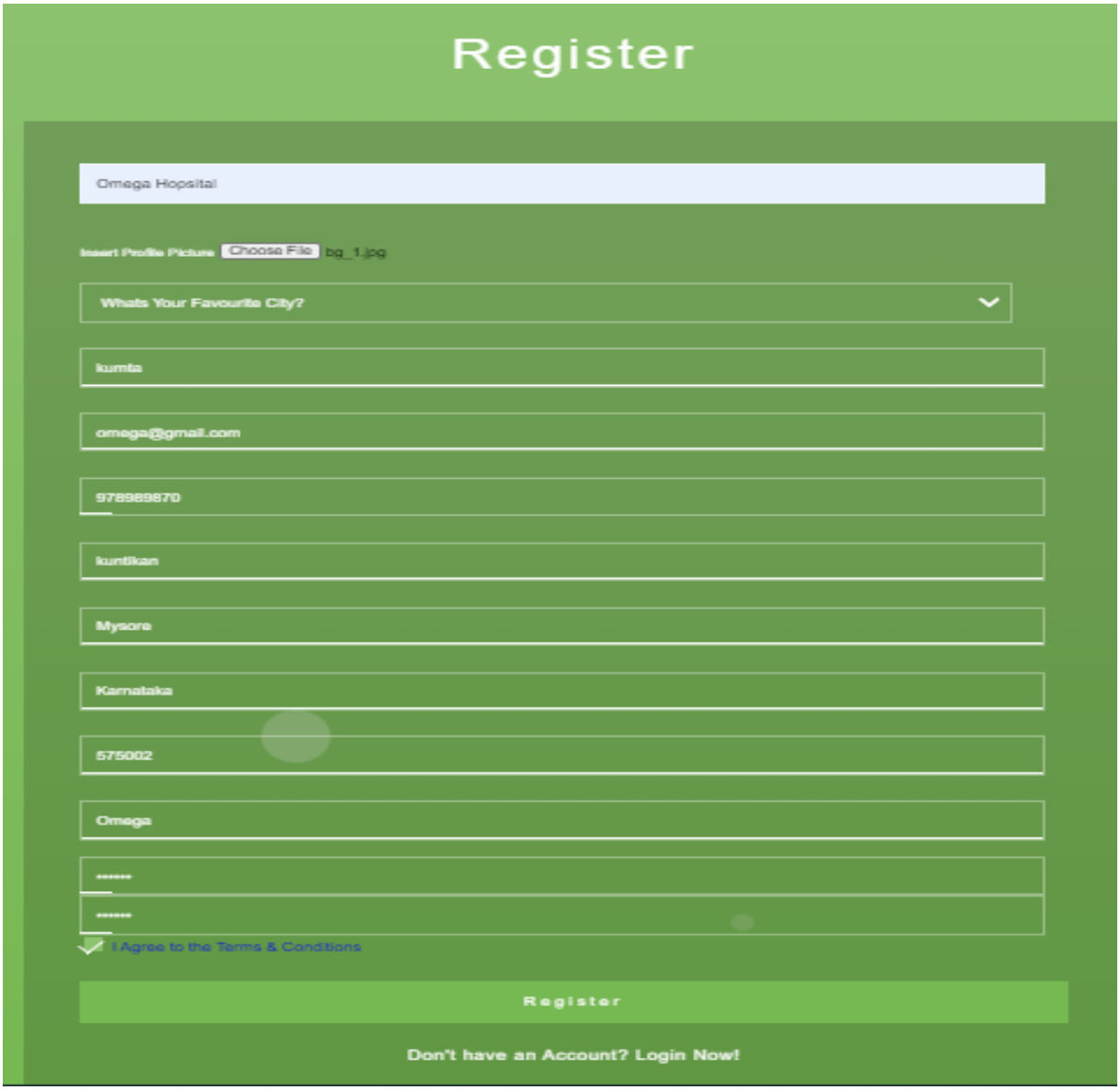
Notification



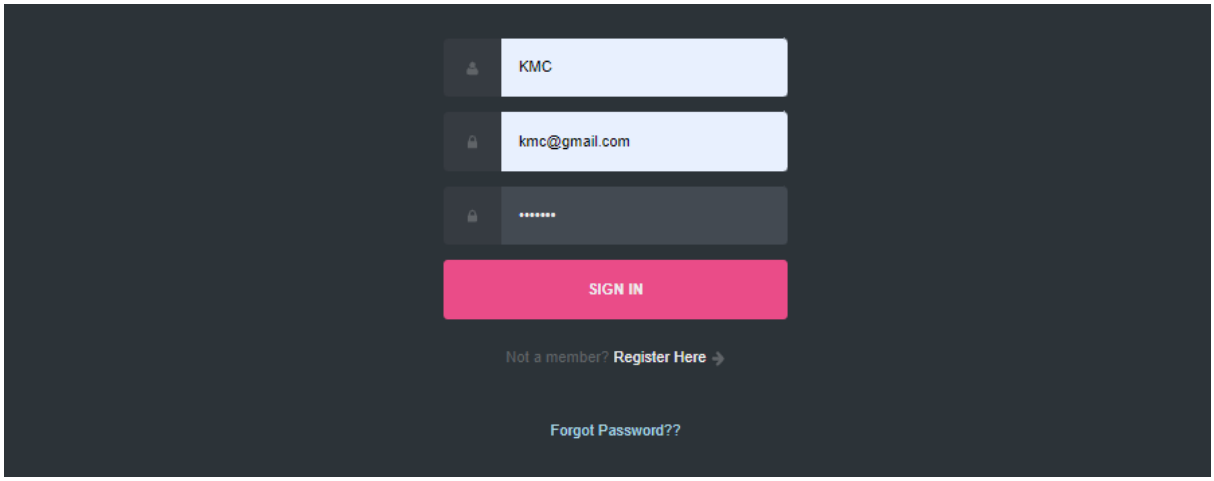
Hospital Name	Patient Name	Adhar Number	Phone Number	DOB	Slot	Email	cost	Payment Status	Result	download
KMC	charlie	1243	6758847654	2021-09-03	2021-09-24	tom@gmail.com	700	paid	Negative	Download Certificate

Hospital

Register

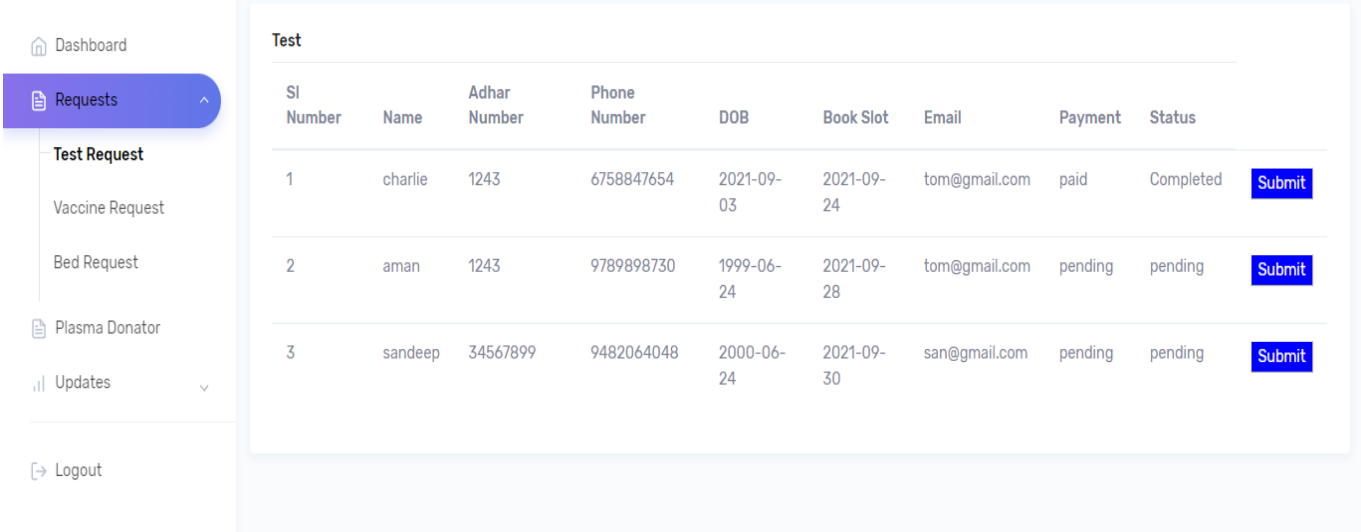


Login

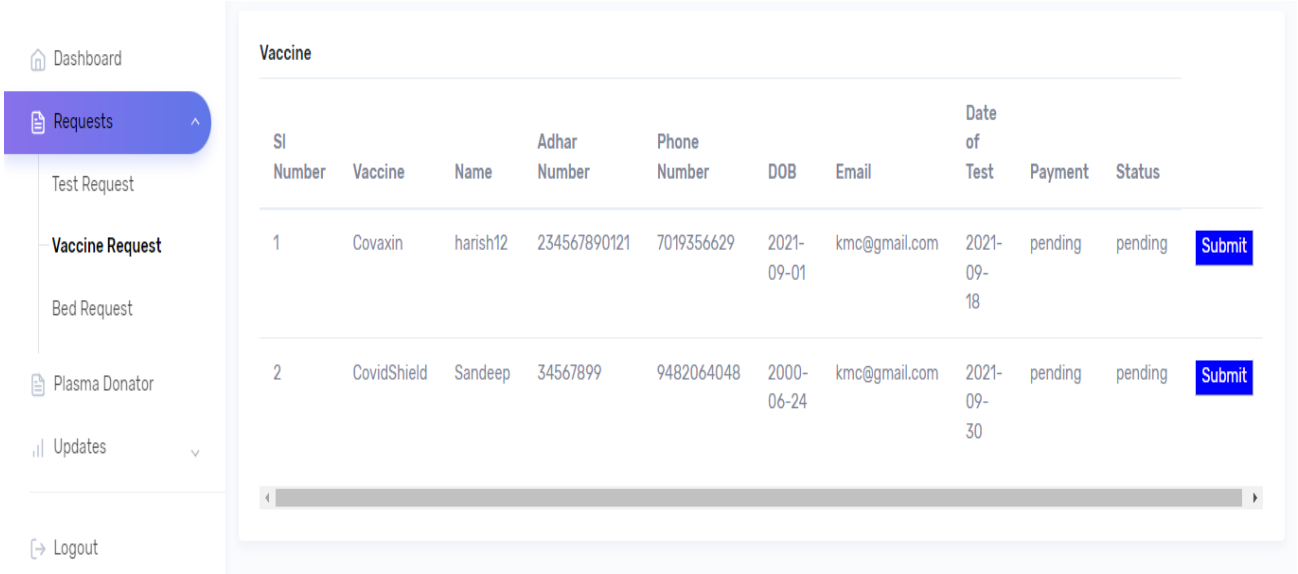


Home-Page

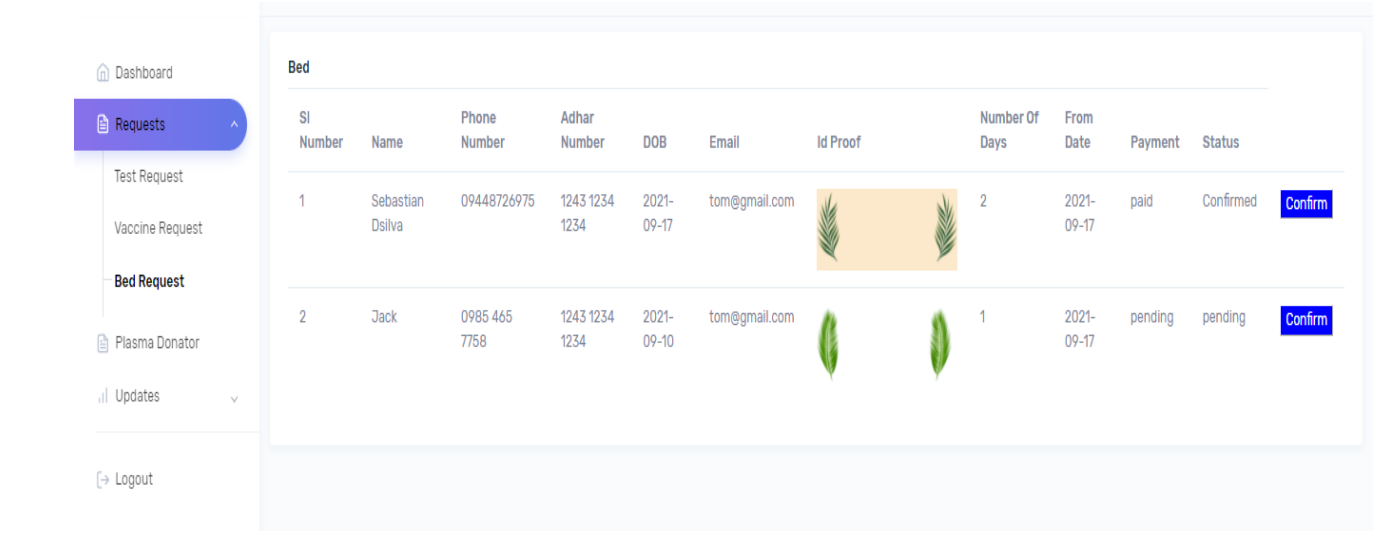
Test-Request



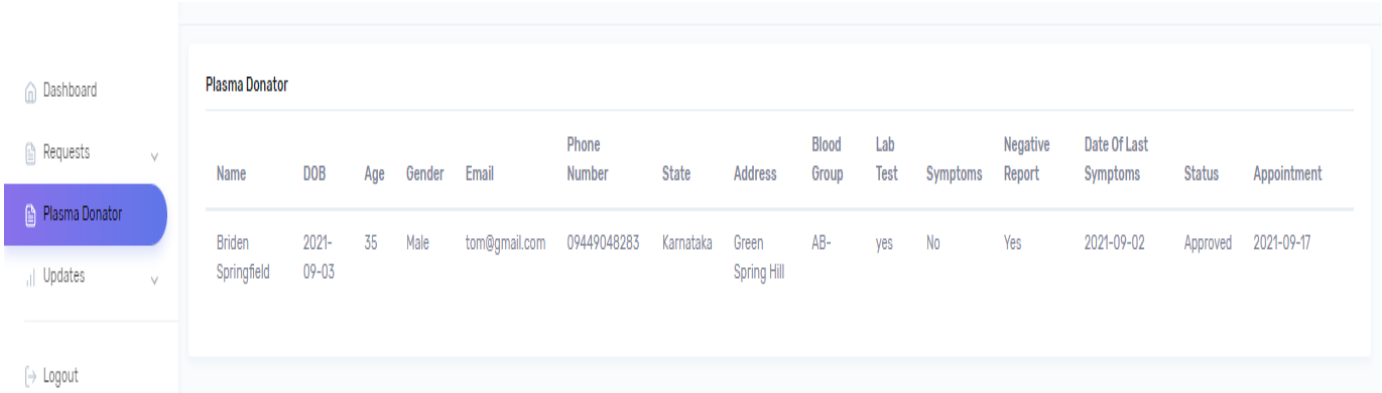
Vaccine-Request



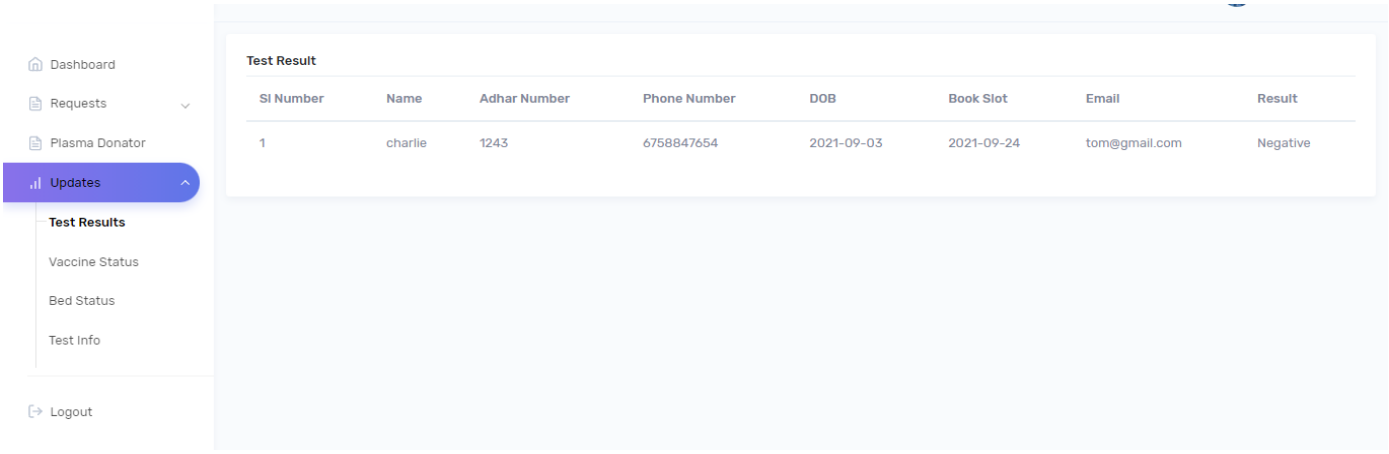
Bed-Request



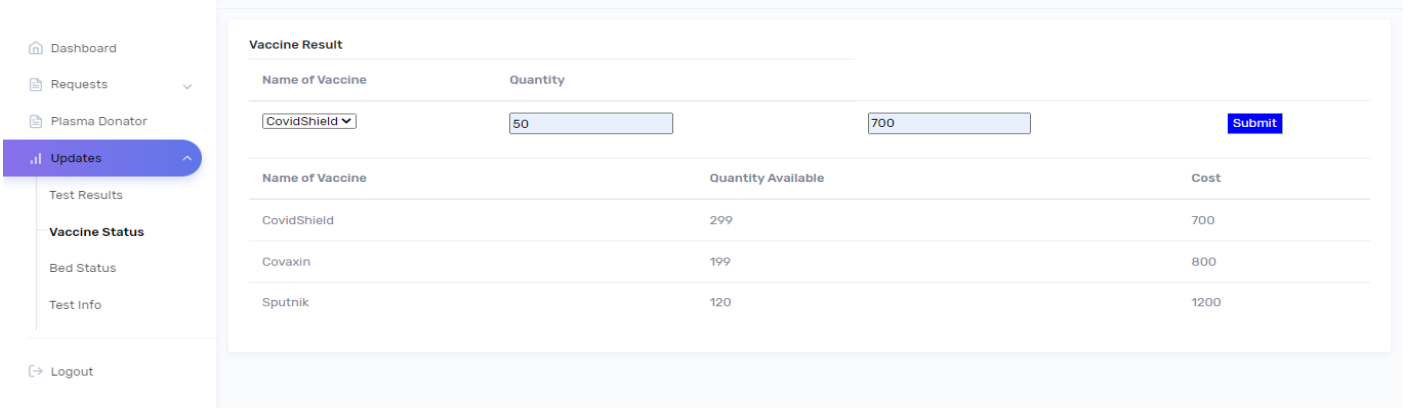
Plasma Donors



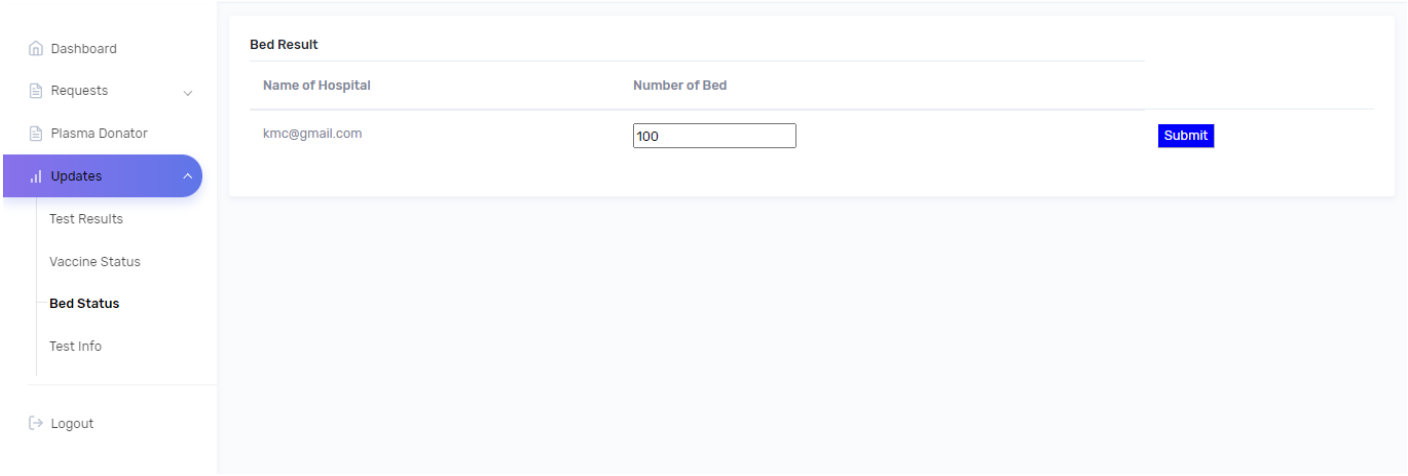
Test Results



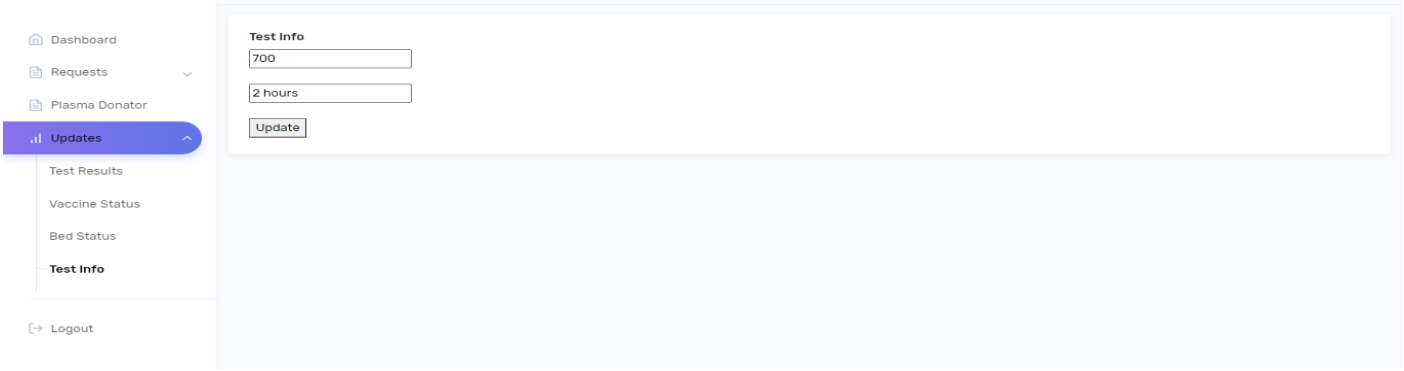
Vaccine Status



Bed Status



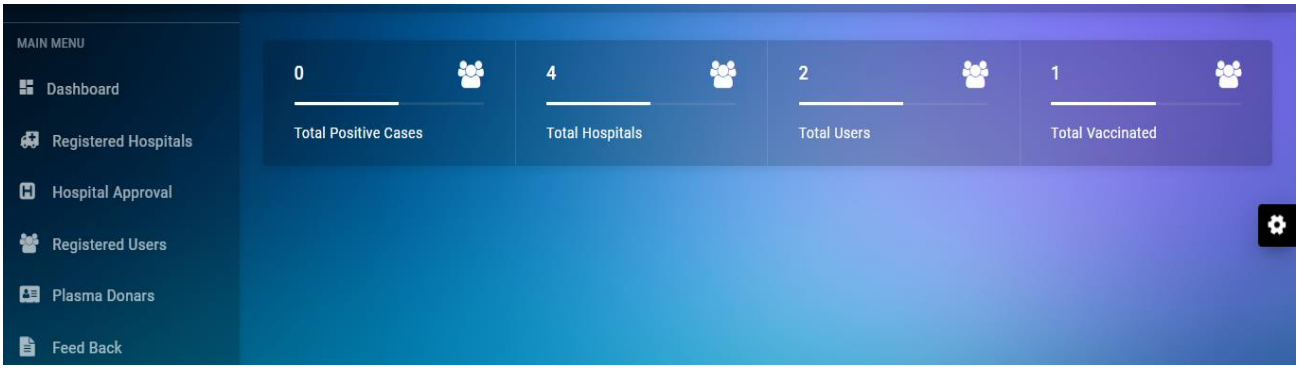
Test Info



Admin

Login

Home Page

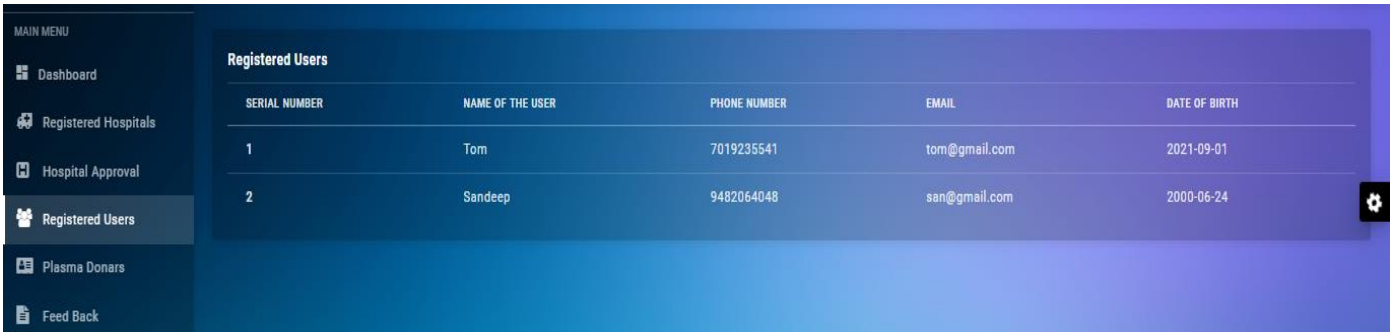


Registered Hospital

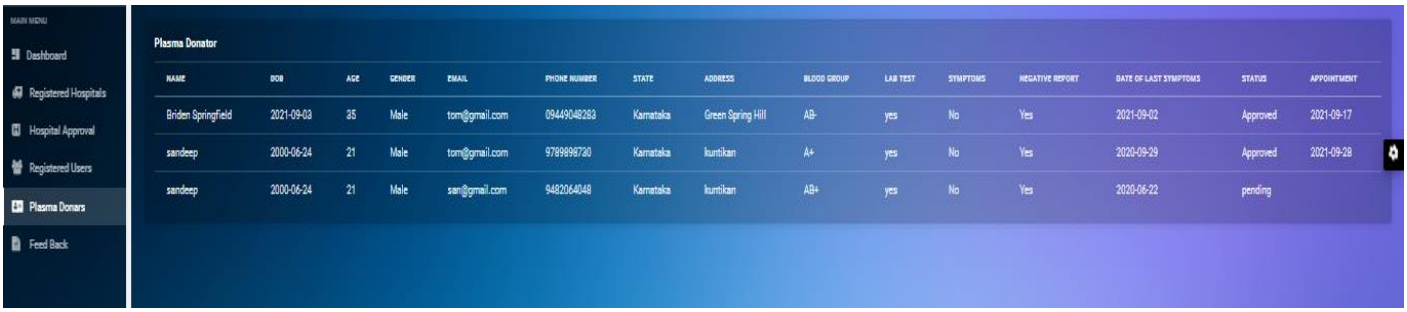
Hospital Approval

Covid – Shield

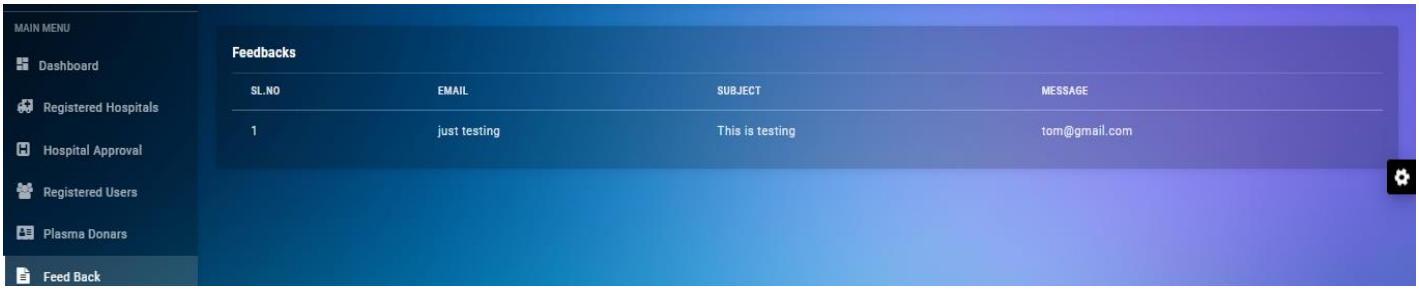
Registered Users



Plasma Donors



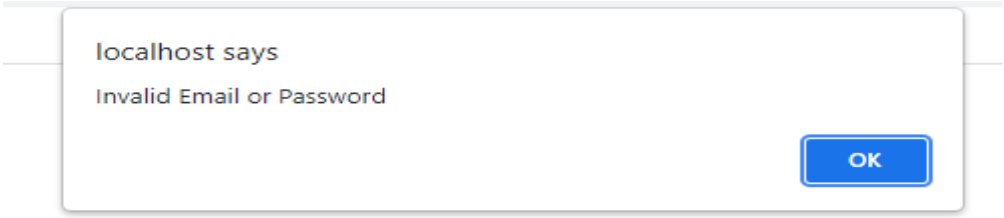
Feedback



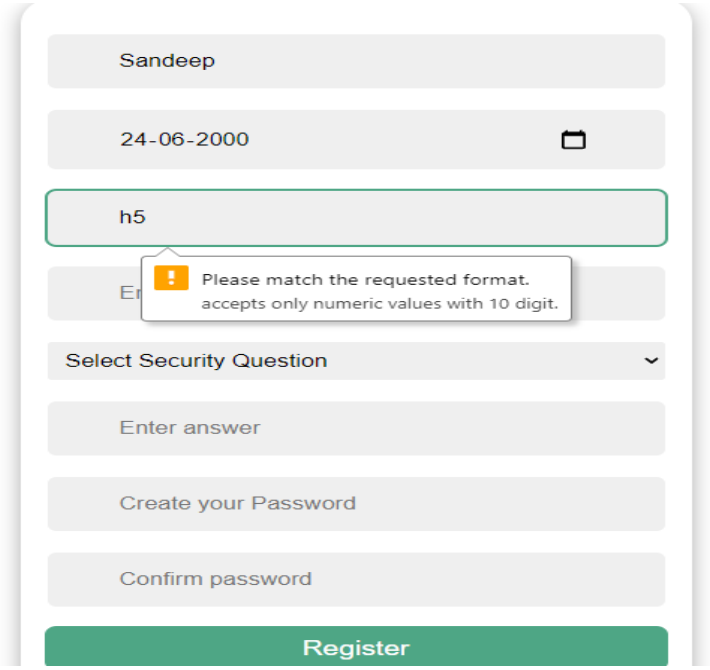
Covid – Shield

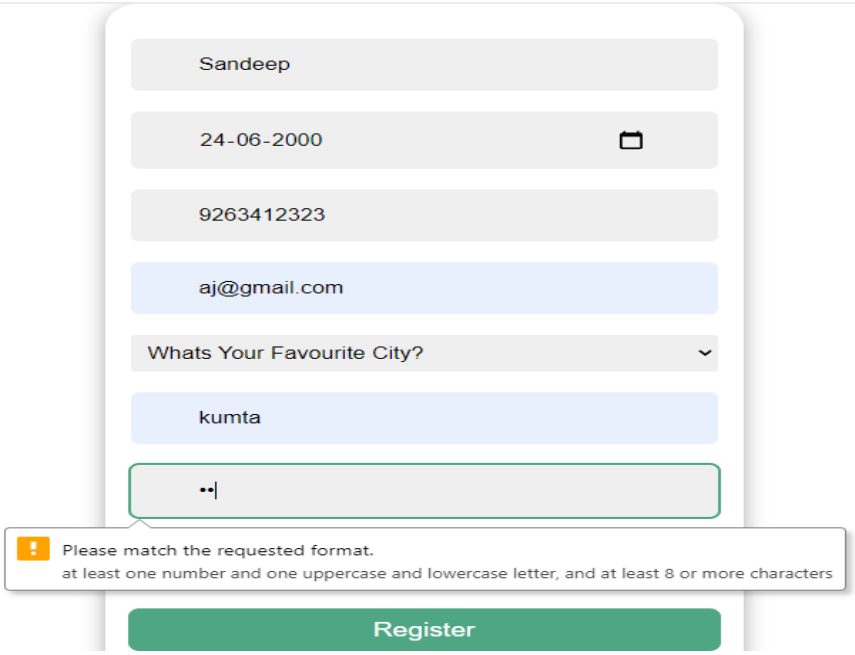
Validation

Authentication

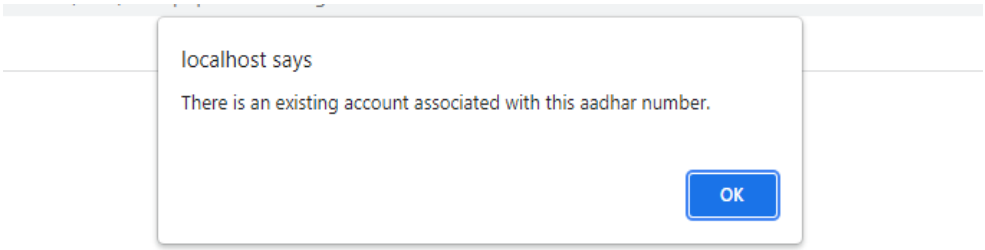


Security

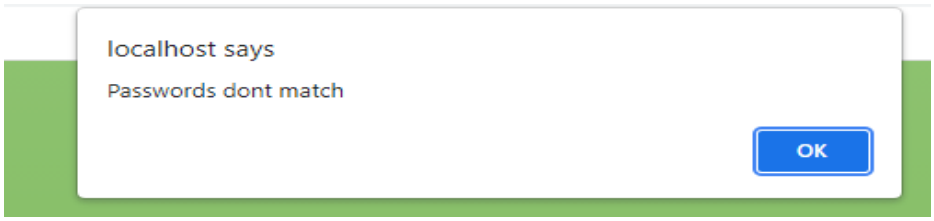




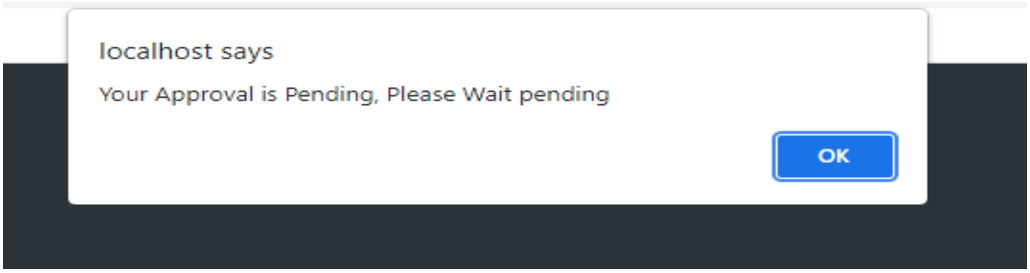
Uniqueness



Cross check



Permission from Admin



"Conclusion and Future Enhancement"

9.1 Conclusion:

- This project satisfies all the requirements and also capable to provide facilities to the users in this critical situation.
- This web-based application helps to reduces the waiting time of the users.
- Irrespective of areas(city/village) this application helps every needy people.

9.2 Future Enhancement

- This application can be used and handled in the future pandemic and health related issues.
- This application will help people save their time because all the facilities related to medical care can be implemented in this one application.
- This application is useful if the community adopts.
- This application is user friendly and reduces the time and human errors.
- In future, this web-based application can be implemented on android based application also.

Bibliography

Websites

- https://www.w3schools.com/
- https://getbootstrap.com/
- https://github.com/
- https://fontawesome.com/