

REDUCING THE EXPOSURE OF NON-COVID PATIENTS

PROJECT REPORT

*Submitted in fulfillment for the J-Component of CSE3002 – Internet and
WebProgramming*

B.Tech. (Computer Science)

By

1. NITISH GUPTA [19BCI0189]
2. RAHUL KUMAR [19BCE0965]

Under the guidance of

Prof. Jayakumar

SadhasivamSCOPE



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Fall Semester 2021-2022

ABSTRACT

The growth of Covid patients in India in second wave was exponential and the no. of deaths was also increasing and there is no way to stop the widespread of the virus other than by staying at home. In the second wave Since, the hospitals were filled with Covid patients thus, a non-COVID patient has a huge exposure to COVID in any hospital, on the other hand as we saw in the second wave, the hospital was full with patients and lack of availability of hospital beds.

The aim of this project is to solve the issue for both sides of the problem by limiting the time, for a patient to stay inside the hospital, to as low as possible

INTRODUCTION

1. The major reason for choosing this topic is upcoming third wave of COVID, many researchers have predicted that a third wave is yet to come and as we have seen in second wave, the situation of our country was not good, there were huge number of deaths and the pressure on hospitals was also very high.
2. Not only the COVID patients, but the non-COVID patients also suffered a lot. The second wave is a huge learning for us and we don't want situation to be this bad in third wave.
3. In this website, patients can login to their account and can directly consult about their problems to management staff. The management staff will then allocate a doctor to the patient based on his/her symptoms.
4. The patient will then message to the doctor and send him all of his/her problems. The doctor may give some medical test for patient.
5. Patient has to go to hospital once and there, staff will take all the requested tests of the patient. As soon as the tests are done the patients are supposed to leave the hospital and the reports of the test will be send to them online by the management, so there is no need for the patient to go to hospital to collect their reports.
6. Once the reports are received by patient, he needs to send it to the doctor. The doctor will go through the report and give him a prescription.
7. In the case of emergency Doctor will report it to the management team and the management team will arrange a room and ask the patient to report to the hospital as soon as possible.
8. If the doctor is not responding to the patient or patient is having any kind of problem with the doctor, he can send his complain to the management team. The management team will have the access to see all the doctor's and patient's page, he can see how the doctor is dealing with a patient and how patient is

communicating with the doctor. . The management team can create, remove or edit any doctor.

9. In this way patient have to go to the hospital for as minimum time as possible and hence it will significantly reduce the exposure of non-COVID patient and there will be a better crowd management in the hospital

PROBLEM STATEMENT

HOSPITAL PROBLEM :

The major problem in the second wave was lack of hospital bed because of which they faced huge problem in giving proper treatment to the patients.

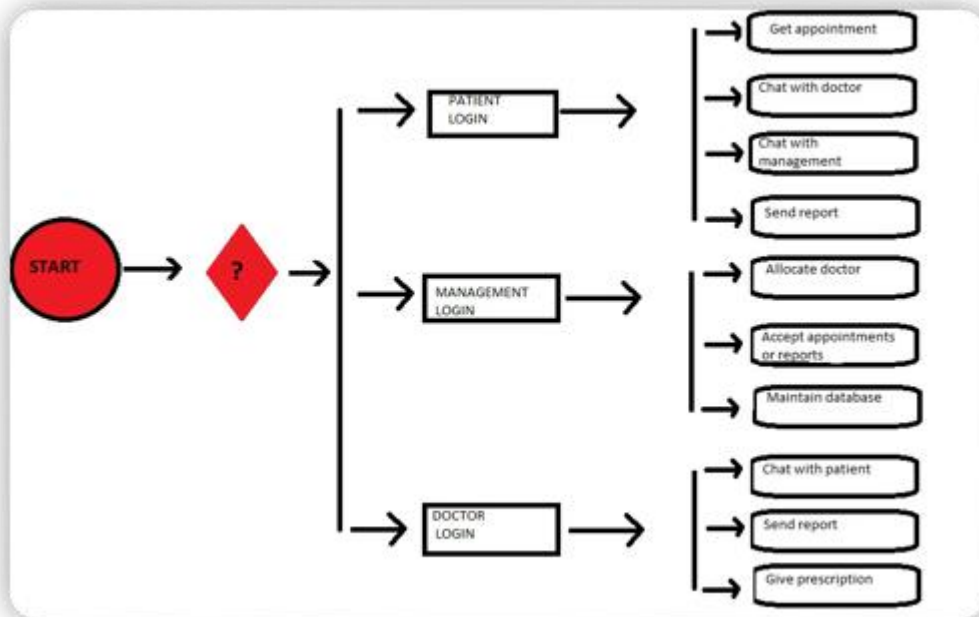
PATIENT PROBLEM :

Most of the hospitals were filled with Covid patients thus, a non-COVID patient has a huge exposure to COVID in any hospital, so the longer they stay at the hospital the higher are the chances for them to catch COVID.

LACK OF DOCTOR'S :

Since the COVID is highly communicable, many of the doctors catch COVID while treating them as a result of which we have seen that many doctors were avoiding to treat patients for their safety. Due to which there was lack of doctors as well.

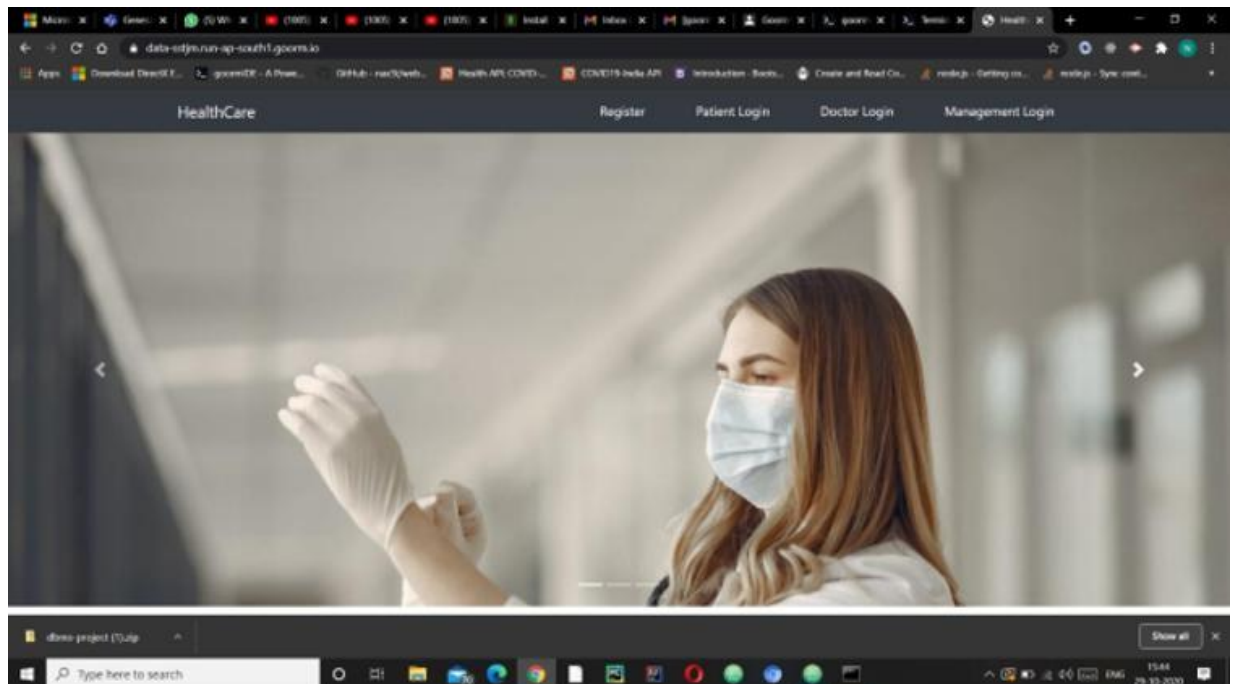
PROCESS FLOW DIAGRAM



MODULES

- Login System for patients, Doctor and Management.
- Patient can chat with Management Team and tell him his problems.
- Management Team allocates patient a doctor.
- Patient can chat with doctor and tell him all his problems in detail in chat
- Doctor can respond to all the queries of the patients and can ask them for all the tests that he needs to analyze the problem

PROTOTYPE DESIGN

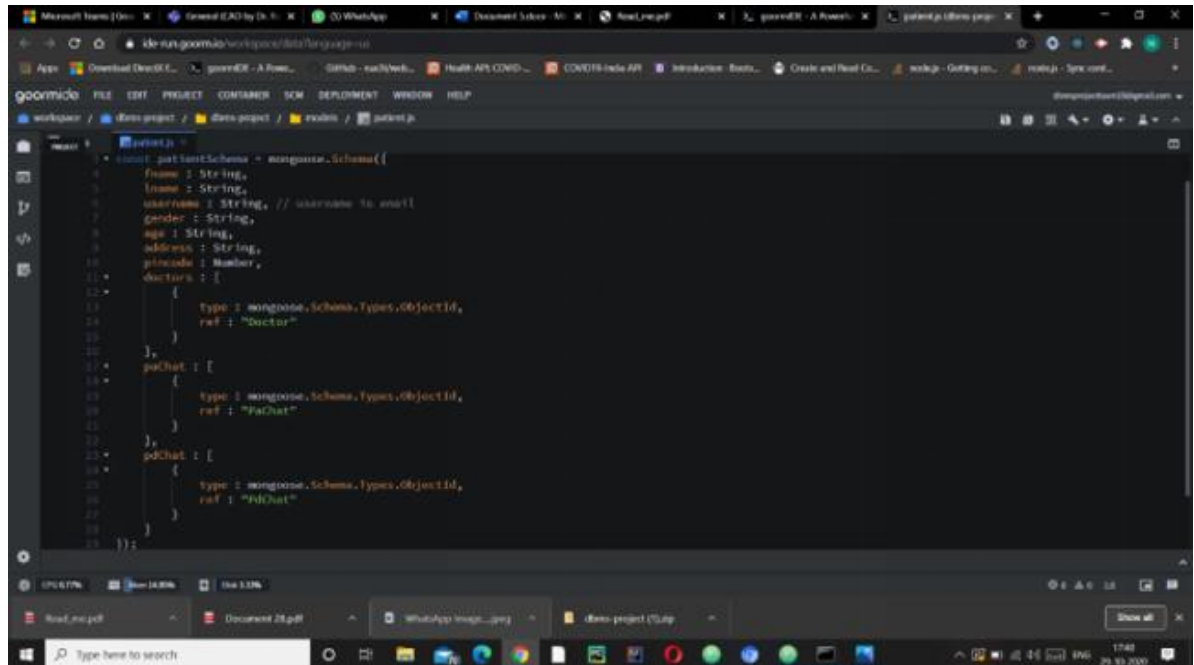


DATABASE DESIGN

User Database

```
4 * const userSchema = mongoose.Schema({
5   username : String,
6   password : String,
7   isPatient : {
8     type : Boolean,
9     default : false
10  },
11  isDoctor : {
12    type : Boolean,
13    default : false
14  },
15  isAdmin : {
16    type : Boolean,
17    default : false
18  }
19 });
```

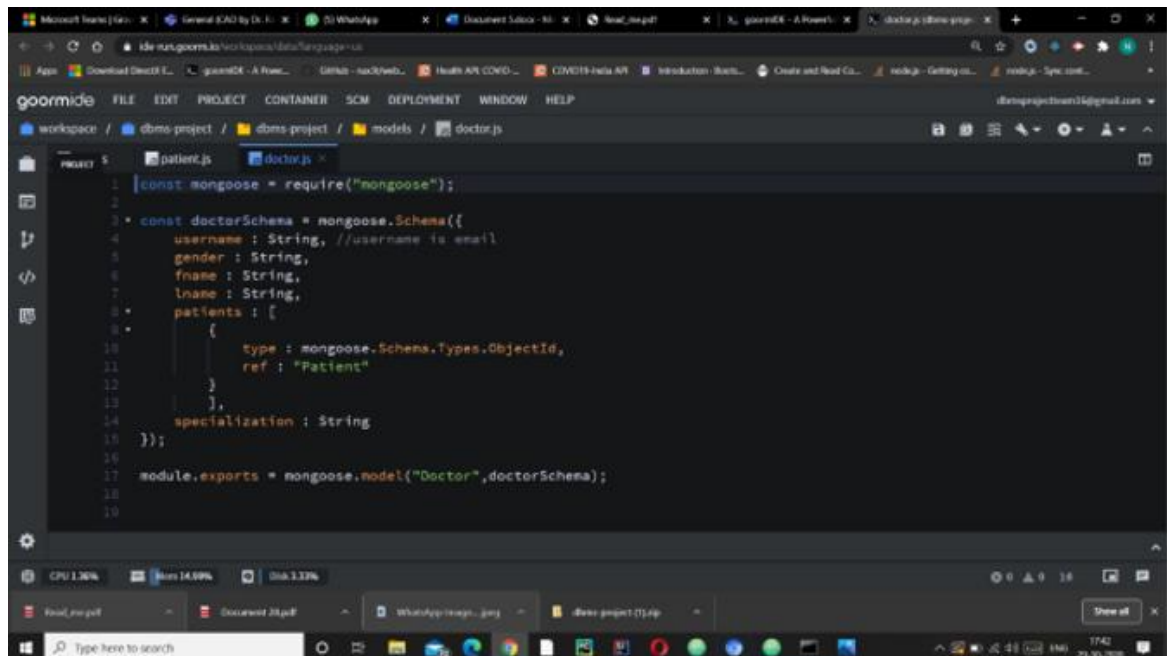
PATIENT DATABASE



The screenshot shows a VS Code editor window with the file `patient.js` open. The code defines a Mongoose schema for a patient, including fields for name, username, gender, age, address, pincode, and a nested array of doctors. It also includes a `paChat` array and a `pdChat` array, both referencing a `PaChat` model. The schema is named `patientSchema` and is exported as `patient`.

```
1 const patientSchema = mongoose.Schema({
2   name: String,
3   lname: String,
4   username: String, // username is email
5   gender: String,
6   age: String,
7   address: String,
8   pincode: Number,
9   doctors: [
10    {
11      type: mongoose.Schema.Types.ObjectId,
12      ref: "Doctor"
13    }
14  ],
15   paChat: [
16    {
17      type: mongoose.Schema.Types.ObjectId,
18      ref: "PaChat"
19    }
20  ],
21   pdChat: [
22    {
23      type: mongoose.Schema.Types.ObjectId,
24      ref: "PdChat"
25    }
26  ]
27 });
28 module.exports = patientSchema;
```

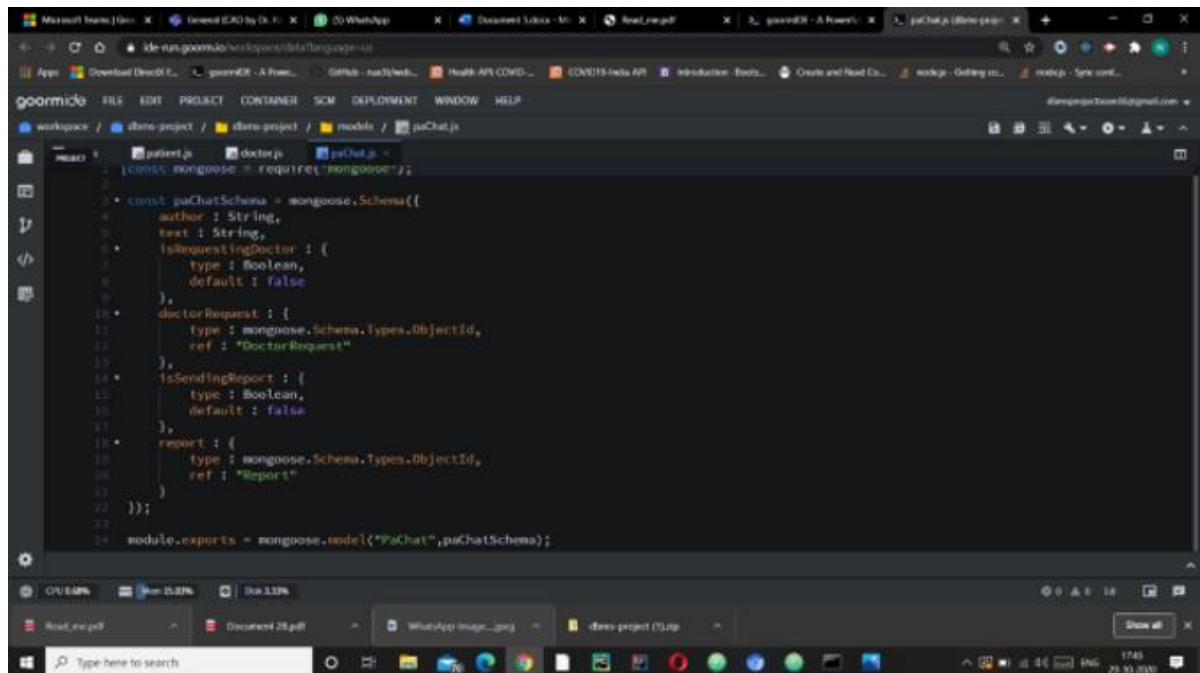
Doctor Database



The screenshot shows a VS Code editor window with the file `doctor.js` open. The code defines a Mongoose schema for a doctor, including fields for username, gender, lname, and a nested array of patients. It also includes a `specialization` field. The schema is named `doctorSchema` and is exported as `doctor`.

```
1 const mongoose = require("mongoose");
2
3 const doctorSchema = mongoose.Schema({
4   username: String, //username is email
5   gender: String,
6   lname: String,
7   patients: [
8     {
9       type: mongoose.Schema.Types.ObjectId,
10      ref: "Patient"
11     }
12  ],
13   specialization: String
14 });
15 module.exports = mongoose.model("Doctor", doctorSchema);
```

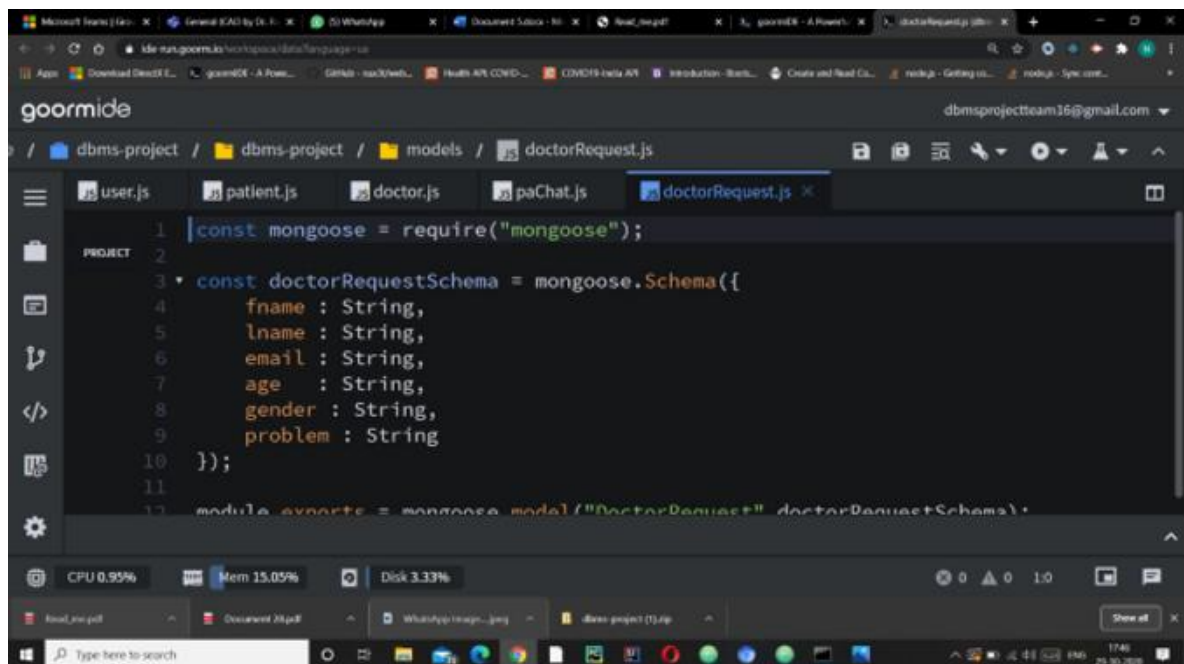
PATIENT-ADMIN CHAT DATABASE (PACHAT)



The screenshot shows the Goormide IDE with the file `patient.js` open. The code defines a Mongoose schema for the PACHAT database. The schema includes fields for `author` (String), `text` (String), `isRequestingDoctor` (Boolean, default: false), `doctorRequest` (Mongoose Schema.Types.ObjectId, ref: "DoctorRequest"), `isSendingReport` (Boolean, default: false), and `report` (Mongoose Schema.Types.ObjectId, ref: "Report"). The schema is named `paChatSchema` and the model is named `PaChat`.

```
1 const mongoose = require("mongoose");
2
3 const paChatSchema = mongoose.Schema({
4   author : String,
5   text : String,
6   isRequestingDoctor : {
7     type : Boolean,
8     default : false
9   },
10  doctorRequest : {
11    type : mongoose.Schema.Types.ObjectId,
12    ref : "DoctorRequest"
13  },
14  isSendingReport : {
15    type : Boolean,
16    default : false
17  },
18  report : {
19    type : mongoose.Schema.Types.ObjectId,
20    ref : "Report"
21  }
22 });
23
24 module.exports = mongoose.model("PaChat", paChatSchema);
```

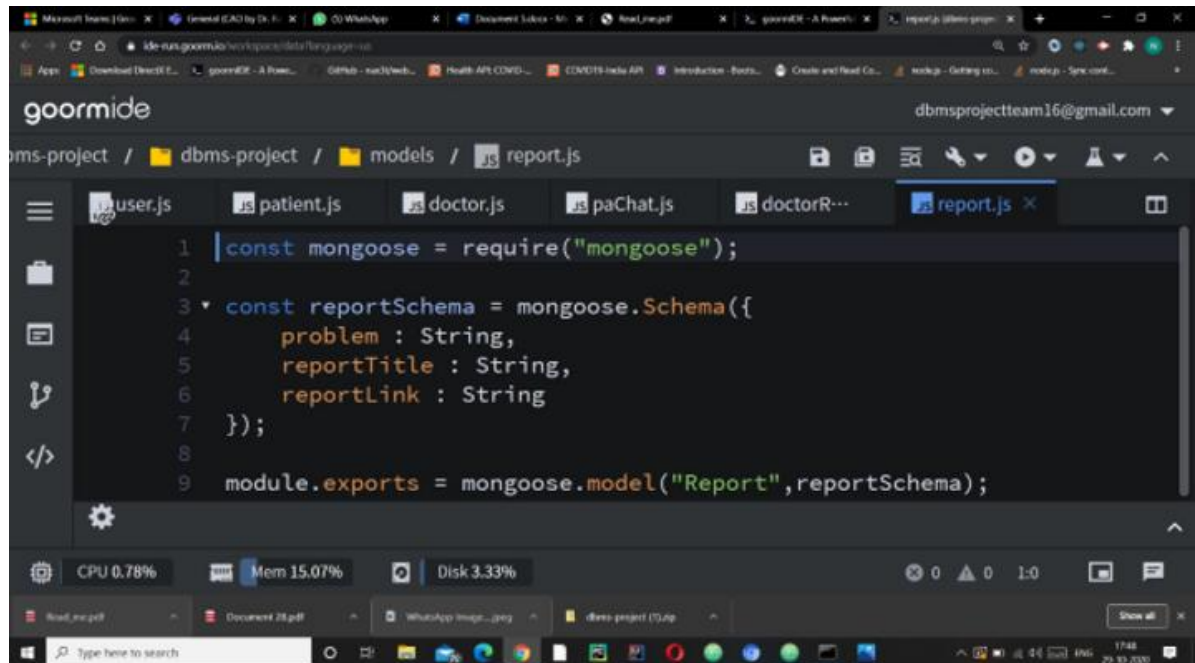
DOCTOR REQUEST DATABASE



The screenshot shows the Goormide IDE with the file `doctorRequest.js` open. The code defines a Mongoose schema for the DOCTOR REQUEST database. The schema includes fields for `fname` (String), `lname` (String), `email` (String), `age` (String), `gender` (String), and `problem` (String). The schema is named `doctorRequestSchema` and the model is named `doctorRequest`.

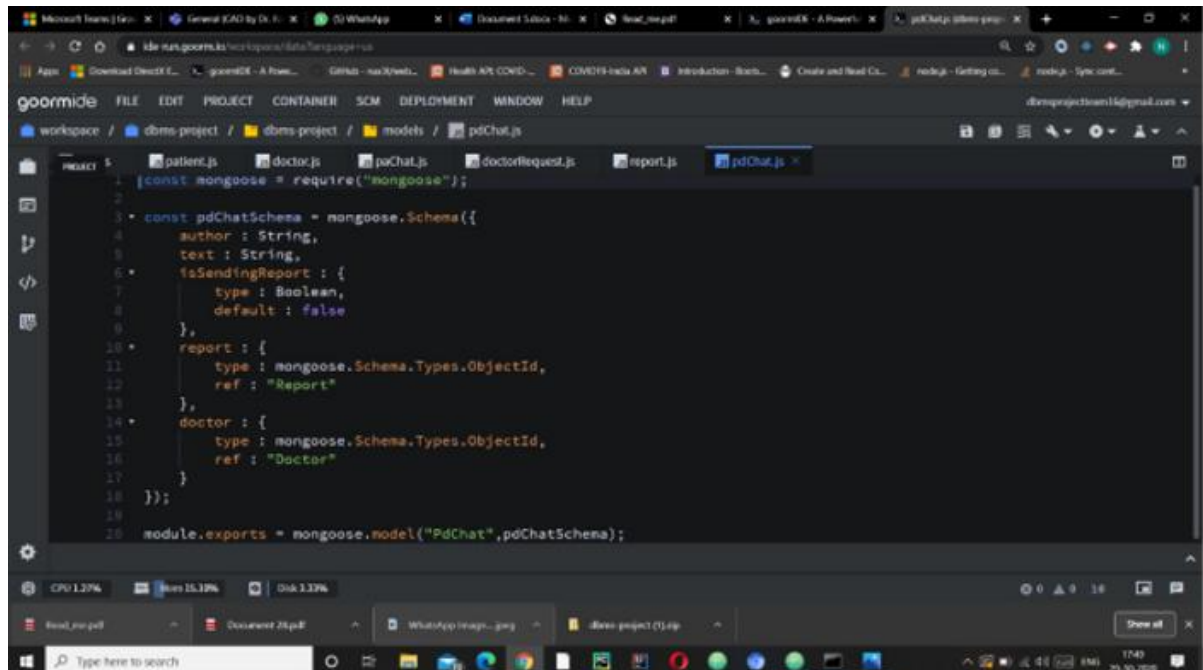
```
1 const mongoose = require("mongoose");
2
3 const doctorRequestSchema = mongoose.Schema({
4   fname : String,
5   lname : String,
6   email : String,
7   age : String,
8   gender : String,
9   problem : String
10 });
11
12 module.exports = mongoose.model("DoctorRequest", doctorRequestSchema);
```


REPORT DATABASE



```
1 const mongoose = require("mongoose");
2
3 const reportSchema = mongoose.Schema({
4   problem : String,
5   reportTitle : String,
6   reportLink : String
7 });
8
9 module.exports = mongoose.model("Report",reportSchema);
```

PATIENT-DOCTOR CHAT DATABASE (PDCHAT)



```
1 const mongoose = require("mongoose");
2
3 const pdChatSchema = mongoose.Schema({
4   author : String,
5   text : String,
6   isSendingReport : {
7     type : Boolean,
8     default : false
9   },
10  report : {
11    type : mongoose.Schema.Types.ObjectId,
12    ref : "Report"
13  },
14  doctor : {
15    type : mongoose.Schema.Types.ObjectId,
16    ref : "Doctor"
17  }
18 });
19
20 module.exports = mongoose.model("PdChat",pdChatSchema);
```


CONCLUSION:

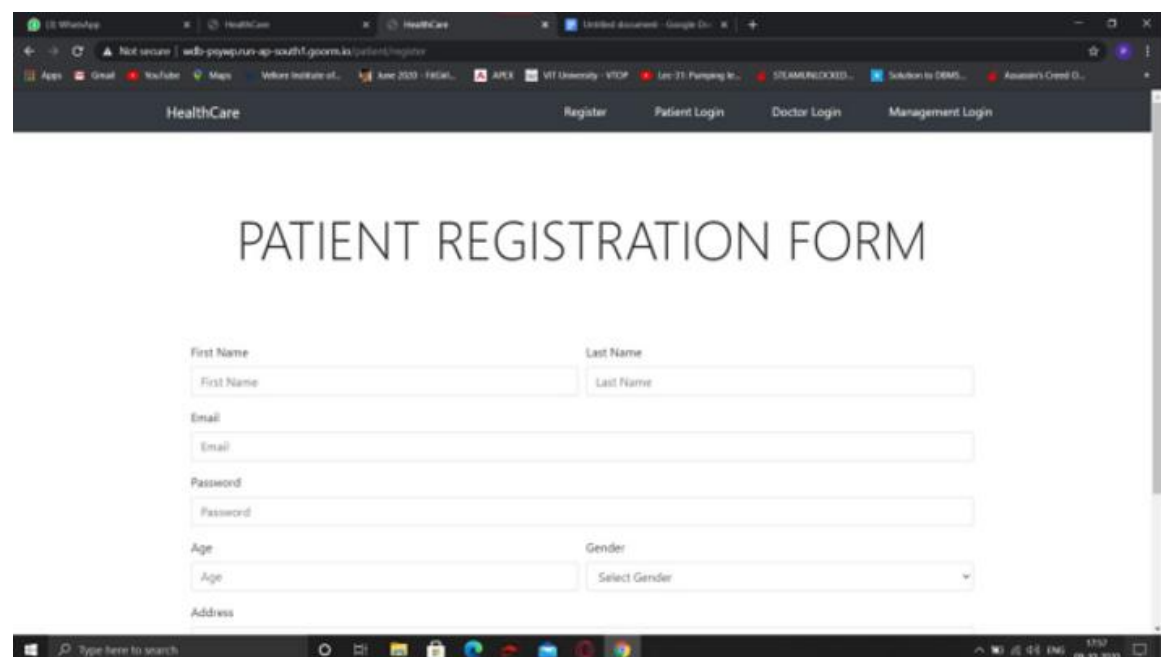
In this website, patients can login to their account and can directly consult about their problems to management staff. The management staff will then allocate a doctor to the patient based on his/her symptoms.

Patient has to go to hospital once and there, staff will take all the requested tests of the patient. As soon as the tests are done the patients are supposed to leave the hospital and the reports of the test will be send to them online by the management, so there is no need for the patient to go to hospital to collect their reports.

REFERENCES:

<https://www.w3schools.com/html/>
<https://docs.mongodb.com/manual/tutorial/>
<https://nodejs.dev/learn>
<http://www.passportjs.org/docs/>
<https://getbootstrap.com/docs/4.0/getting-started/introduction/>
<https://www.w3schools.com/css/>

SNAPSHOTS:



The screenshot shows a web browser window displaying a "PATIENT REGISTRATION FORM". The browser's address bar shows the URL "http://www.pyppurup-ap-south1.goorm.io/patient/register". The page has a dark header with the "HealthCare" logo and navigation links: "Register", "Patient Login", "Doctor Login", and "Management Login". The form itself is white and contains the following fields: "First Name" and "Last Name" (text inputs), "Email" (text input), "Password" (text input), "Age" (text input), "Gender" (a dropdown menu with "Select Gender" as the placeholder), and "Address" (text input). The Windows taskbar is visible at the bottom of the screen.

WhatsApp HealthCare HealthCare Untitled document - Google Docs

Not secure | wdr-pywpur-ap-south-1-gooms-13.patient/register

Apps Gmail YouTube Maps Vellore Institute of Technology June 2020 - FACET... APCL VIT University - VTOP Lec 31: Pumping It... STEAMUNLOCKED... Solution to DBMS... Assassin's Creed O...

First Name Last Name

First Name Last Name

Email

Email

Password

Password

Age Gender

Age Select Gender

Address

Appartment, floor or landmark

Pincode

6-digit Pincode

Submit

Type here to search

WhatsApp HealthCare HealthCare Untitled document - Google Docs

Not secure | wdr-pywpur-ap-south-1-gooms-13.patient/register

Apps Gmail YouTube Maps Vellore Institute of Technology June 2020 - FACET... APCL VIT University - VTOP Lec 31: Pumping It... STEAMUNLOCKED... Solution to DBMS... Assassin's Creed O...

HealthCare Register Patient Login Doctor Login Management Login

PATIENT LOGIN

Email address

modrocks@gmail.com

We'll never share your email with anyone else.

Password

Submit

Untitled document...pdf

Type here to search

Show all

18:16 09-11-2020

How to extract... How to Extract To... Kik Start - Google... A.com/madison... Course 1 - Part 4... WhatsApp... HealthCare... TensorFlow in Py...

Not secure | web-pywp-run-ap-south1.gloom.io/patient/register/edit

EDIT PATIENT

First Name Last Name

Sathwik Goyal

Email

sathwik2000@gmail.com

Password

Age Gender

20 Male

Address

VIT VELLORE

Pincode

Type here to search

3:00 PM 09-10-2020

How to extract... How to Extract To... Kik Start - Google... A.com/madison... Course 1 - Part 4... WhatsApp... HealthCare... TensorFlow in Py...

Not secure | web-pywp-run-ap-south1.gloom.io/doctor/register

DOCTOR REGISTRATION FORM

First Name Last Name

First Name Last Name

Email

Email

Password

Password

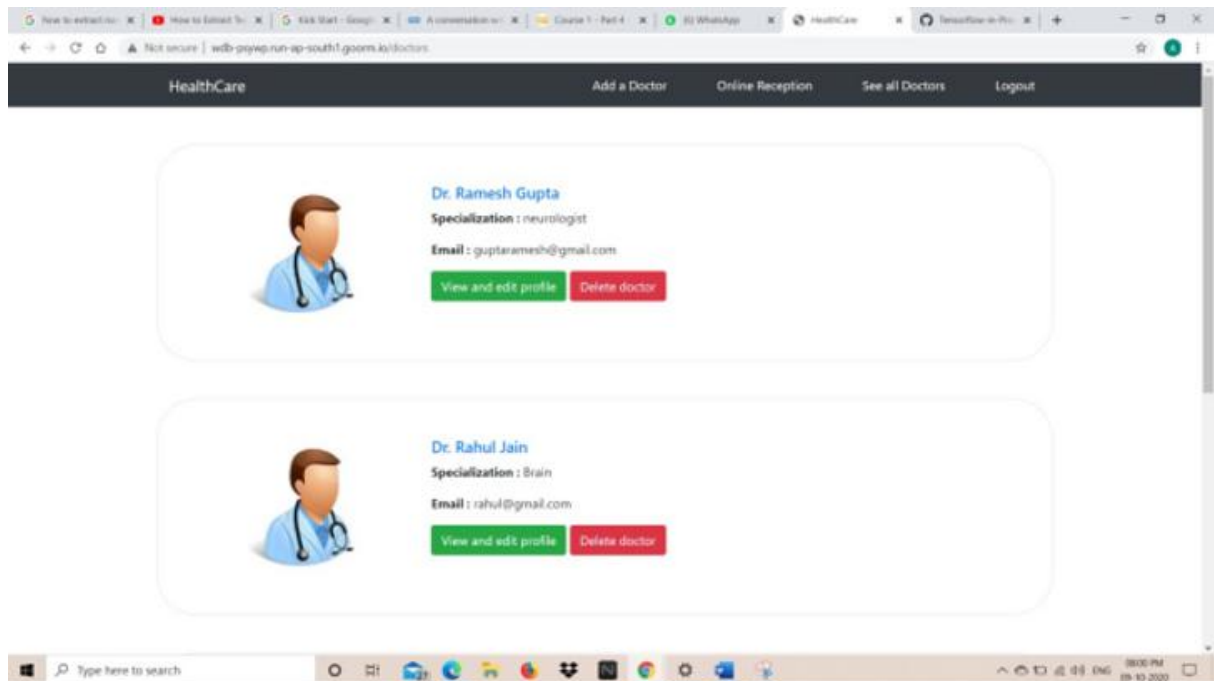
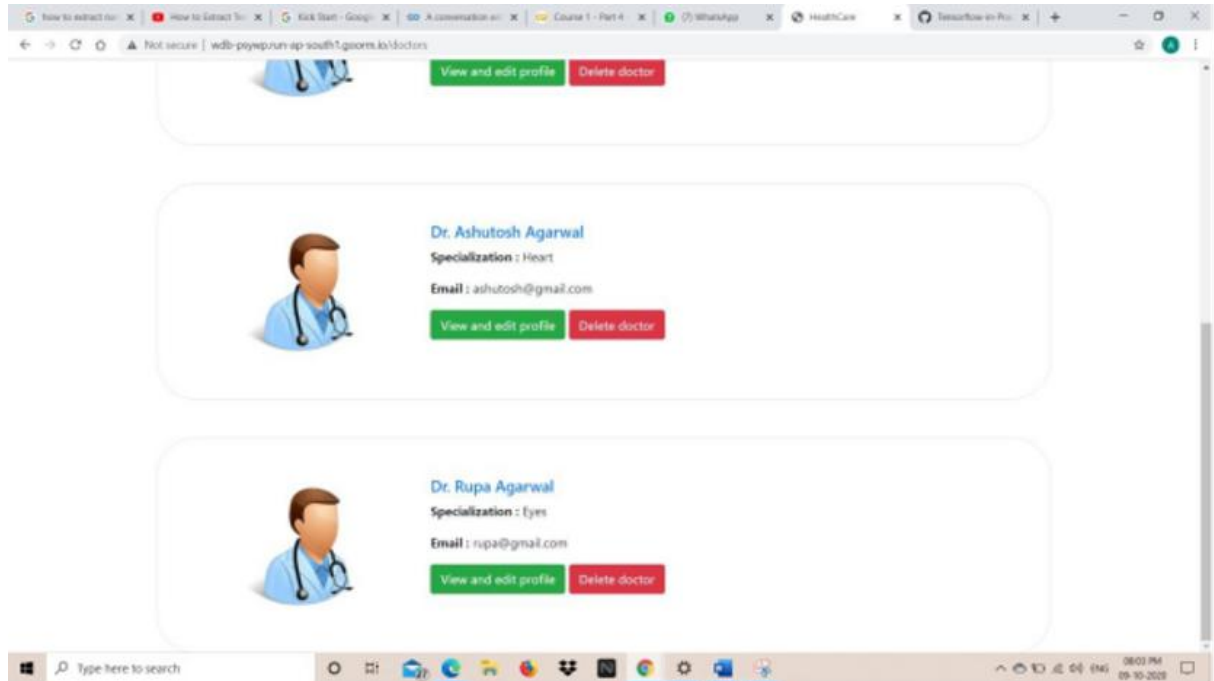
Specialization Gender

Specialization Select Gender

Submit

Type here to search

3:01 PM 09-10-2020



How to extract no... x How to Extract To... x Kick Start - Design... x A conversation co... x Course 1 - Part 4... x WhatsApp x HealthCare x TensorFlow in Pro... x

Not secure | wdb-guywp-run-ap-south1.g0orm.io/Doctors/5f37e5d276281021dfe190/register/edit

EDIT DOCTOR

First Name Last Name

Ramesh Gupta

Email

guptaramesh@gmail.com

Password

Specialization Gender

neurologist Male

Edit

Back

Type here to search

08:01 PM 09-10-2020

How to extract no... x How to Extract To... x Kick Start - Design... x A conversation co... x Course 1 - Part 4... x WhatsApp x HealthCare x TensorFlow in Pro... x

Not secure | wdb-guywp-run-ap-south1.g0orm.io/patient/chat/5854547ad1887223ba76bd

HealthCare Add a Doctor Online Reception See all Doctors Logout

ONLINE RECEPTION

PATIENT
Doctor Request
Name : Satwik Goyal
E-mail : satwik2000@gmail.com
Age : 20
Gender : Male
Problem : headache and cold
Allocate doctor

MANAGEMENT
Dr Ramesh Gupta has been allotted to you. Discuss your problem with the allocated doctor

PATIENT
Doctor Request

Type here to search

08:09 PM 09-10-2020

HealthCare

Allocated Doctors Ask for a Doctor Online Reception View and Edit profile Logout

ASK FOR A DOCTOR

Email address

modrocks@gmail.com

Problem

Write your problem or symptoms here in detail.

Submit

HealthCare

Allocated Doctors Ask for a Doctor Online Reception View and Edit profile Logout

ASK FOR A DOCTOR

Email address

modrocks@gmail.com

Problem

I have serious Head ache from past 4 days

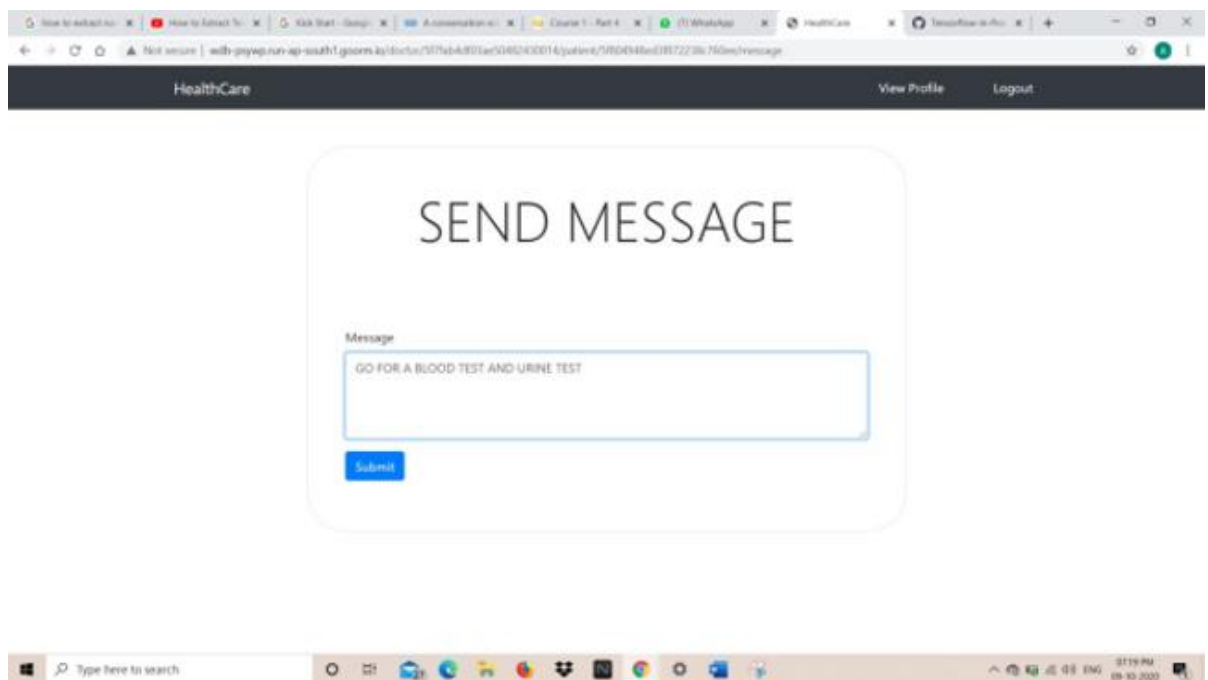
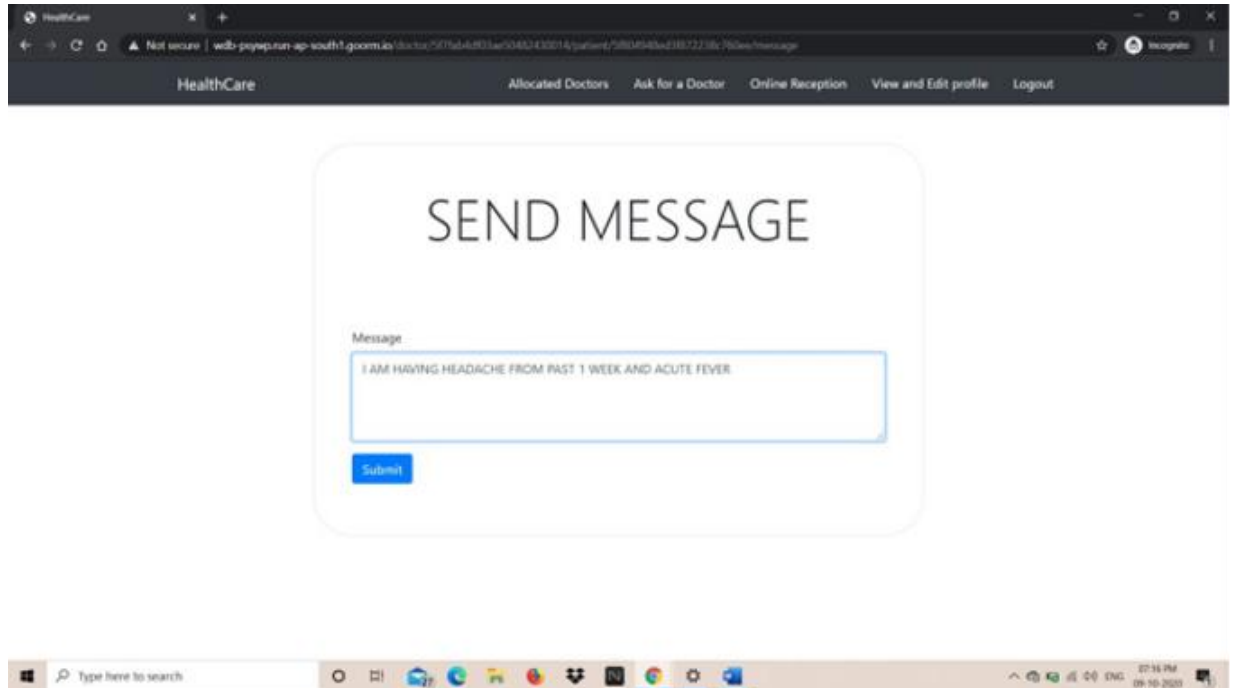
Submit

Untitled document...pdf

Type here to search

Show all

18:55 09-10-2020



MANAGEMENT

Dr Ashutosh Agarwal has been allotted to you. Discuss your problem with the allocated doctor

PATIENT

when will i get my blood report

MANAGEMENT

it will 1 week

HealthCare

MANAGEMENT

Report

Problem : heart problem

Title : blood test

Medical Report

Name : _____ Date : _____

When did your problem start? : _____ Describe Problem : _____

Cause of Current Problem :

☐ Car Accident ☐ Work Injury ☐ Gradual onset ☐ Other

Did this Problem require Surgery : ☐ No ☐ Yes ☐ No Date of Surgery

Past Medical History Do you have a history of the following problems?

<input type="checkbox"/> Breathing Problems	<input type="checkbox"/> Stroke	<input type="checkbox"/> Depression
<input type="checkbox"/> Pregnancy	<input type="checkbox"/> Bone/Joint Problems	<input type="checkbox"/> Bowel/Bladder
<input type="checkbox"/> Heart Problems	<input type="checkbox"/> Kidney Problems	<input type="checkbox"/> History of heavy alcohol use
<input type="checkbox"/> Current Wound/Skin Problems	<input type="checkbox"/> Gallbladder/Liver	<input type="checkbox"/> Drug use
<input type="checkbox"/> Pacemaker	<input type="checkbox"/> Electrical implants	<input type="checkbox"/> Smoking
<input type="checkbox"/> Tumor/Cancer	<input type="checkbox"/> Anxiety attacks	<input type="checkbox"/> Headaches
<input type="checkbox"/> Diabetes	<input type="checkbox"/> Sleep Apnea	

☐ No Surgeries

Surgeries/Hospitalizations	Year	Complications

☐ No Medication

SOURCE CODE :

https://github.com/Rahul-dotcom-beep/IWP_PROJECT.git