|  |
| --- |
| Name: Hembadura De Silva |
| Student Reference Number:  10707162 |



|  |  |  |
| --- | --- | --- |
| Module Code: SOFT255SL | Module Name: Software Engineering for the Internet using Java | |
| Coursework Title:  Web-based software application for large-scale hospital using JSP, servlets, JDBC and Java beans | | |
| Deadline Date:  28/04/2021 | | Member of staff responsible for coursework:  Kaneeka Vidanage |
| Programme: BSc(Hons) Software Engineering | | |
| Please note that University Academic Regulations are available under Rules and Regulations on the University website [www.plymouth.ac.uk/studenthandbook](http://www.plymouth.ac.uk/studenthandbook). | | |
| Group work: please list all names of all participants formally associated with this work and state whether the work was undertaken alone or as part of a team. Please note you may be required to identify individual responsibility for component parts.   |  |  |  | | --- | --- | --- | | ID Number | Name | Degree Programme | | 10707162 | Hembadura De Silva | BSc (Hons) Software Engineering | | 10707163 | Arumadura De Silva | BSc (Hons) Software Engineering | | 10707347 | PDS Rathnayaka | BSc (Hons) Software Engineering | | 10707151 | WNS Botheju | BSc (Hons) Software Engineering | | 10707245 | Udawaththage Kumara | BSc (Hons) Software Engineering |   ***We confirm that we have read and understood the Plymouth University regulations relating to Assessment Offences and that we are aware of the possible penalties for any breach of these regulations. We confirm that this is the independent work of the group.***  Signed on behalf of the group:  A picture containing text, whiteboard, businesscard  Description automatically generated | | |
| Use of translation software: failure to declare that translation software or a similar writing aid has been used will be treated as an assessment offence.  I \*have used/not used translation software.  If used, please state name of software………………………………………………………………… | | |
| **Overall mark \_\_\_\_\_% Assessors Initials \_\_\_\_\_\_ Date\_\_\_\_\_\_\_\_\_** | | |

\*Please delete as appropriateSci/ps/d:/students/cwkfrontcover/2013/14

1. **INTRODUCTION**
2. **Hospital Management System**

Scenario made after requirement analysis

**HMS28** is a project where we have planned to develop a web-based software application for a large-scale hospital. The software is developed based on the JSP, CSS, servlets, Java beans, JDBC, MySQL database management system technologies and concepts. Hospital consists of several departments and we have made our own assumptions while developing the system. We have developed the system mainly focusing on out-patient management and as further improvements we are hoping to expand the system with in-patient management, pharmacy, and the radiology department. Considering the system requirements, the scenario is as follows.

The system users will be the receptionist, cashier, medical officers (doctors), nurses, laboratory technician, system administrator and patients. System administrator will be given the facility to login to the system, and they are the ones who register the doctors, nurses and other staff users into the system after going through the records in the manual system. Each and every user will get a unique user id. Created user credentials will be given manually to the staff users to ensure the security of the system. The related user is given the facility to change their passwords later, as they prefer. System admin will upload the doctors time schedules to the system as well.

A patient can make an appointment at the reception through a phone call or physically. Receptionist can register the patient by creating the patient’s profile and make the appointment by selecting the doctor and time by checking through the time schedules. Receptionist will get a warning message as the maximum count of appointments per doctor exceeds. An appointment will go through 4 states. Receptionist creates/ updates (If the patient is having a profile already) the profile. The appointment is created. Cashier can view the created appointments. After payment is done, cashier updates it as paid and it will be displayed on doctor’s appointment list when he /she views. By any chance if a doctor cancels an appointment or couldn’t make on time it will be a cancelled appointment and cash will be refunded to the patient at the cashier when they give their reference id. Cashier can view the cash refund requests. After the cashier settles the payment refund, they can remove the requests.

State transition diagram for an appointment:

Created

Paid

Started

Completed

Cancelled

When a patient is about to meet the doctor, they will be issued a token after the payment is done. Doctors can view their appointments. They can view a patient's profile, view and update the patients’ medical records, upload prescriptions where the diagnosis per visit and drugs to be taken are entered. They can request lab reports.

A lab report will go through 4 states.

Requested

Paid

Completed

Results Pending

Cashier can view the lab requests. When the payment is done nurses can view the paid lab requests. After the patients’ samples to be tested are given by the nurses to the lab technicians it will be analyzed and completed lab reports will be uploaded to the database by the lab technician. Then a notification will be sent to the patient. A patient can view their lab report. They can view their profile, prescriptions, medical records as well.

Functions:

Receptionist:

* View Patient Profiles
* Patient Registration - Create Patient Profile
* Update Patient Profile
* Search for a patient (using their id)
* Make an appointment.
* View appointments
* View doctor’s availability

System Admin:

* Register Staff Users ->

-Receptionist

-Doctor

-Nurse

-Laboratory Technician

-Cashier

* Upload work schedule of doctors

Doctor:

* Search for the patient and
* View patient profile
* Upload prescriptions
* View medical records.
* Update medical records.
* View their own appointment list.
* Request lab reports
* Cancel an appointment.

Nurse:

* View “paid” lab requests. (when ref.id is given nurse will take the samples)

Lab Technician:

* Upload the completed lab reports.

(In this function make sure to update the “lab report” table’s state field from paid to completed)

Cashier:

* View lab requests (also update state as “paid”)
* View created appointments (also update state as “Paid”)

Patient:

* View profile
* View medical record.
* View prescriptions.
* View lab report.

User Login:

* System Admin
* Doctor
* Nurse
* Lab technician
* Receptionist
* Cashier
* Patient

user type assigned int -

system admin - 0

doctor - 1

receptionist - 2

nurse - 3

lab tech - 4

cashier - 5

patient - 6

lab report state (int)-

requested - 0

paid - 1

results pending - 3

completed - 4

Appointment state(int assigned) -

created - 0

paid - 1

started - 2

completed - 3

cancelled - 4

1. **DESIGN OF HMS28 DATABASE**
2. ER Diagram

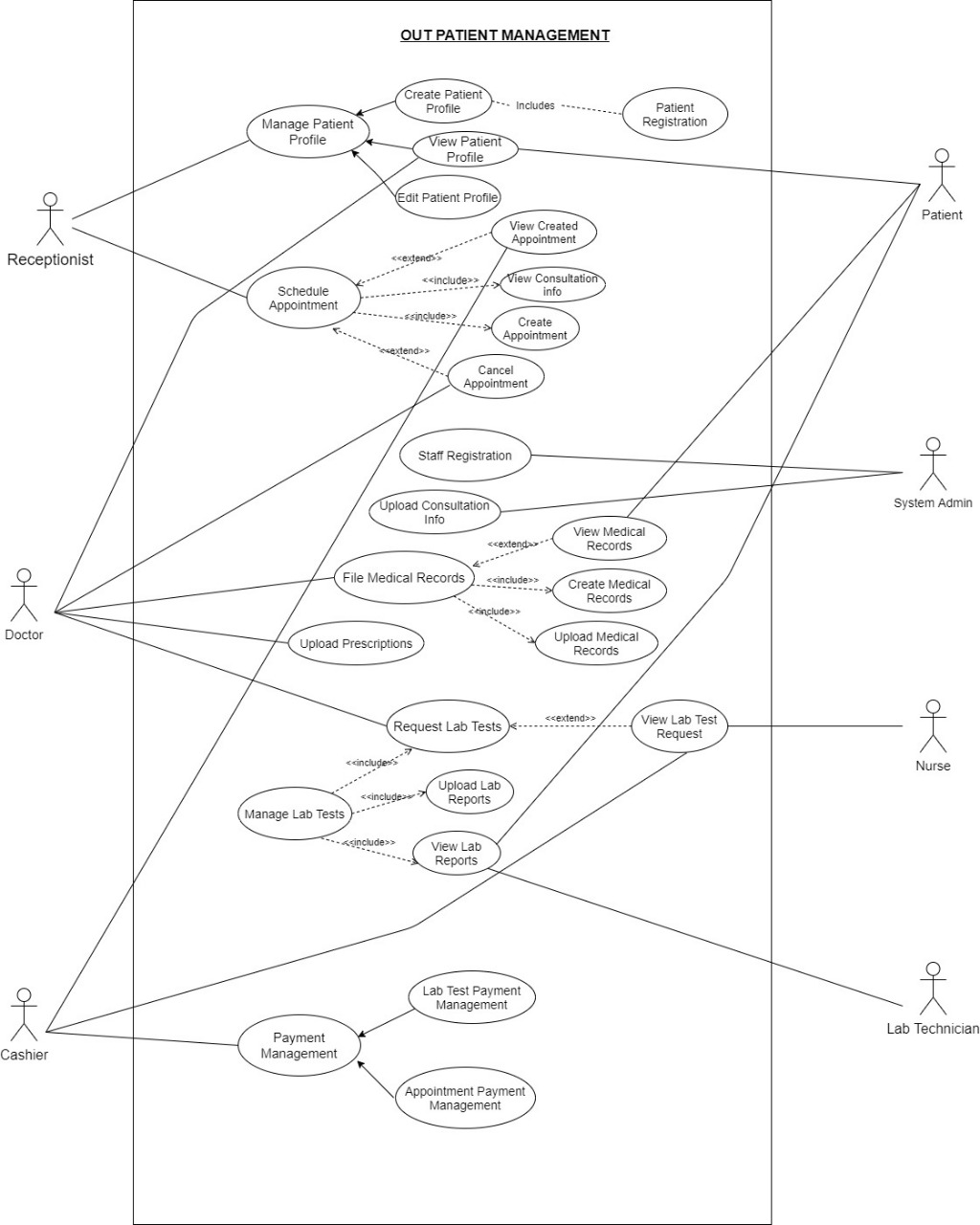
A picture containing diagram

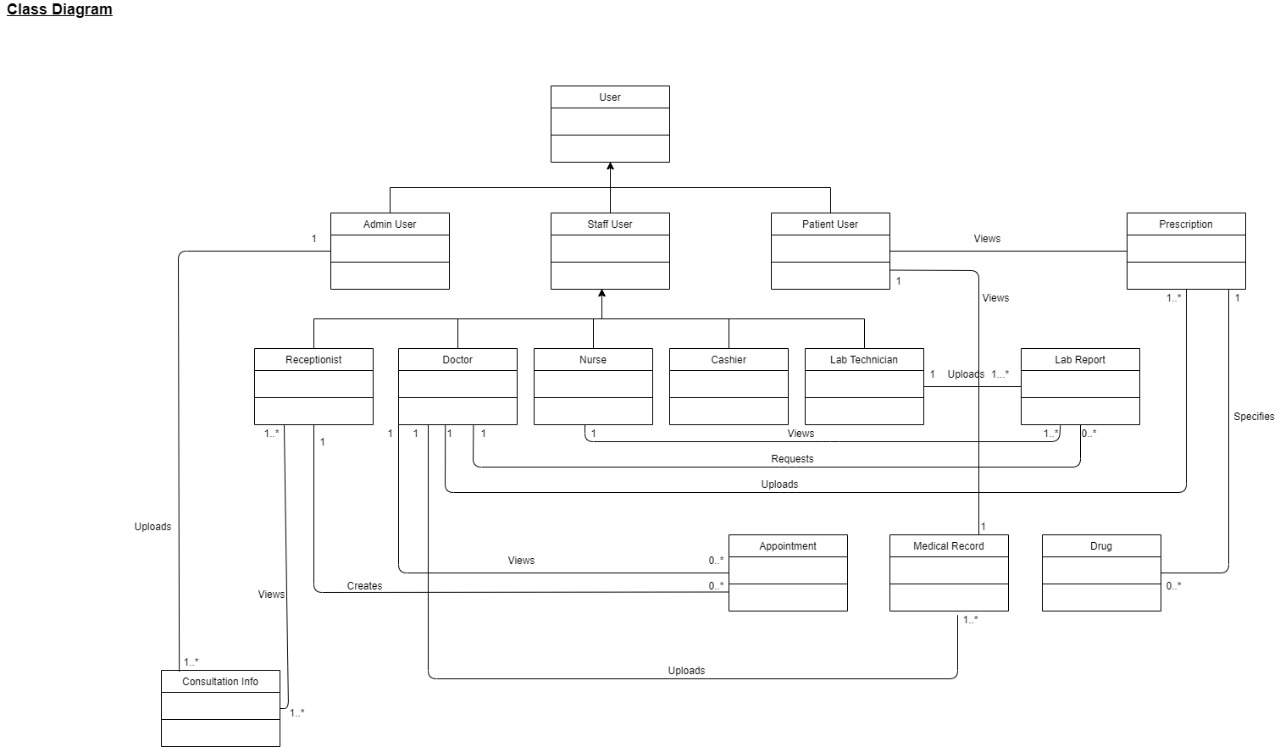
Description automatically generated

**3.UML DIAGRAMS**

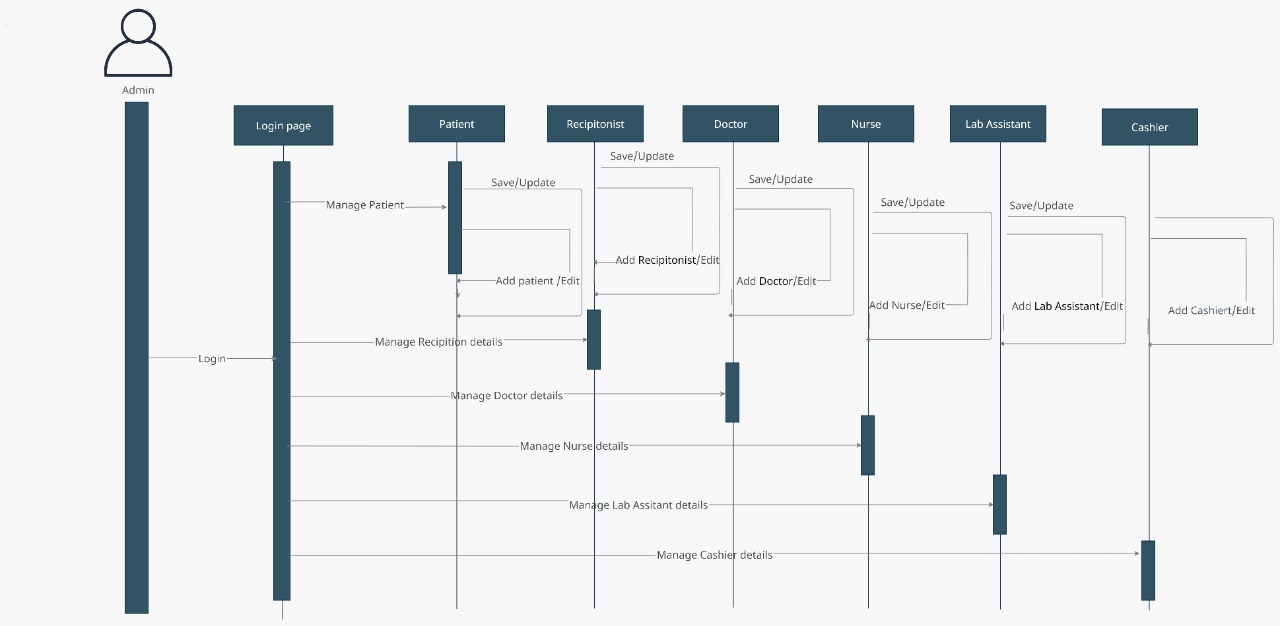
1. USE CASE DIAGRAM

HMS28 Hospital Management System



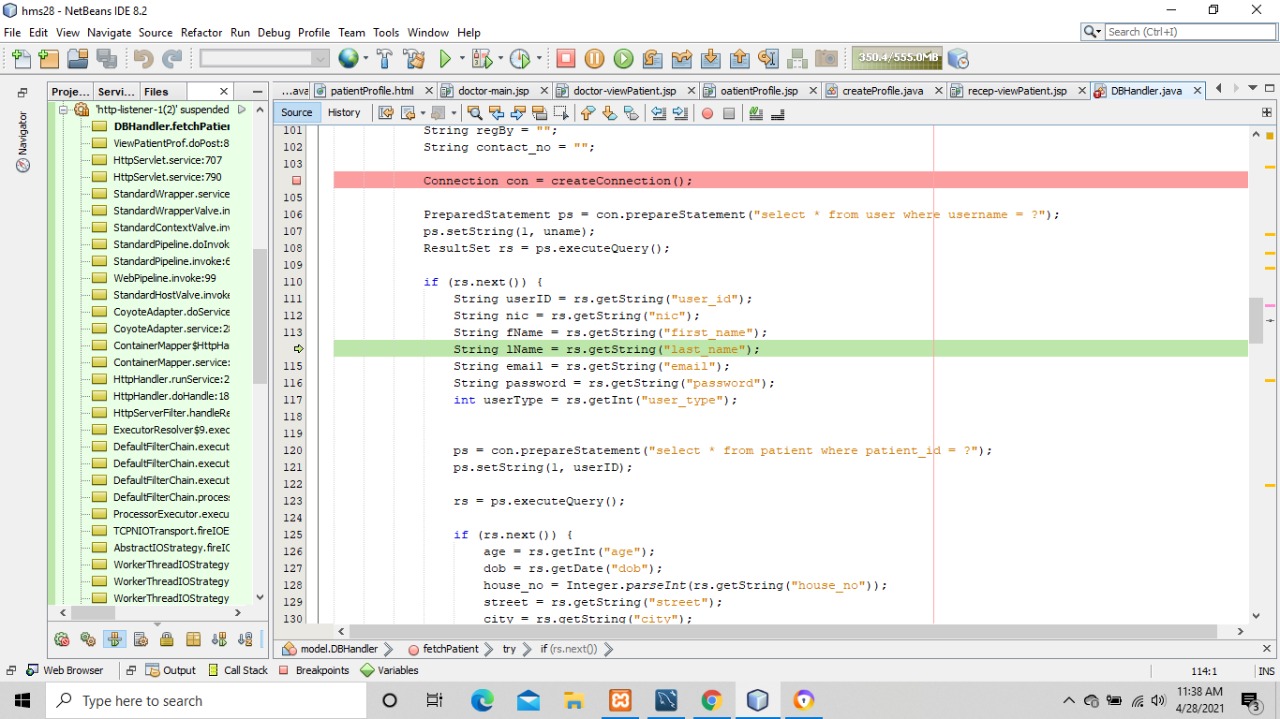
1. CLASS DIAGRAM

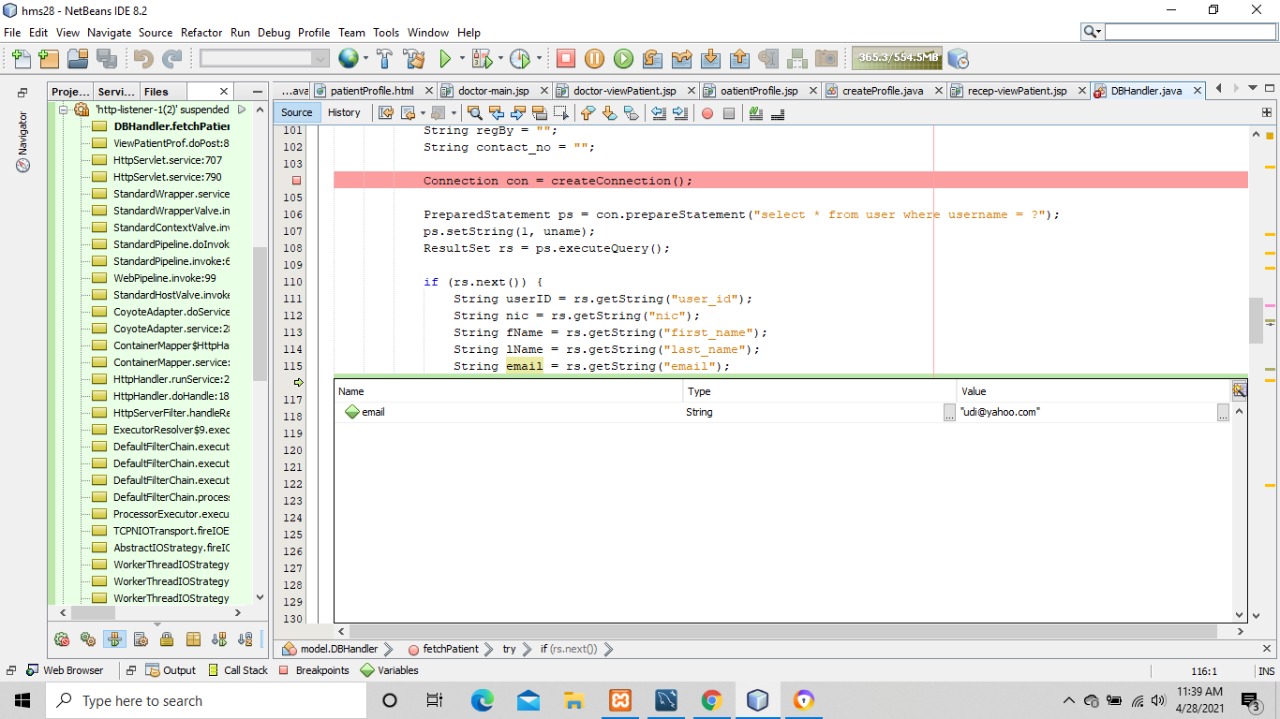
3.Sequence Diagram



**4. Test Cases**

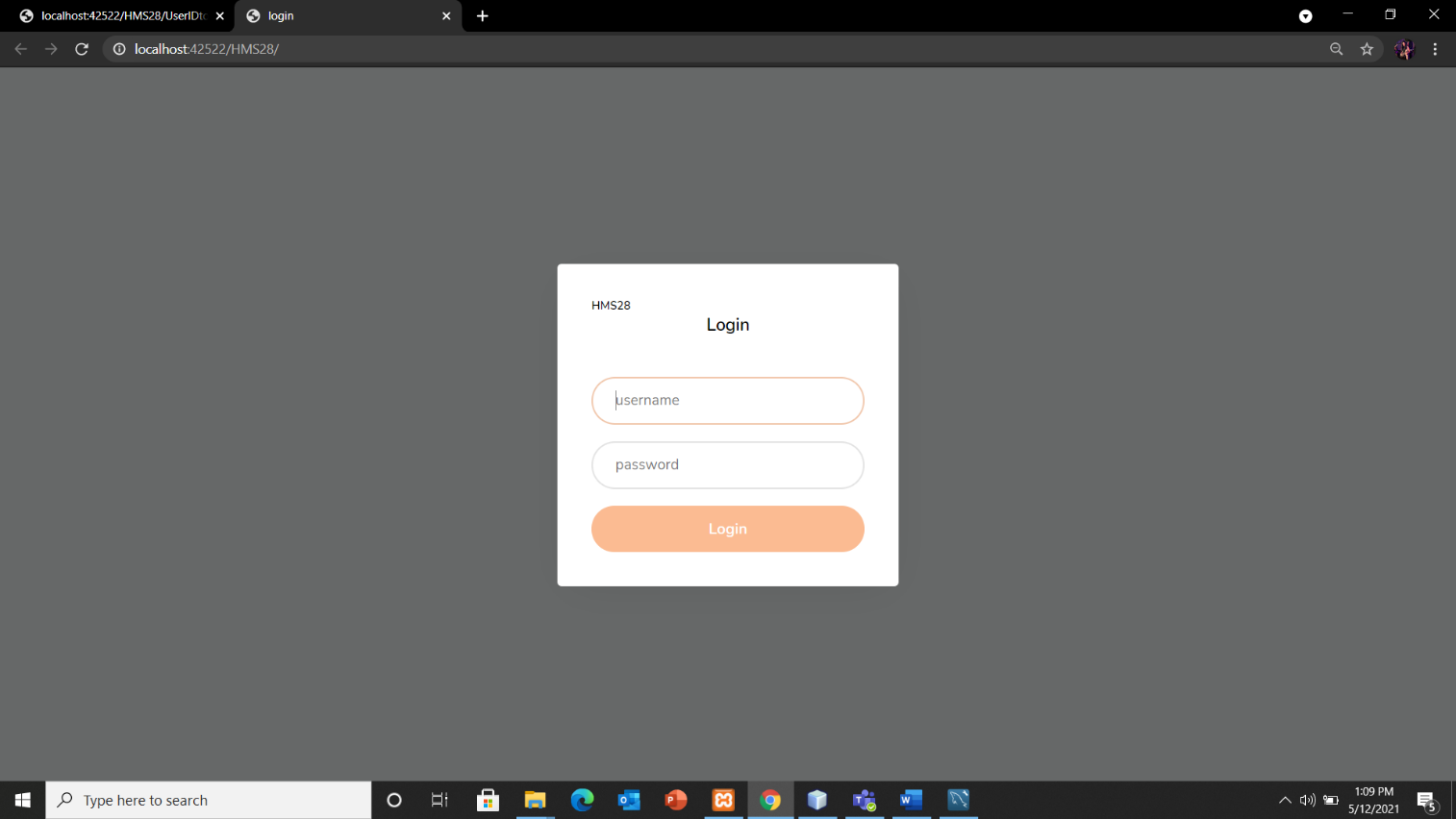






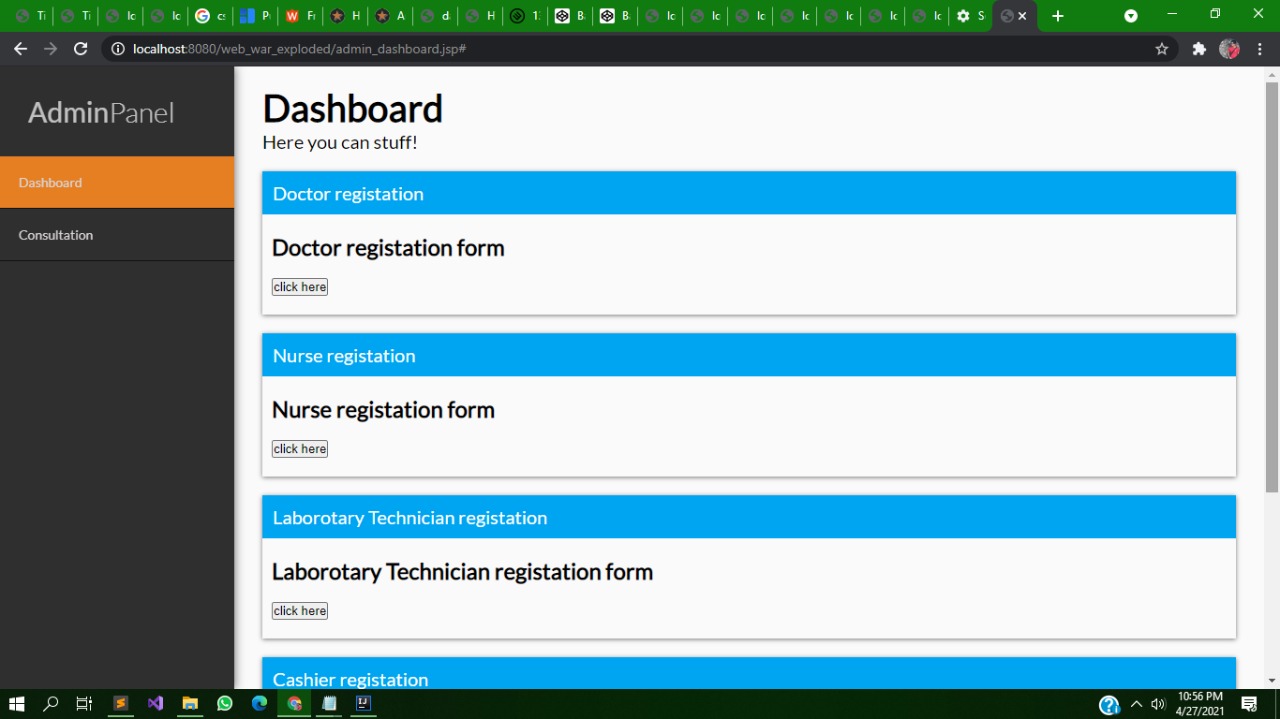
5. SCREENSHOTS

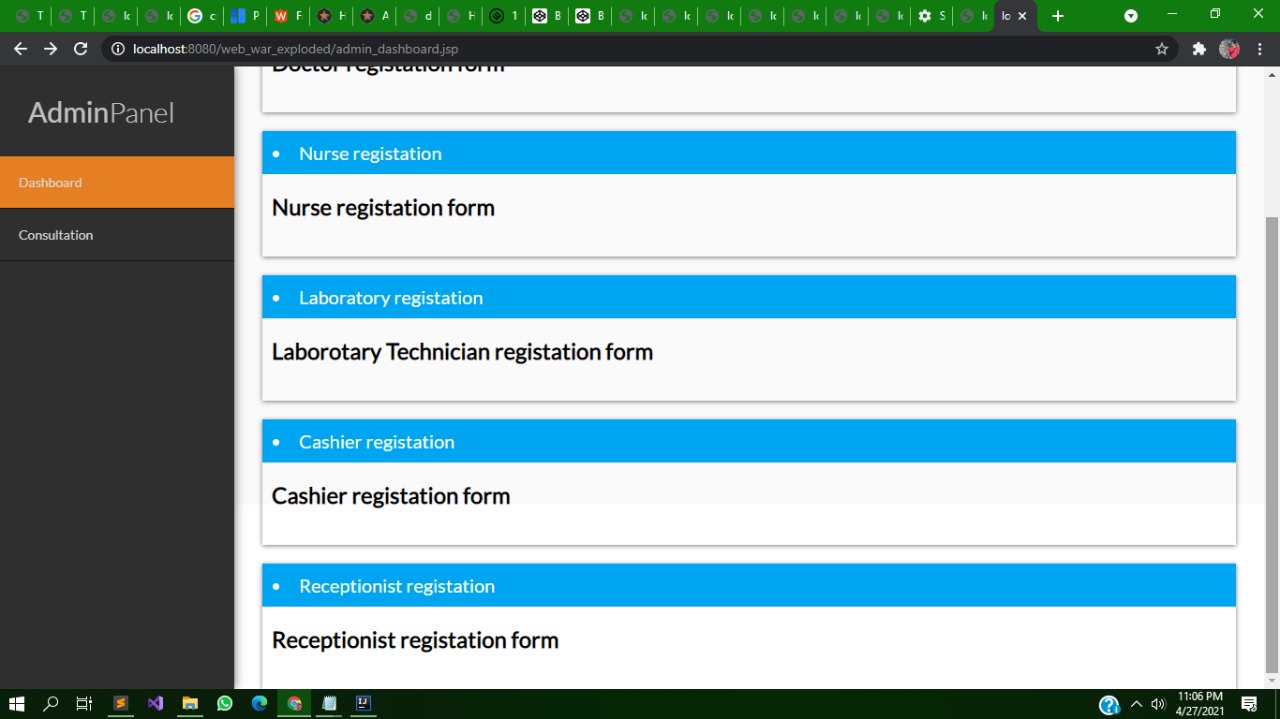
1. Login



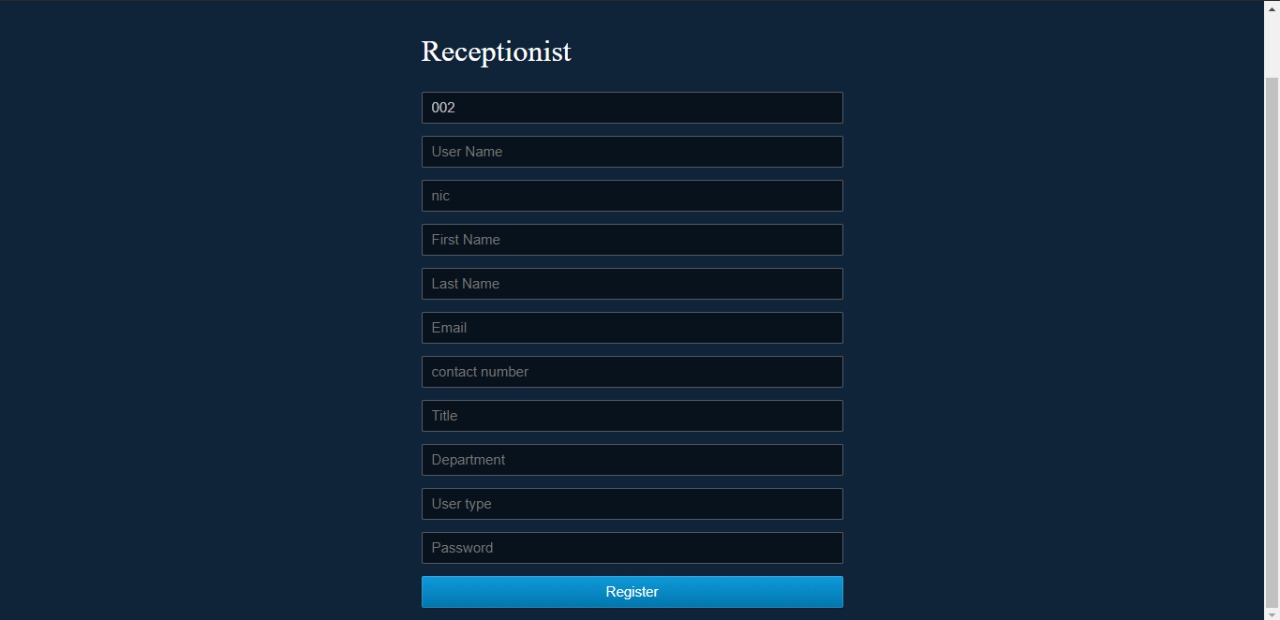
1. Admin Panel

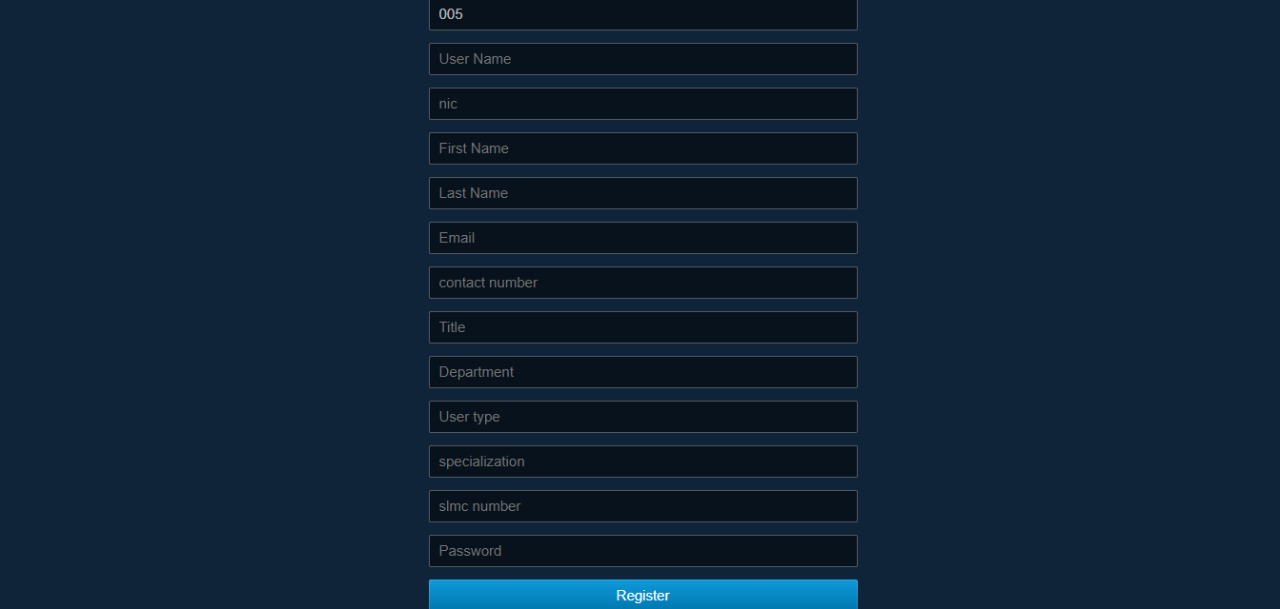
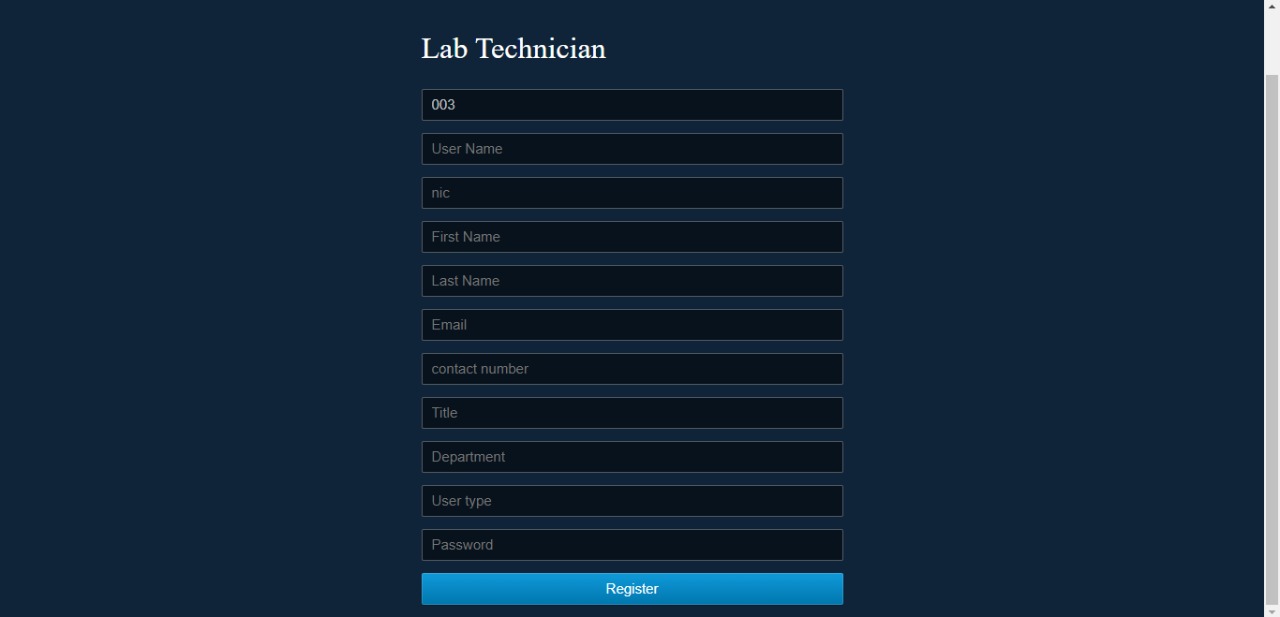
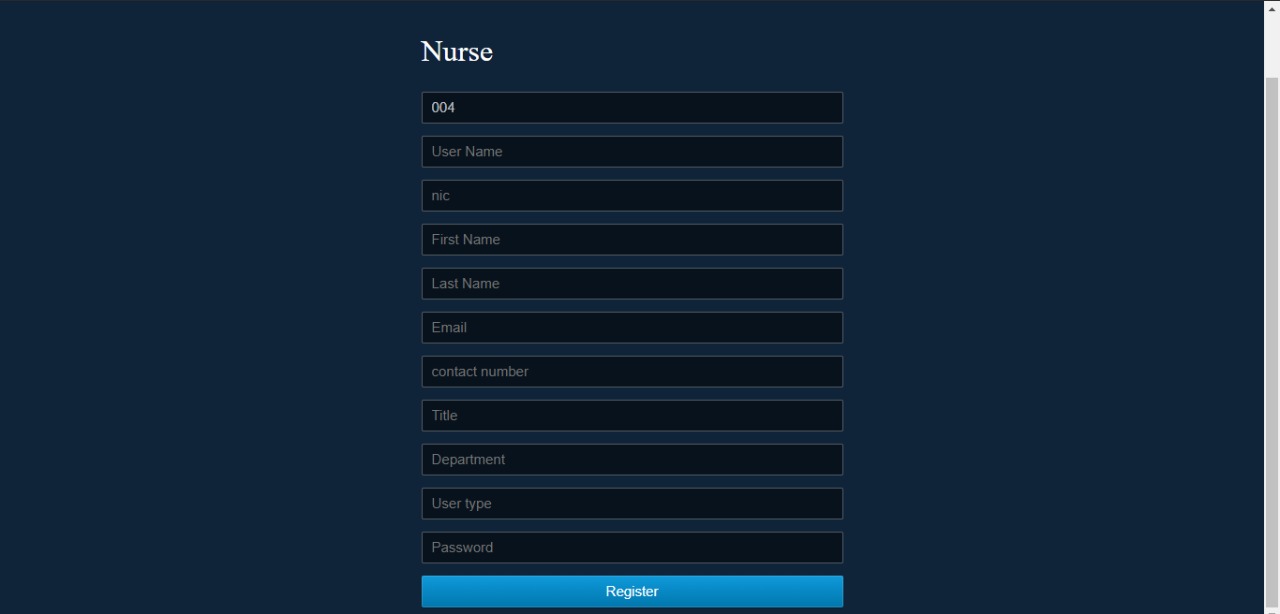
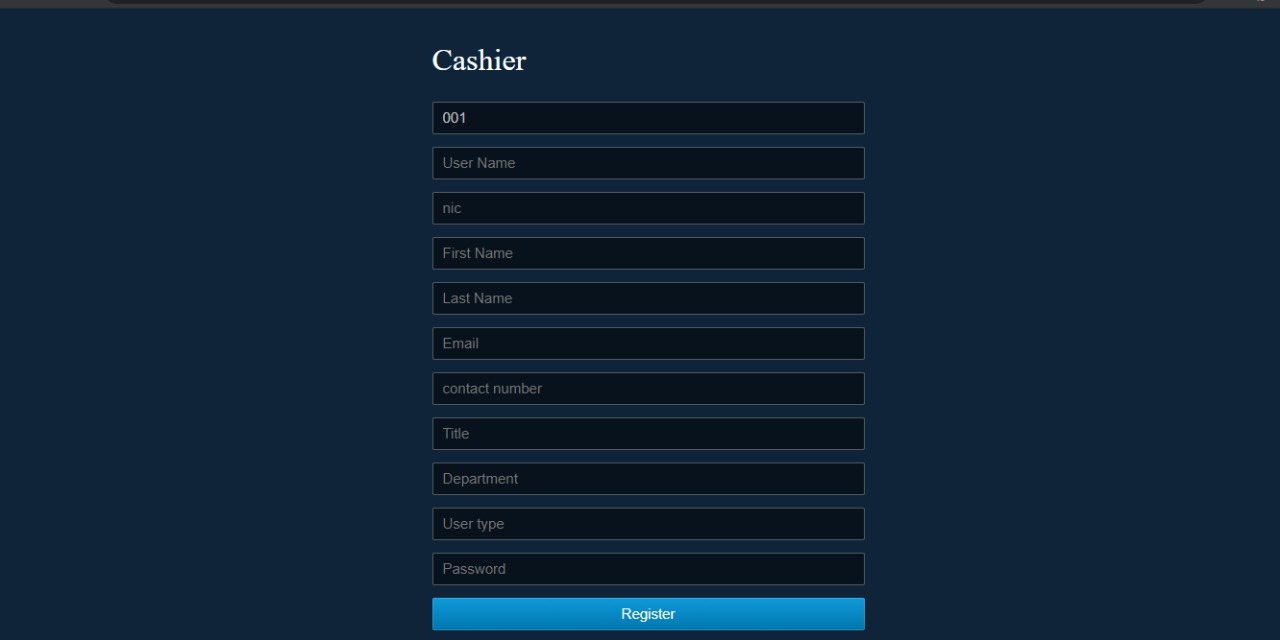
Dashboard:





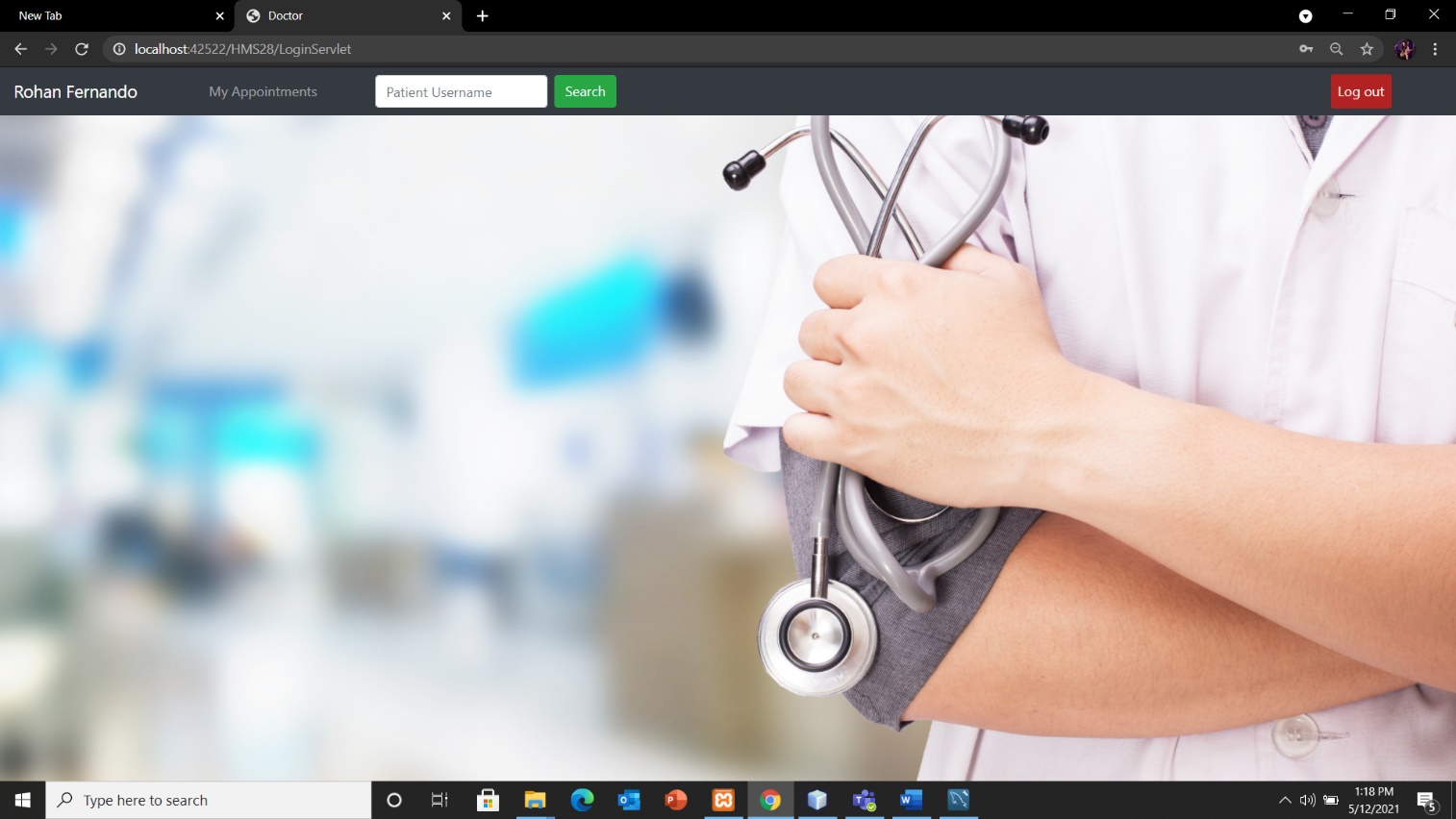
Registration Forms:

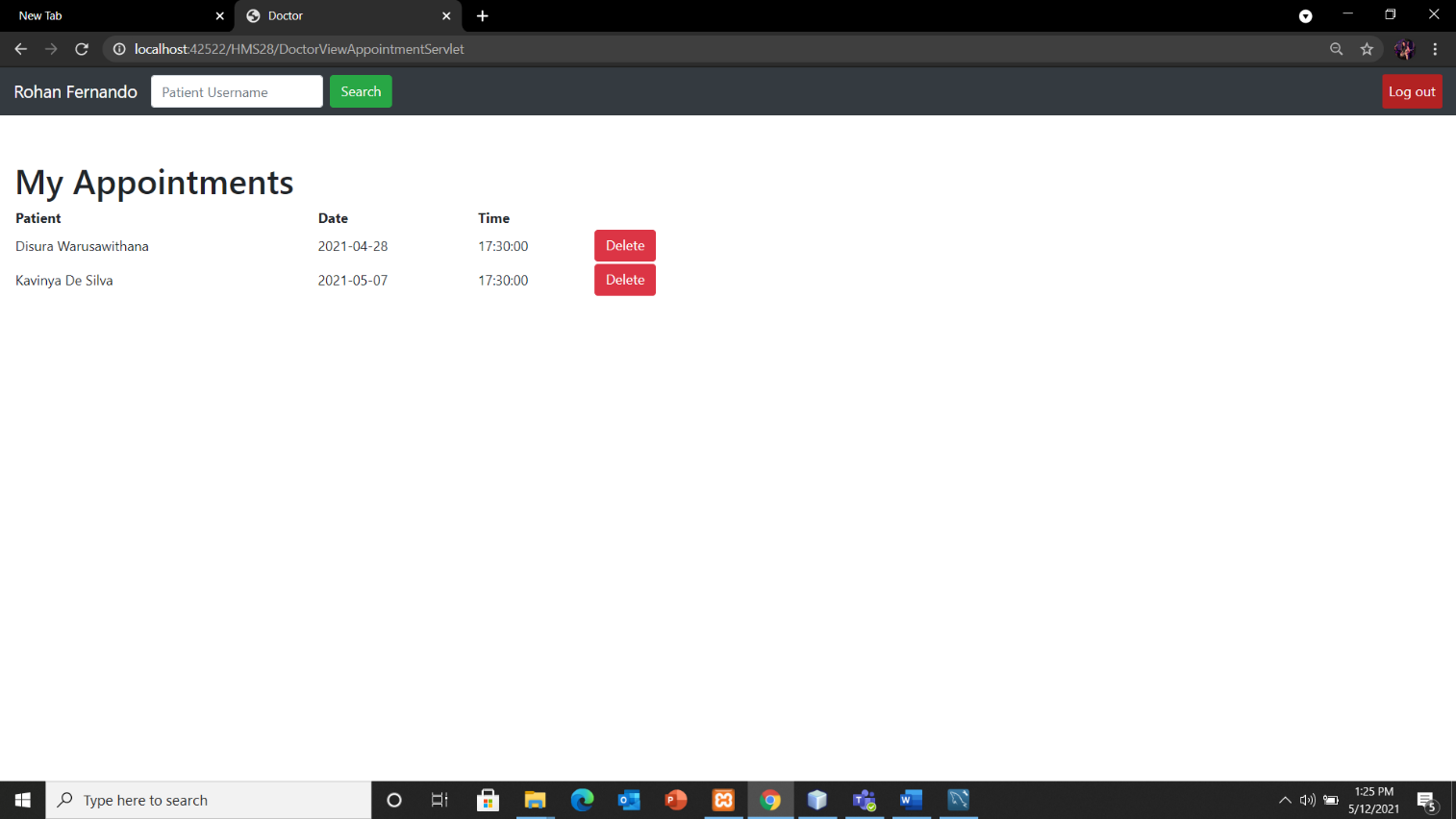




Doctor

1.Doctor Main



Doctor – View My Appointments and Cancel Appointments if needed.

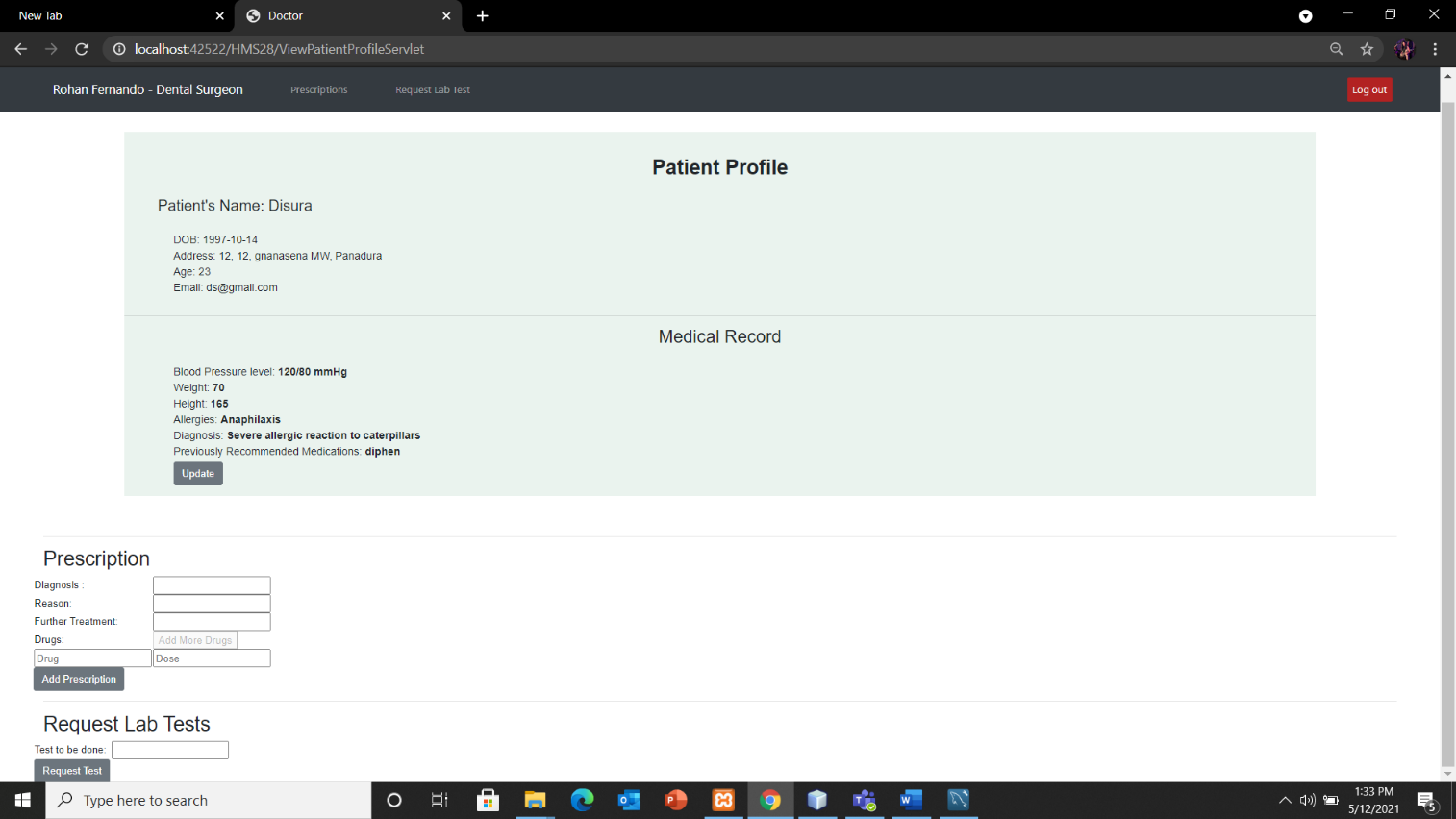


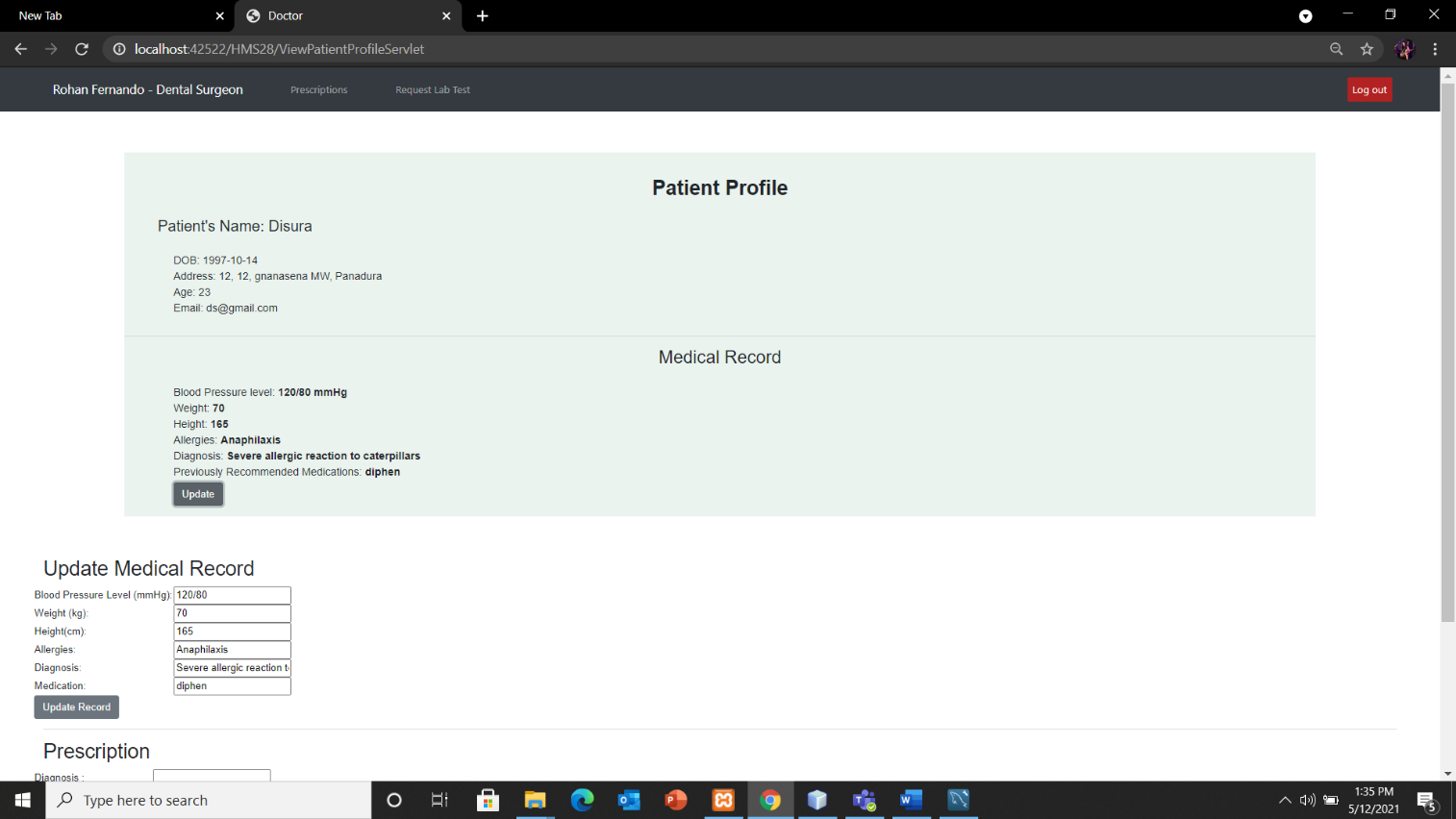
Doctor - Search for Patient and View Patient profile

View patient’s medical record

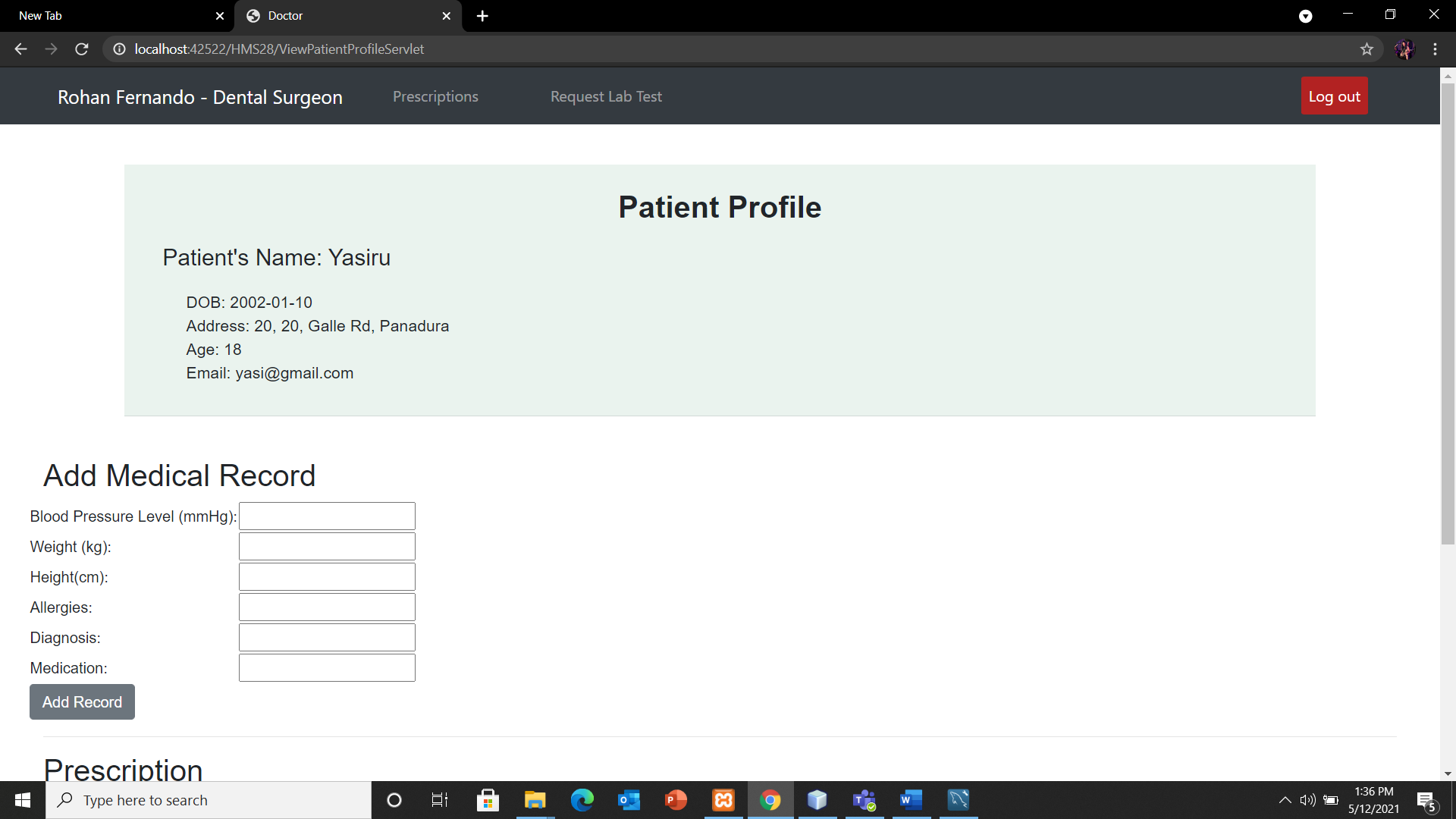
Upload Prescription for patient

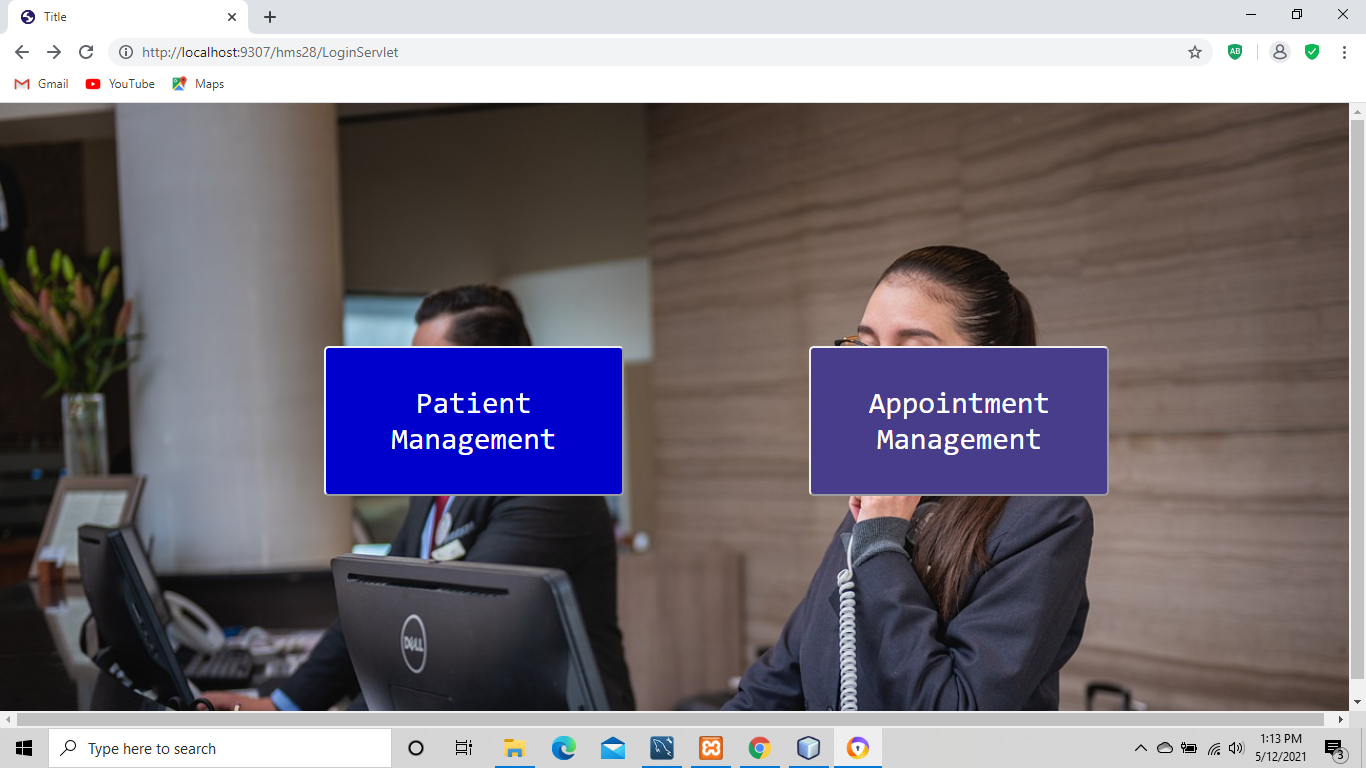
Request Lab Test



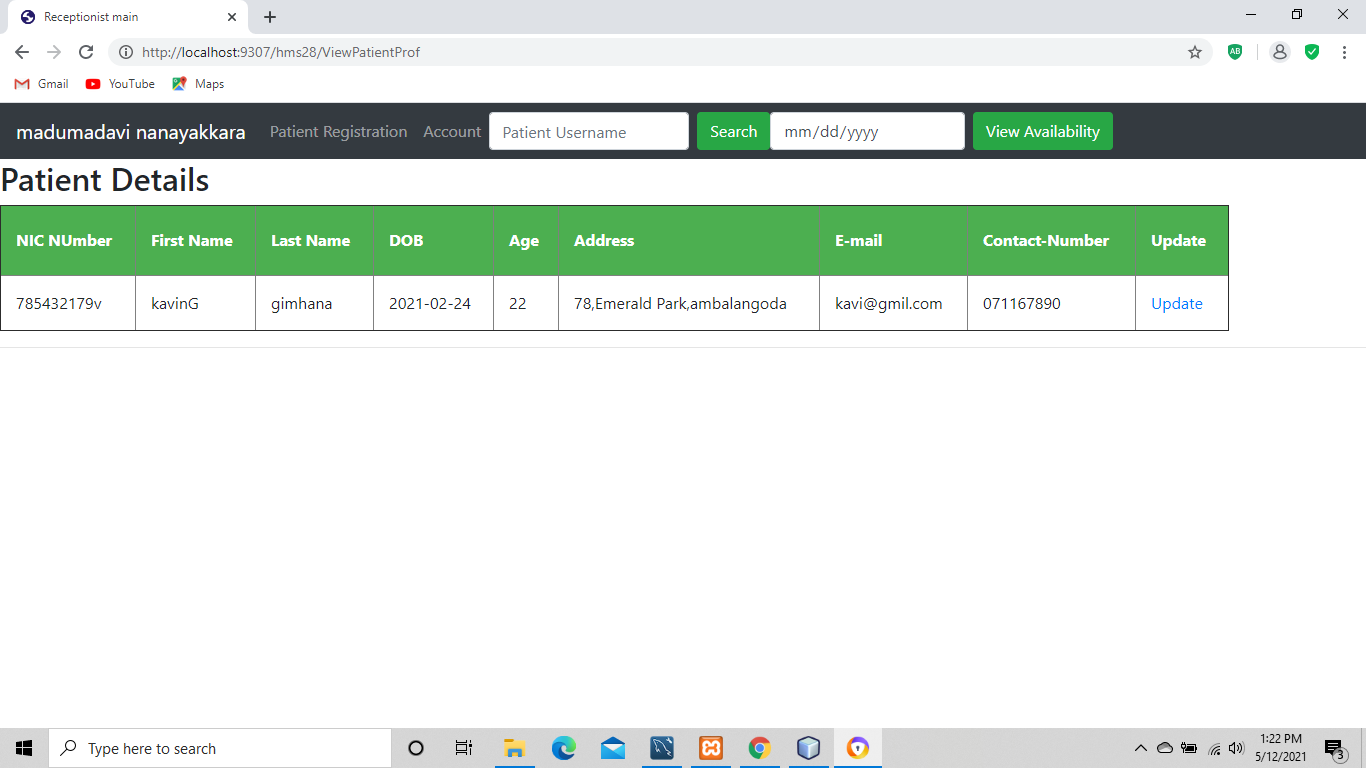
Update Medical Record:

Doctor – Add medical record for a newly registered patient.

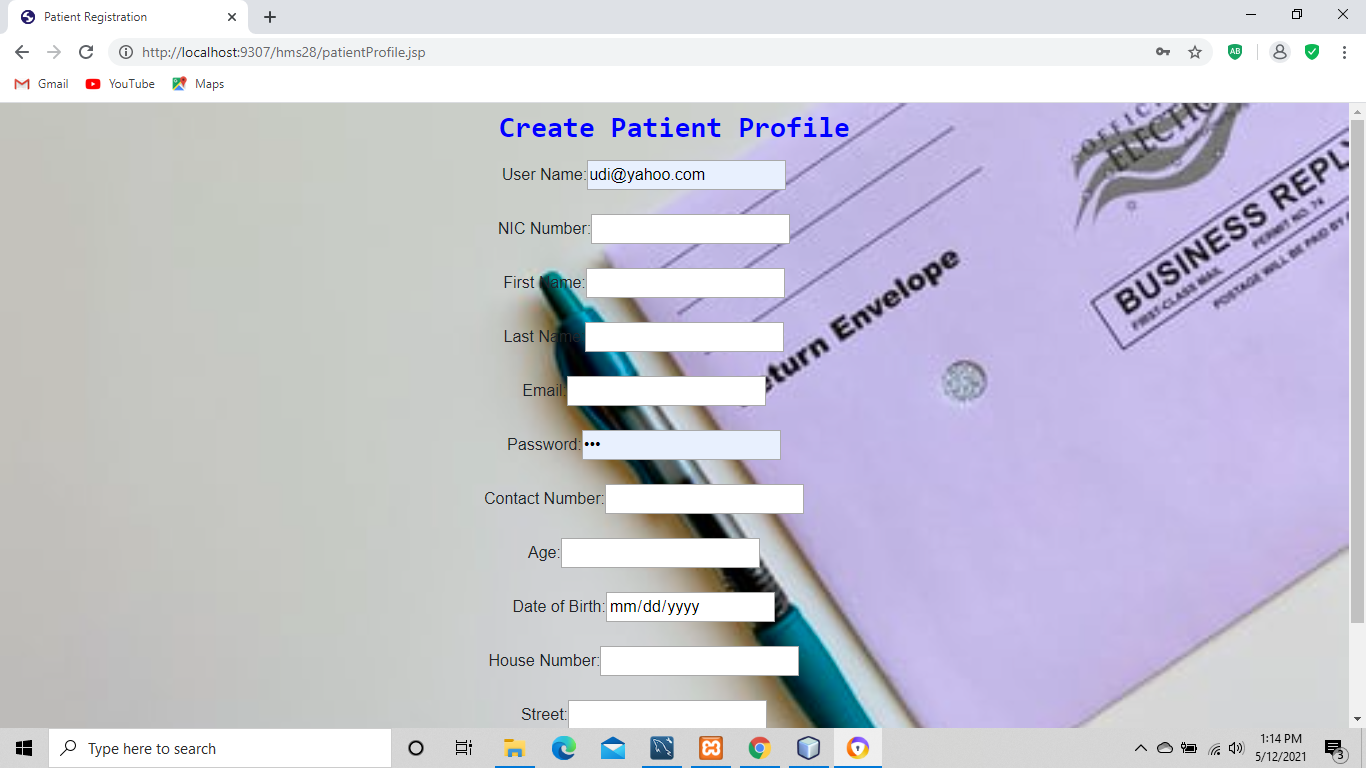




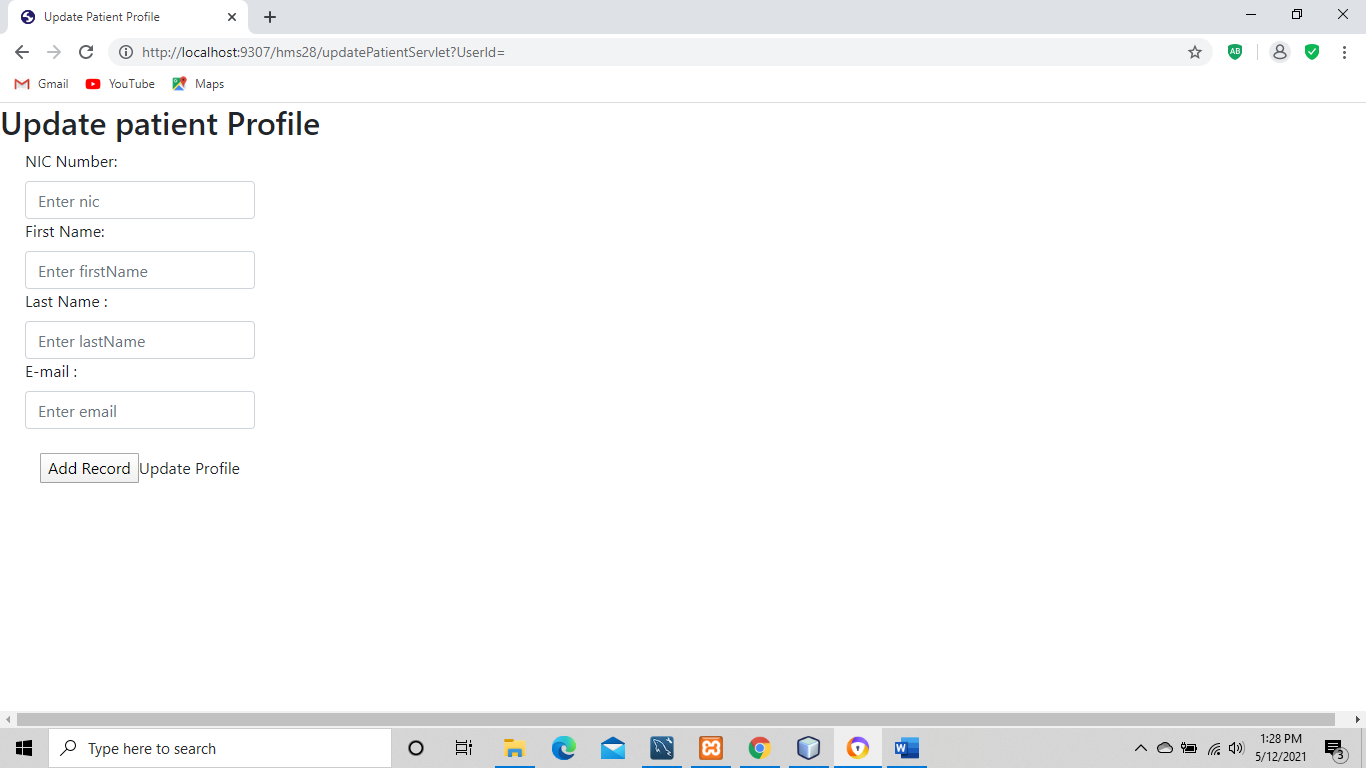
1. Receptionist Dashboard
2. Patient Management- Main Page



03.Patient Registration

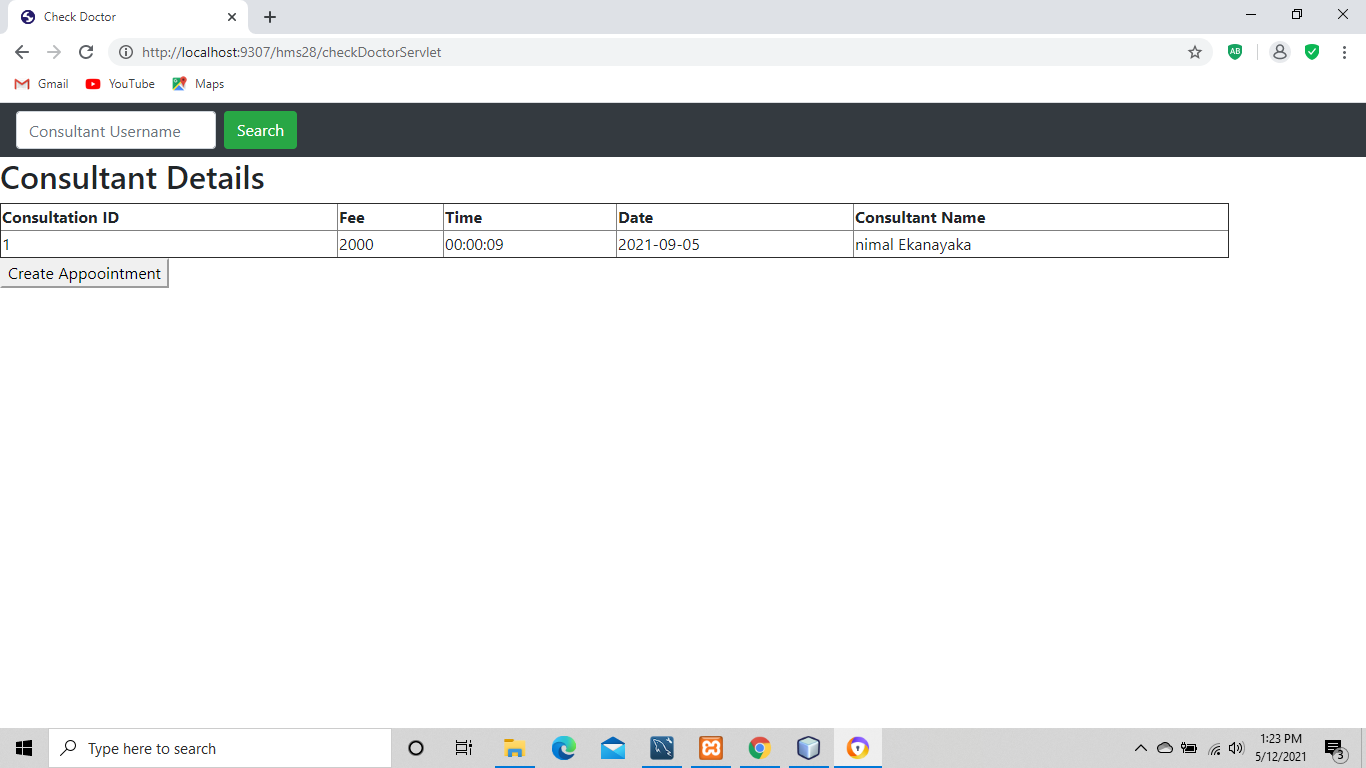
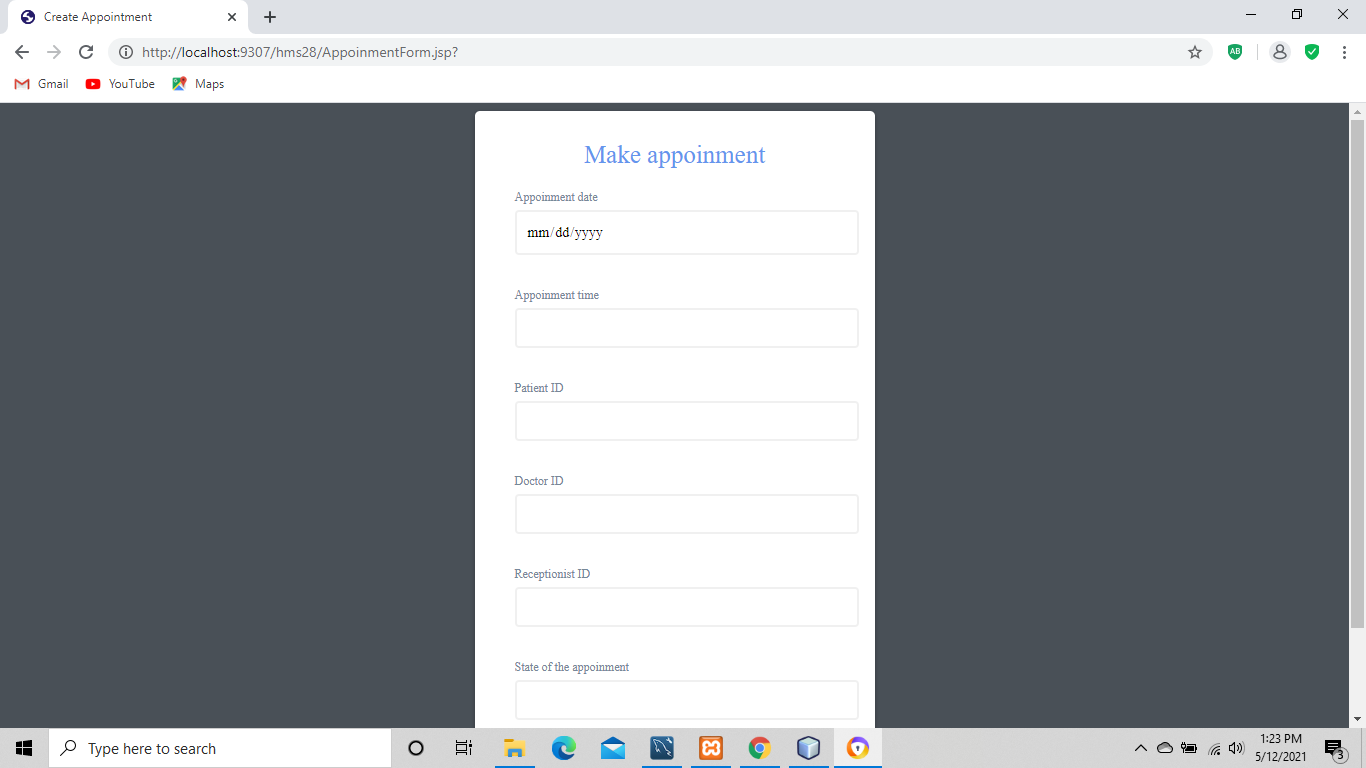


1. Update Patient Profile

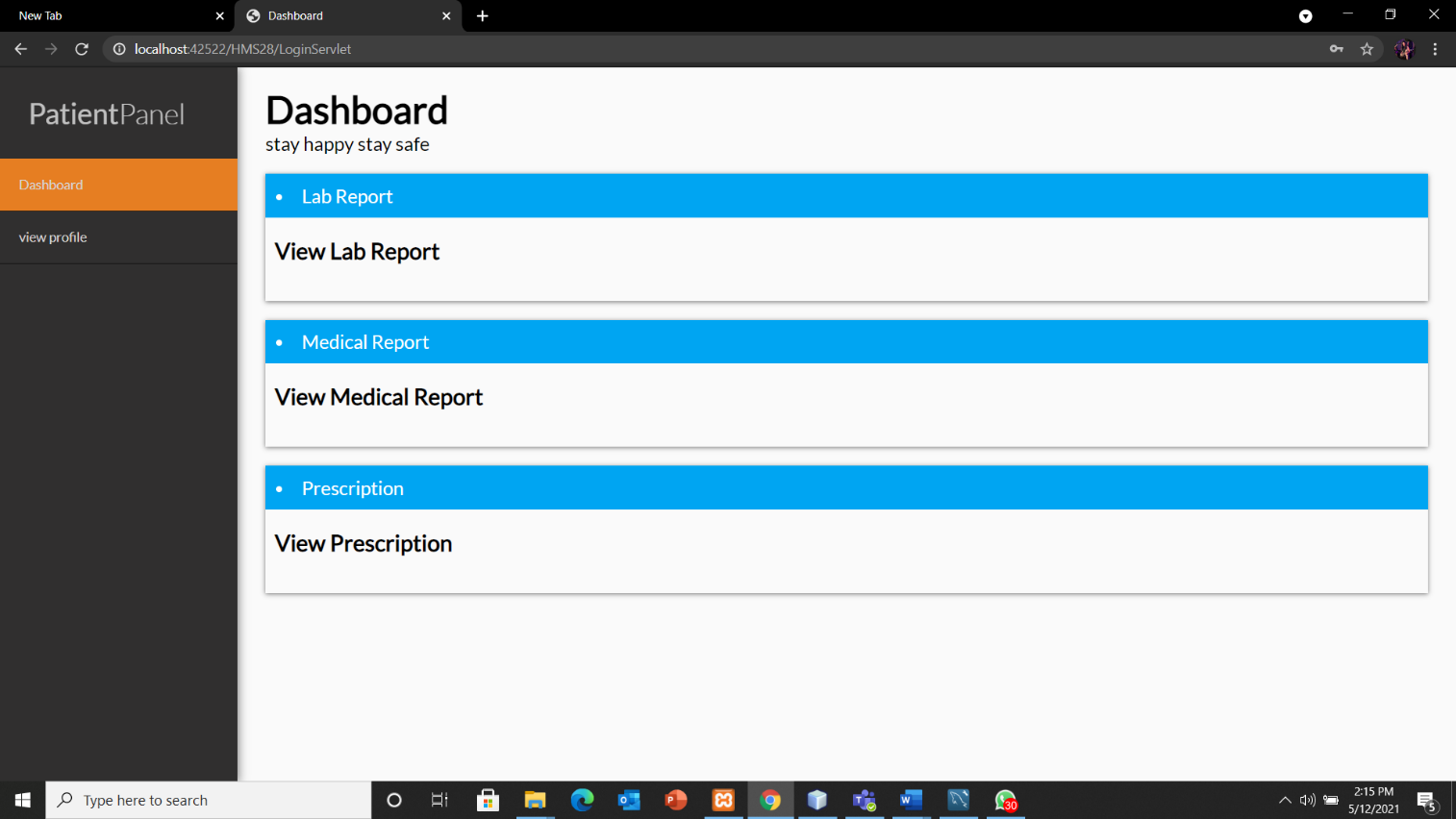


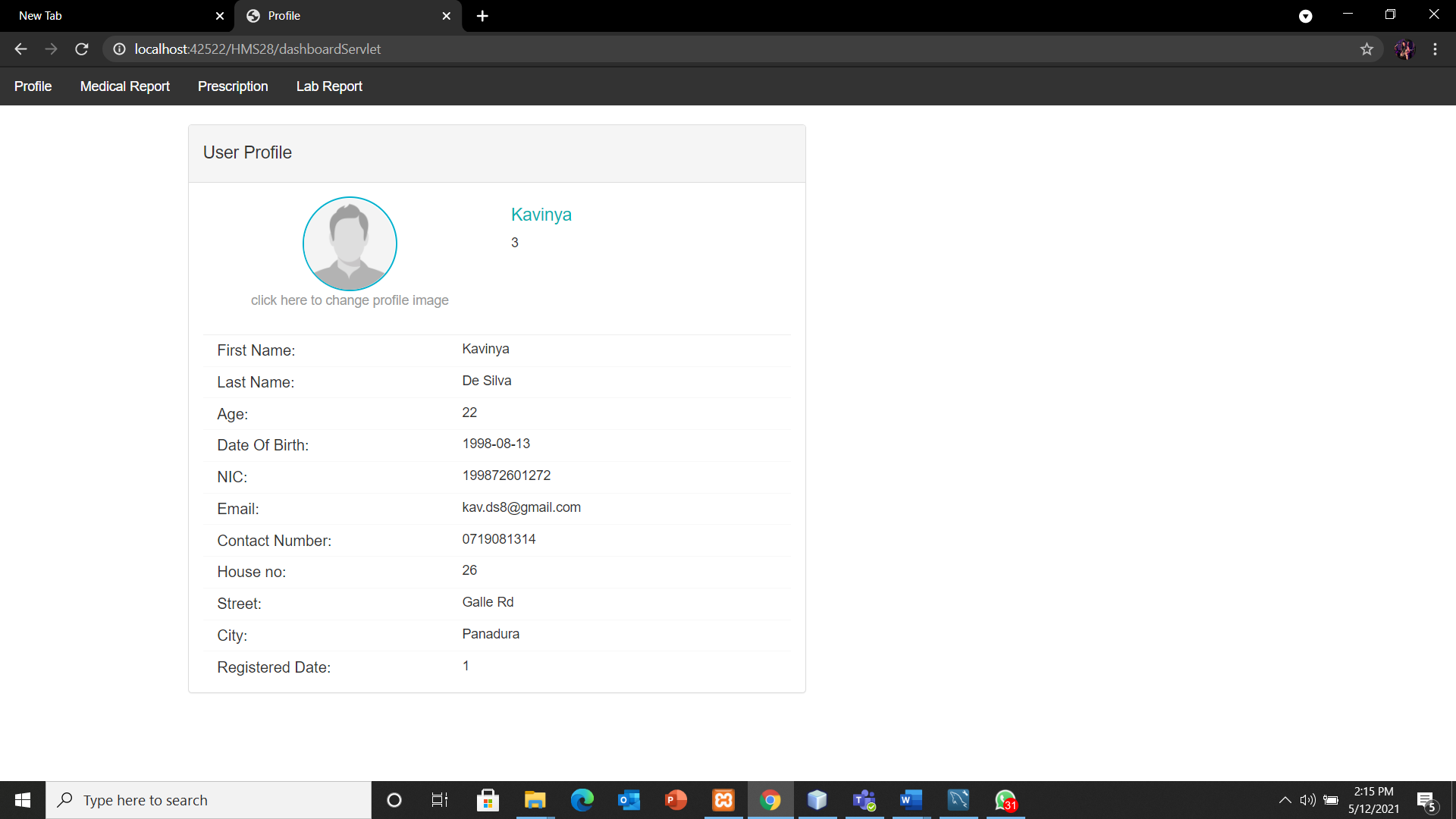
Appointment Management

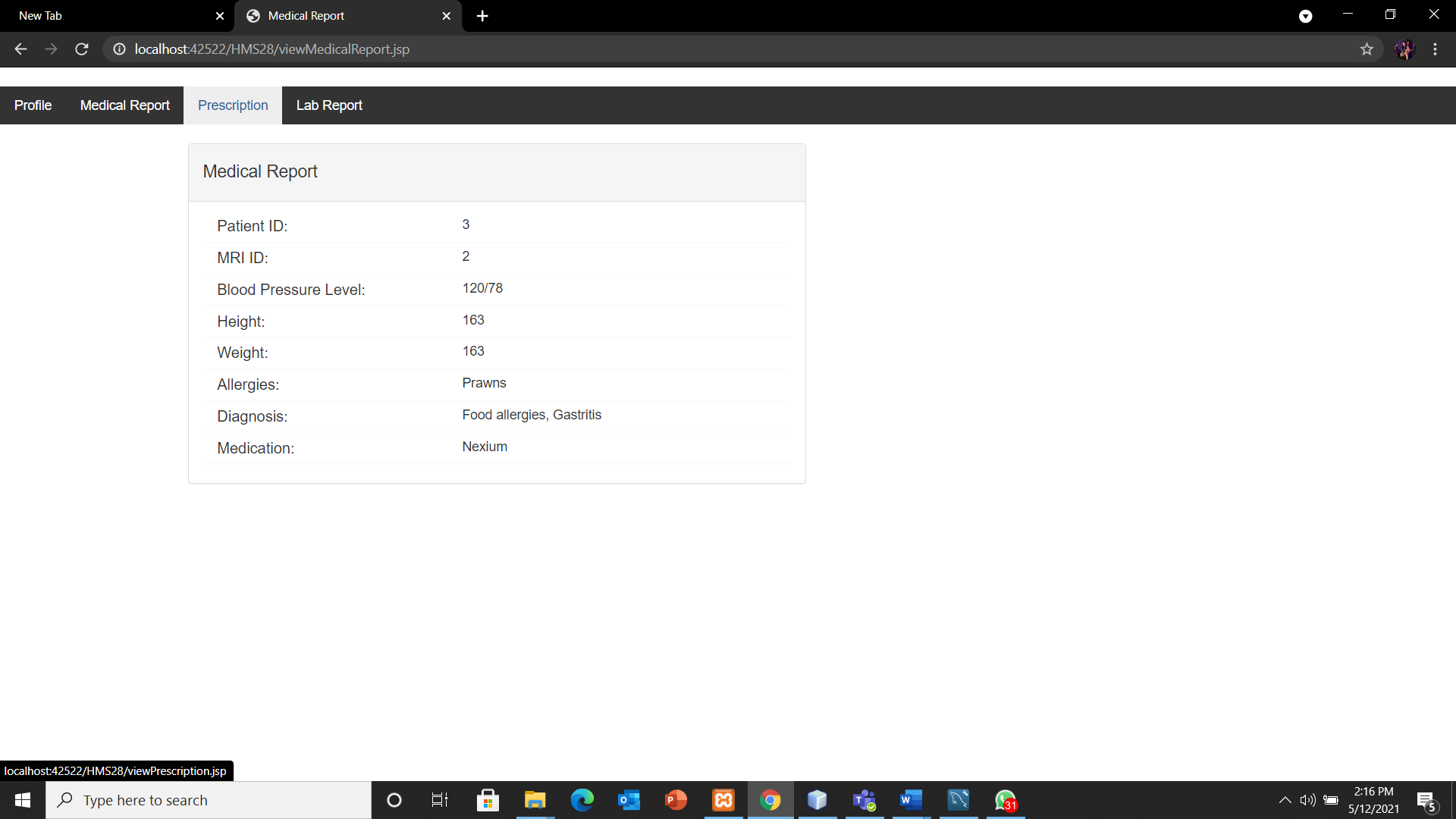
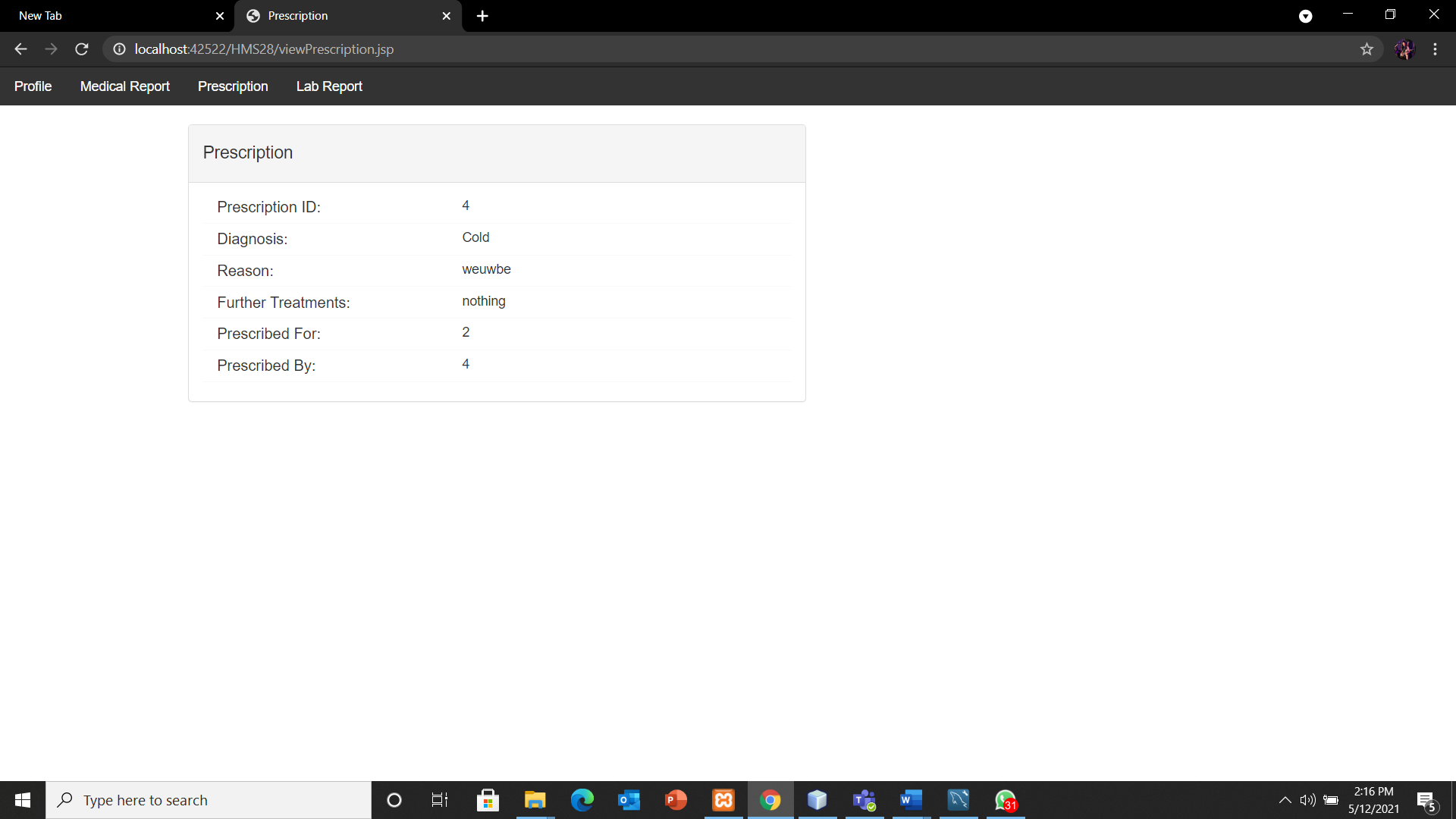
1. View Doctors Availability

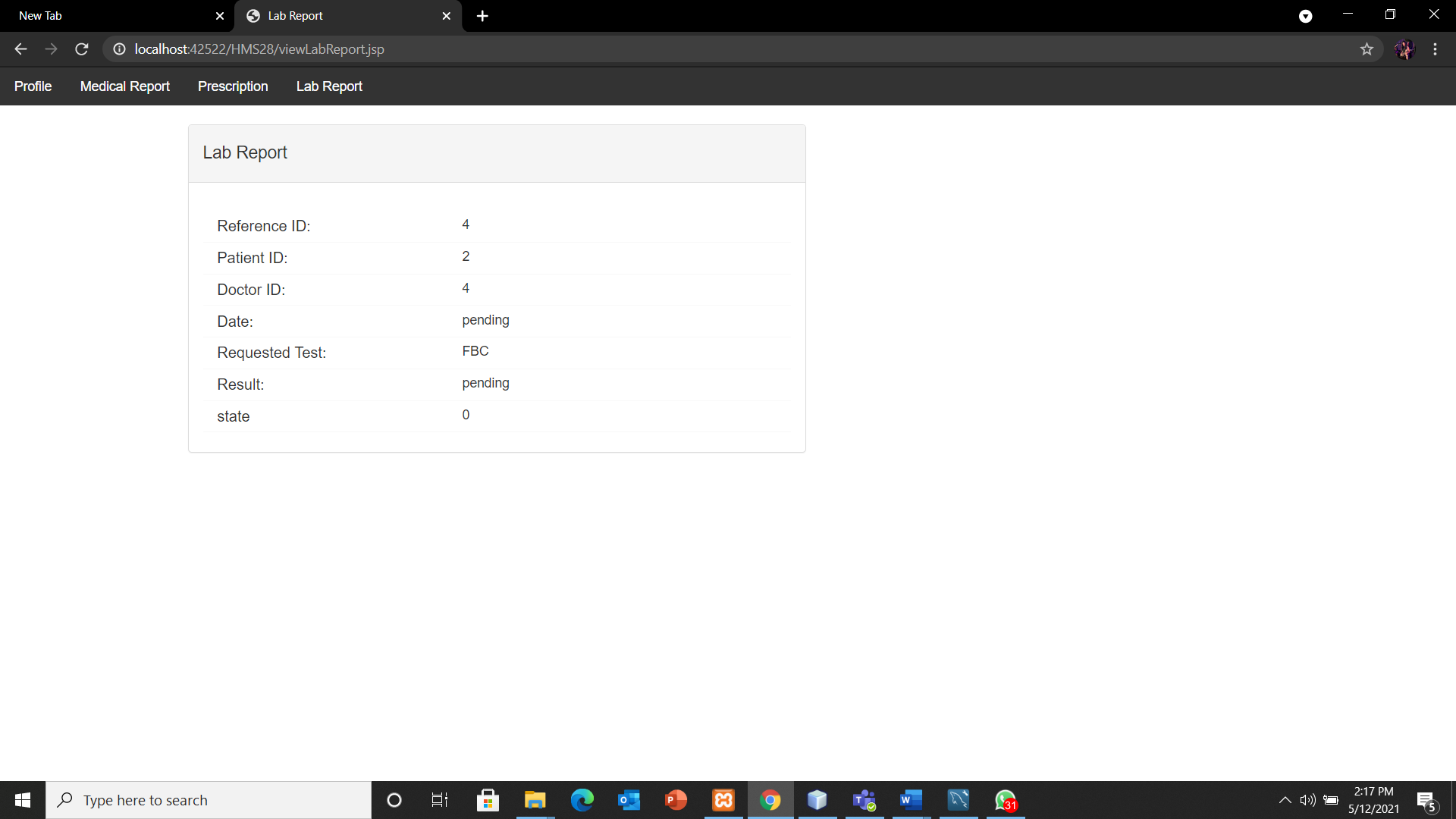
Create Appointment

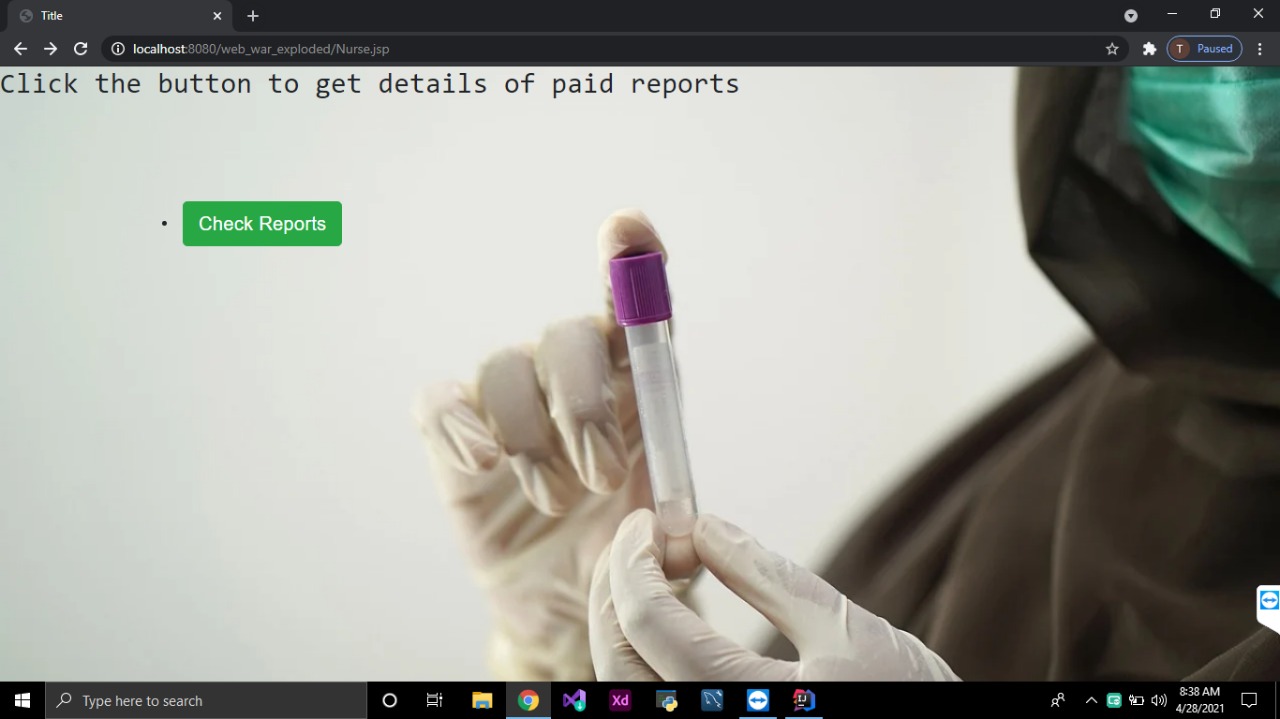
Patient:

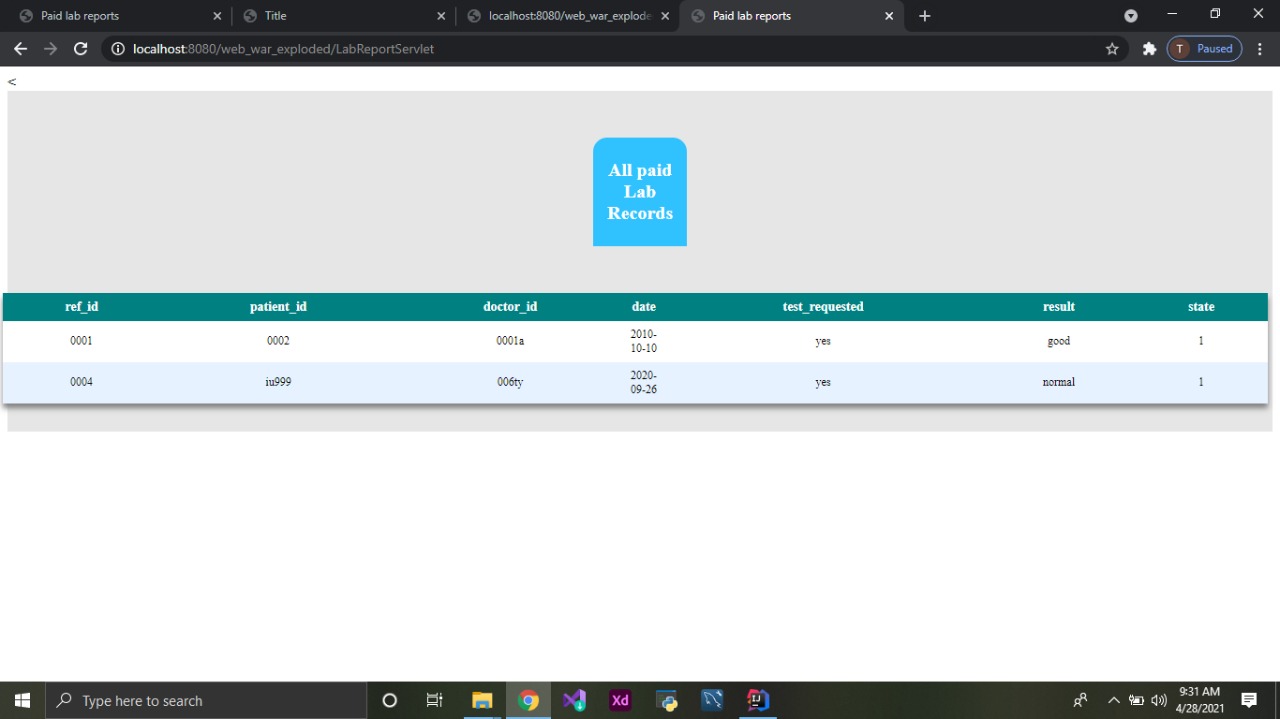






Nurse -



|  |  |  |
| --- | --- | --- |
| Plymouth Index | Name | Completed Tasks |
| 10707162 | Hembadura De Silva | Requirement analysis  Database  Doctor - related jsp, servlets, DBhandler methods |
| 10707163 | Arumadura De Silva | Relational Mapping  Receptionist - related jsp, servlets, DBhandler methods  Testing |
| 10707245 | Udawaththage Kumara | Sequence Diagram  Send Lab Request  Nurse - related jsp, servlets, DBhandler methods |
| 10707151 | W N S Botheju | Patient - related jsp, servlets, DBhandler methods  Use Case Diagram |
| 10707347 | P D S Rathnayake | User Login  Class diagram  System Admin -related jsp, servlets, DBhandler methods |