**Patient appointment & Hospital Management**

**Functional Scope :**

1.**Patient**:

* Patient once register using details,
* After login patient can book appointment based on doctor availability, using book an appointment. In that appointment doctor name, patient name, appointment date and time.
* Patient can view book appointment history
* Patient can also view prescribed medications

2.**Admin**:

* Admin have individual, login authentication,
* After admin login,
* Admin can add/register new doctor and doctor details
* Admin can do approve or reject patient appointments
* Admin can also add, edit, delete prescribed medications

3.**Doctor**

* Doctor have individual, login authentication
* Doctor can also view patient appointment details like patient name, date, time
* Doctor can add prescription to patient, can edit save , it will reflect in patient page as well.

**Technical Scope:**

* Created Controller for doctor, patient, admin
* Handle common exception handling
* Created Repository for doctor, patient, admin
* Created service for individual for doctor, patient, admin followed by implementation for all
* Handle all login implementation handled here,
* Handle user login, view edit, update,
* Used Html, CSS, java script for front End side
* Used Spring Boot, hibernate, java for backend
* Using Rest Controller, Mapping concept
* Used authentication for user login, admin login

**API End Points :**

**Home:**

<http://localhost:63342/HospitalManage/DoctorAppointmnet(FrontEnd)/index.html>

**Admin**:

<http://localhost:63342/HospitalManage/DoctorAppointmnet(FrontEnd)/mgmt_login.html>

**Patient Registration:** <http://localhost:63342/HospitalManage/DoctorAppointmnet(FrontEnd)/signup.html>

**Patient Login:**

<http://localhost:63342/HospitalManage/DoctorAppointmnet(FrontEnd)/login.html>

**Doctor:**

<http://localhost:63342/HospitalManage/DoctorAppointmnet(FrontEnd)/mgmt_login.html>

**Data Validation:**

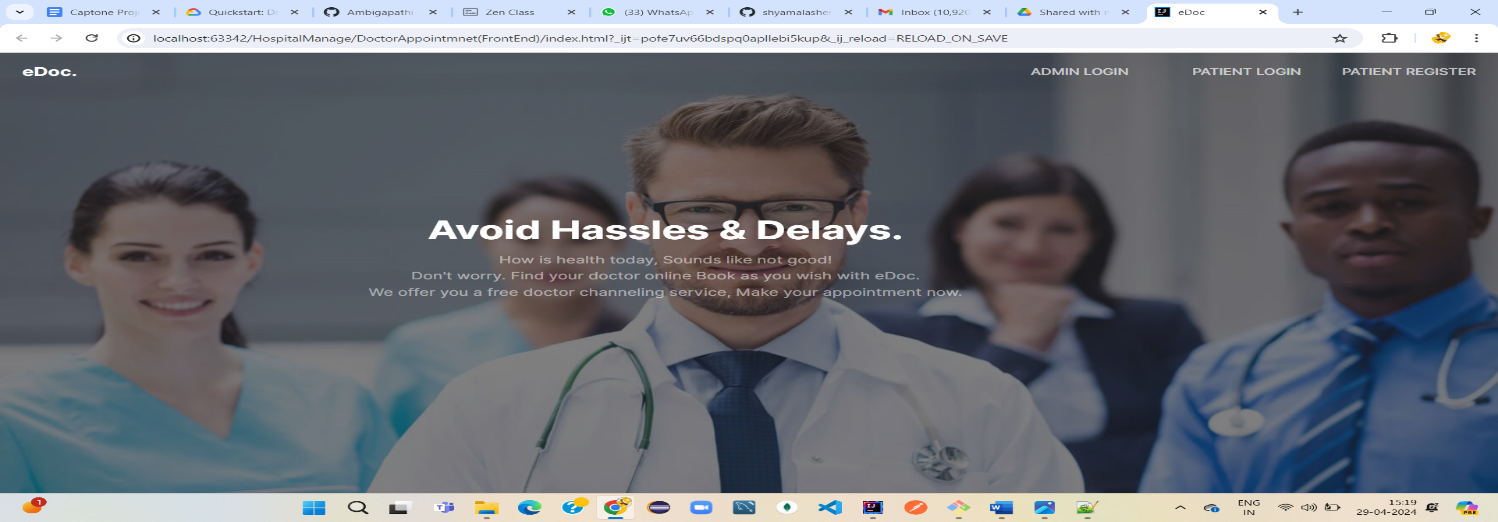
* Only register patient can login and book a appointment
* Only authorized admin can view admin page and add doctor
* Only registered doctor can add, edit prescription
* Only admin can add or remove doctor
* Patient can view prescription, cannot do add or edit

**Testing and Quality Assurance:**

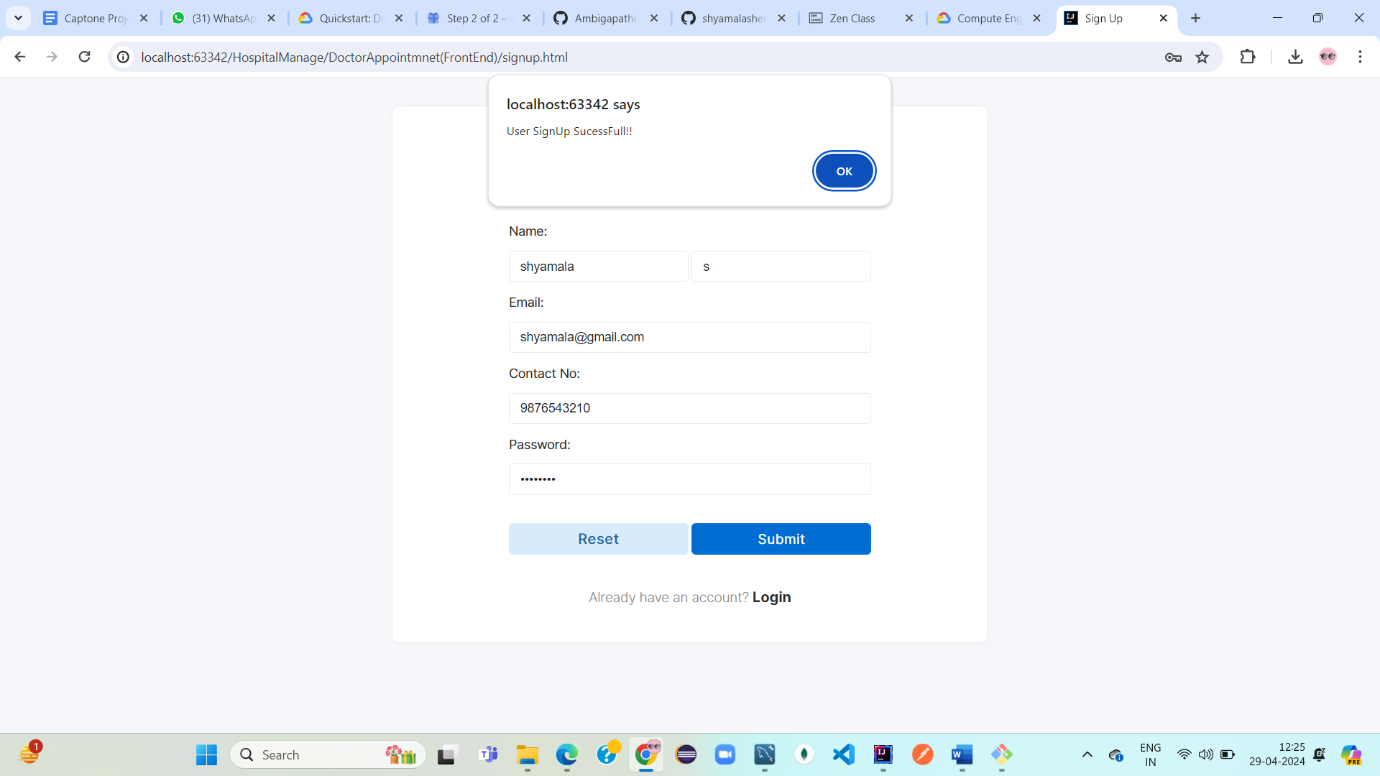
* Tested no other user can login admin page
* Tested no other user can login doctor page
* Tested only admin can add doctor details
* Tested only doctor can add , edit prescription
* Only registered patient can book a appoitnment

**Sample Output:**

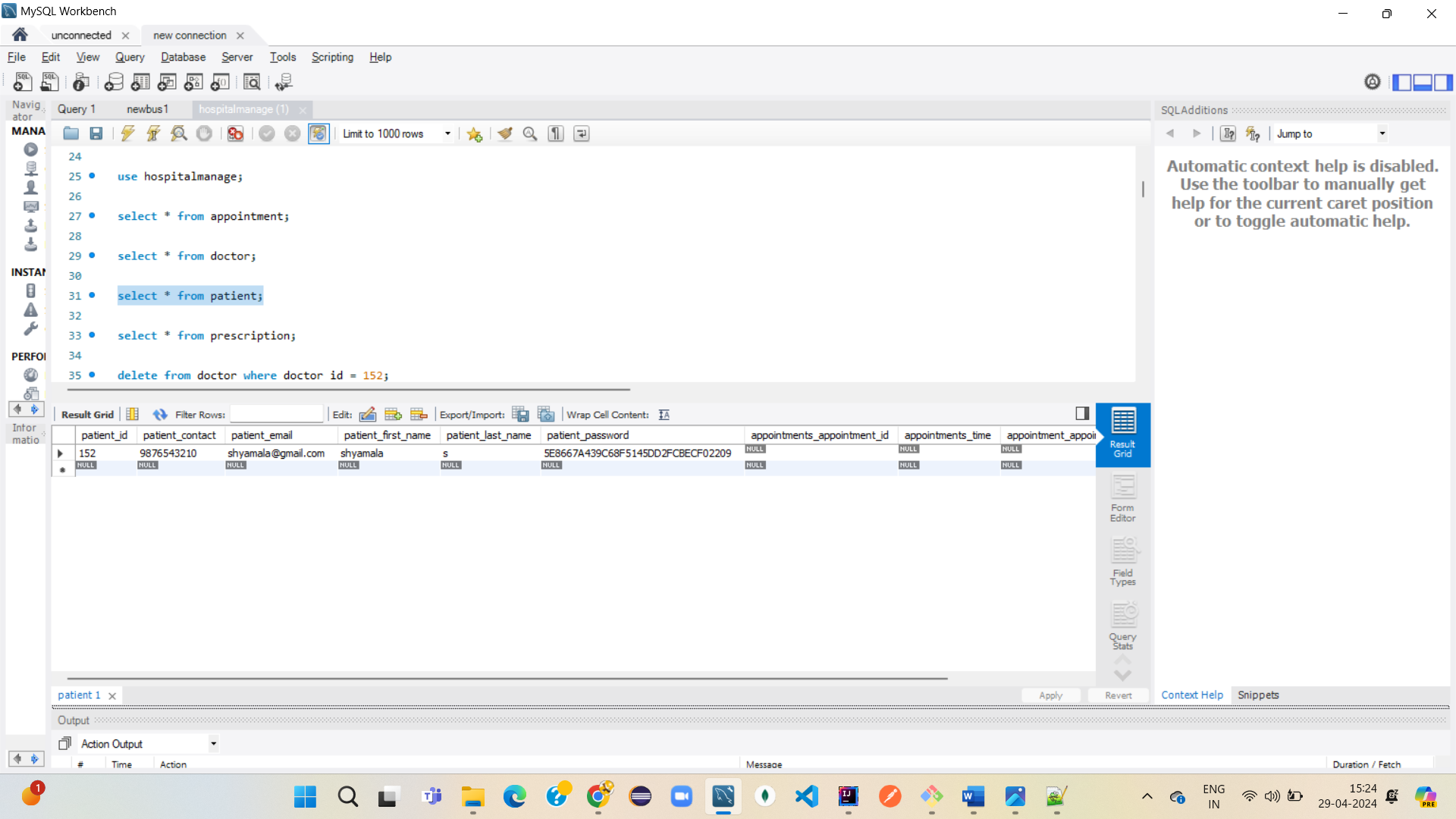
* **Home page for eDoc.**



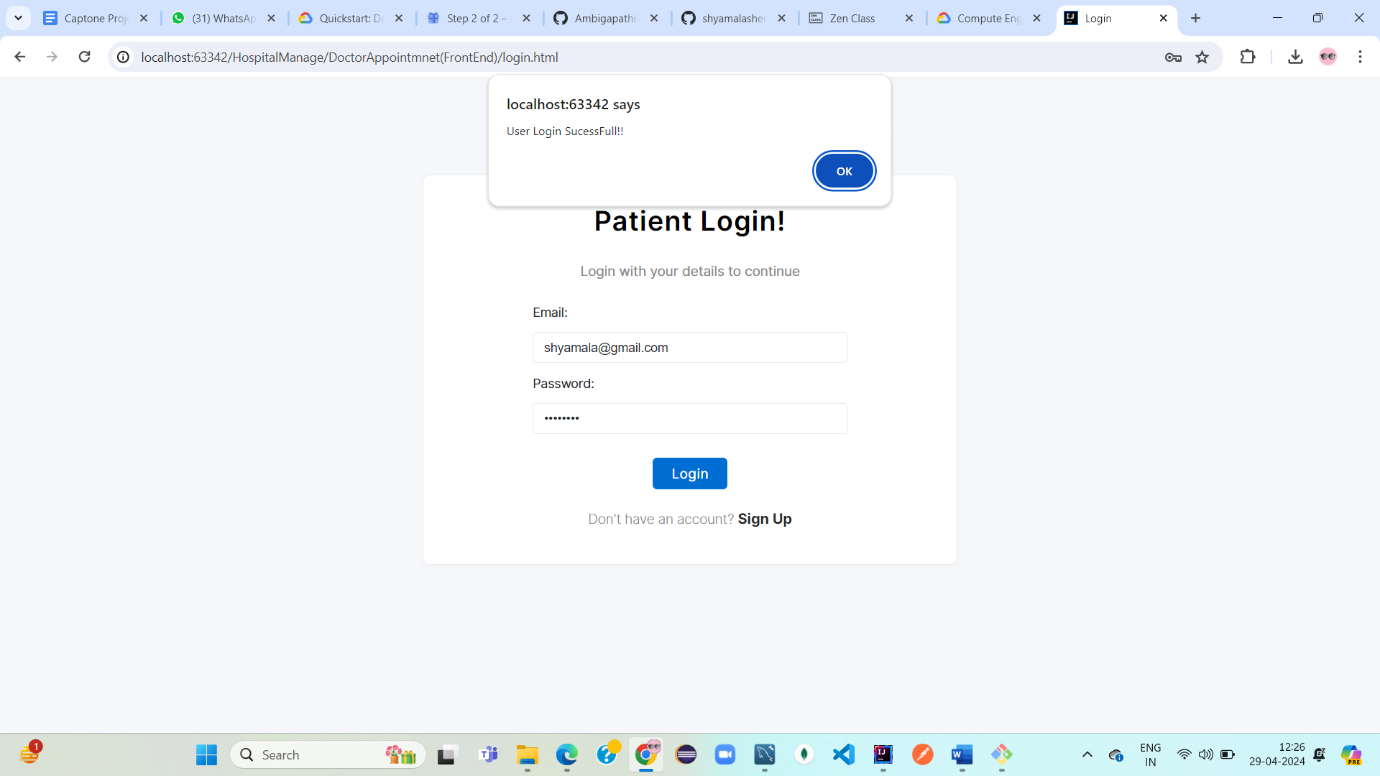
* **Patien Register with details:**



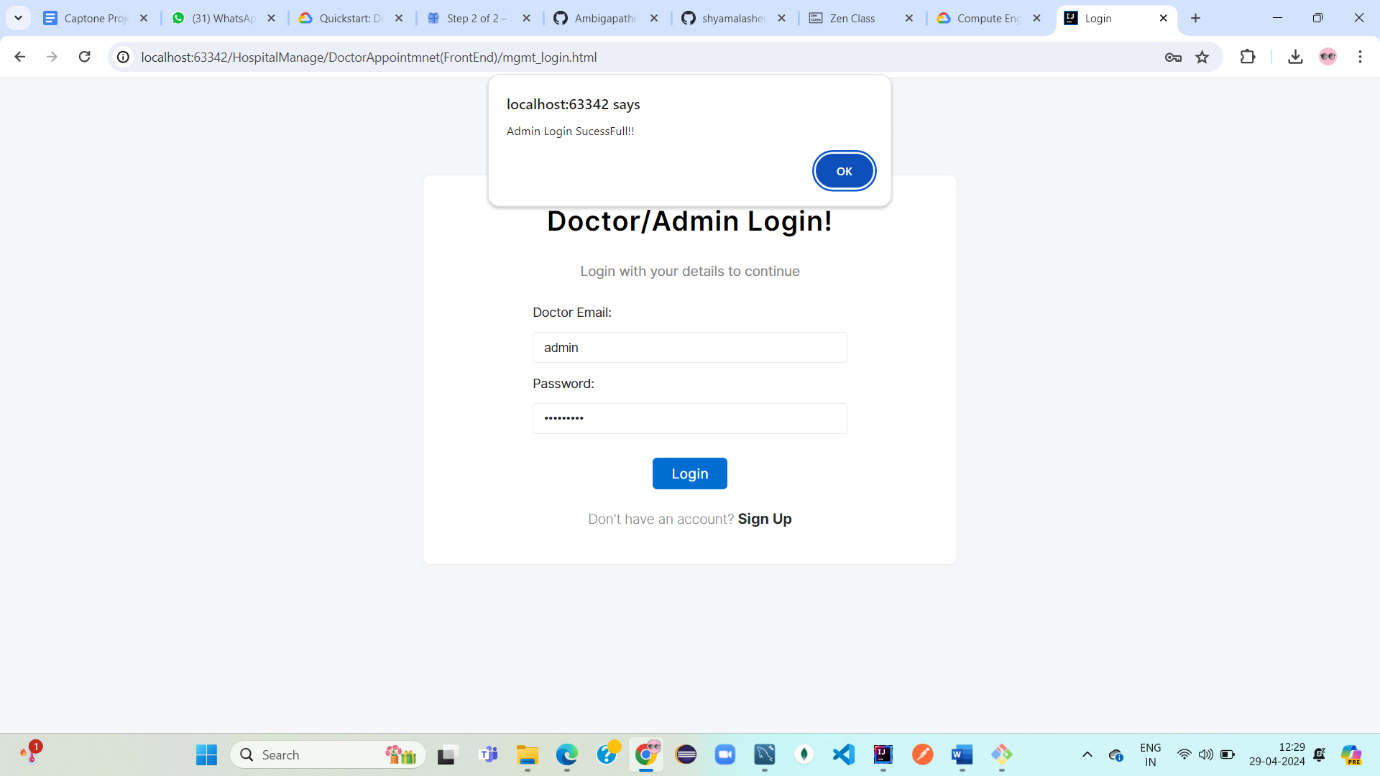
* **Patient Register Details in DB:**



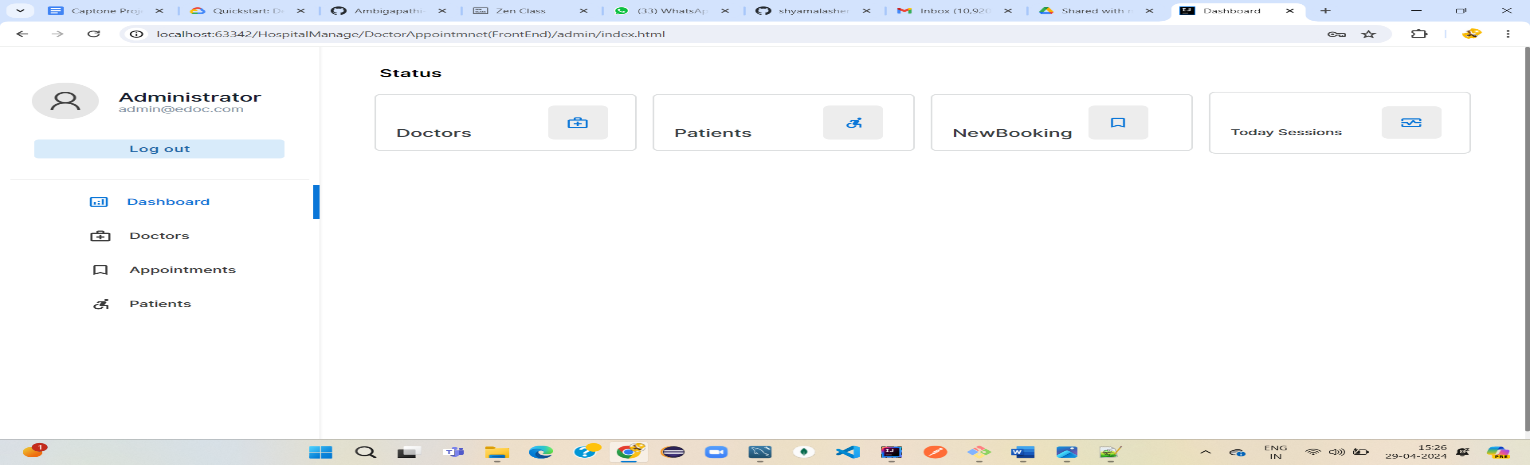
* **Patient Login using register details:**



* **Admin login using authorized details:**

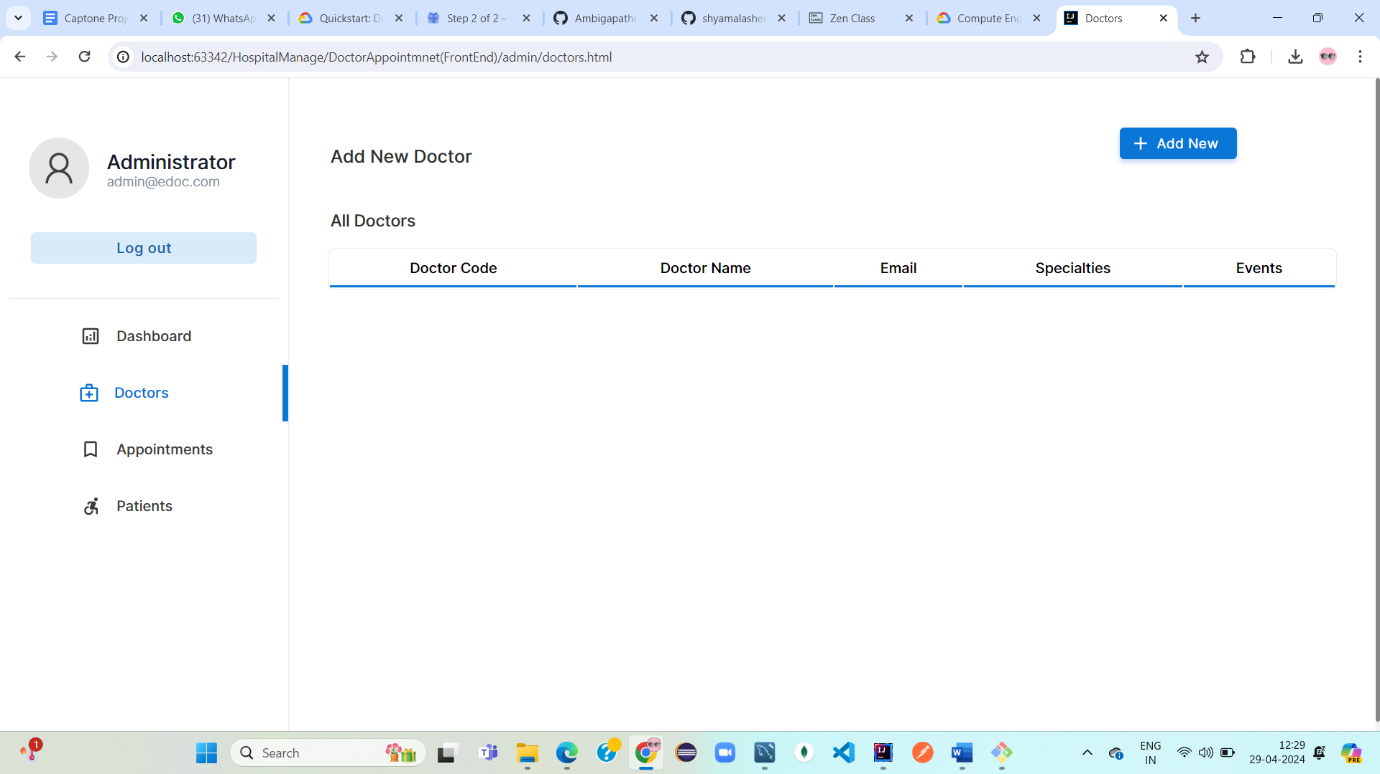


* **Admin Page:**

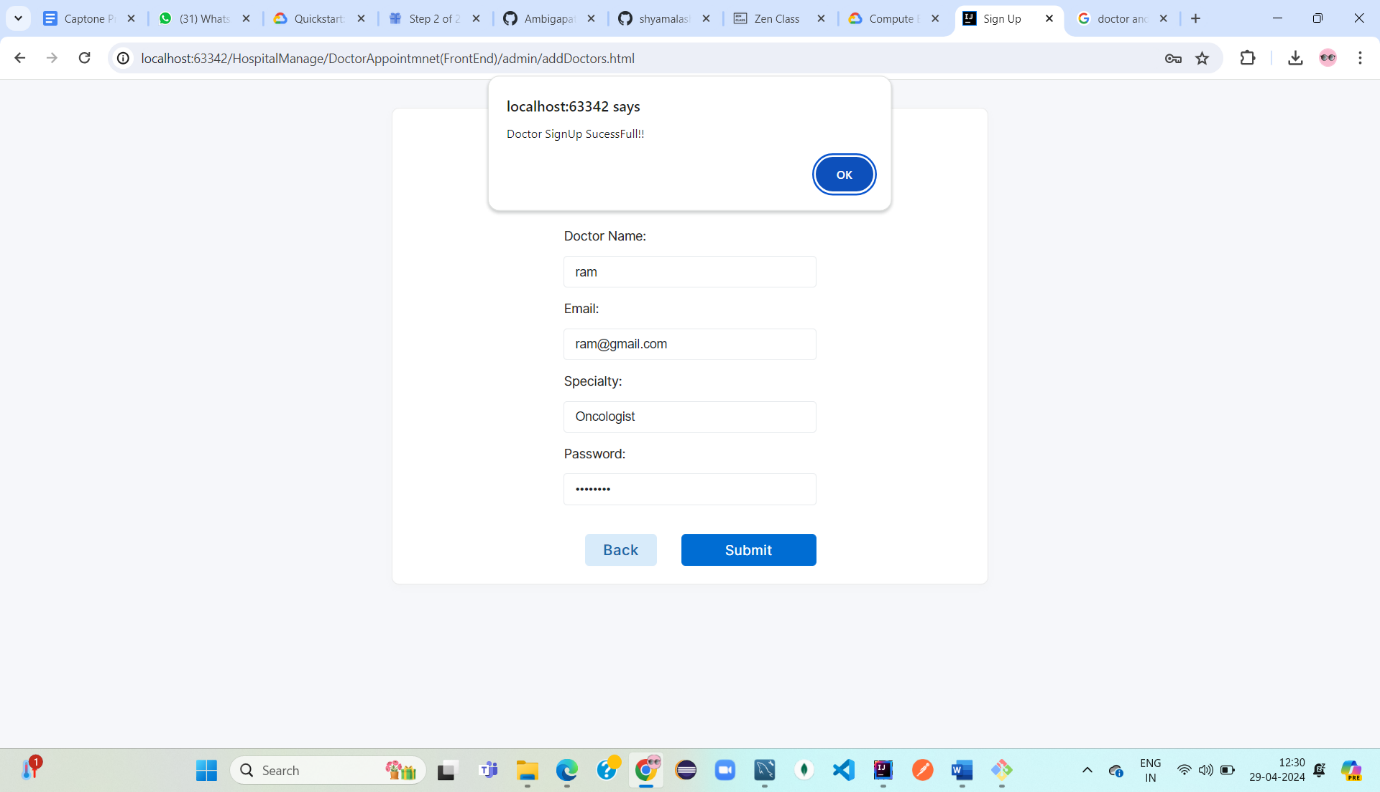


* **Admin can add doctor with doctor details:**

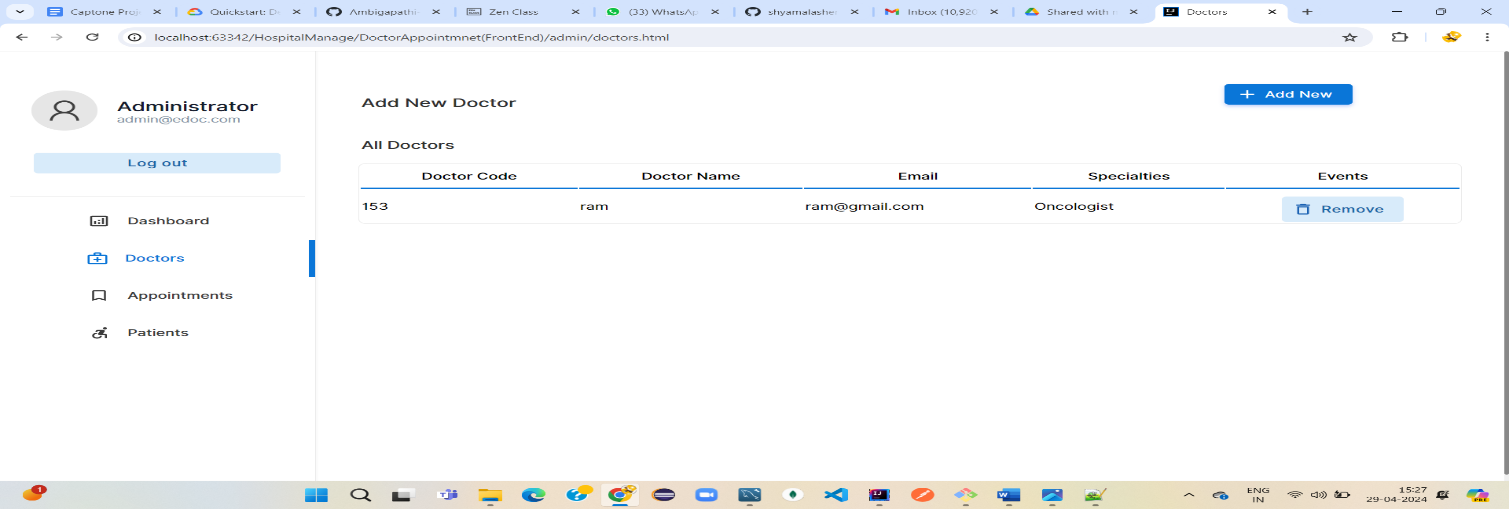
**Before add:**



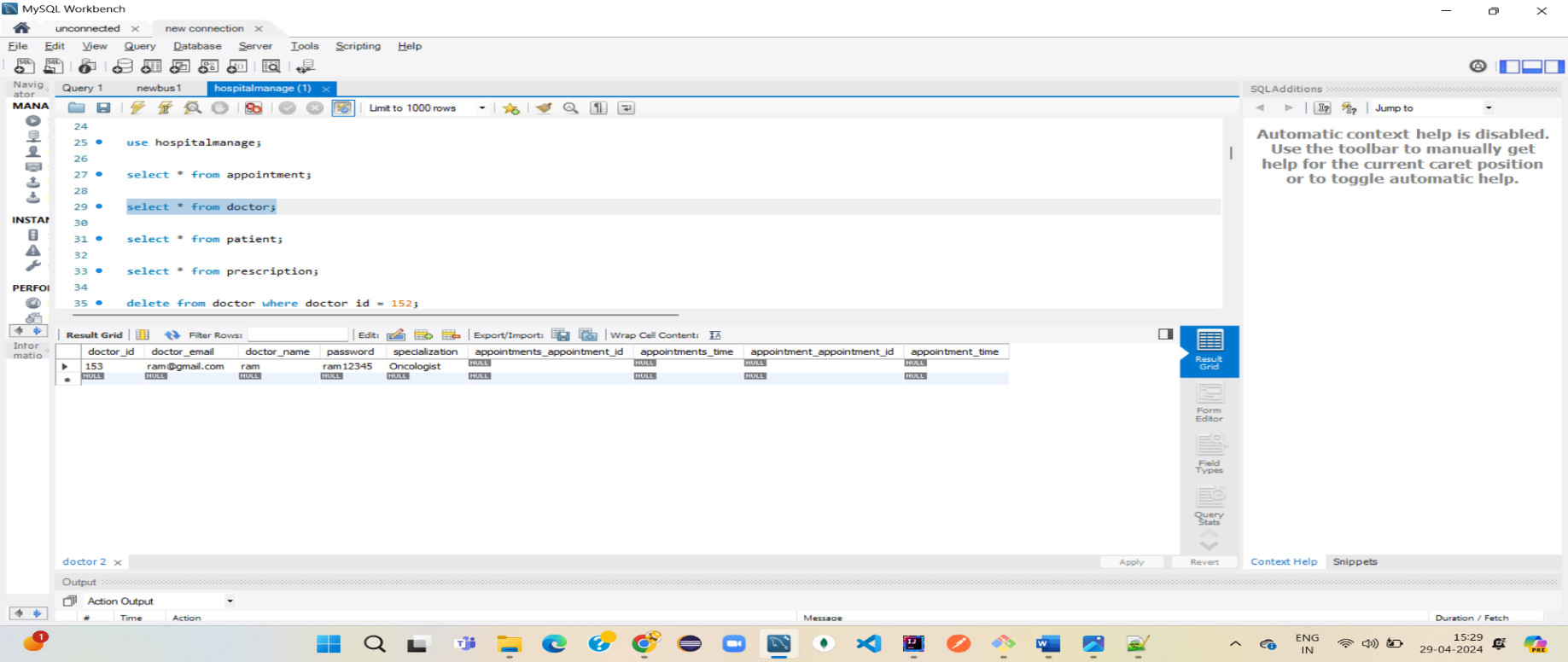
* **After add:**
* **Register doctor with details:**



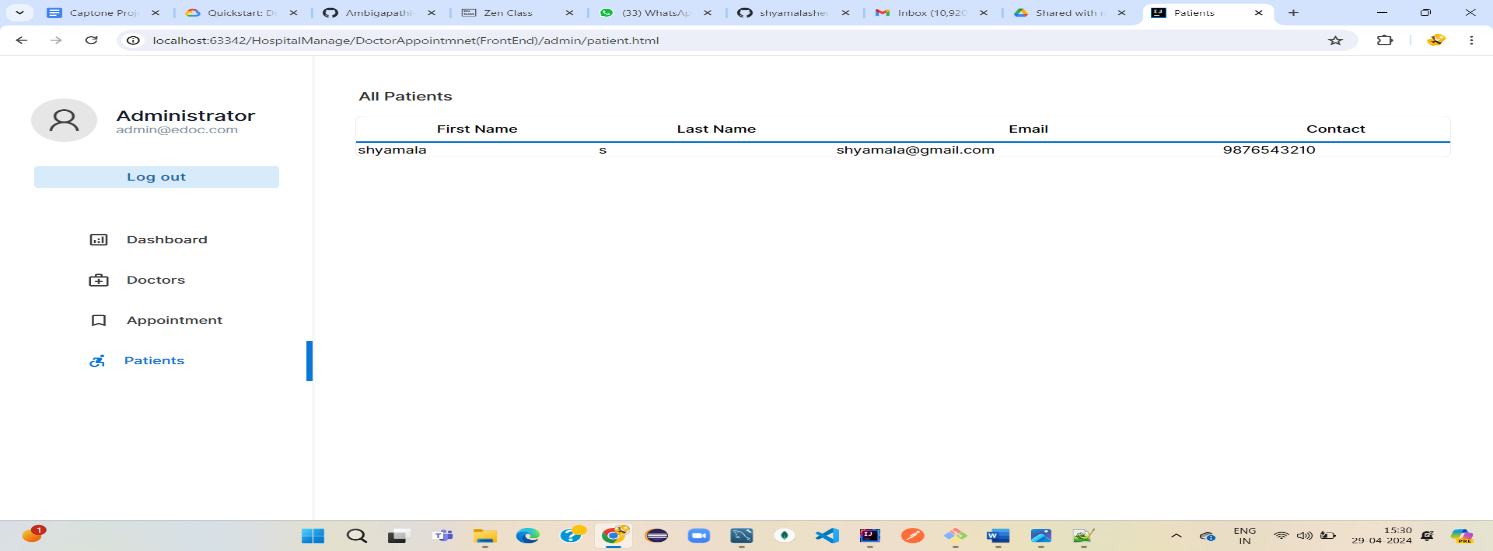
* **Admin added doctor in Doctor List:**



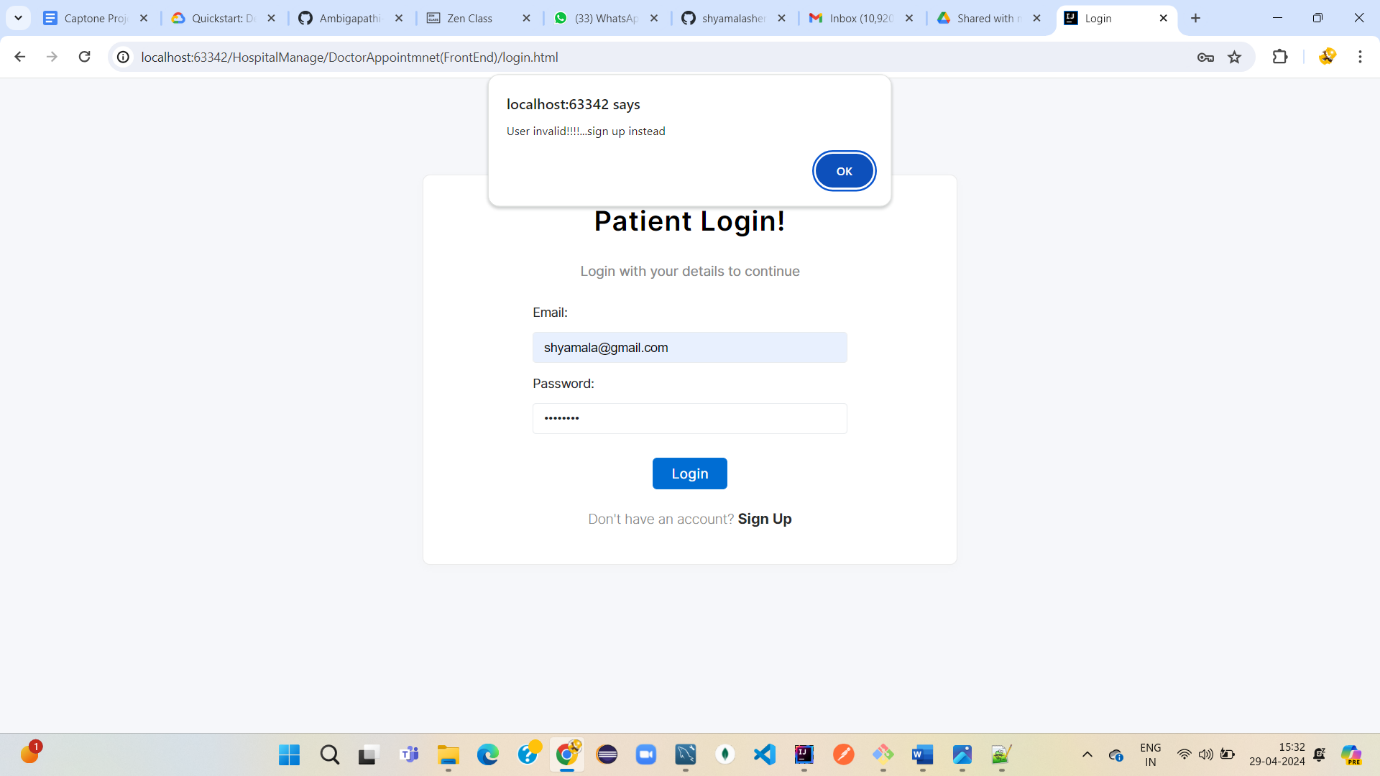
* **Doctor details in DB:**



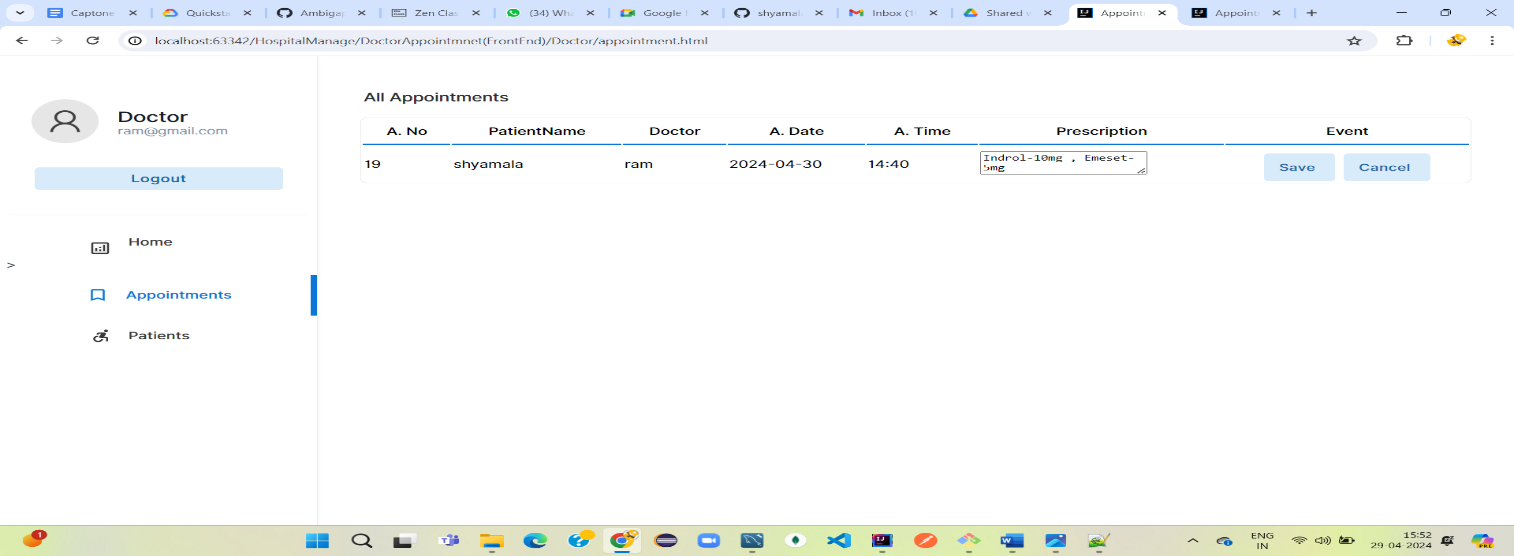
* **Admin can view Registered patient:**



* **If Patient entered wrong password:**



* **Doctor added prescription for patient :**



* **Patient got prescription once doctor added prescription :**

