# How to install the application?

## Required software

* Java 1.8
* Maven 3.3.9
* MySQL 5.7.17 server and MySQL Workbench
* Google Chrome (or another Web Browser)
* NodeJS 6.9.4, NPM 3.10.10 (They’re installed together)
* Angular CLI (can be obtained by running ‘npm install -g @angular/cli’ )

## How to install the required software

### Installing Java

Go to the website (<https://java.com/en/download/>) , download the installer and run it. The project will run on Java 1.8 or newer.

### Installing Maven

* Download Maven from the official website (<https://maven.apache.org/download.cgi>). Version 3.3.9 was used but newer ones work as well
* Extract the downloaded archive
* Add the M2\_HOME and MAVEN\_HOME variables to your environment. Make them point to the path where you just extracted Maven
* Add %M2\_HOME%\bin to your path so you will be able to run maven from the command line

### Installing MySQL

* Download MySQL installer from the website (<https://dev.mysql.com/downloads/installer/>)
* Run the installer and follow the instructions
* When asked chose to install MySQL Workbench alongside with the MySQL server
* To start the MySQL server (on Windows) open the Task Manager, go to the Services tab and look for the MySQL57 process. Right-click that process and select Start

### Installing NodeJS

* Go to <https://nodejs.org/en/> and download the latest version of the installer
* Run the installer and follow the instructions. When asked if you want to install NPM too chose to do so (by default this option is already checked)
* Open a terminal and run “npm install npm -g –upgrade” to update NPM to the latest version
* Run “npm install -g @angular/cli” to also install angular-cli

## How to install the application

* Use MySQL Workbench to connect to the MySQL Server
* Run the init.sql initialization script
* Create a new MySQL user with username=”assignment” and password=”softwaredesign” and give this new user full rights on the newly created schema (assignment3)
* Go into the folder containing the backend application of the project (the ‘implementation/backend’ folder) in IntelliJ
* Open a terminal in that folder and run “mvn install”
* Wait for maven to build the project
* Run the newly created jar (it will be created in the /target folder)
* Go into the ‘implementation/frontend’ folder and open a terminal
* Run ‘npm install’ to install the required node modules (they will be installed based on package.json)
* Run the ‘ng serve’ command
* Wait for the front-end application to be ready
* Open a web browser and go to localhost:4200

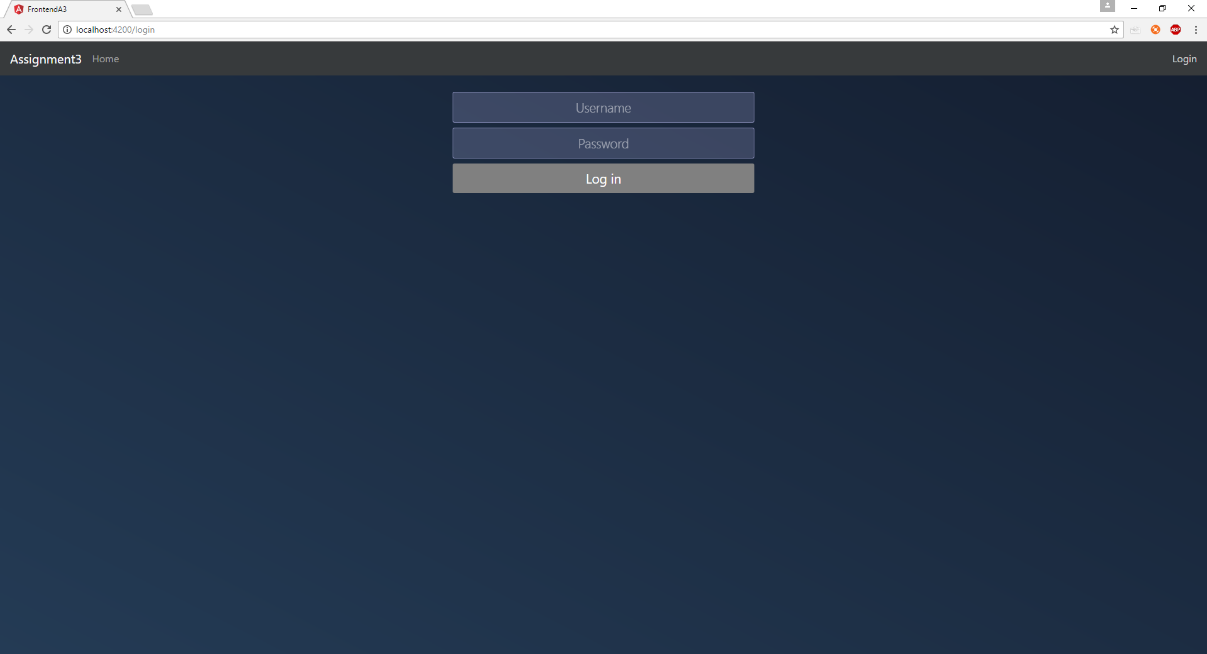
# How to use the application?

The home page of the application looks like this

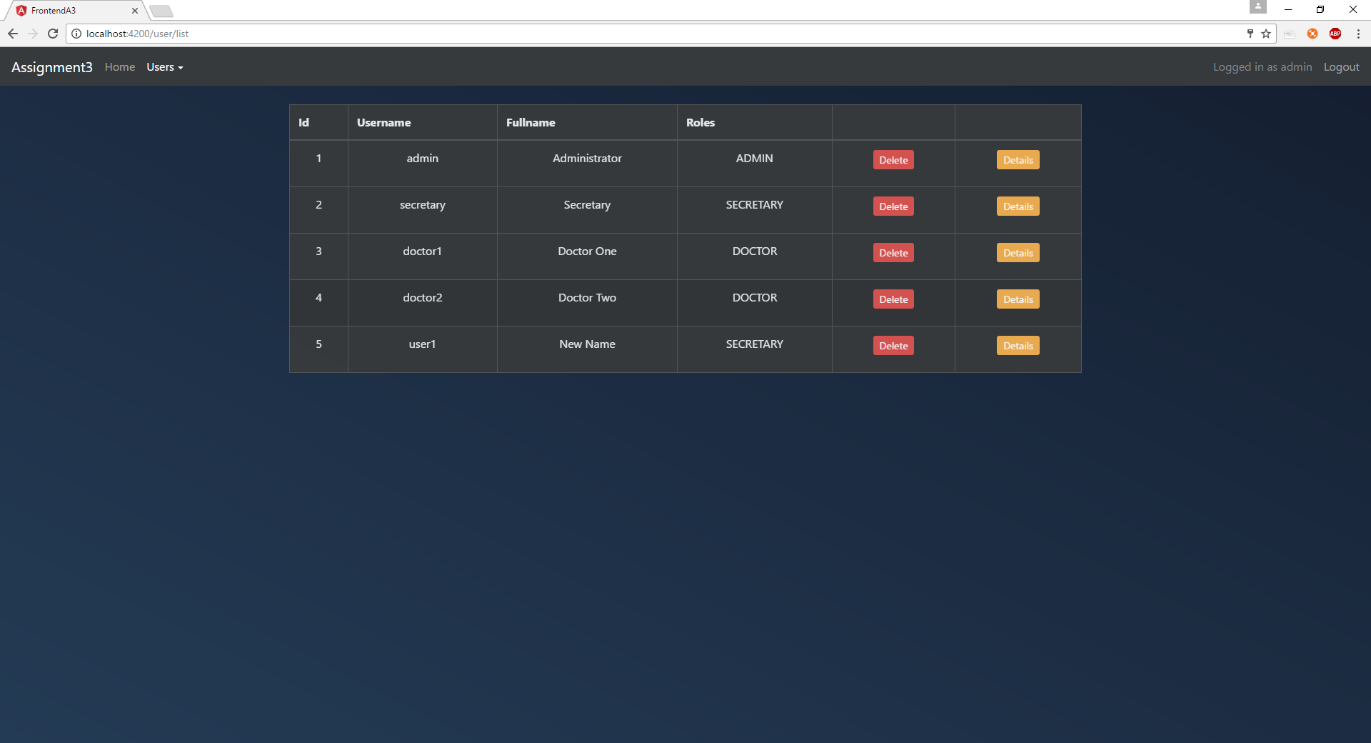
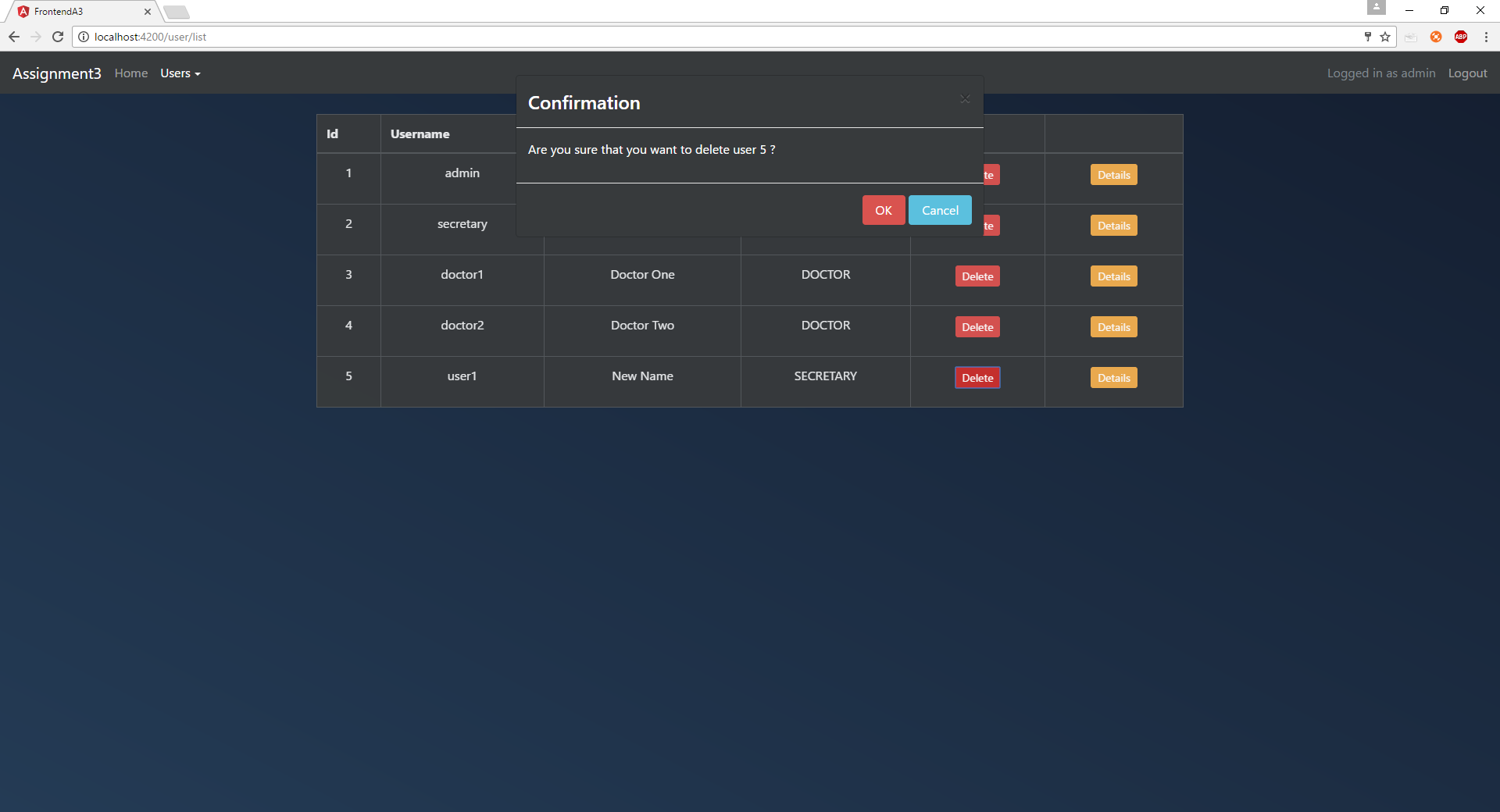
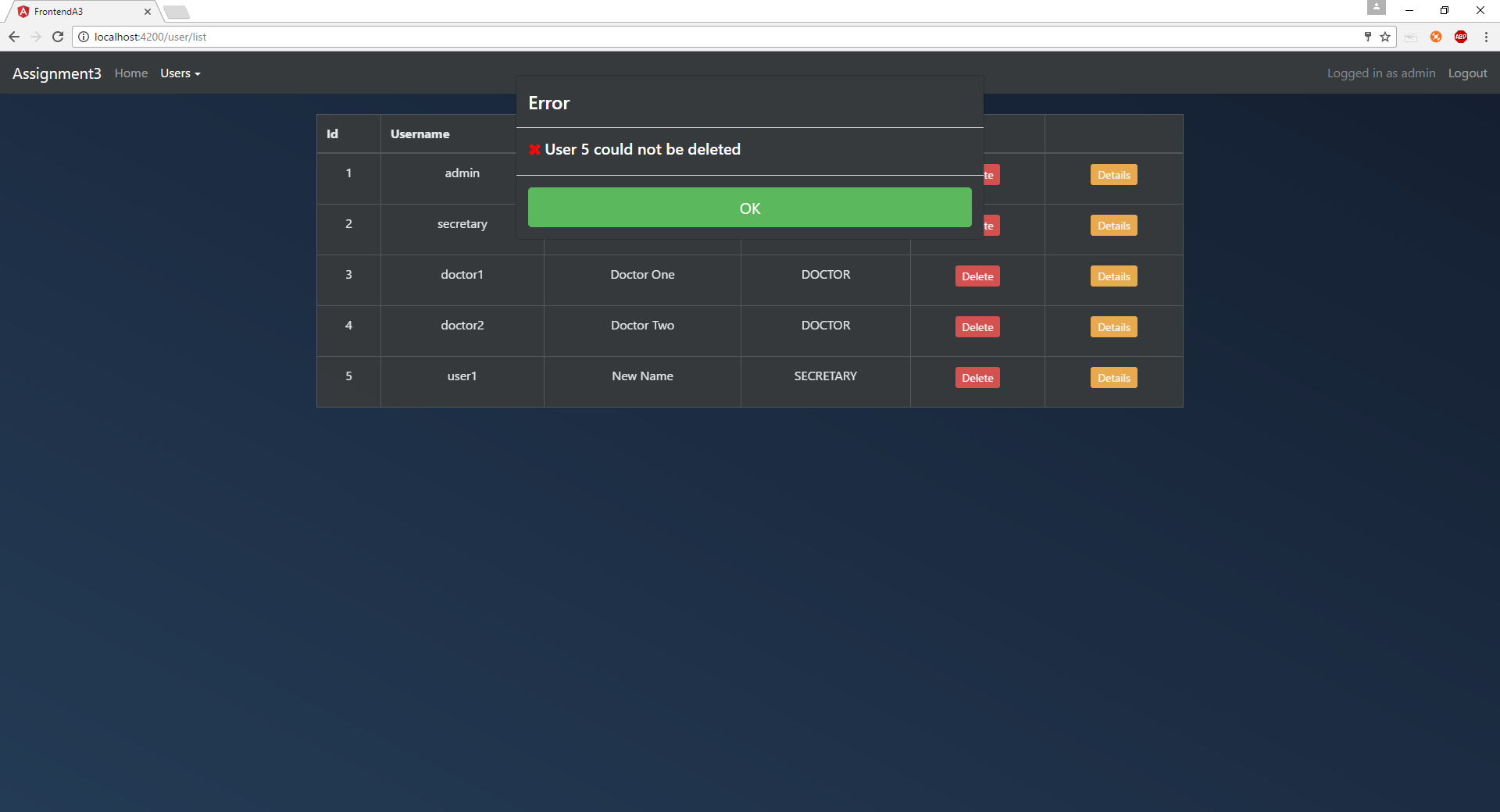
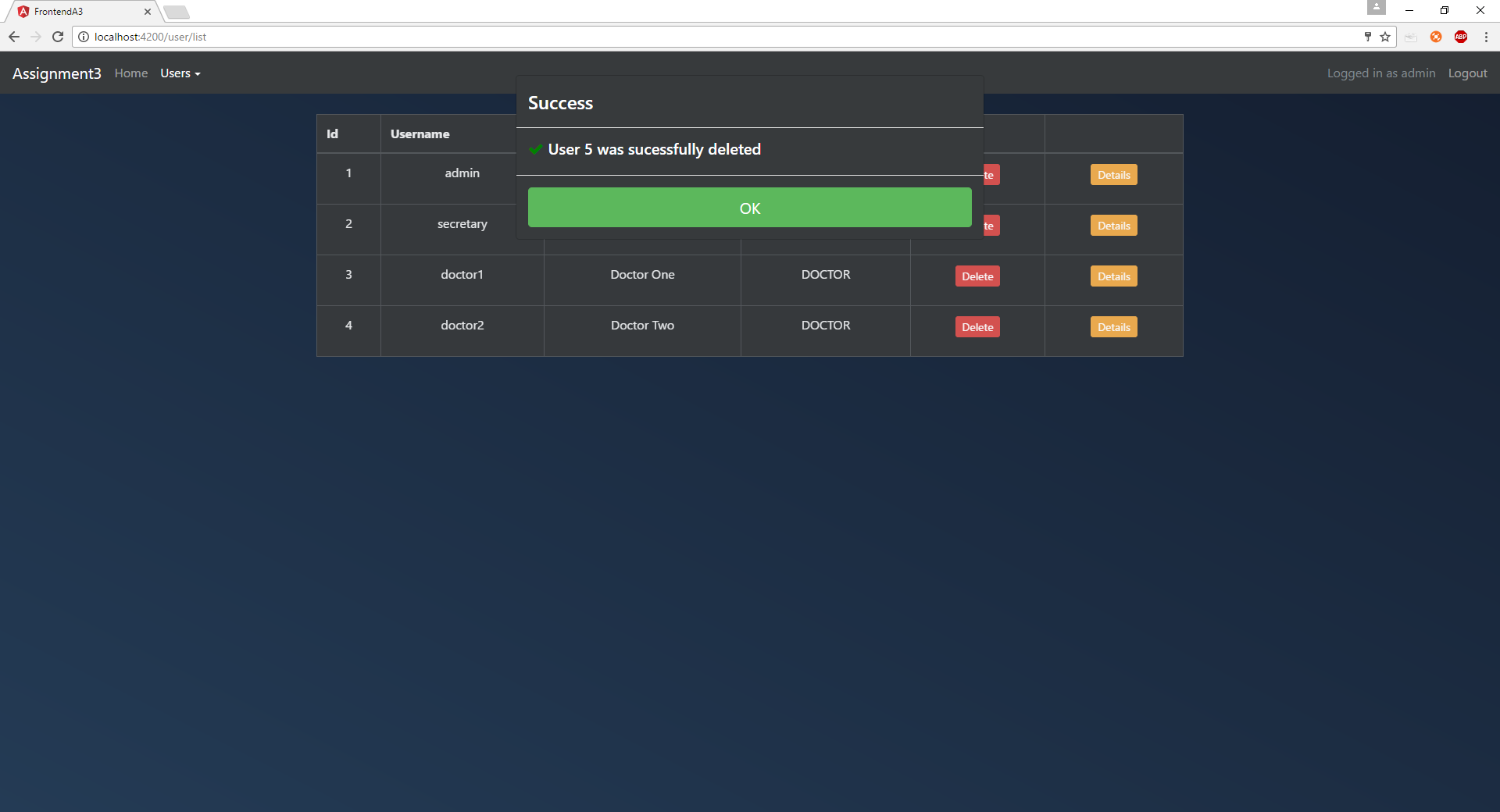
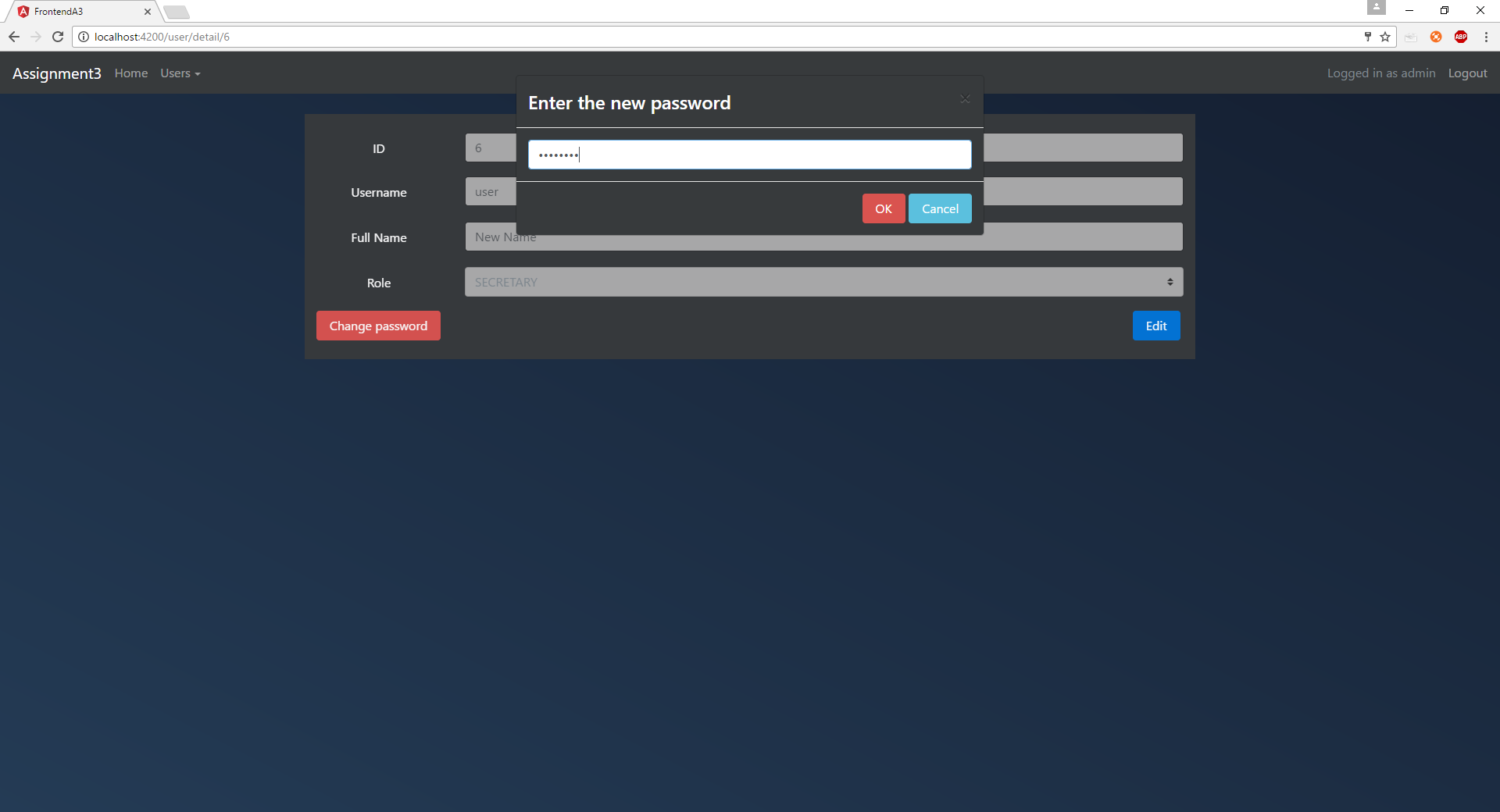
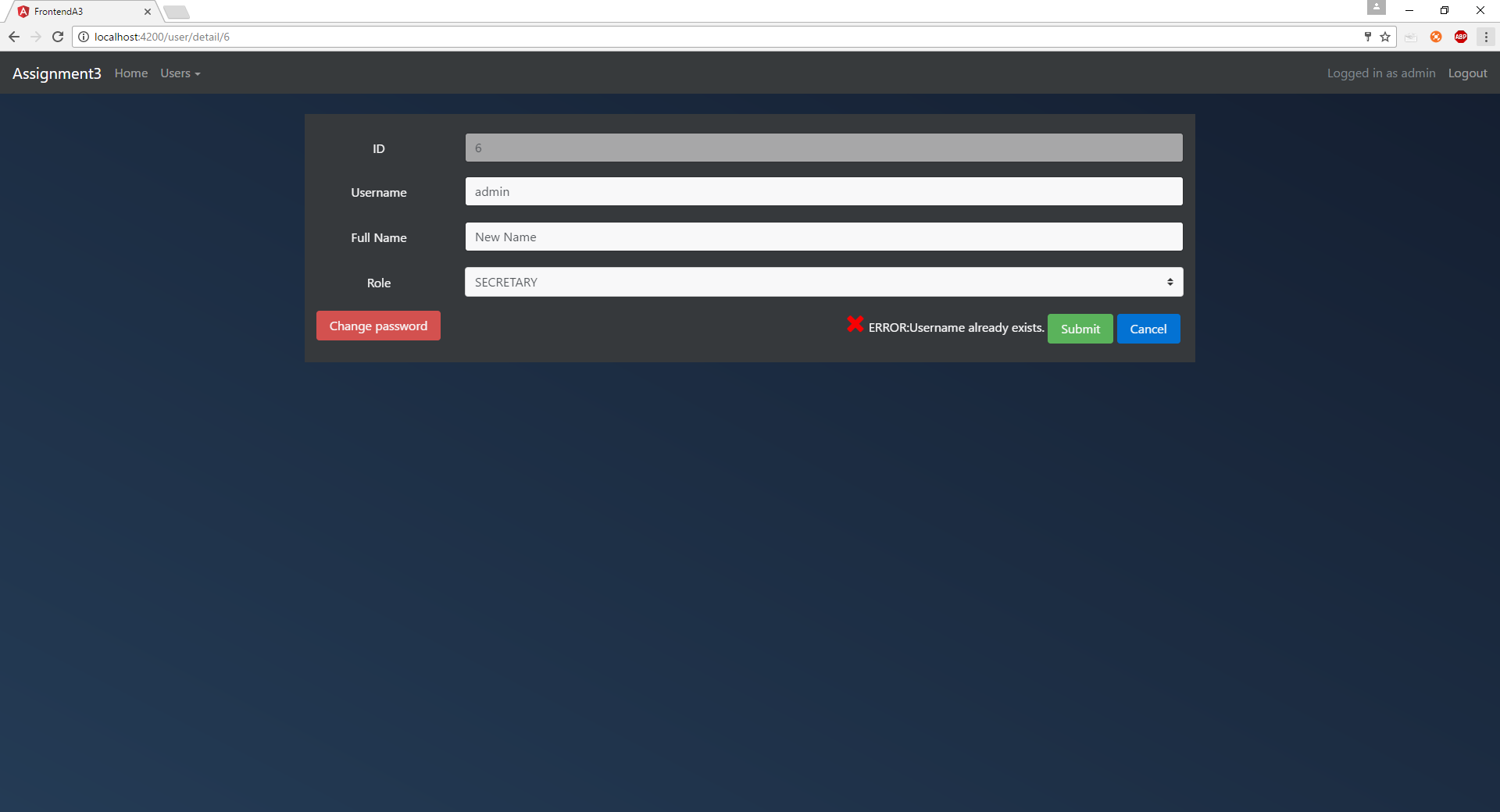
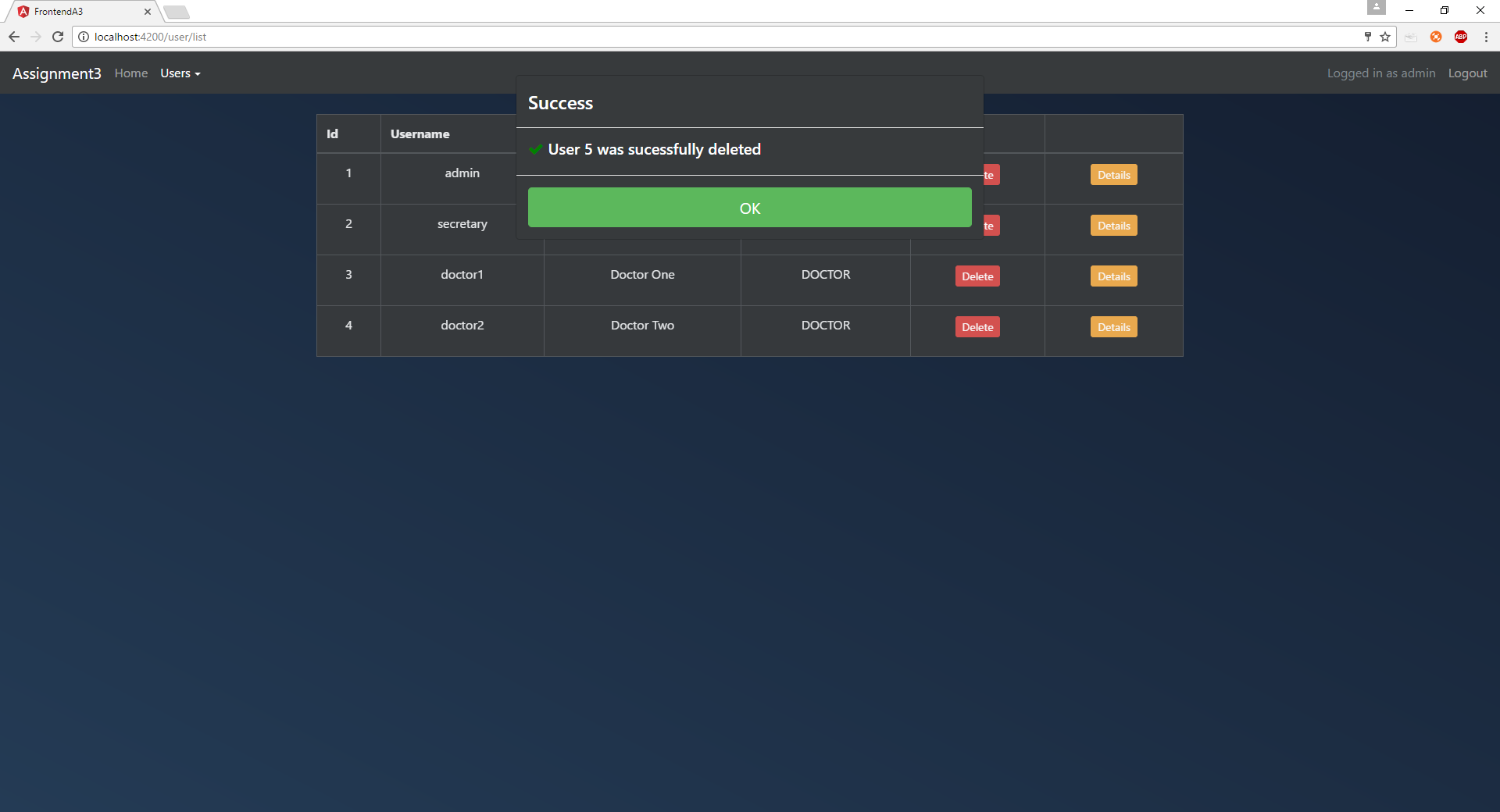
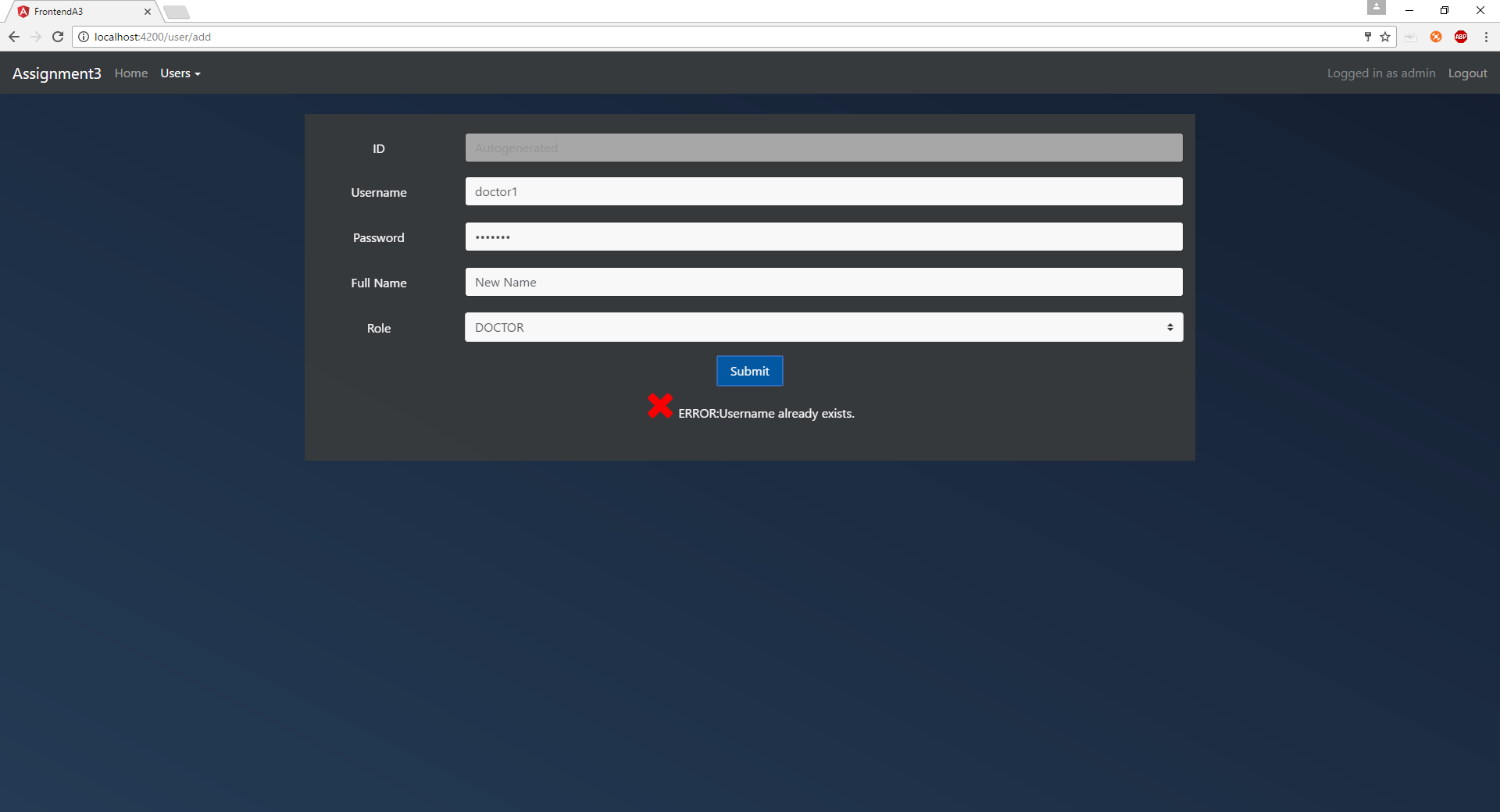
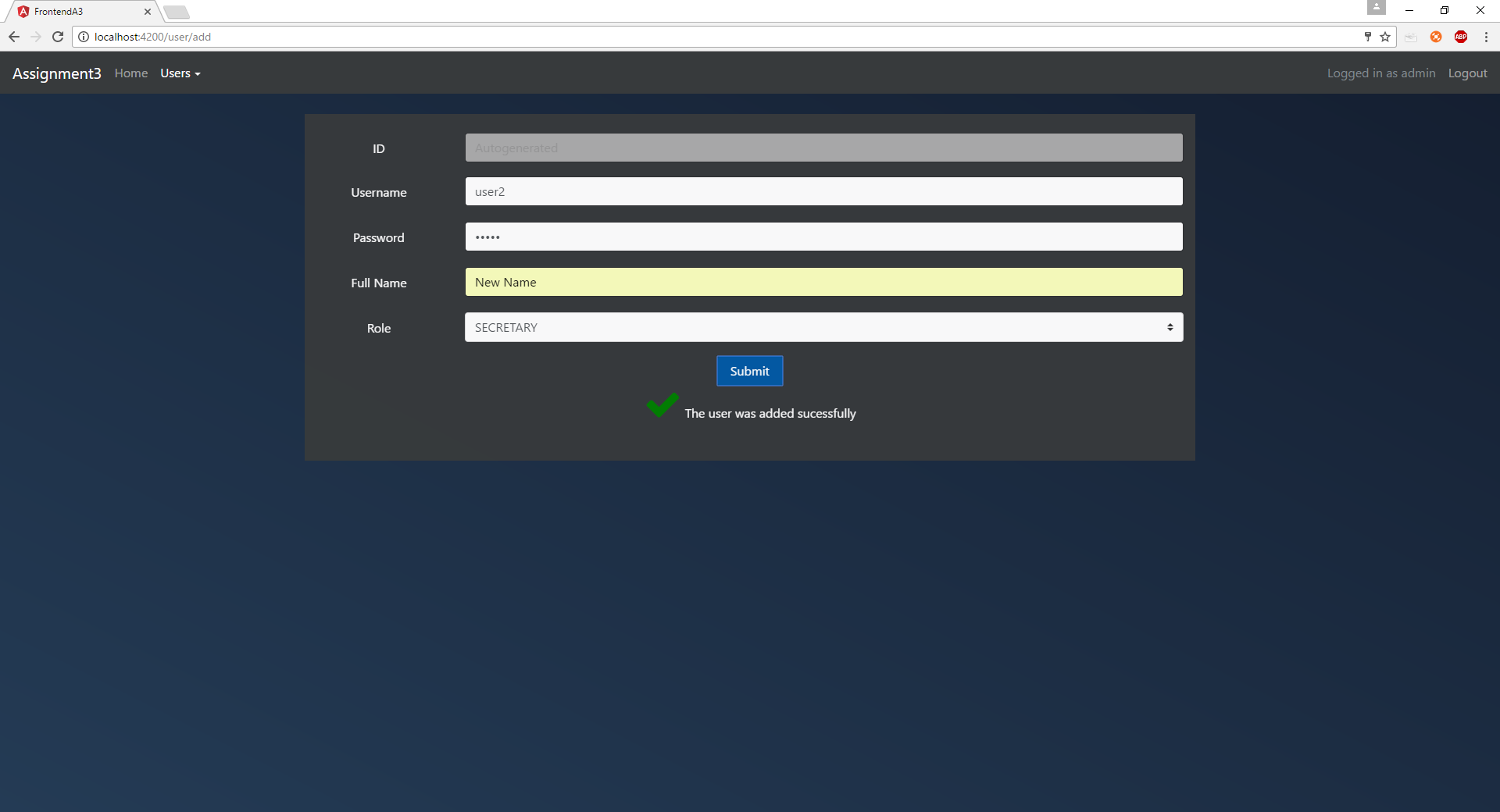
The login page can be reached by clicking the “Login” button in the upper-right corner. The users are able to log into the application by entering their credentials and clicking the “Log In” button.

The initialization script created four users that can be used for logging in:

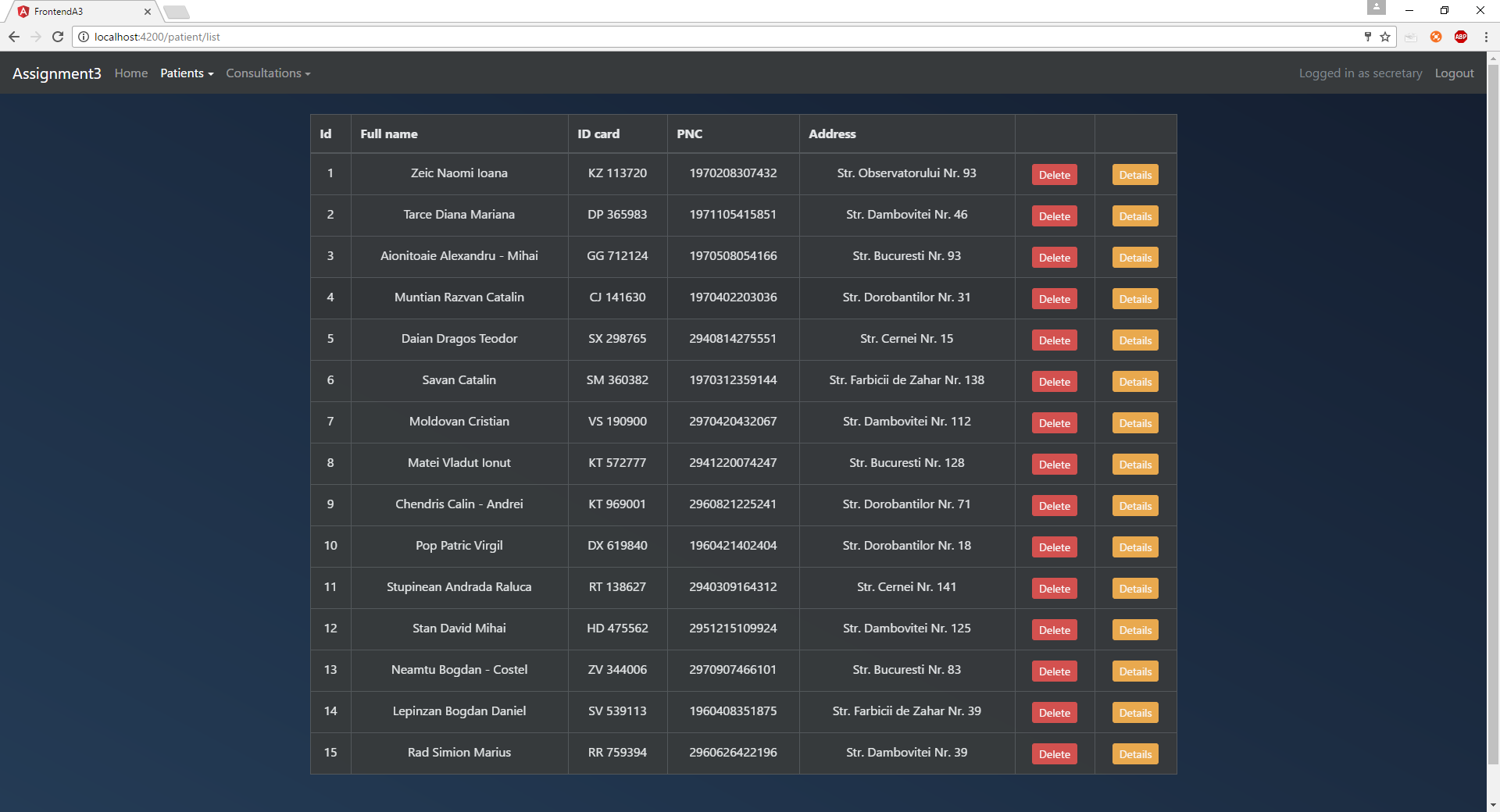
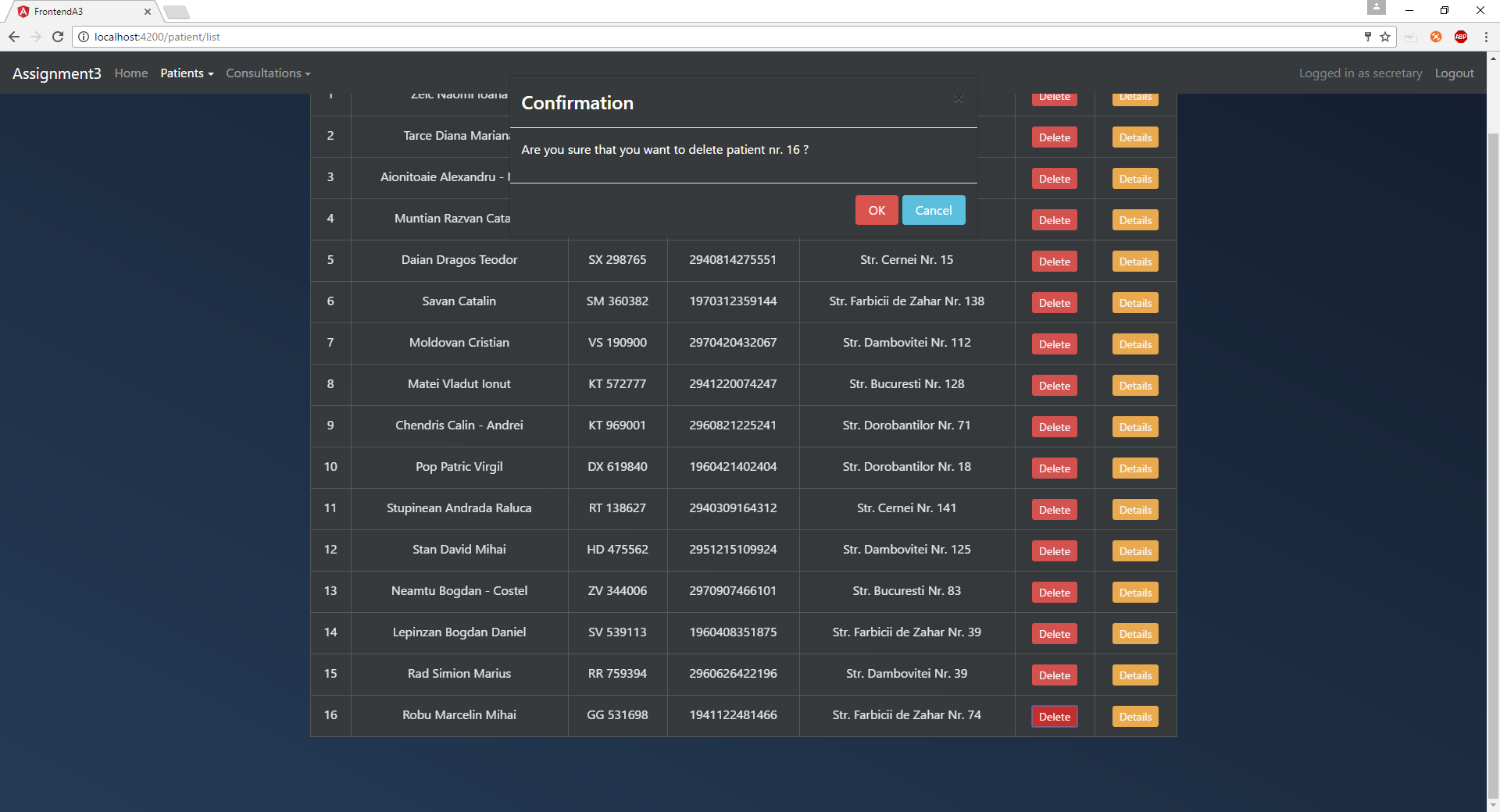
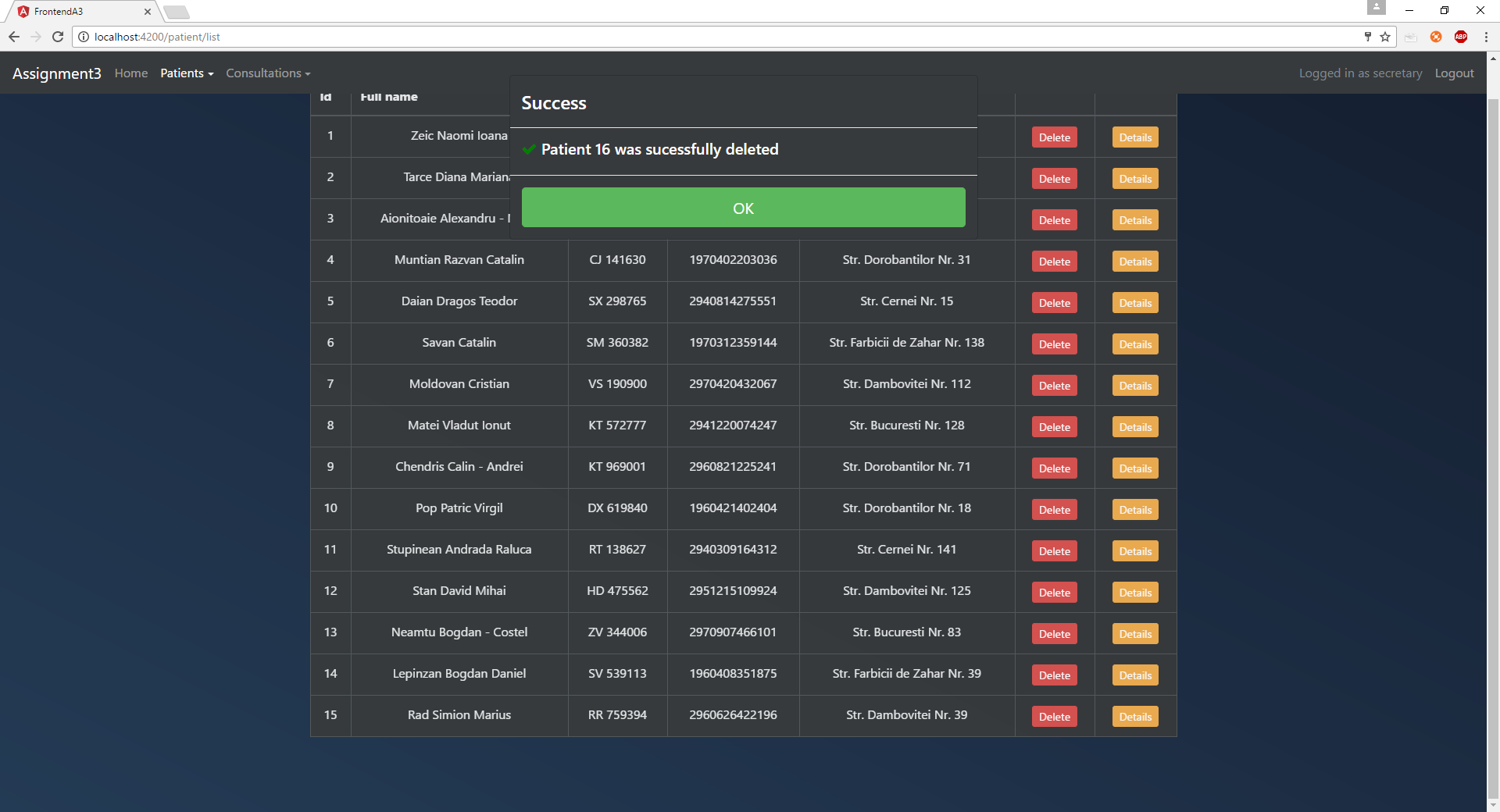
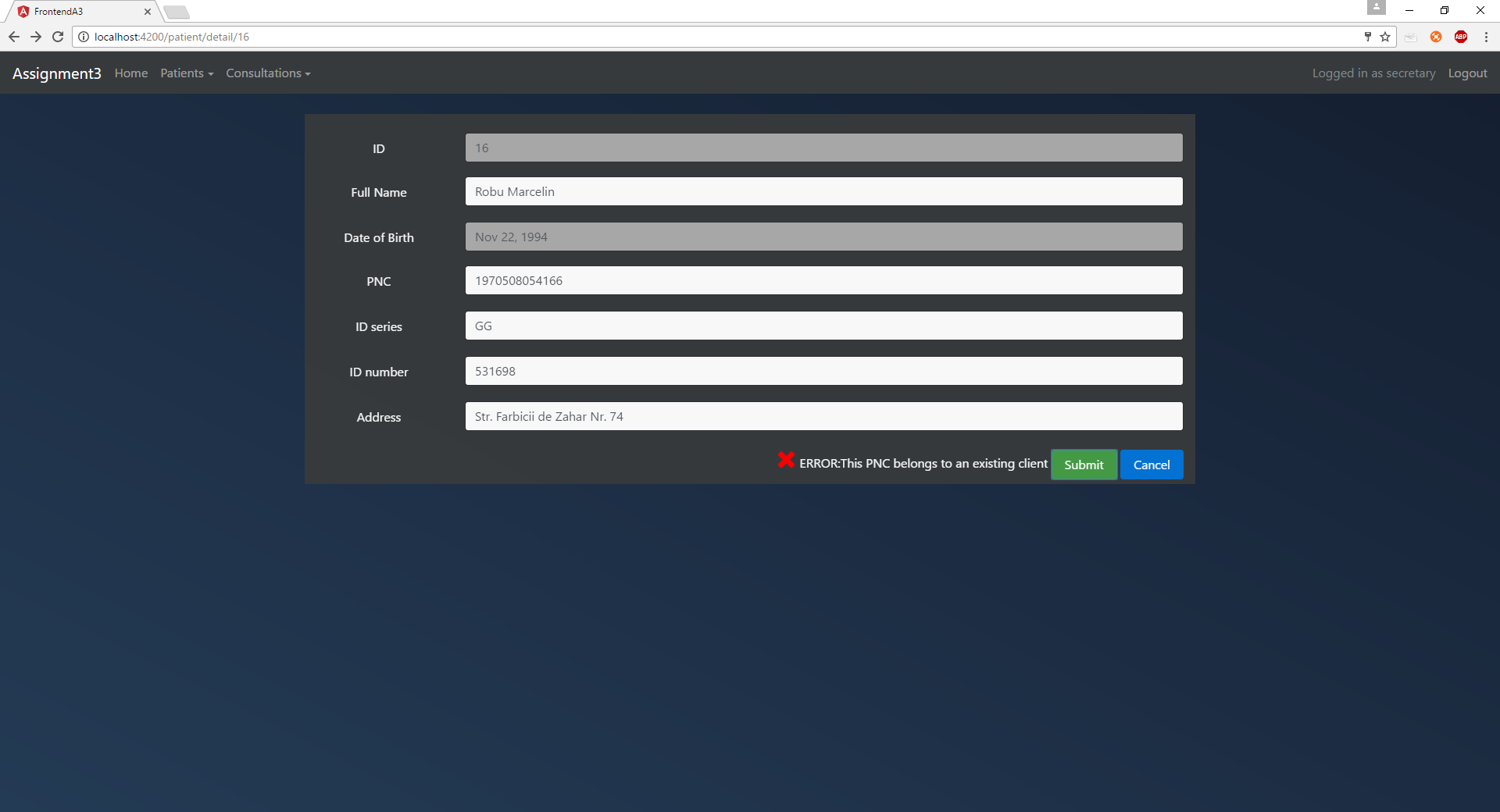
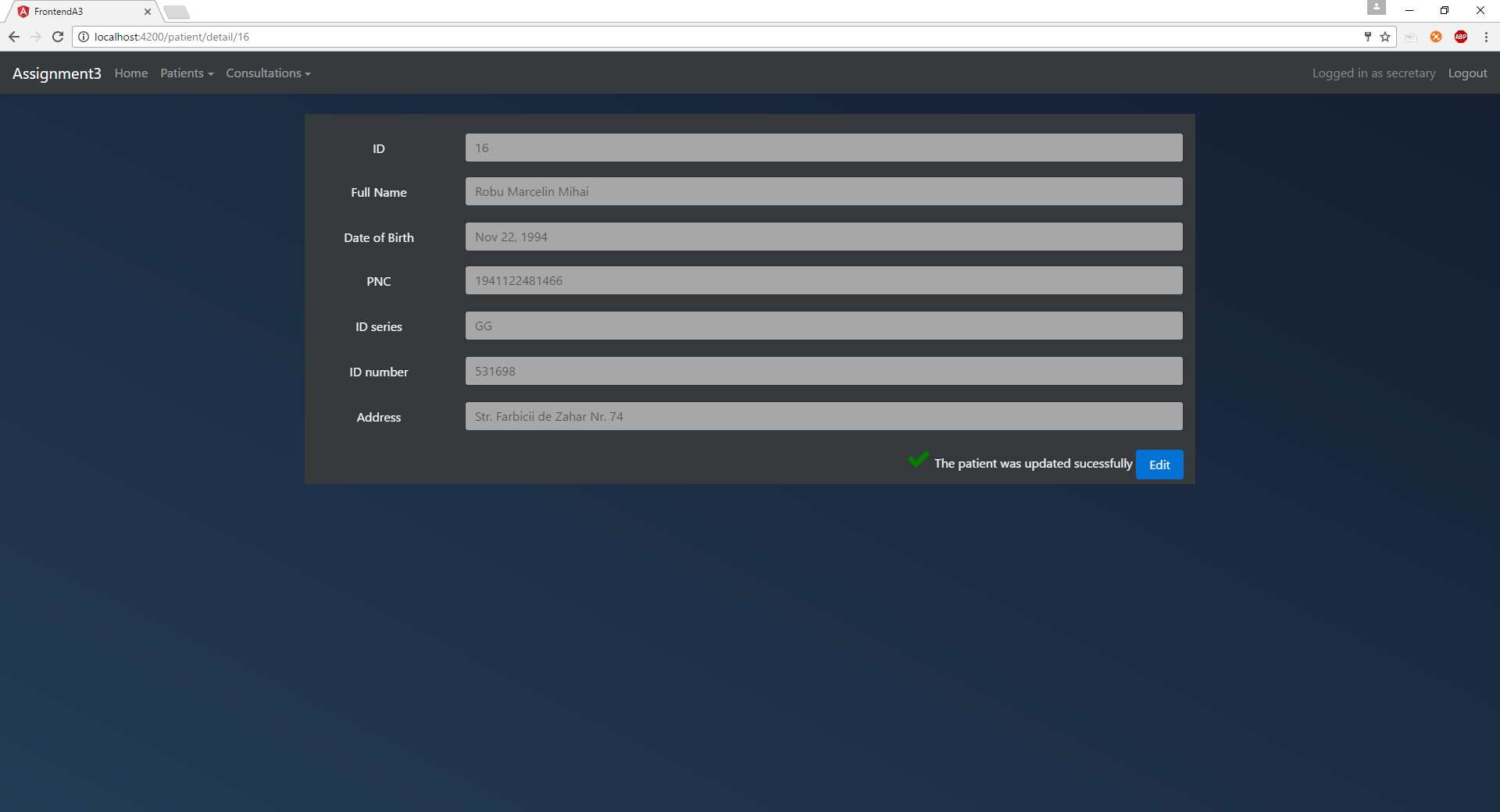
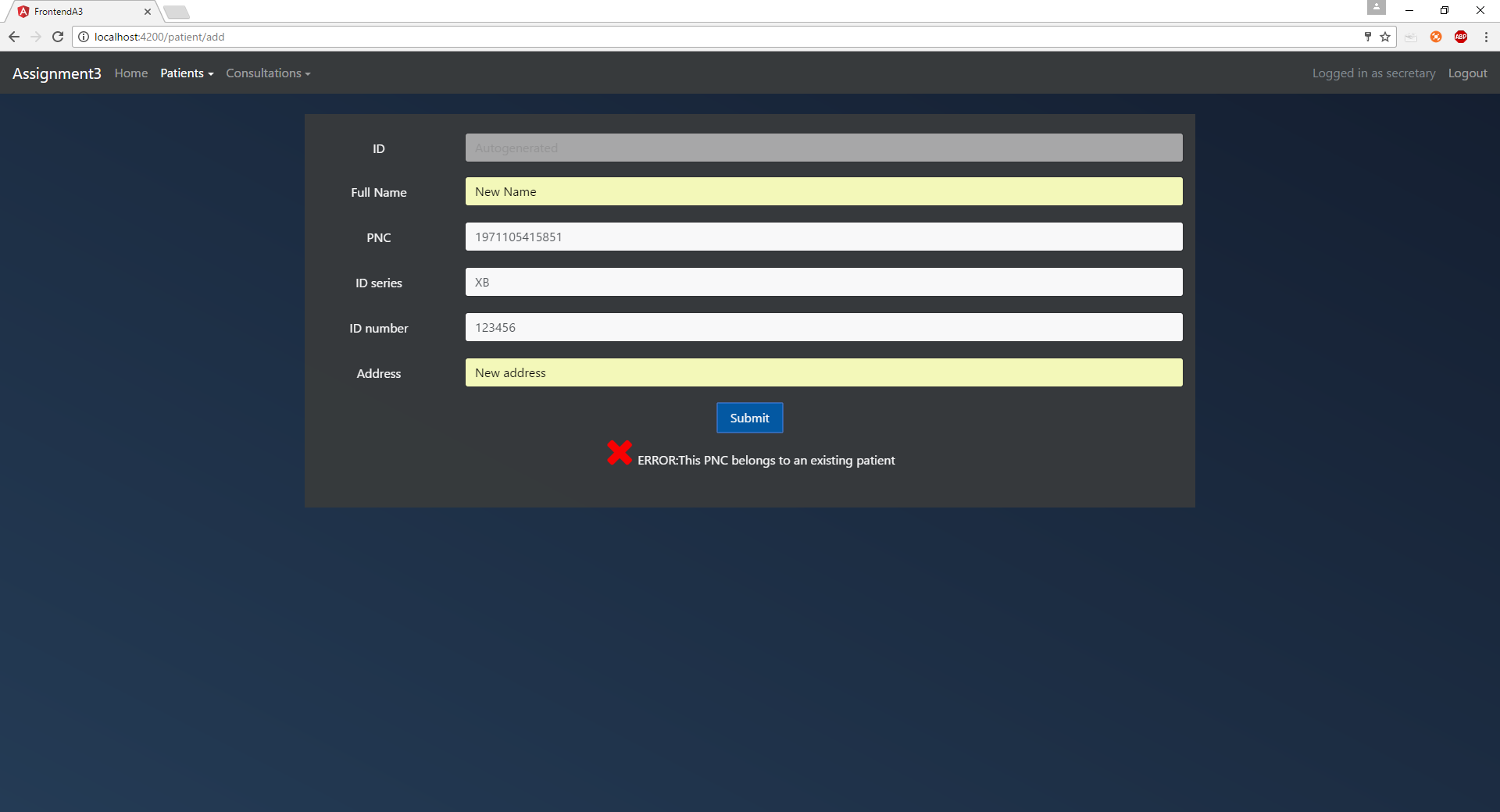
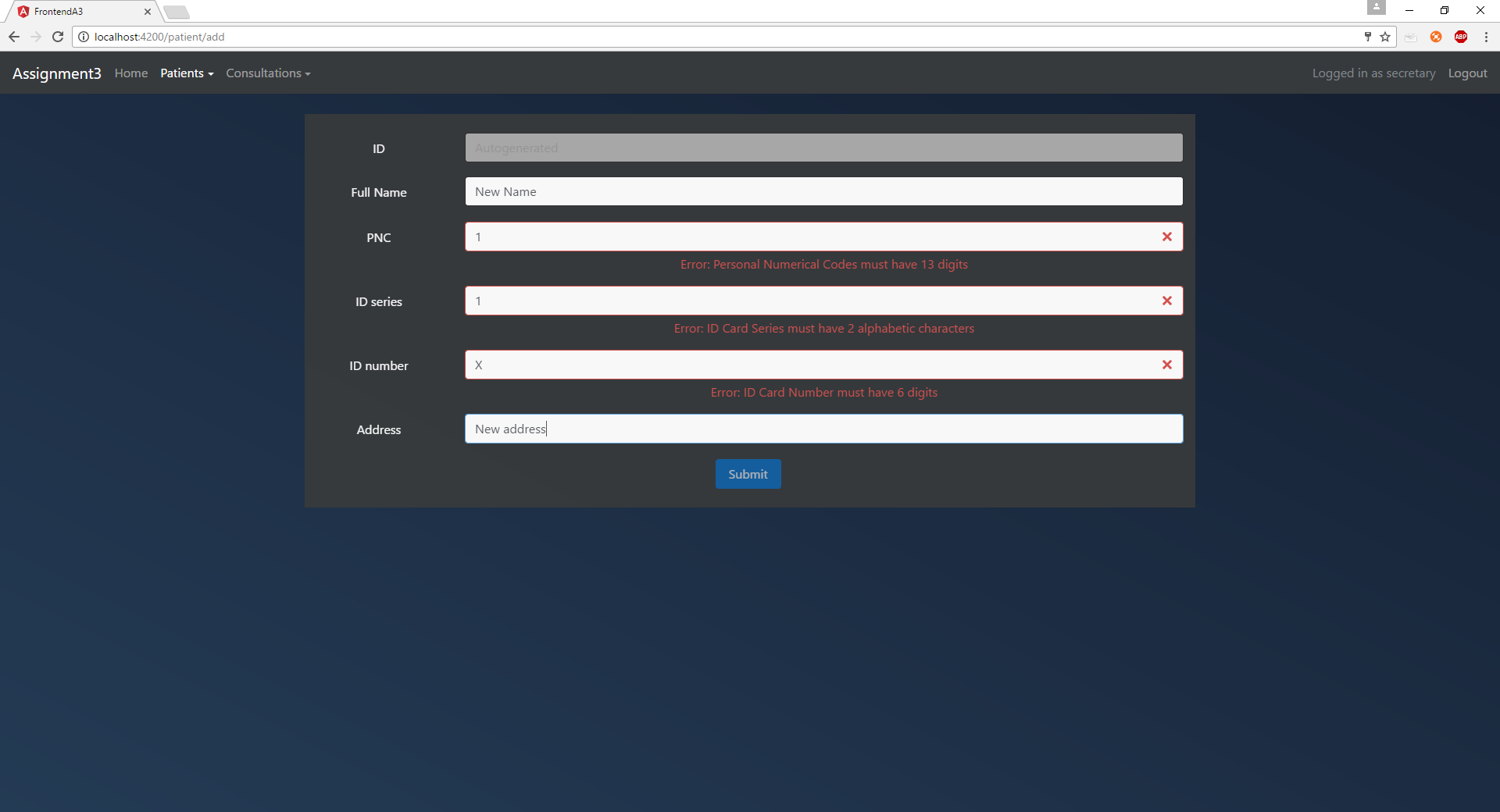
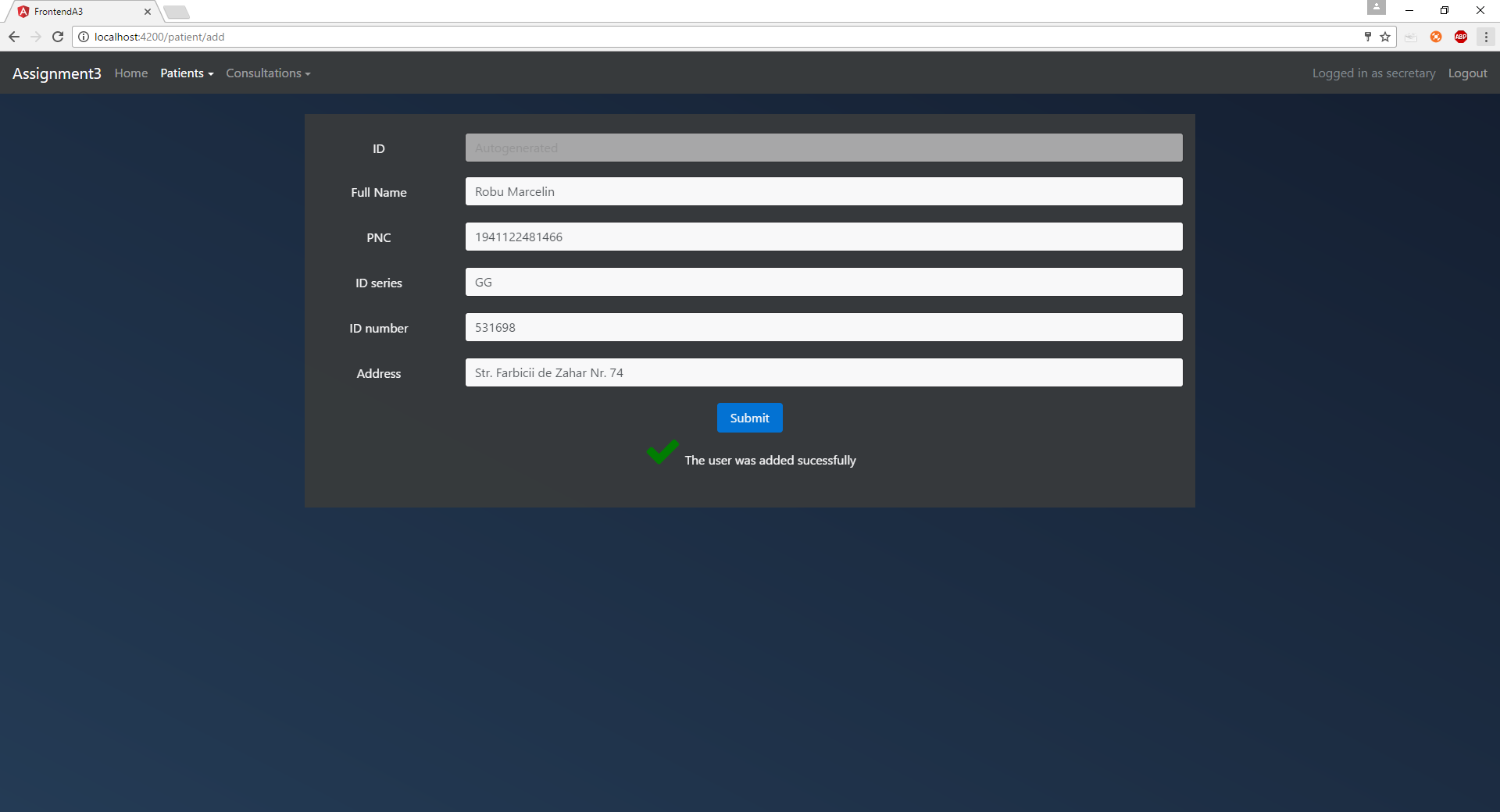
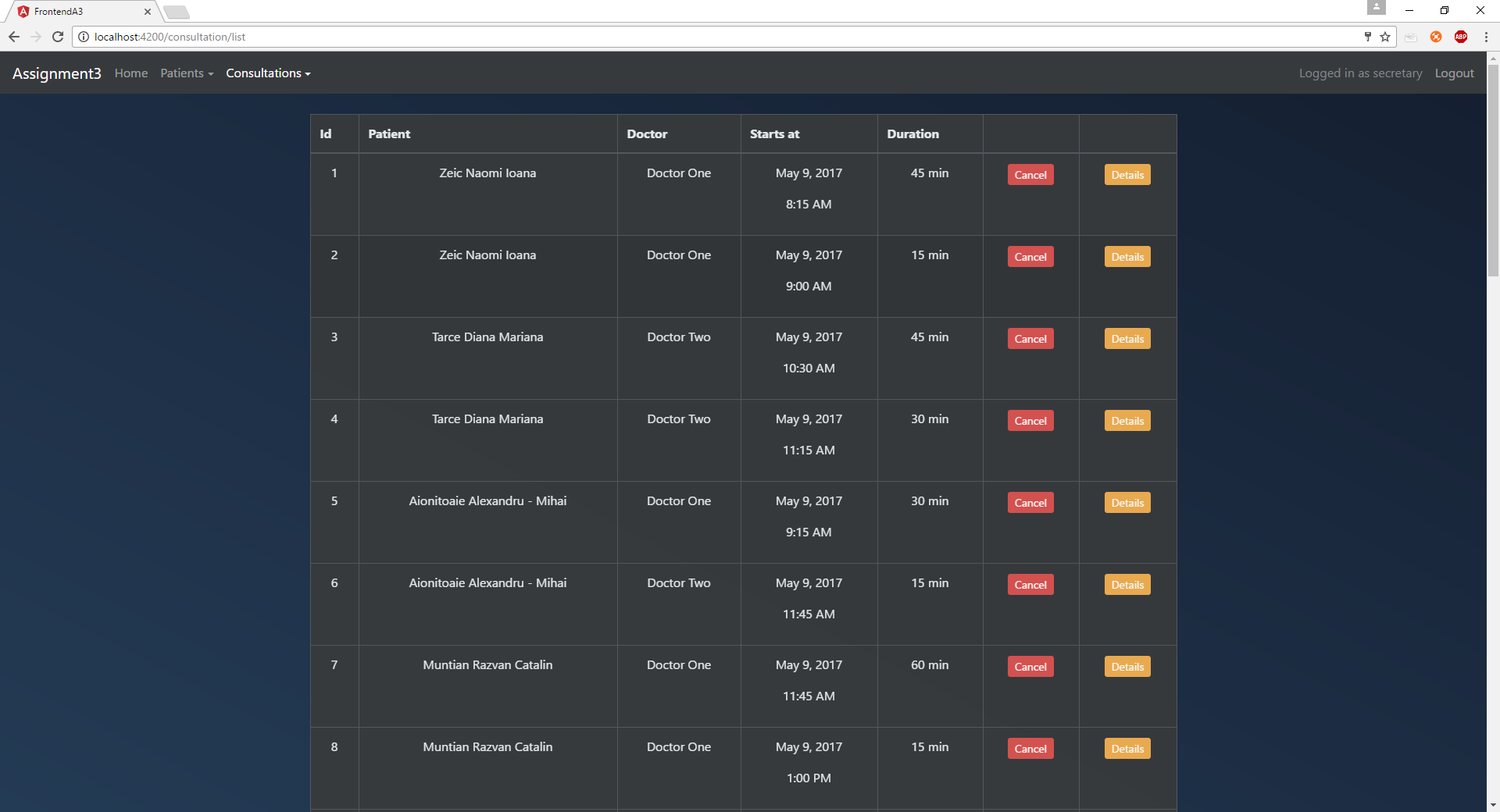
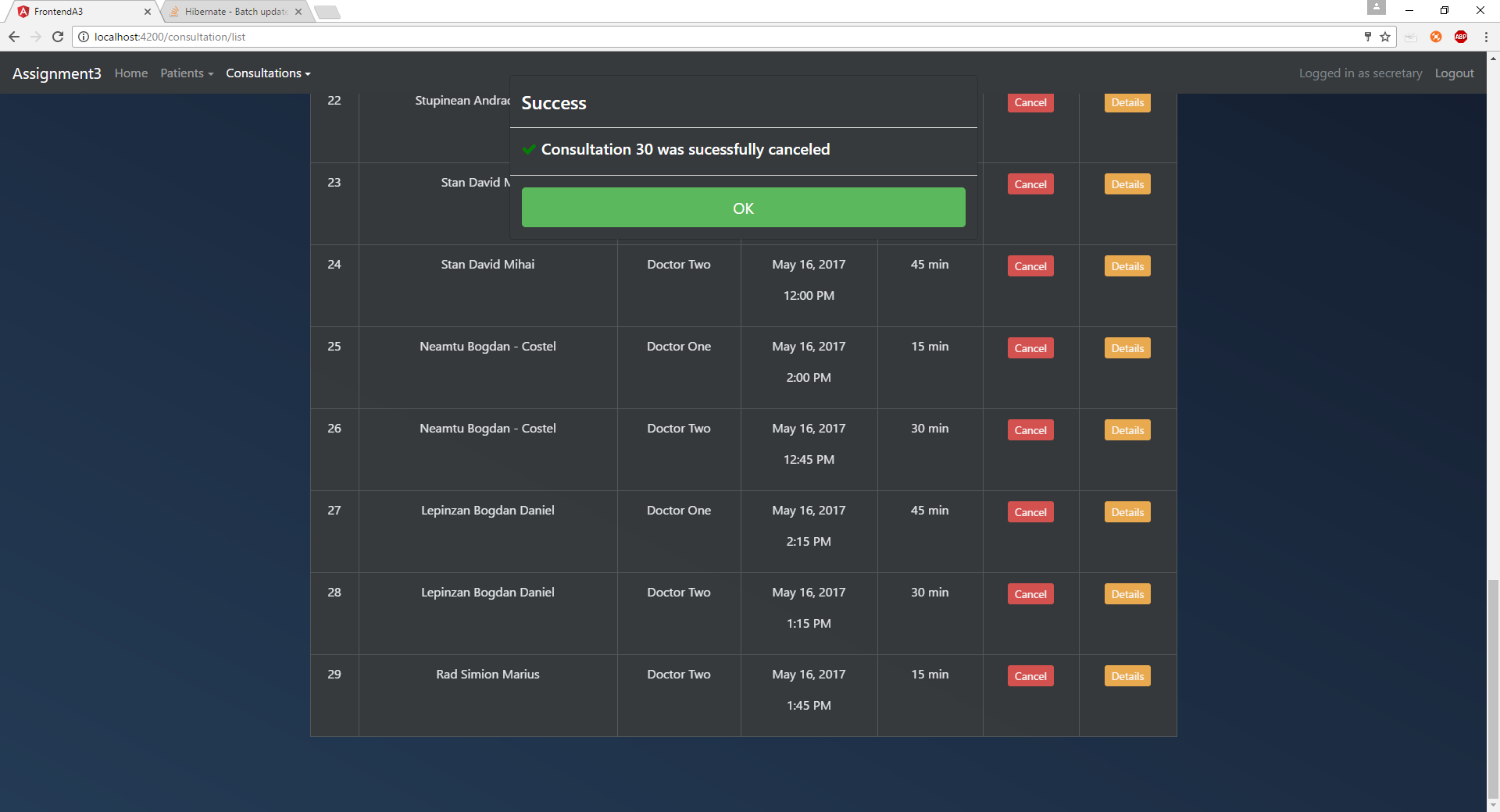
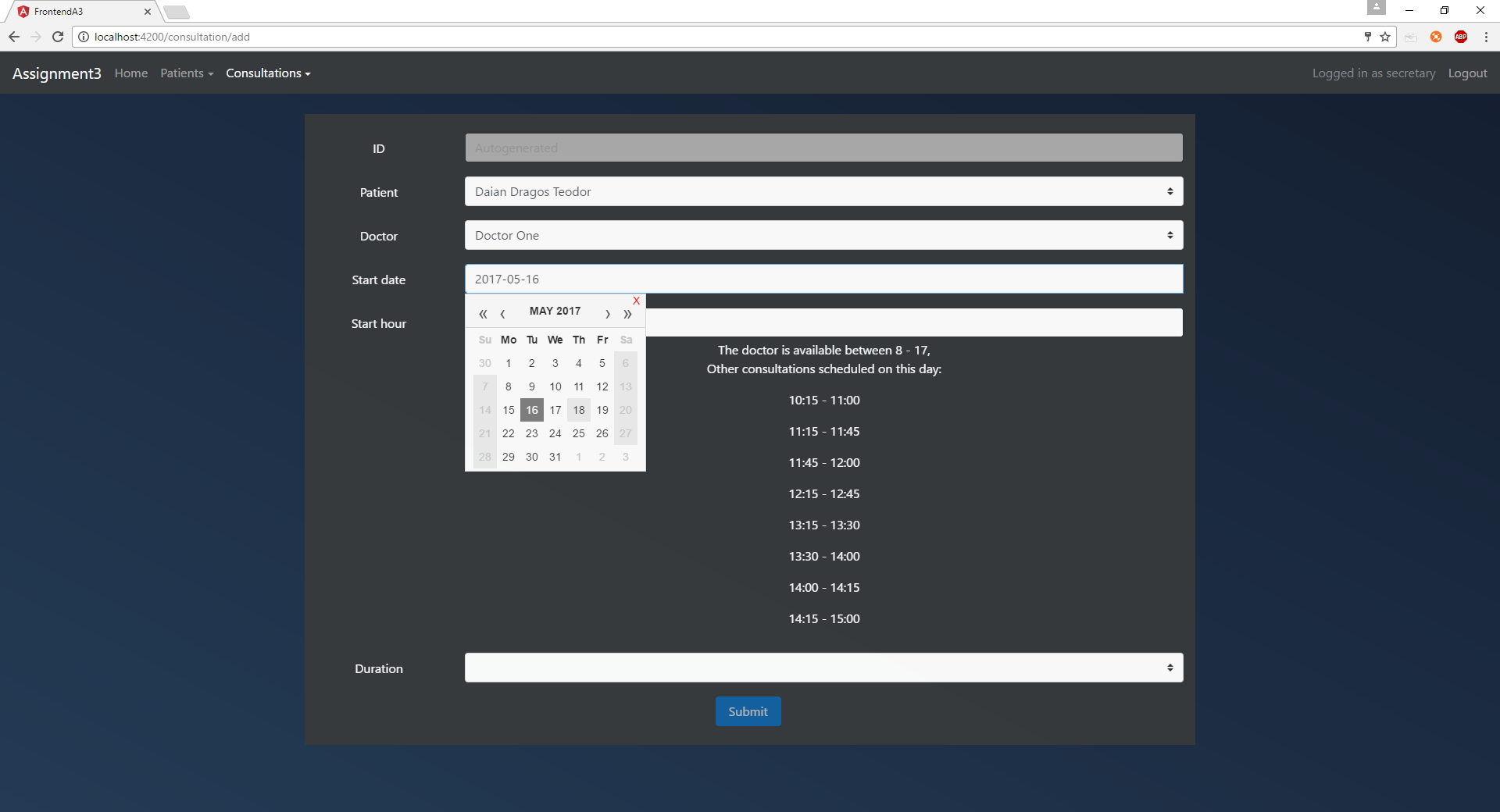
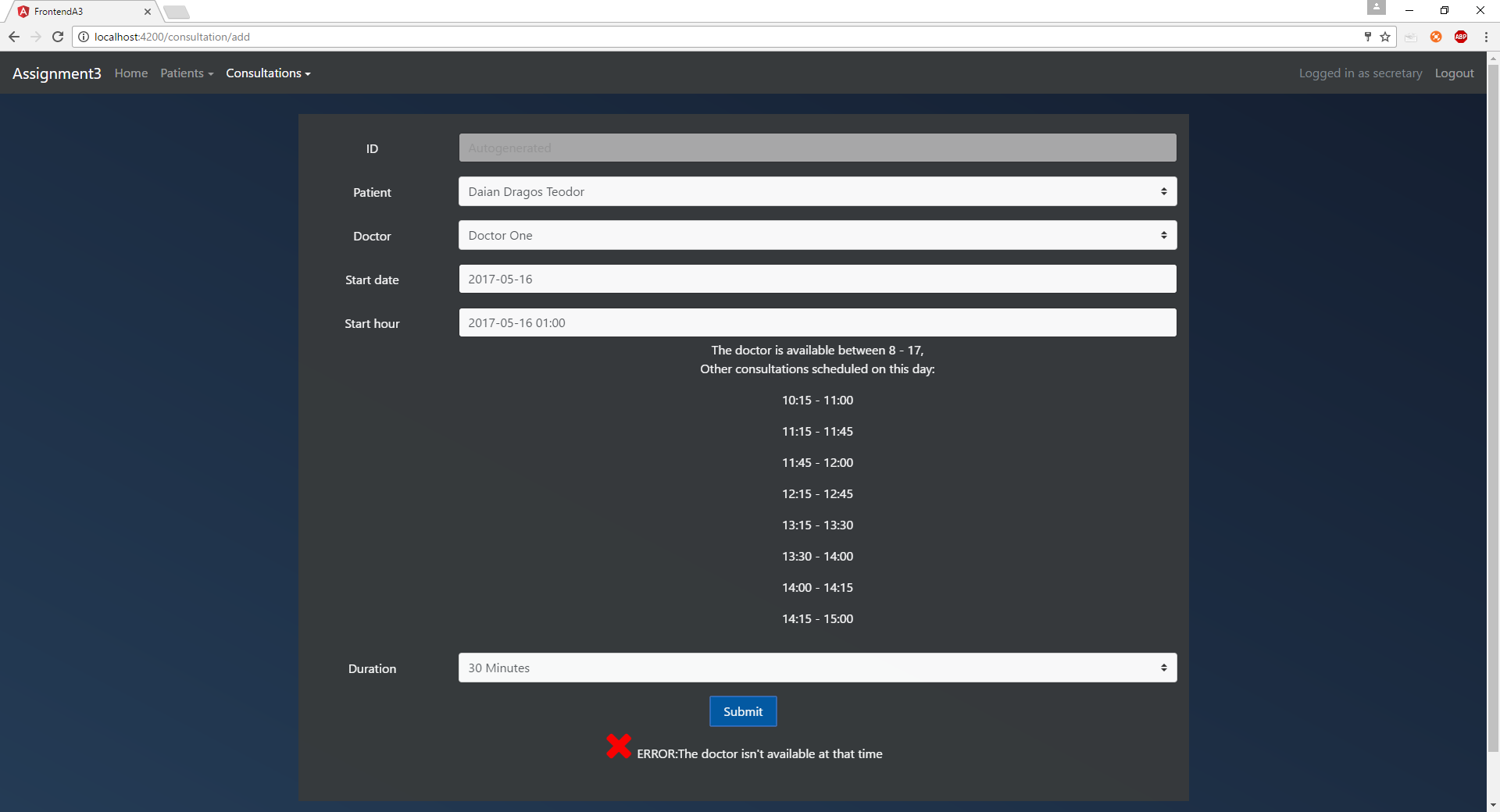
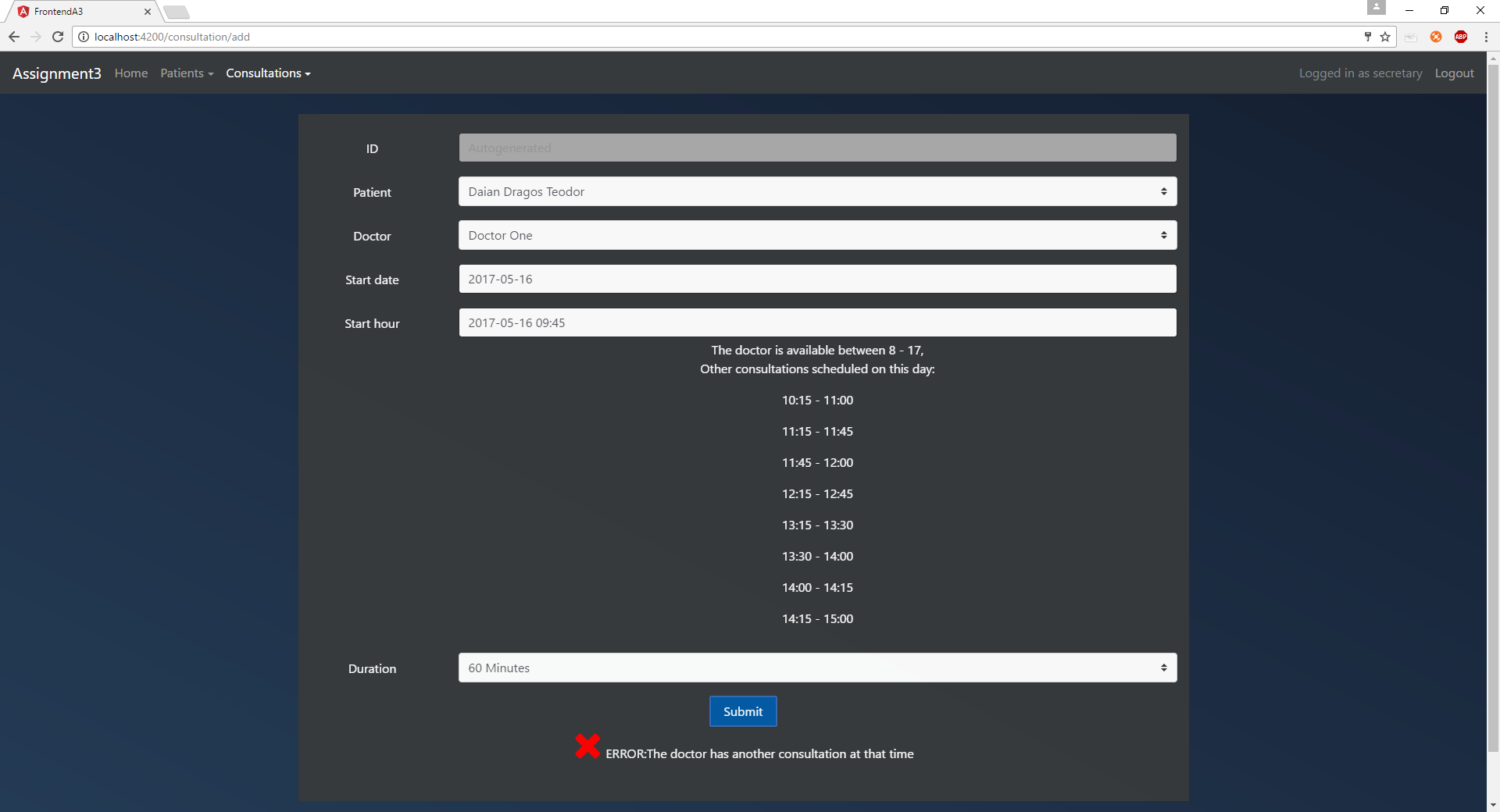
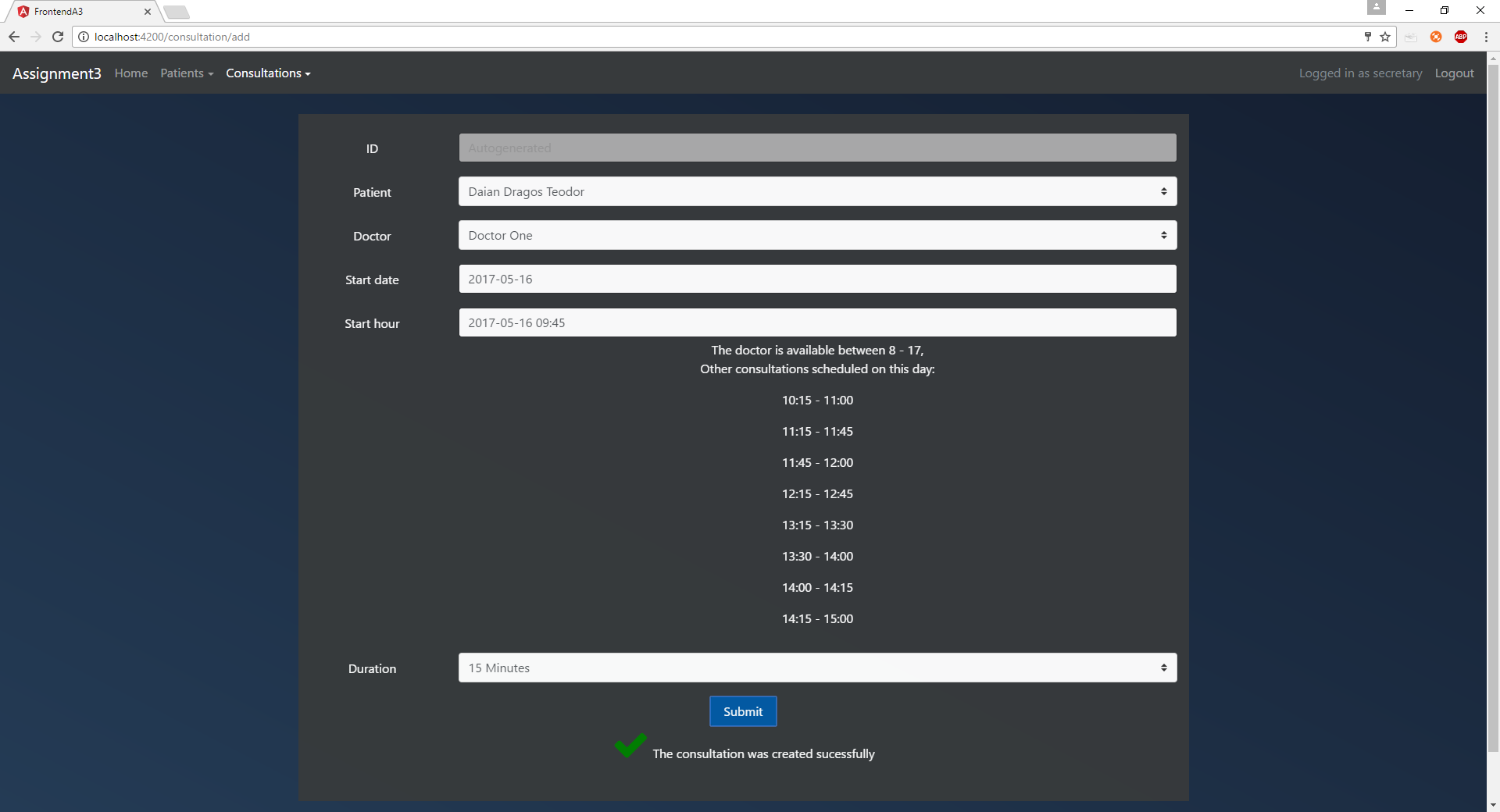
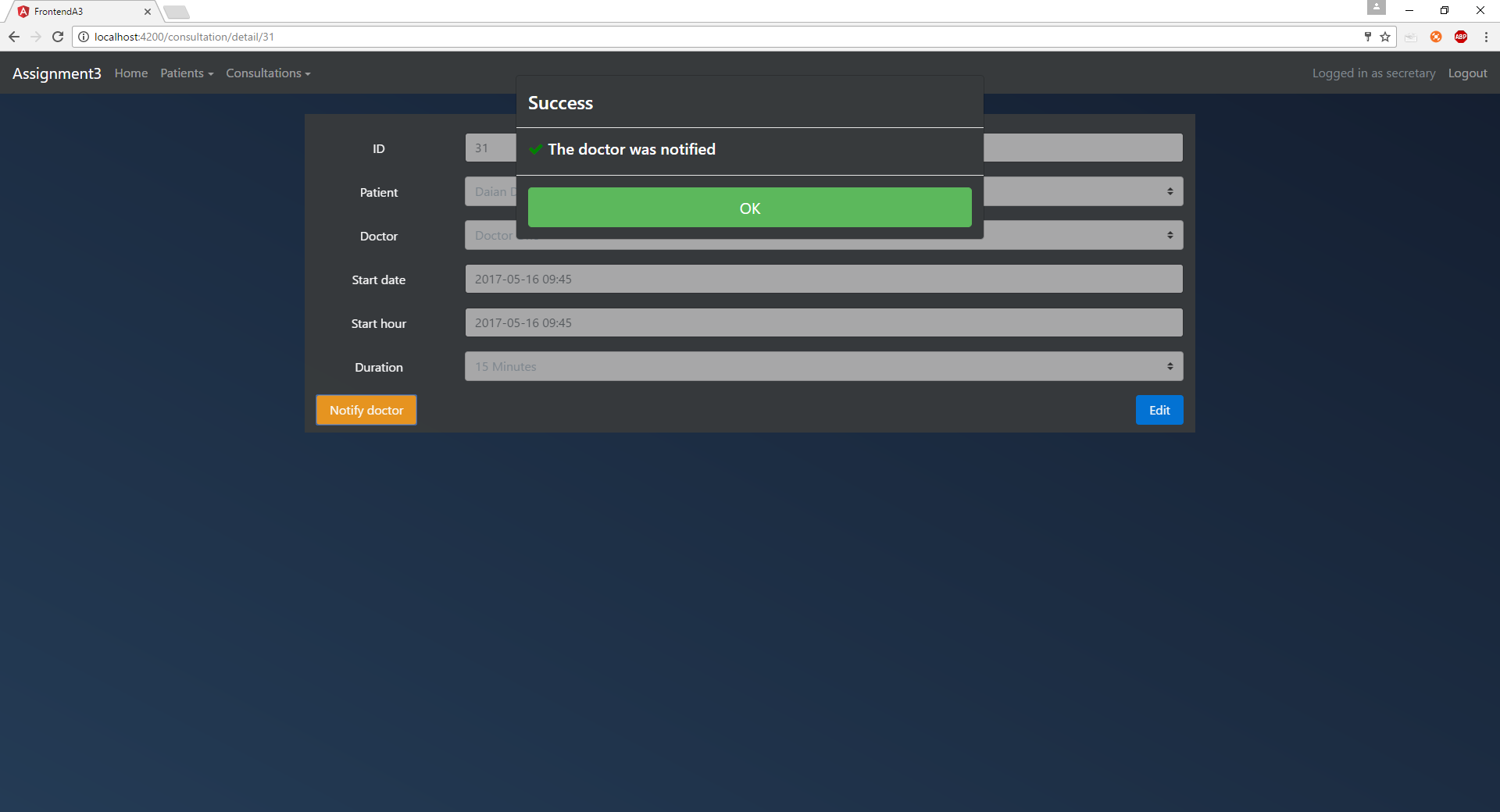
* user: admin (pass: admin) – administrator user
* user: secretary (pass: secretary) – secretary user
* user: doctor1 (pass: doctor1) – doctor user
* user: doctor2 (pass: doctor2) – doctor

The resources that be accessed by each of these users are described below.

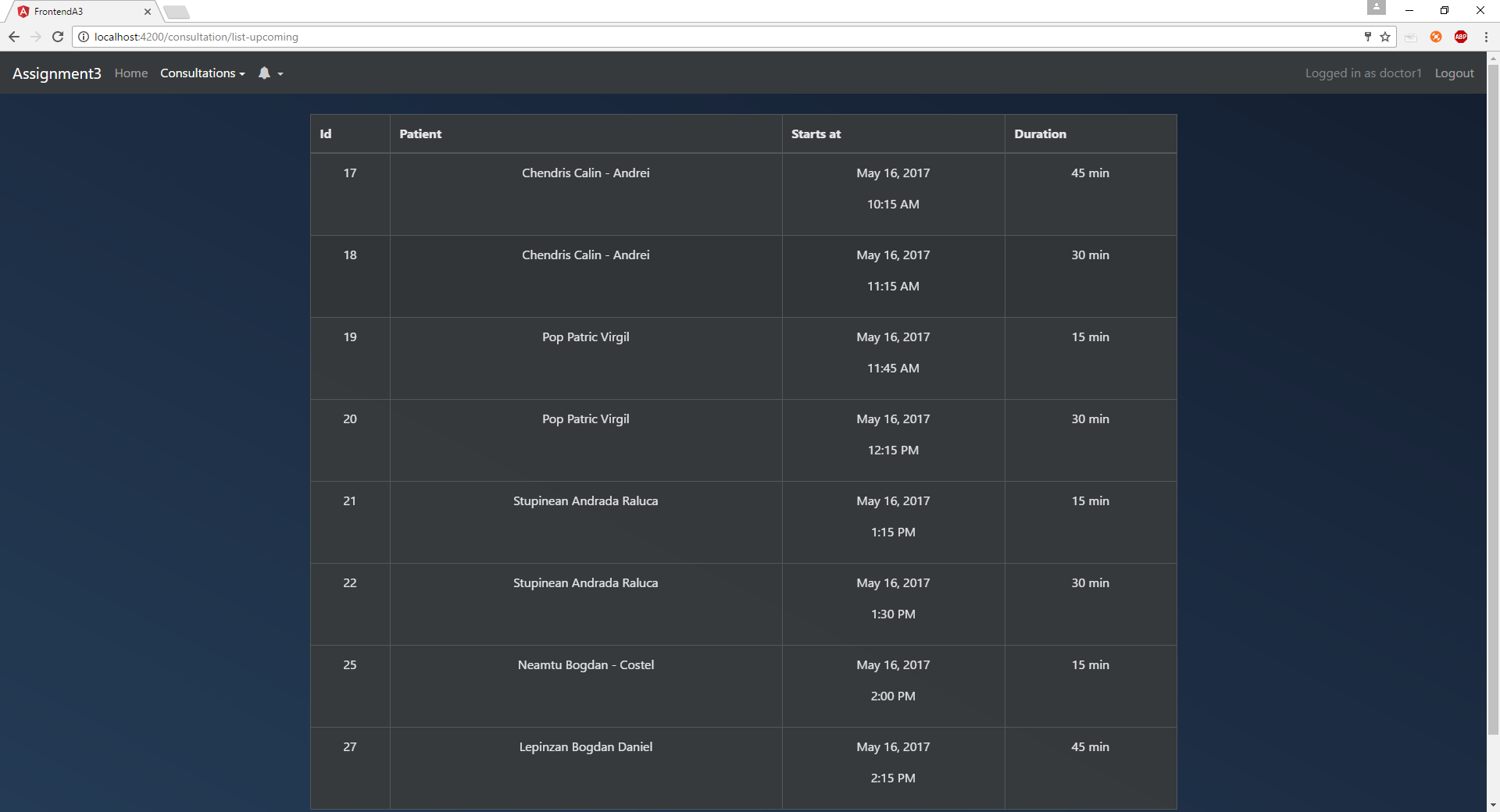
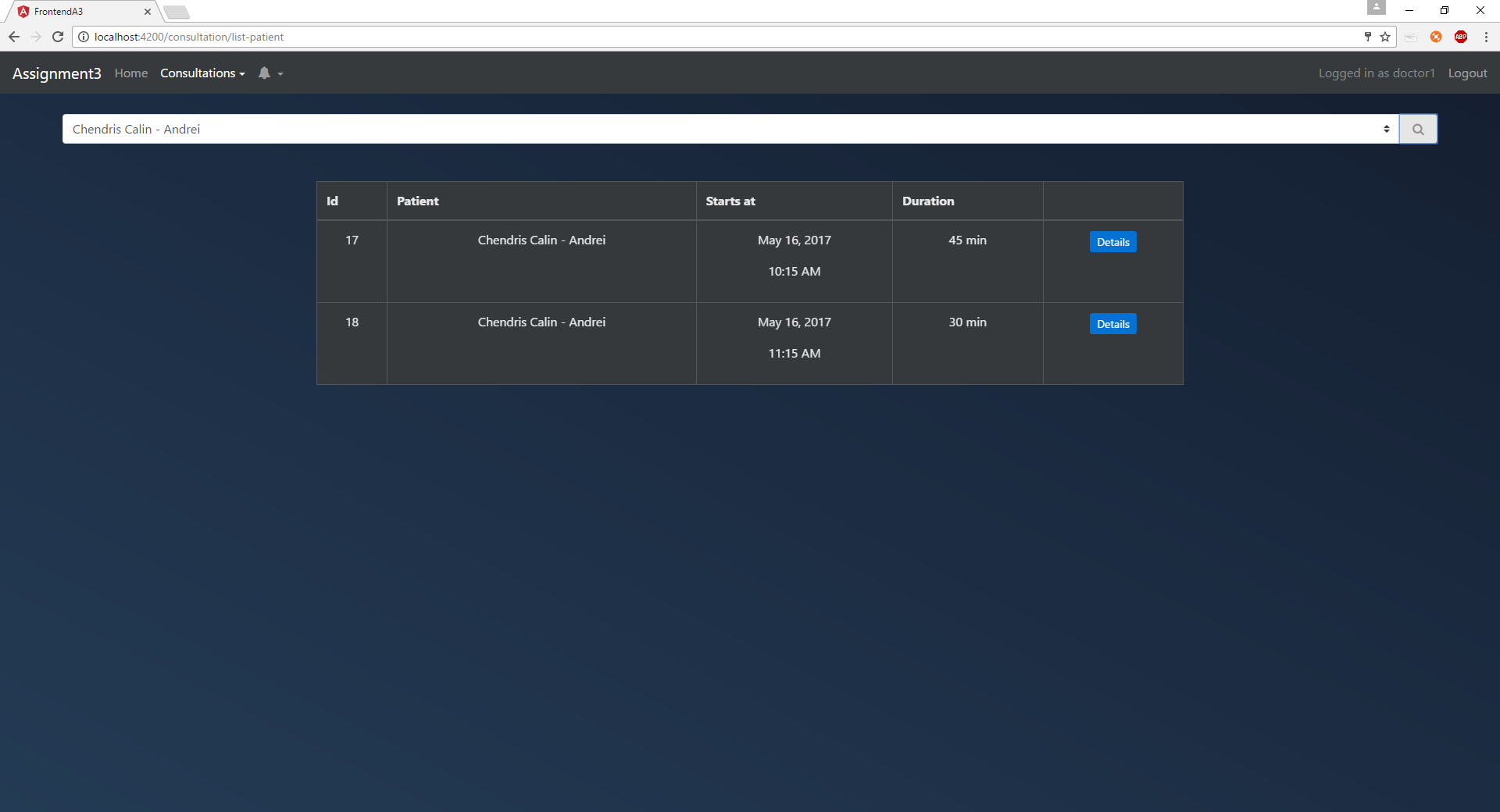
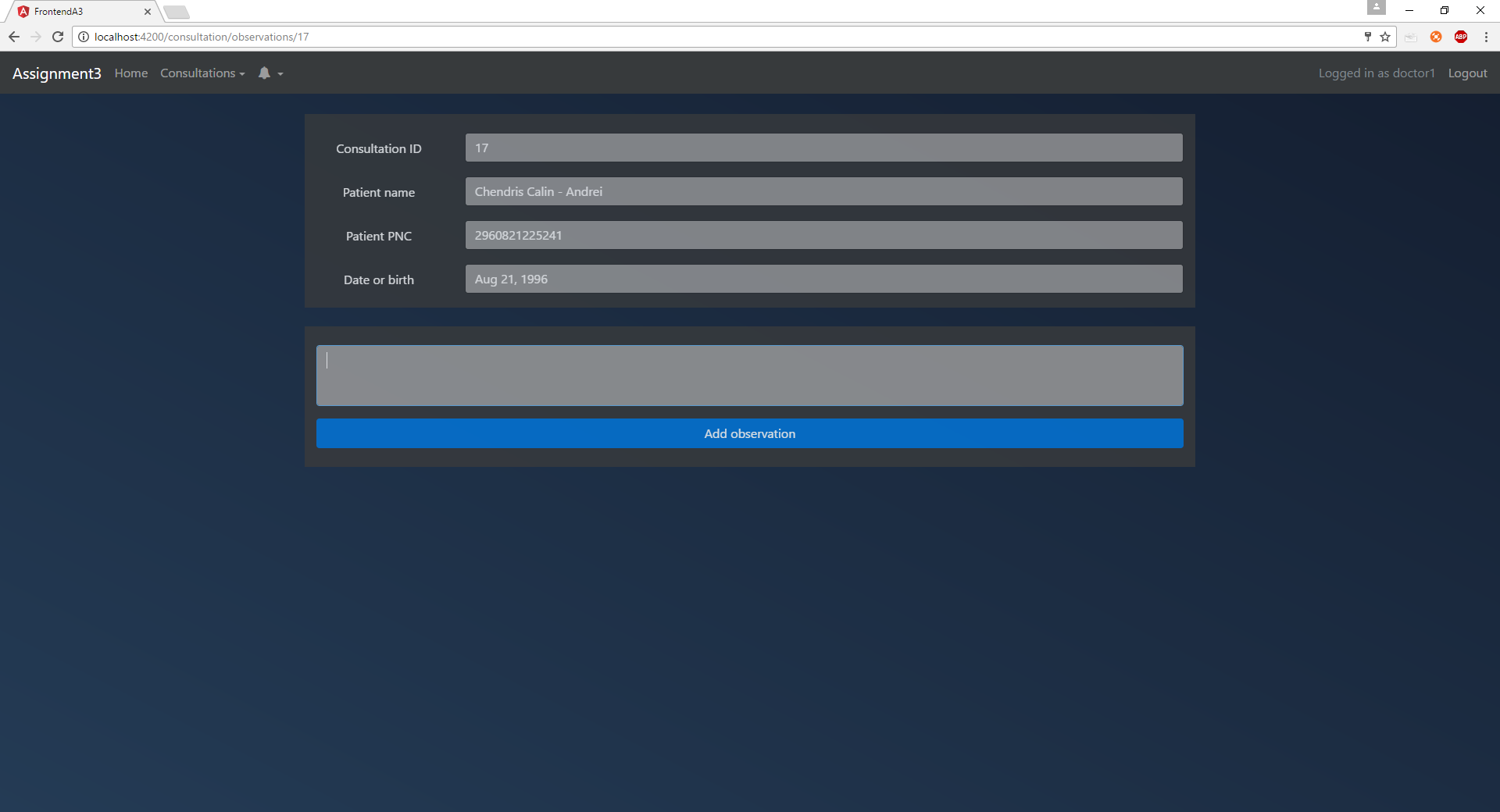
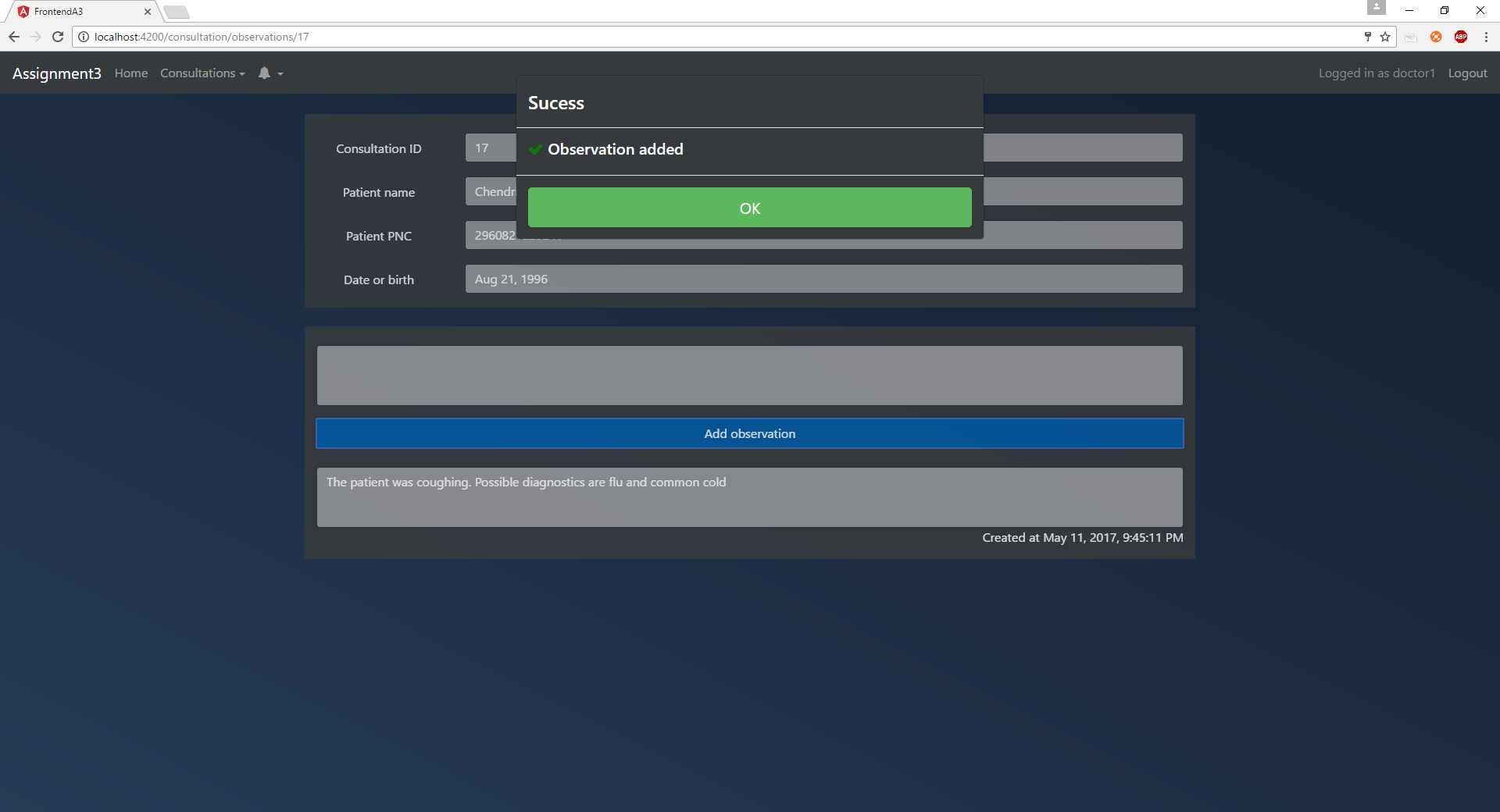
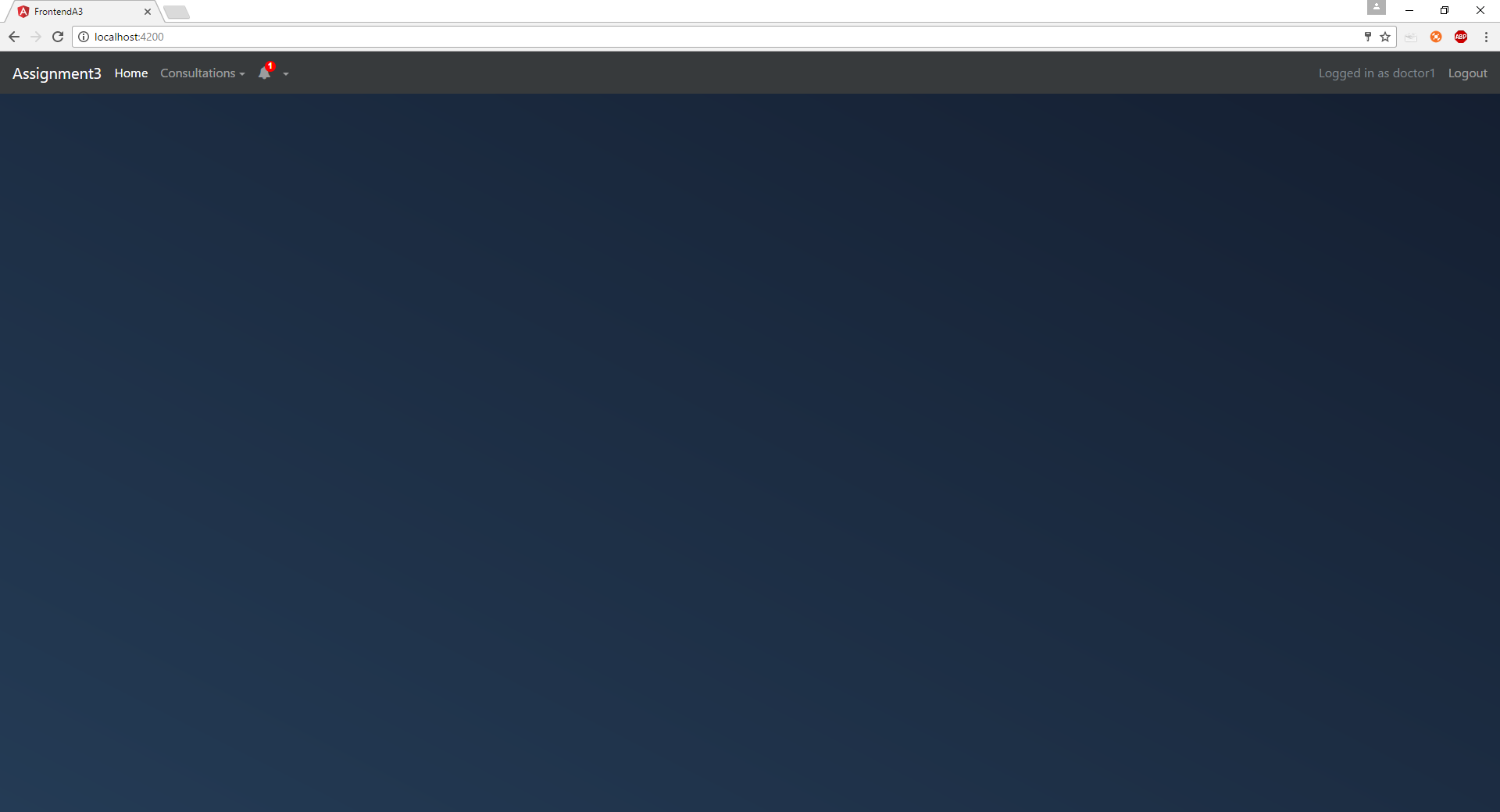
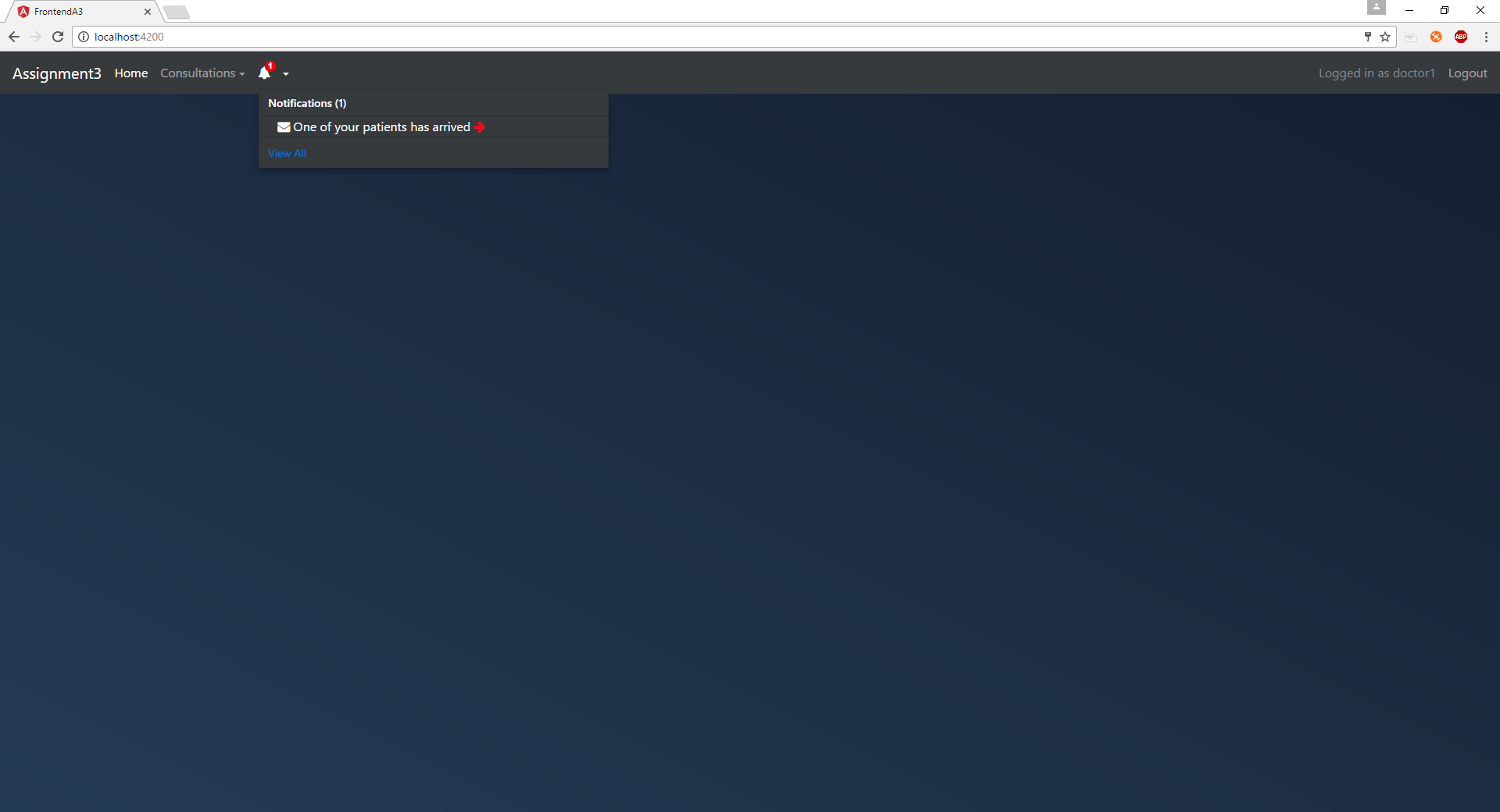
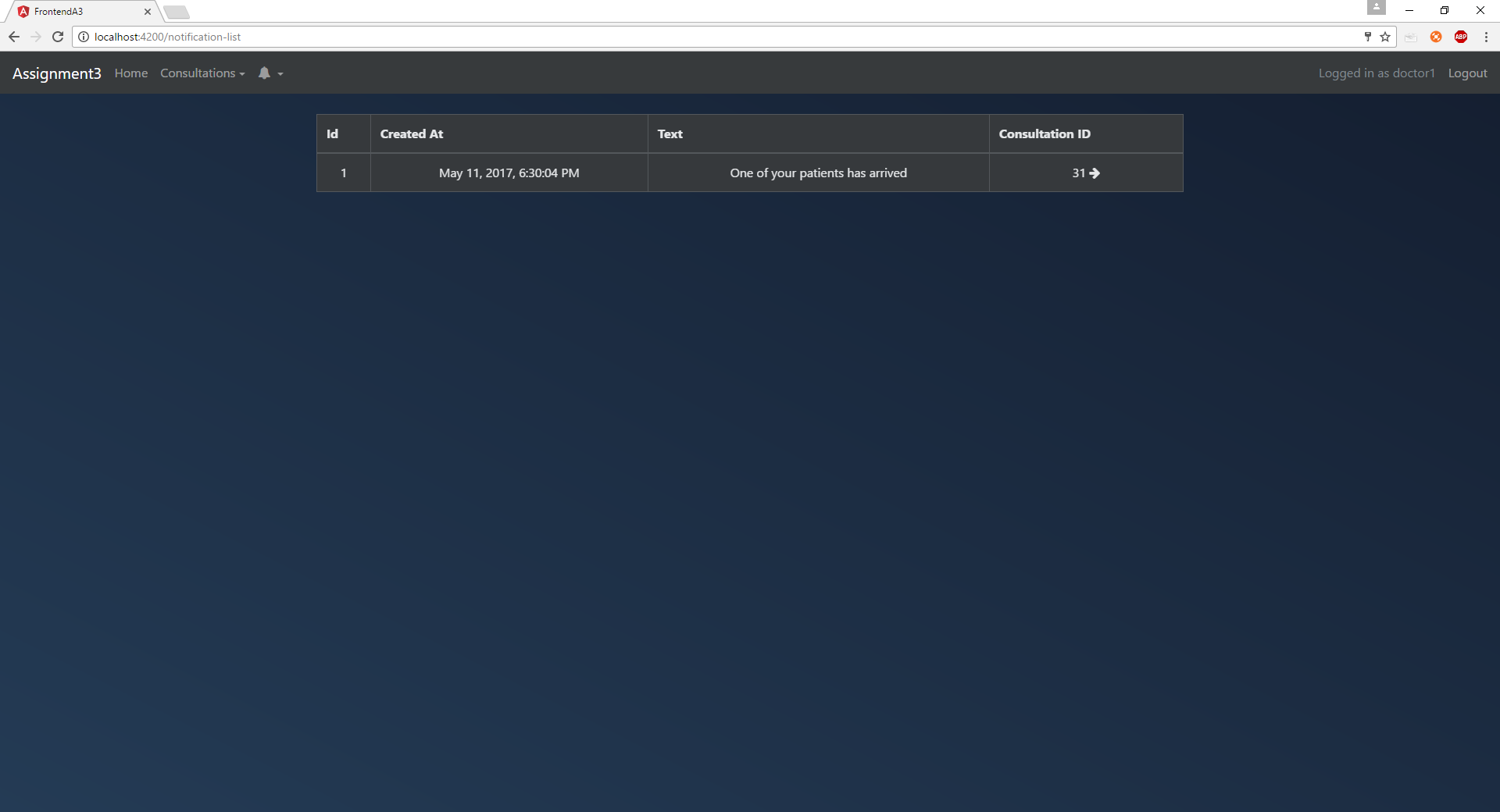
If the logged in user is an administrator they will be able to do the following:

* See a list containing all users by going to Users->List users
* Delete a certain user by clicking on the “Delete” button from the corresponding column
  + Confirmation prompt
  + Error
  + Warning: You won’t be able to delete or edit yourself (you will receive an error message stating exactly that)
  + Success
* Edit the user data by clicking on the “Details” button and then selecting “Edit” to make the fields editable. When the user is done editing they should click the “Submit” button
  + Prompt for changing password of a user
  + Error (duplicate username)
  + Success
* Add new user accounts by going to “Users->Add User”
  + Error (username already in use)
  + Success

If the logged in user is a secretary they will be able to do the following:

* See a list containing every patient by going to “Patients->List patients” from the navbar
* Delete a certain patient by clicking on the “Delete” button from the corresponding column
  + Confirmation prompt
  + Success
* Edit a patient by clicking on the “Details” button and then selecting “Edit” to make the fields editable. When the user is done editing they should click the “Submit” button
  + Error
  + Success
* Add new patient by going to “Patients->Add patient”
  + Error (personal numerical code is already used)
  + Invalid fields
  + Success
* See a list containing every consultation by going to “Consultations->List Consultations” from the navbar
* Cancel a certain consultation by clicking on the “Cancel” button from the corresponding column
  + Confirmation prompt
  + Success
* Creating a new consultation by going to “Consultations->New consultation” from the navbar
  + After a doctor is selected the user will be able to select a date
  + After a date is selected the user will be able to select an hour and the schedule of the selected doctor for that day will be displayed
  + If other consultations exist on that day they will also be displayed
  + Error (The doctor is not available at that time)
  + Error (The doctor is having another consultation at that time)
  + Success
* Edit the details of the consultation by clicking the “Detail” button then making the fields editable by clicking “Edit”. In edit mode the behavior of the form is similar to the one described for adding consultations
* Notifying the doctor that the patient arrived for the consultation by clicking the “Detail” button for that consultation and clicking “Notify doctor”. A doctor cannot be notified twice for the same consultation

If the logged in user is a doctor they will be able to do the following:

* See a list of every consultation that was attributed to them by going to “Consultations->List consultations” from the navbar
* See a list of consultations of upcoming consultations (the start date is in the future) by going to “Consultations-> Upcoming consultations” from the navbar
* See consultations for a certain patient by going to “Consultations->See consultations for patient” and selecting a patient
* By clicking “Details” on a consultation the doctor will be able to see some details about the consultation and also they will be able to view observations left on that consultation or add new observations
  + Observation added
* See that they were notified by looking at the bell icon from the navbar
* They can click the arrow from the notification text to go to the consultation referred by that notification (and at the same time mark the notification as seen). The page that will be loaded is the details page presented above
* Click on “View all” to see all their past notifications