In this coding assignment you will create a sample web application. You must use the AngularJs for front-end. This application should exhibit best practices for application development where possible.

Application Requirements:

1.         Using the AngularJs, build a screen which shows a list of hour long slots from 9am to 5pm.

2.         When one time slot is clicked, load details screen (another page/screen) which asks for first name, last name and phone number. Screen will have 2 options - Cancel and Save.

3.         When the name and phone number is submitted/saved, the app should redirected back to the main screen and selected timeslot should change to red, indicating the time slot is no longer available.

4.         If the red time slot is clicked on again, redirect again to the details screen with the name and phone number for that appointment pre-populated. Users will be able to edit the name and phone number to change the user for the appointment. Don’t worry about credentials. Any user can update any timeslot, irrespective, it is booked by same user or not.

5.         Implement fake REST endpoints for fetching the booked slots initially as well as store newly booked and edited slots.

6.         Create a simple local storage to persist the data as a fallback for the booking data since REST endpoints don’t exist.

Bonus Points:

1. Make the app responsive using Twitter Bootstrap.
2. Showcase the use of Angular Directive.
3. Implement actual REST endpoints using NodeJs that stores the information in MongoDB or as a local nodejs variable.
4. Make the app look nice :)

Completion:

Please check in your code into a repository we can access – Git, etc. along with the instruction on how to run it locally and send us an email saying it is complete.

ReadMe:

C:\Program Files\MongoDB\Server>cd 3.2

Start mongodb:

mongod.exe --dbpath "d:\mean\demoApp\data"

Execute mongo.exe on another cmd

Create collection

use timeslotTracker

> db.mycol.insert({timeSlot:'9 AM – 10 AM',firstName:'Poonam',lastName:'Gokani',phoneNumber: '8460283784'})

db.mycol.find().pretty()

> db.mycol.insert([{timeslot:'11 AM - 12 PM'},{timeslot:'12 PM - 1 PM'},{timeslot:'1 PM - 2 PM'},{timeslot:'2 PM - 3 PM'},{timeslot:'3 PM - 4 PM'},{timeslot:'4 PM - 5 PM'}])

> db.mycol.update({'timeslot':'11 AM - 12 AM'},{$set:{'timeslot':'11 AM - 12 PM'}});

Start node JS Server

node server.js

To update collection column name:

db.collection\_name.update({}, {$rename: {"oldname": "newname"}}, false, true);

To add new column in collection:

db.myCollection.update({}, {$set: {"isOpen": false}}, false, true)

create backup of database in mongodb you should use **mongodump**

mongodump /out:d:\mean\demoApp

To restore backup data mongodb's **mongorestore**

mongorestore -d timeslotTracker /dir:d:\mean\demoApp\timeslotTracker

mongorestore /dir:d:\mean\demoApp\timeslotTracker

To Drop the database:

db.dropDatabase()

**Time Slot Tracker App**

Web application for time slot tracking

**Running the app**

First, make sure to have installed following:

1. Node Version ()
2. MongoDB Version (3.2)

Clone this repo to a local directory and run npm install to install dependencies:

$ git clone git@github.com:sport195/s195-mobile.git

$ cd s195-mobile

$ npm install

Execute following commands on your command prompt or you can create batch files for all commands:

Used To start mongodb and specify path where you want to store data:

**mongod.exe --dbpath "path where you want to store mongodb data"**

Restore the dump of mongodb data ( create database, collection and document for timeslot)

**mongorestore -d timeslotTracker /dir: path of timeSlotTracker Dir of above repository**

To start node server move to dir where server.js file and execute following command:

**node server.js**

You will receive following output once server starts:

Connected & Listen to port 8080

Execute index.html file and enjoy Time Slot Tracker Application.

**Validation**

FirstName : Required Field (only Enter Alphabet, maximum length:30 Digit)

LastName : Required Field (only Enter Alphabet, , maximum length:30 Digit)

Phone Number: Required Field (Only Enter Digit, minimum and maximum length : 10 Digit as per Indian mobile number)

**Application Bonus Point:**

Make the app responsive using Twitter Bootstrap.

Showcase the use of Angular Directive by implementing validation

Implement actual REST endpoints using NodeJS Express Framework that stores the information in MongoDB.