**Hospital Management System Api Version 3.0**

**Made by Anmol hira**

*.POM DEPENDENCIES*

I have used the following:-

* spring-boot-starter-jpa for Hibernate ORM support.
* spring-boot-starter-web basic of spring boot app that includes base dependencies.
* spring-boot-devtools to automatically reload/update the application without restarting it.
* Mysql-connector-java for jdbc driver that is automaticalley added by spring boot application.
* spring-boot-maven-plugin for packaging out app and running it.

***Hibernate Configuration***

* For setting up Hibernate datasource we configure it via our application.properties file.
* Dao Interfaces and it's implementations are defined in in.latticeInnovation.springboot.hospitalmanagementsystemapi.dao package
* Entity classes are defined in in.latticeInnovation.springboot.hospitalmanagementsystemapi.entity
* Service interface and It’s implementation is saved in in.latticeInnovation.springboot.hospitalmanagementsystemapi.service.

***Rest Controller***

1. Rest Controller is defined in in.latticeInnovation.springboot.hospitalmanagementsystemapi.rest package .

***Other details***

I have used ***Postman*** to check for client Request and Server response.

There are Two Rest Controllers and their respective Service and Dao classes defined in separate packages name as follows:-

1. PatientRestController
2. BookRestController

***PatientRestController***

1. **Used to create simple entry for patients with all the validations specified in your provided PDF.**
2. **All CRUD Features are designed as per the instructions.**
3. **I have added extra feature to find one patient by giving id and find the list of all the patients.**

**Request type for PatientRestController**

* **GET –** To get 1 record by passing the id or all records

**Ex -** <http://localhost:8080/api/get-all-patients>

http://localhost:8080/api/patients/1

* **POST –** for inserting new patient record

**Ex-** <http://localhost:8080/api/add-patients>

{

"name" : "Anmol hira",

"email" : "k621@gmail.com",

"address" : "5034 Roshanara raod opp roshaanara bagh",

"phone": "+919899916878",

"password" :"Anm@123fsdhfds",

"date\_of\_admission" :"2019/09/2"

}

* **PUT-** For updating a customer record by passing id

Ex:- <http://localhost:8080/api/update-patient/>

{

"id" : 1,

"name" : "Anish Sharma",

"email" : "k621@gmail.com",

"address" : "5034 Roshanara raod opp roshaanara bagh",

"phone": "+919899916878",

"password" :"Anm@123fsdhfds",

"date\_of\_admission" :"2019/09/2"

}

* **DELETE –** For deleting a customer record

Ex- <http://localhost:8080/api/del-patients/5>

**BookRestController**

**Used to book appointments for the patients.** When you fetch patient record you will get it in date wise order as per your instructions.

**GET**: <http://localhost:8080/book/all-bookings> - GET all the list of patients sorted by date.

**POST:** http://localhost:8080/book/add-appointment Used to add new appointment record in the database.

Ex:

http://localhost:8080/book/add-appointment

{

"patient\_id" : 1007,

"patient\_name" : "Anmol",

"date" : "30/12/7",

"doctor\_name" : "Ram prakash sharma",

"status" : "progress"

}