

**CDAC BANGALORE**

PROJECT Documentation

New Life Hospital management system

### Submitted By:

**Ananya Augustin – 200950181009**

**Anjali Varma – 200950181010**

**Mridul Mathew Mohan – 200950181056**

**Nitin – 200950181061**

**Shivam Tripathi -- 200950181096**

**1.Project Functionality**

* Main aim of the project is to store information about the patients’ details and provide lab facility and details of the doctor that are present in the hospital.
* A new patient can register by entering the details such as name, email, address, age and phone number and choose a password of its choice.
* Patient can to login with the email and password in the main page.
* The patient then able to view its details and also be able to book lab facility and appointment with a doctor.
* The patient can also view it booked appointments and lab facility.
* Similarly, the doctor can also login into the web site with email and password.
* A doctor has the option to check patient records and delete any record.

2. **USE CASE**

**2.1 PRESENTATION LAYER:**

* When homepage is opened it contain details about the hospital and have some options like create a new account, login as a patient or as a doctor.
* The registration of the patient contains field like name, email, address, age, gender, and password and phone number.
* On the login page, the patient needs to enter his email and password.
* The patient then redirected to the users page, where the options like book an appointment, book a lab facility, check appointment and check lab booking.
* When you login as a admin(doctor), you have the option to view all the patient details and also have the option to delete a patient records.

**2.2 SERVICE LAYER:**

* When the patient enters it details in the form the spring boot take the details and save them in the database.
* Whenever a patient enter the email and password, the spring boot check these details with the database records and if it is a match then the patient can login otherwise an error message is displayed.
* Once login the patient can book appointments and lab facilities and these details get stored in the database so that when the same patient login after some time then he/she be able to view its booked lab facility and appointment.
* A doctor details also are already been saved in the database so the doctor can login in the same way a patient login in the website.
* The doctor have the option to retrieve a patient details from the database and also he/she can delete any record of a patient.

**2.2 Database LAYER:**

* The database stores all the details of the patient and doctor and to do so we take help of the jpa repository interface.
* With the help of annotation like entity we are able to create a table for patient, doctor, admin, lab and appointment modules.

3. **PROBLEMS FACED DURING PROJECT:**

PROBLEM 1:

* This being an online course the first issue that we face is regarding the communication as we are not physically present there to work together.

RESOLUTION:

* We overcome this issue by having regular meeting with the platform google meet platform.

PROBLEM 2:

* The second issue we faced is while connecting the back-end with the front-end in the logins as we are unable to get the token from the database and pass it to react.

RESOLUTION:

* We overcome this issue with the help react-cookies library as it does a great job of abstracting access to cookies from the document with approachable functions and options.

4. **LEARNINGS DURING THE PROJECT**

* We learnt a number of technologies like CSS bootstrap, Reactjs and many more.
* Despite the methodology used, there must be some amount of documentation available as resource should a team member leave or a new person is assigned to a role.
* We learnt a lot about GITHUB which simplified the process of working with other people and made it easy to collaborate on project. It helped us to easily merge our changes in with the master branch of the project.
* We also used Jira as our scrum board to organize tasks, assign specific tasks to team members, and track each task through its lifecycle.