CMPE 275
Enterprise Application Development
Team 7

SJSU ID	**TEAM MEMBER**
015252003	Sumeet Gupta
015260232	Bharath Jagini
015275546	Harsh Sheth

1. Motivations:

To learn and apply Java, SpringBoot, MySQL, ReactJS technologies to develop a vaccine management system which will be scalable, durable and reliable. For Authentication, the application uses Google OAuth.

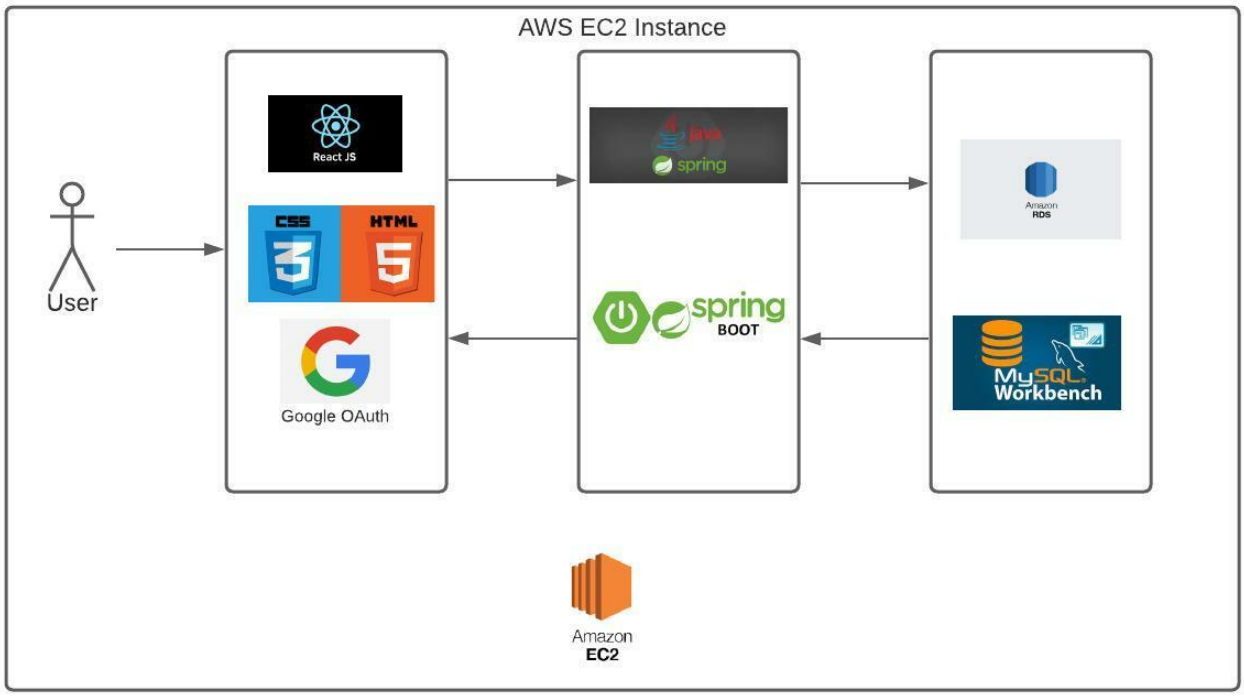
2. Introduction:

The objective of the project is to develop a Vaccine Management System Application. Vaccine Management System is an application based on a portal for tracking the vaccination status updated by the people.

This application is divided into 2 personas Patient and Admin. The User can create an account using name, email, password, first name, last name, date of birth, Address, Gender. In order to make a vaccination appointment, the Patient must login. The Patient can view and edit all their appointments under the appointments tab. Patients can also view the vaccination history under the past appointments tab. Patients are allowed to check in only 24 hours before the appointment start time. The Admin is allowed to view the appointment available in different clinics. Admin can also add new diseases, clinics and vaccine details under Add disease, Add Clinics, Add Vaccine tabs respectively. This application enables users to fetch the details regarding the vaccination.

We have included validations on every user input of the application. Server is tested for myriad concurrent users at a time. The test plan was made to verify the testing of the applications for various test cases.

3. System Architecture Design:



4. Technology Choices:

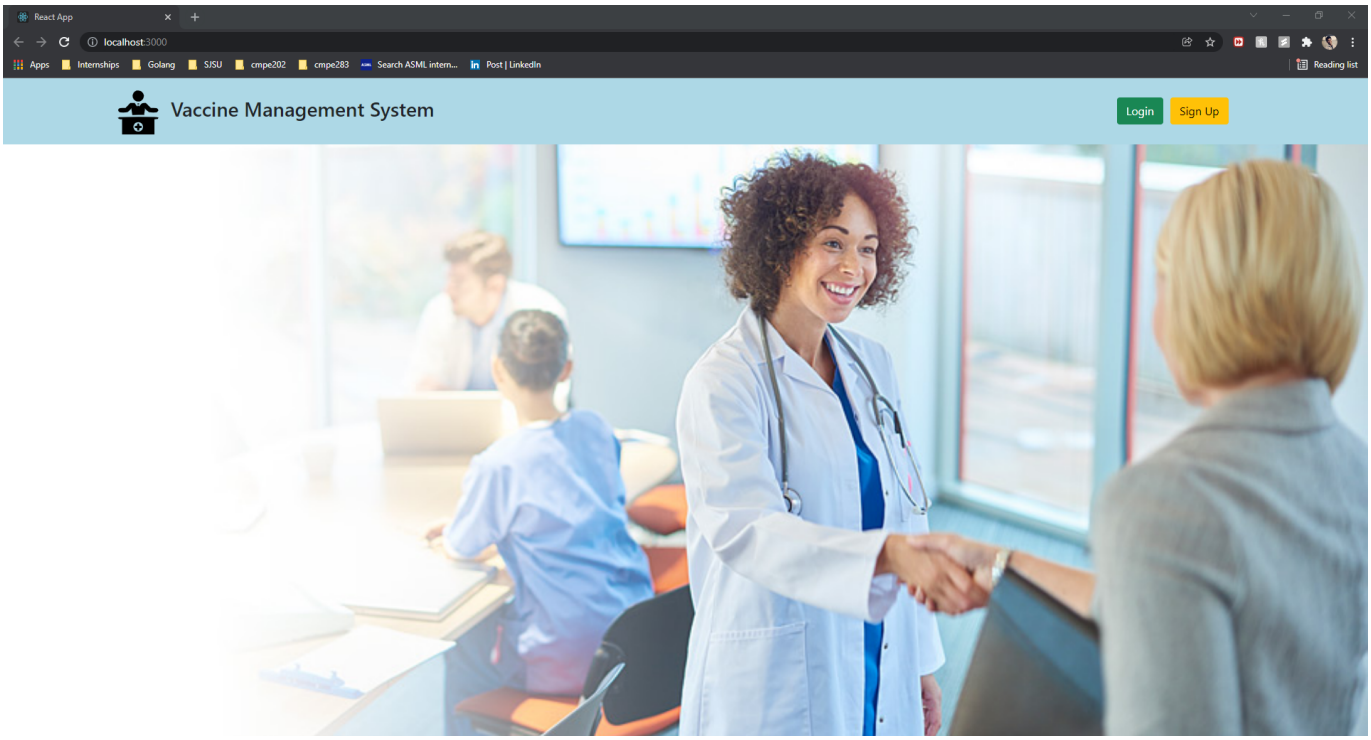
Since the Application has to be user friendly, we used the modern technologies which can help scale the application and make the application UI/UX better. Below are the list of technologies used for developing the application:

- Frontend: React, HTML5, CSS, Bootstrap.
- Backend: Java, Spring Boot.
- Database: AWS RDS MySQL.
- Cloud Technologies: AWS EC2.

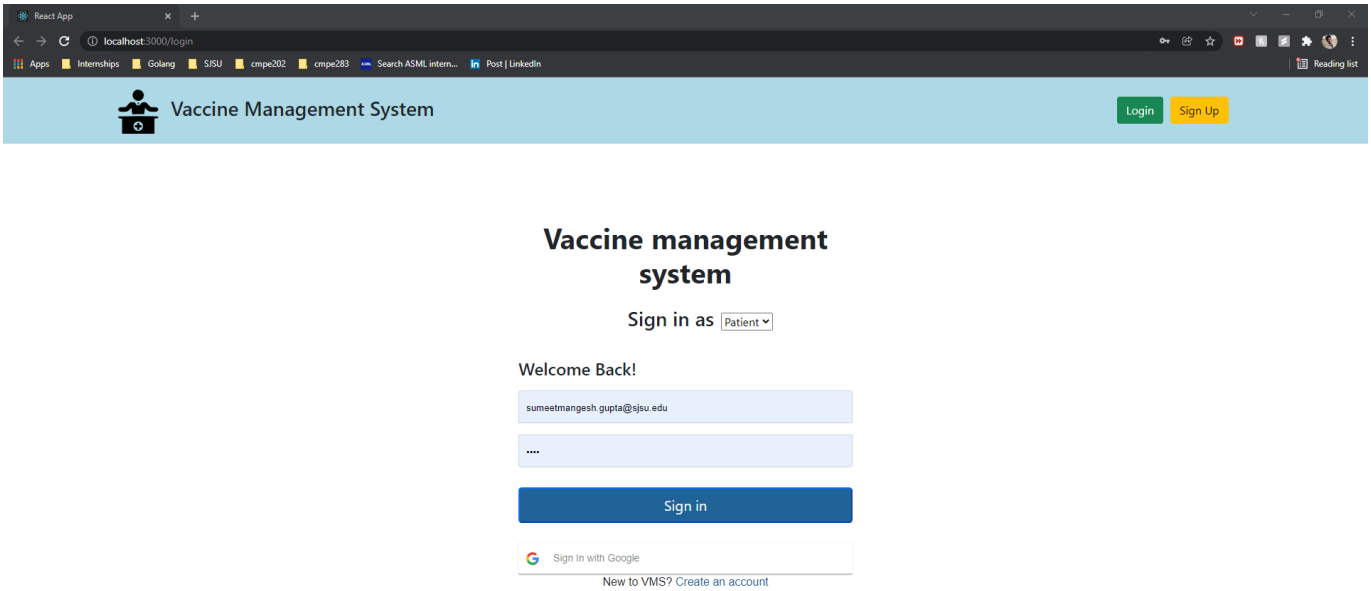
These technologies helped in improving the user experience making application user friendly.

5. Application Screenshots:

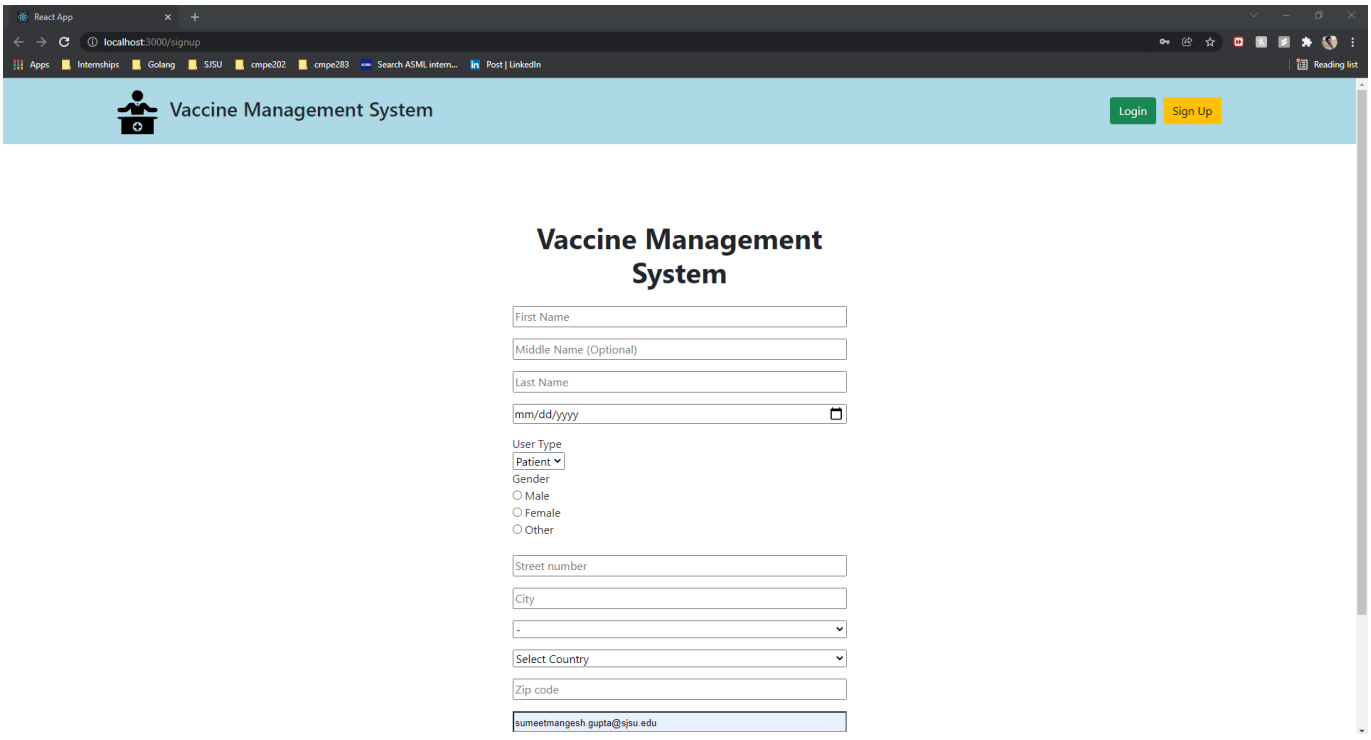
Landing page



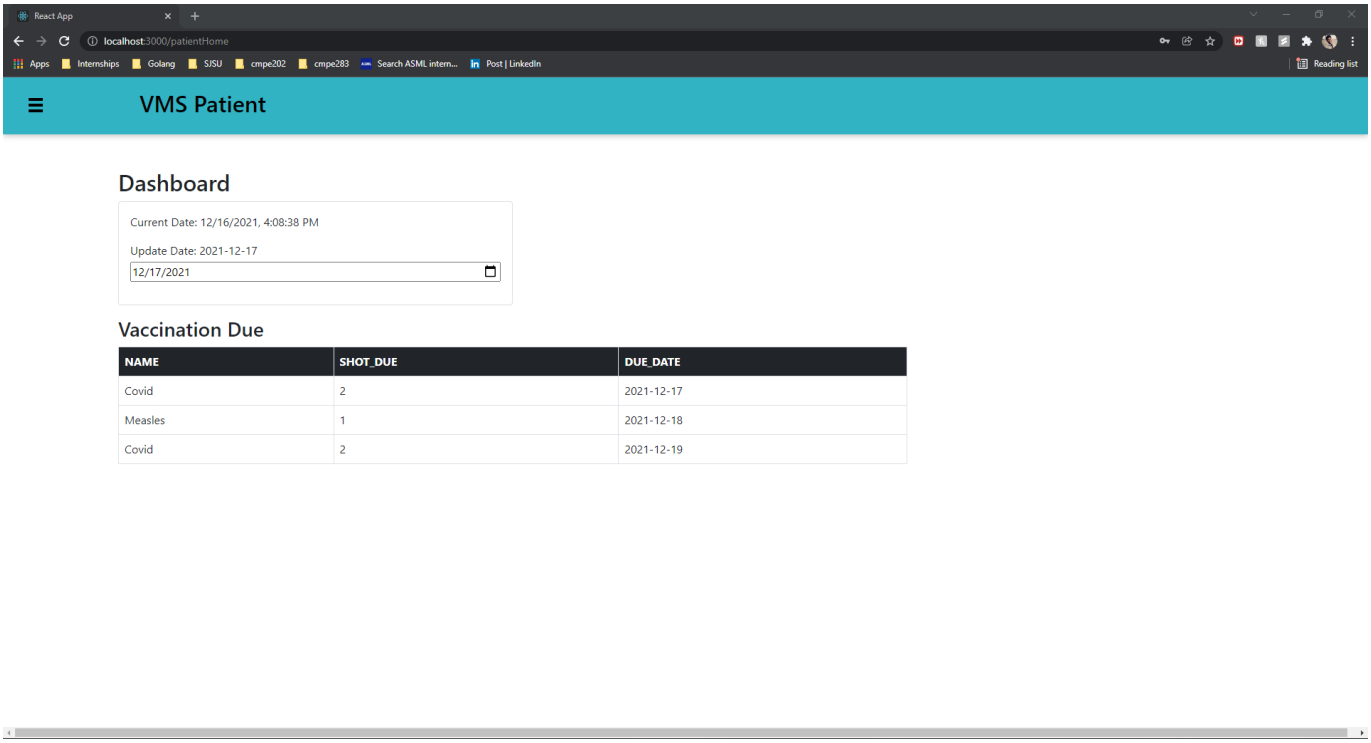
Login page



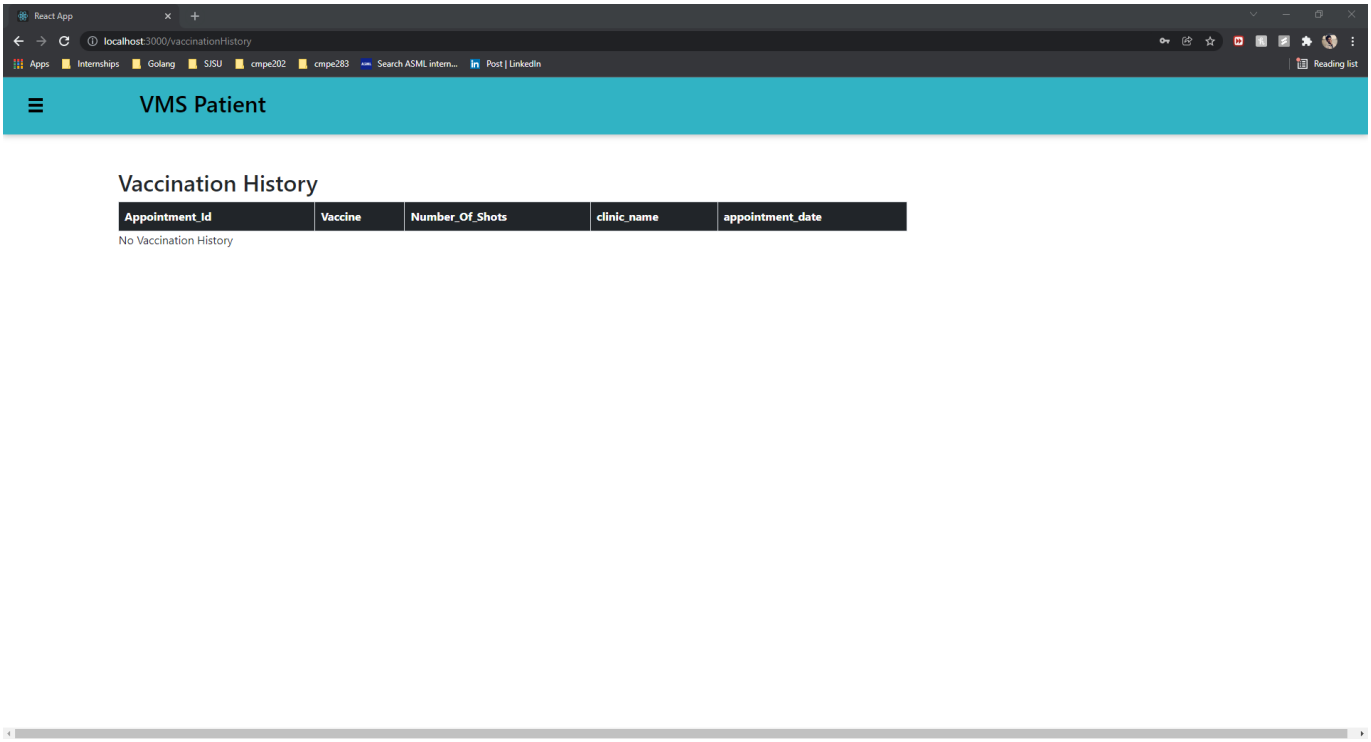
Signup page



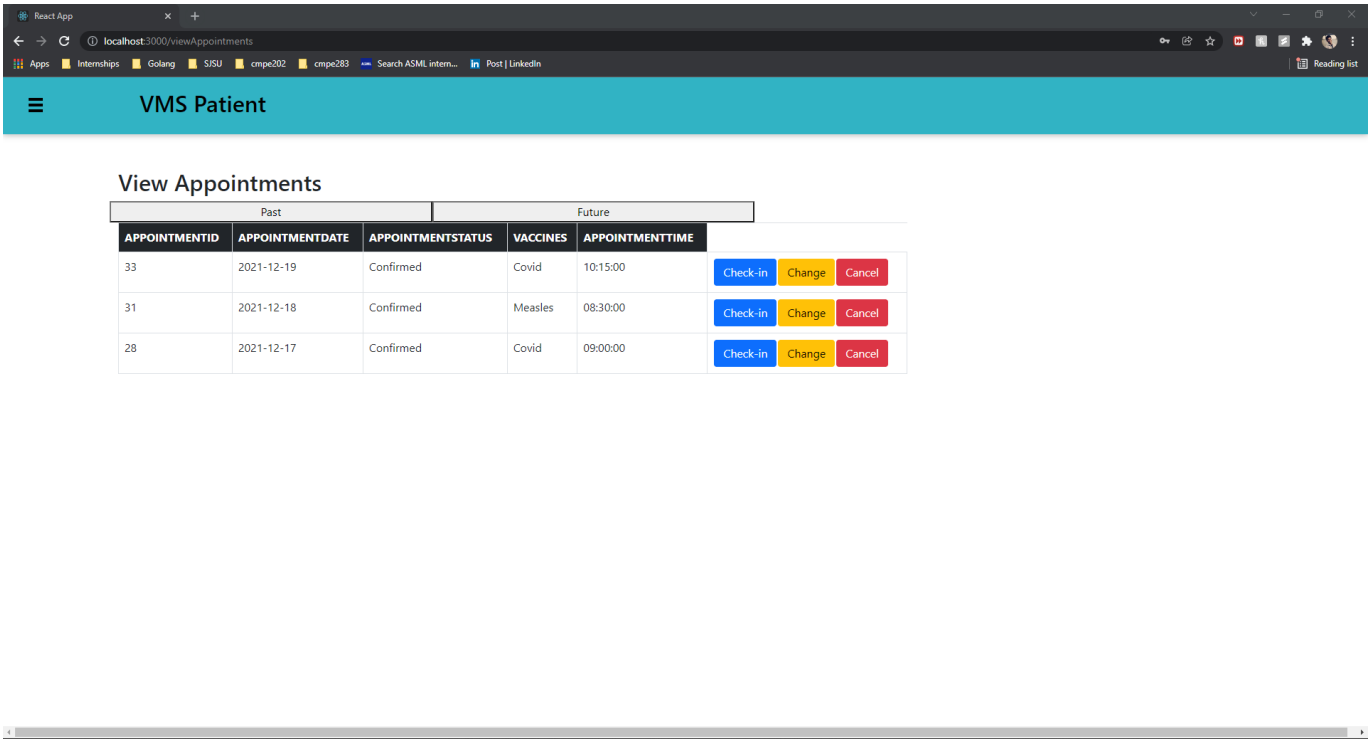
Patient Home page



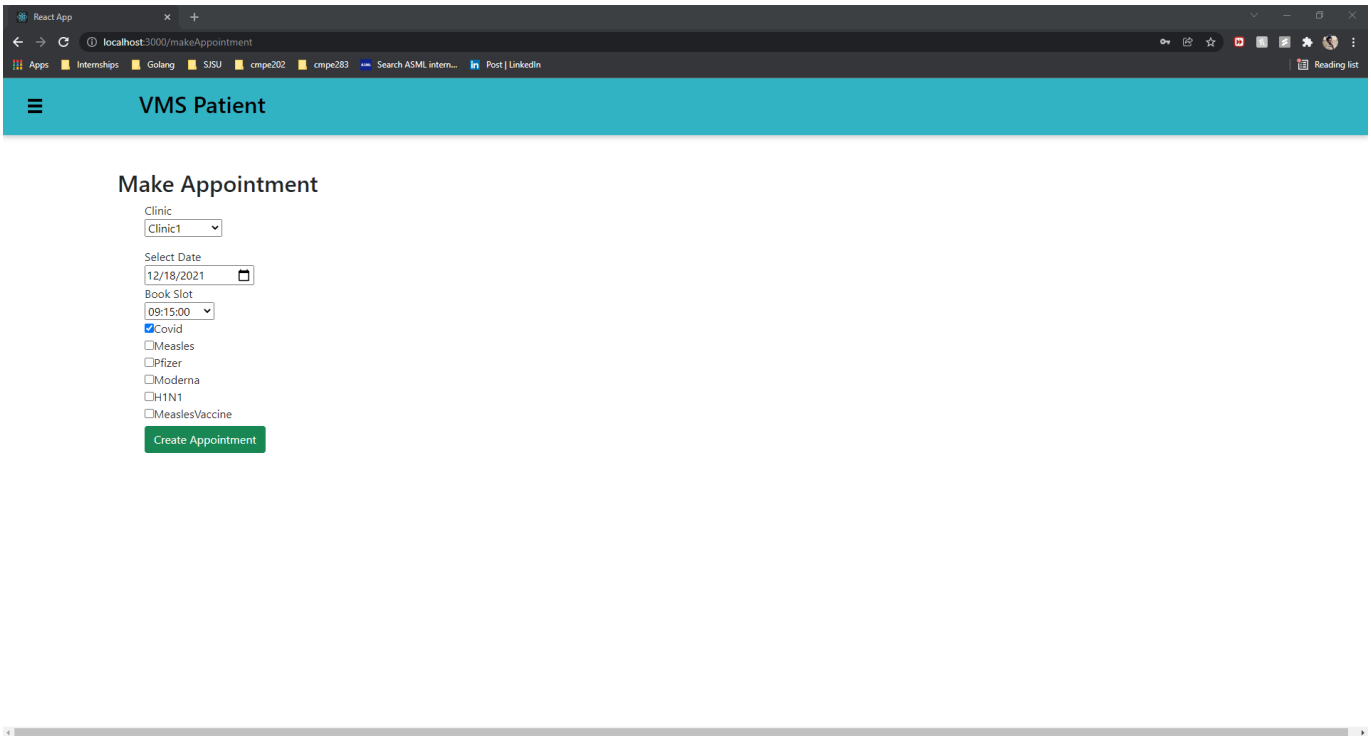
Vaccination History page



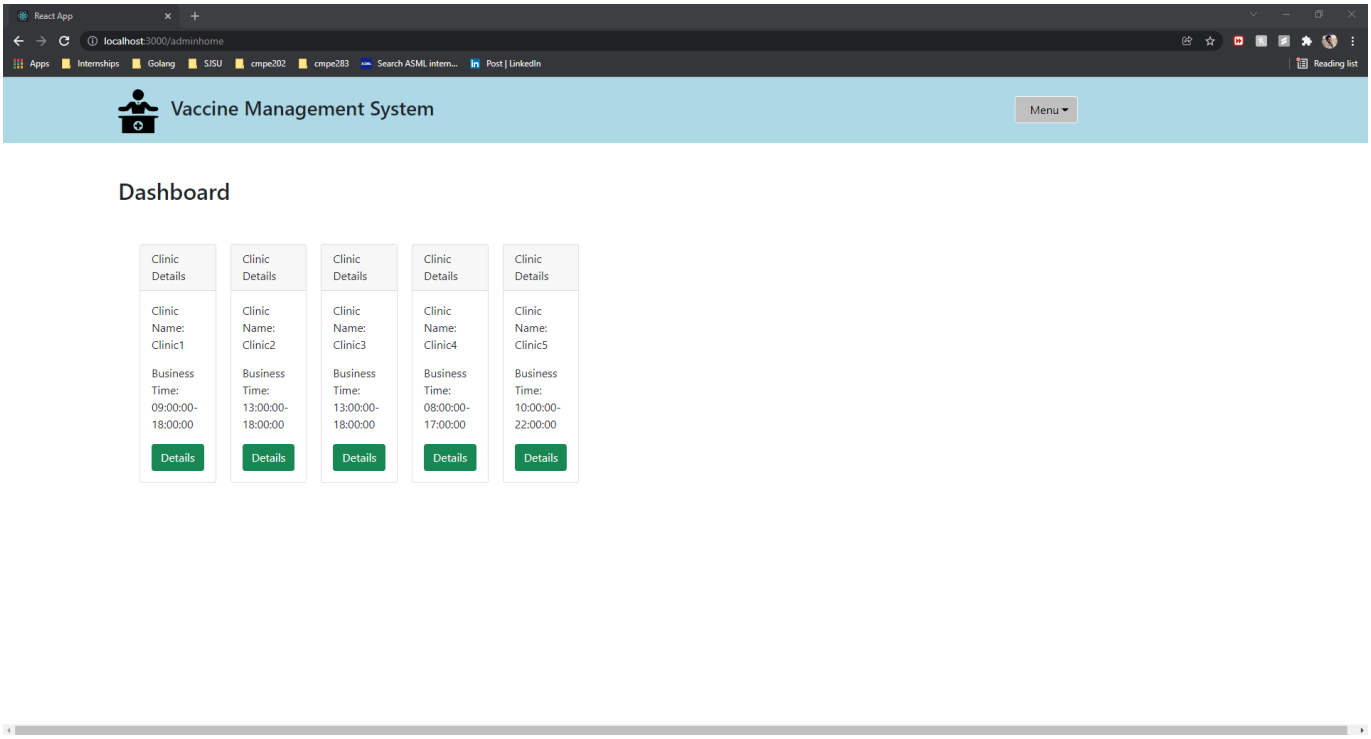
View Appointment page



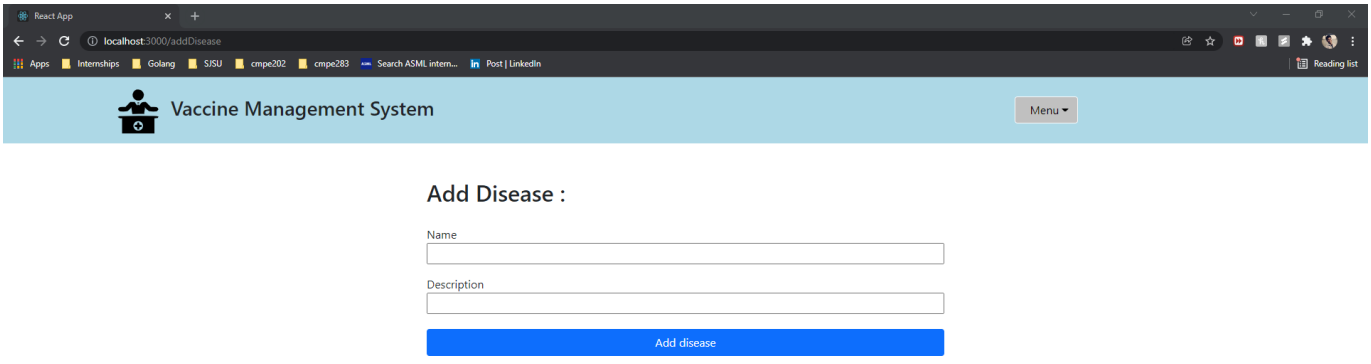
Make Appointment page



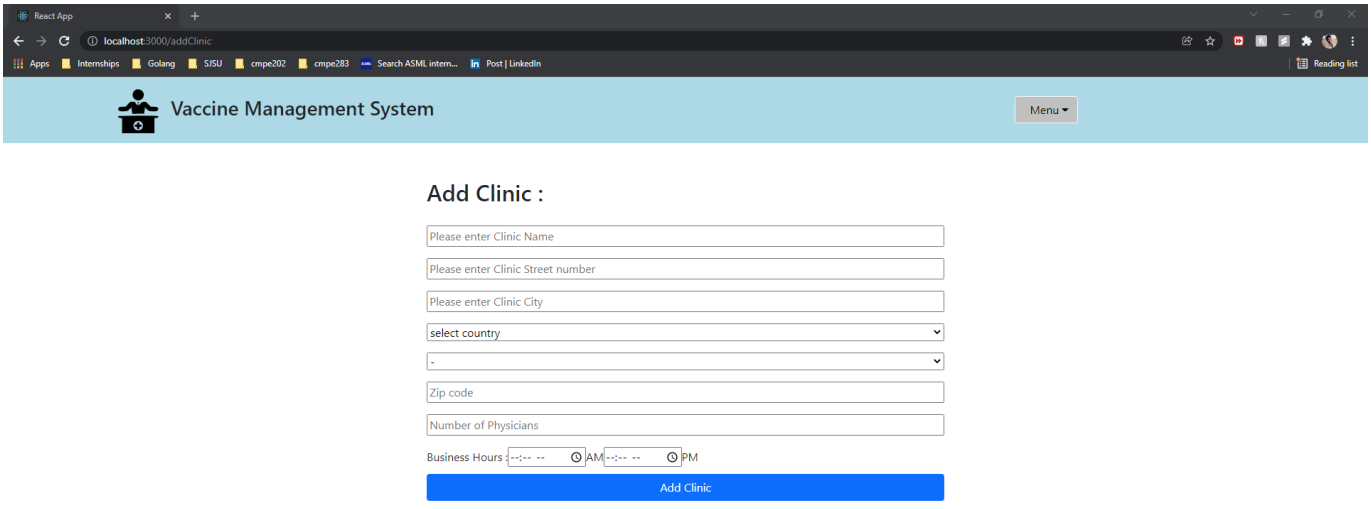
Admin Home page



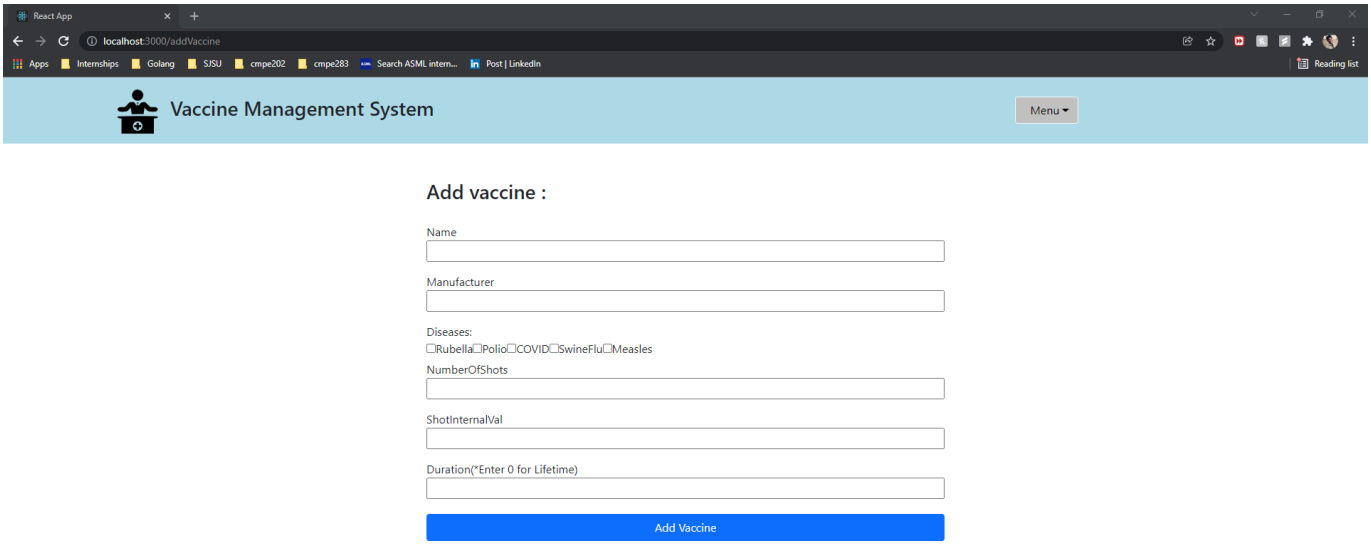
Add disease page



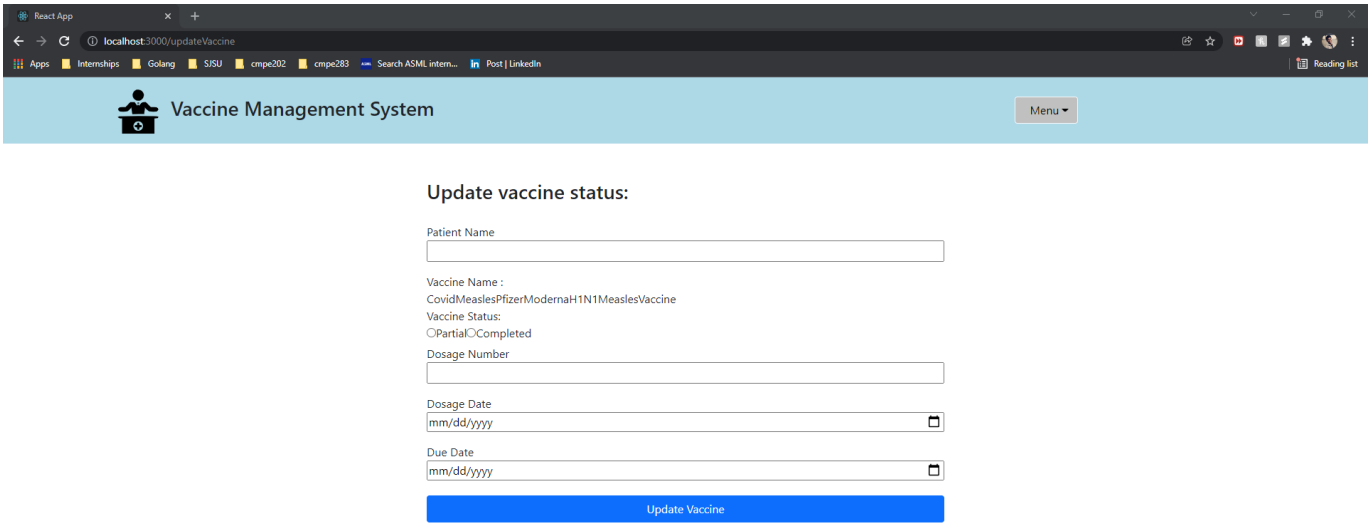
Add Clinic page



Add Vaccine page



Update Vaccine page



6. Testing Plan and results:

We divided the Testing Plan as per pages and created Test Cases for each page. We wrote the JUnit tests and did manual testing for the application and tested the application. Below were results of the same.

Login/Signup Page

srno	Testcase	Expected	Result
1	Do correct login for Patient	Should login and redirect to PatientHome	Pass
2	Do incorrect login for Patient	Should show error	Pass
3	Do correct login for Admin	Should login and redirect to AdminHome	Pass
4	Do incorrect login for Admin	Should show error	Pass
5	Try login with null fields for Patient	Should show error	Pass
6	Try login with null fields for Admin	Should show error	Pass
7	Do correct sign up for Patient	Should signup and redirect to PatientHome	Pass
8	Do incorrect sign up for Patient	Should show error	Pass
9	Do correct sign up for Admin	Should signup and redirect to AdminHome	Pass
10	Do incorrect sign up for Admin	Should show error	Pass
11	Try signup with null fields for patient	Should show error	Pass
12	Try signup with null fields for admin	Should show error	Pass

Admin Pages

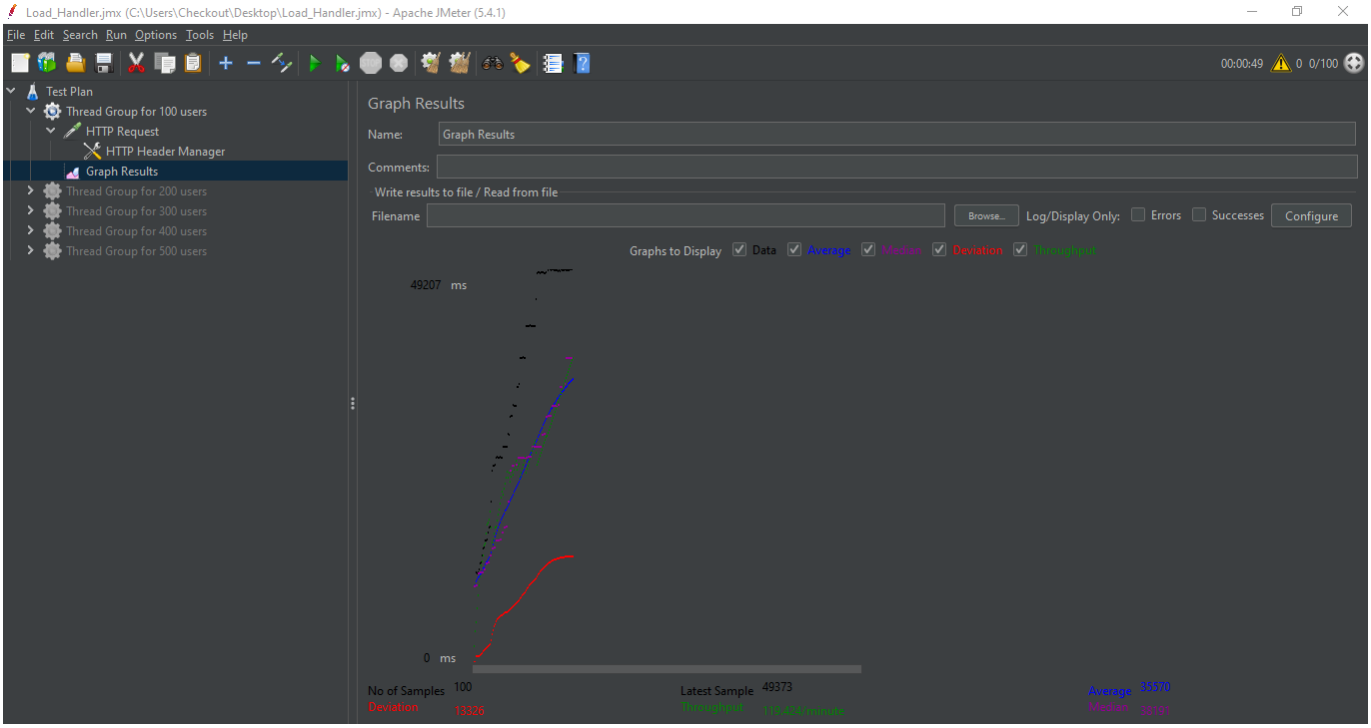
srno	Testcase	Expected	Result
1	Do correct Add disease	Should add new disease in DB	Pass
2	Do incorrect Add disease	Should show error	Pass
3	Do correct Add Vaccine	Should add new vaccine in DB	Pass
4	Do incorrect Add Vaccine	Should show error	Pass
5	Do correct Add Clinic	Should add new clinic in DB	Pass
6	Do incorrect Add Clinic	Should show error	Pass
7	Do correct Update Vaccine	Should update vaccine in DB	Pass
8	Do incorrect update vaccine	Should show error	Pass
9	Check for getAllClinics with data	Should get all clinic list from DB	Pass
10	Check for getAllClinics with no data	Should get no clinic list from DB	Pass
11	Do logout	Should redirect to landingPage	Pass

Patient Page

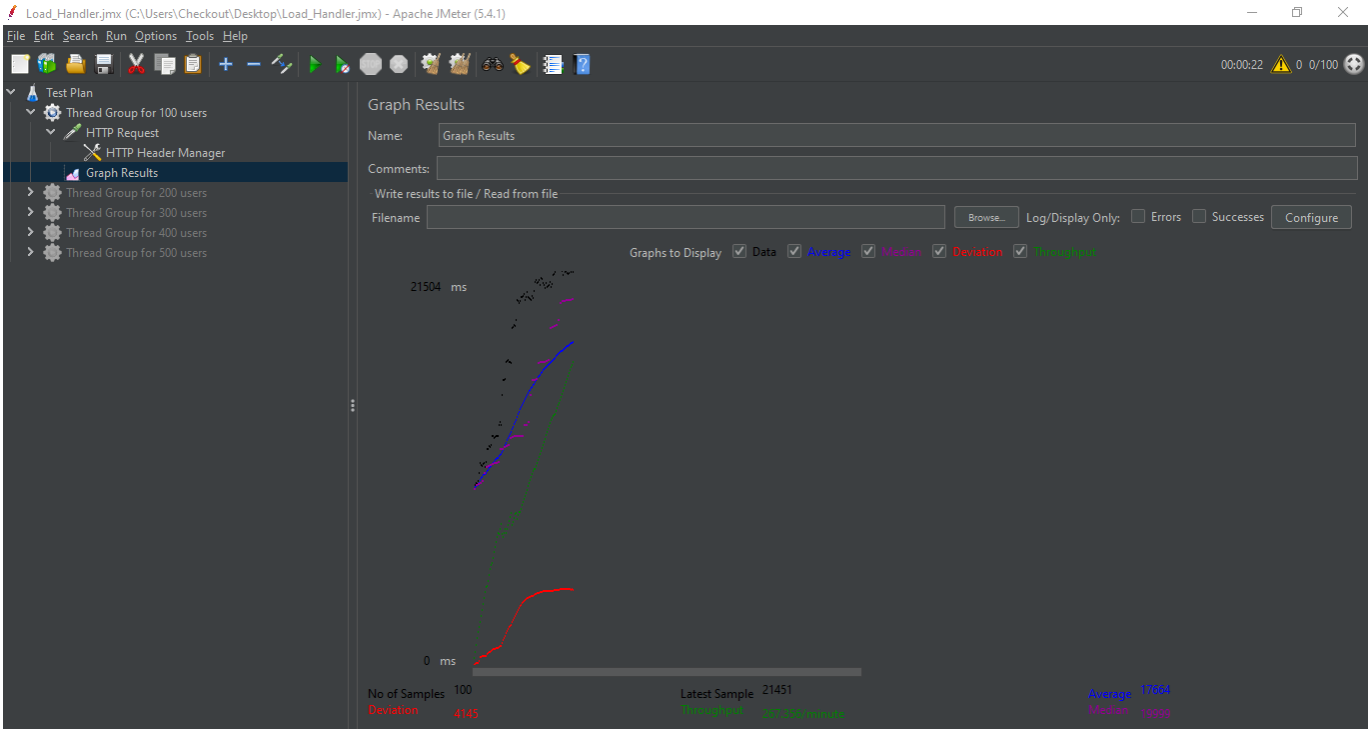
Srno	Testcase	Expected	Result
1	Do correct Add Appointment	Should add new appointment	Pass
2	Do incorrect Add Appointment	Should show error	Pass
3	Do correct Change Appointment	Should change existing appointment	Pass
4	Do incorrect Change Appointment	Should show error	Pass
5	Do delete Appointment	Should delete the appointment	Pass
6	Get Vaccination History	Should get all vaccination history	Pass
7	Get List of Appointments	Should get all list of appointments	Pass
8	Do correct checkIn	Should be able to checkIn	Pass
9	Do incorrect checkIn	Should not be able to checkIn	Pass
10	Do logout	Should redirect to landingPage	Pass

Observations on application performance

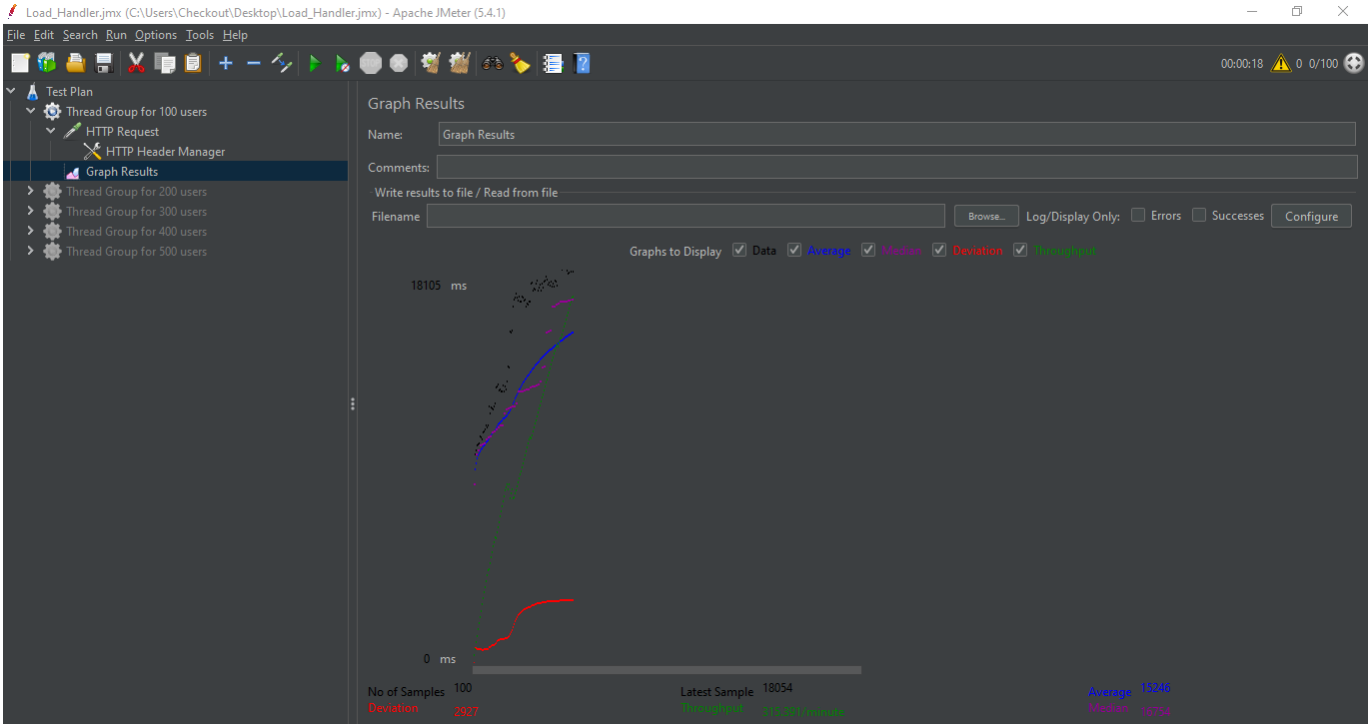
Load Balancer with 100 Threads and 1 services server



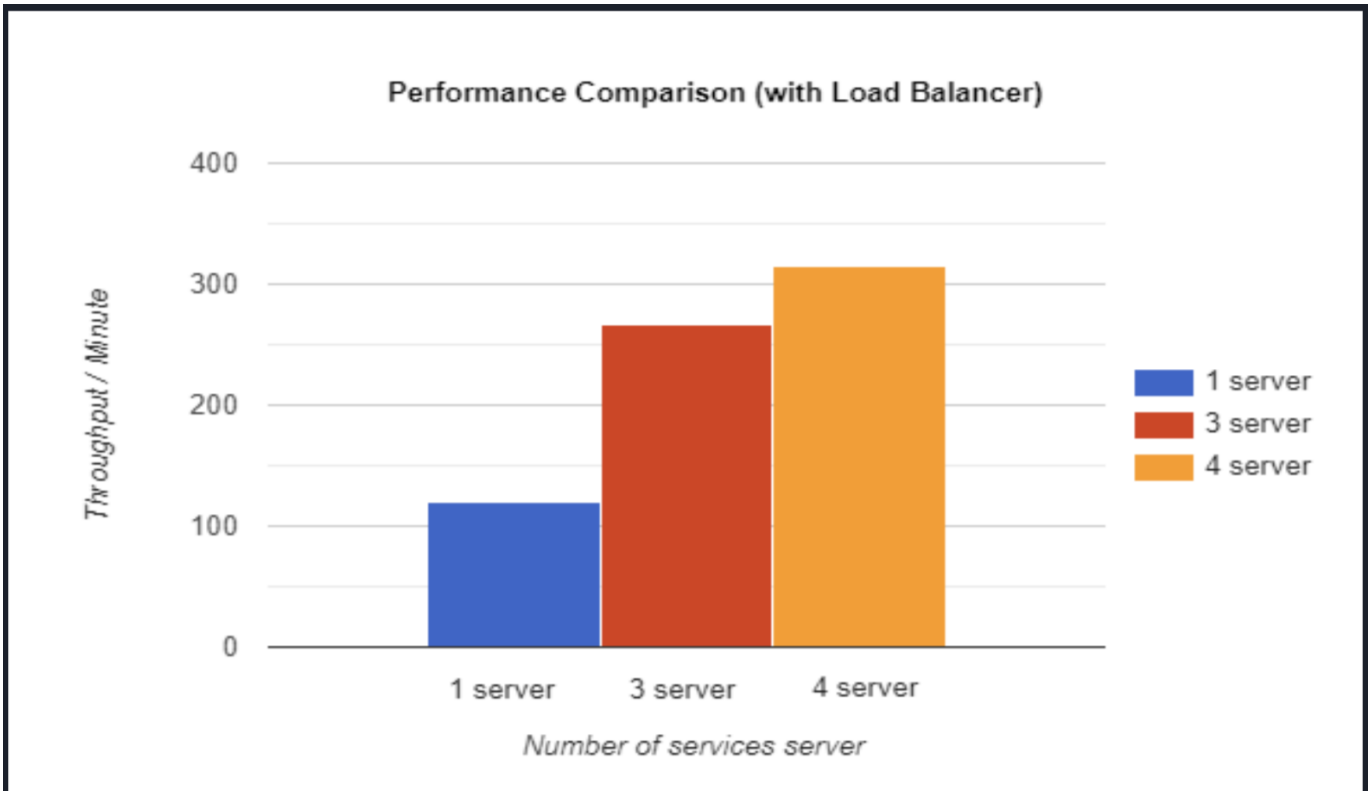
Load Balancer with 100 Threads and 3 services server



Load Balancer with 100 Threads and 4 services server



Comparative analysis of load balancer:



7. Lessons Learned & Possible future work:

The whole world is fighting with the covid pandemic, and the pandemic is lasting and spreading faster with lack of knowledge of the extent of the virus in certain regions. The project mentioned in this document eliminated this cause by providing the vaccination management system which can help users to book appointments for various vaccines. The Team followed scrum process and Agile methodology which helped in finishing the project on time. Currently we are assuming the stocks in the clinic are well equipped for Vaccination. We haven't considered the stock of vaccines at the clinic. The possible future work can involve the inventory of clinics and schedule of patients and doctors which can help in management of vaccination appointments.

-END OF REPORT-