Project Report

On

Covid-19 Vaccine Management System

Co-VMS

Submitted in partial fulfillment for the award of

Post Graduate Diploma in Advance Computing (PG-DAC)
from C-DAC, ACTS (Pune)

Guided by:

Mr. Suleman

Presented by:

Mr. Ankit Hepat	**PRN: 200240120029**
Mr. Anubhav Kumar	**PRN: 200240120035**
Ms. Arshi Saxena	**PRN: 200240120036**
Mr. Ashish Patidar	**PRN: 200240120040**

Centre for Development of Advanced Computing (C-DAC), Pune

ACKNOWLEDGEMENT

This project "**Co-VMS**" was a great learning experience for me and I am submitting this work to Advanced Computing Training School (CDAC ACTS).
I am very glad to mention the name of *Mr. Suleman* for his valuable guidance to work on this project. His guidance and support helped me to overcome various obstacles and intricacies during the course of project work.

I am highly grateful to Ms. Risha P.R. (Manager (ACTS training Centre), C-DAC, for her guidance and support whenever necessary while doing this course Post Graduate Diploma in *Advanced Computing (PG-DAC)* through C-DAC ACTS, Pune.

My heartfelt thanks goes to *Ms. Shilpi Shalini* (Course Coordinator, PG-*DAC*) who gave all the required support and kind coordination to provide all the necessities like required hardware, internet facility and extra Lab hours to complete the project and throughout the course up to the last day here in C-DAC ACTS, Pune.

From:

Mr. Ankit Hepat (200240120029)

Mr. Anubhav Kumar (200240120035)

Ms. Arshi Saxena (200240120036)

Mr. Ashish Patidar (200240120040)

TABLE OF CONTENTS

1. Introduction of Project:

Coronavirus disease (COVID-19) is an infectious disease that spreads rapidly throughout the world. In March 2020, the World Health Organization (WHO) declared the COVID-19 outbreak a pandemic. The pandemic has severely impacted health systems, economic and social progress throughout the world. In India, 1,08,15,687 confirmed COVID-19 cases and over 1,54,964 deaths have been reported as of 06th February 2021. While strong measures were adopted and some progress was made in containing the spread through better public health interventions, diagnostics and treatments, scientists across the world have accelerated the process to develop a safe and effective vaccine that will break the chain of transmission.

Co-VMS is a user centered web application which simulates vaccine administration for the general populous, thus providing a digital interface for the people to consent for vaccination and track their appointments. The various Government Installations to administer the vaccines, manage inventory and generate reports.

While the front-line health workers have been battling the virus head-on displaying high amounts of physical as well as moral courage, it presents immense challenges to manage the operations of containing the spread of the virus and providing adequate medical attention to all in a country such as India. The next foreseeable challenges that comes up are to manage the vaccination drives and logistics. Going digital is the best foot forward to tackle large amounts of data and streamlining administrative tasks. The obvious choice to overcome these challenges would be to develop an integrated web application which collates data at state and district level.

Our application provides an interface for the public to apply for vaccination and track their appointments with the services through the internet, and also help the Government Installations to manage the logistics of the vaccine on a single integrated platform.

The applications also serves as a real-time data point for the statistical analysis of coronavirus metrics throughout the country at different levels, which helps in managing the supply chain of the vaccine inventory.

Co-VMS is a comprehensive suite of services which provides an application that assists with a variety of services management tasks and service beneficiary requests from the time of initial registration to tracking of the vaccinated population.

A Vaccination Drive requires scheduling of appointments to directing the user to the appropriate vaccination centre for the scheduled appointment and managing the inventory and daily vaccination capacity of the centre.

2. Product Overview and Summary

2.1 Purpose:

To apply for vaccination, track scheduled appointments and provide post vaccination feedback. To request for vaccine stock, vaccinate scheduled beneficiary and process submitted feedback. Using this application the beneficiary, vaccination centre, district office can interact with Co-VMS services from anywhere.

2.2 Scope:

Co-VMS is a web application to assist a user(beneficiary) for scheduling appointments for vaccination, tracking their appointments and submitting post vaccination feedback.
It also assists various Government Installations(vaccination centres, district offices) for managing vaccine inventory and summarizing the user data for analysis and decision making.

2.3 Overview:

Section 3.0, the Overall Description, provides an overview of the components and the relationship between them. Section 4.0 provides the Specific Requirements of the application. In the subsection (4.1) and (4.2) of which the various functional requirements and various interface respectively are discussed. Section 5.0 describes Database Design details.

2.4 Feasibility Study

Feasibility is a determination of whether a project is worth doing or not. Before actually recommending the new system it is important to investigate if it is feasible to develop the new system.

Before developing and implementing a system, we have to be sure that our system is feasible in the following ways:

1. **Technical Feasibility:**

In the type of feasibility study, the system analyst has to check whether it is possible or not to develop the requested system with availability of manpower, software, hardware, etc.. The system which we run in Linux as well as windows platform and hence are suitable for the end-user. The system is technically feasible because it does not require too much manpower and runs with the basic available equipment.

2. **Operational Feasibility:**

In this type of feasibility study the operation implementation of the system is considered. Checking is done regarding whether it is feasible for the user department to use the software or will there be any inertial resistance from the users. Thus the proposed system is said to be operationally feasible only of the end users are able to understand the system clearly and correctly and can use the system with ease and with the minimum training. We need to train our staff so that system will be handled efficiently. As the system developed is very user-friendly and easy to operate for any person with minimum computer knowledge of computer is also able to handle our system. It is also easy to operate due to the user-friendly interface developed using Java and Angular.

3. **Economical Feasibility:**

In this type of feasibility study, the benefits of the system to the organization are considered by taking into consideration the cost-benefit analysis. The basic software, which is required for the implementation of the system, is Java which easily available. Also with the basic training user can use this software thus reducing the training cost to the organization. Thus, using this system is feasible for the organization and loading Java and the proposed system is economically feasible for the organization. As our system goes online we will have a lot of customers adding to our publicity. This in turn will increase our profit.

3. Overall Description:

3.1 Product Features

The main feature of this system is the user can apply for vaccination. The system shows the schedule of the appointment. The user must be a registered before applying for vaccination. The vaccination centres can manage vaccine inventory and vaccinate scheduled beneficiaries.
The district offices can approve or reject vaccination centre applications, can allot vaccine stock to respective vaccination centres and track district level coronavirus metrics and vaccination drive metrics.
The administrator(Central Government) can approve or reject district office applications and track country level coronavirus metrics and vaccination drive metrics.

3.2 Technology Used

BACK END
Spring Boot
Spring Data JPA with Hibernate ORM
Spring Security
MYSQL

FRONT END
HTML5
CSS3
Bootstrap 4
Angular 2+

3.2 User Classes

There are four types of user to this system one is the Beneficiary(General Populous) which can apply for vaccination and submit post vaccination feedback. Vaccination Centres which can manage vaccine inventory and vaccinate scheduled beneficiaries. District Offices can approve or reject vaccination centre applications, can allot vaccine stock to respective vaccination centres and track district level coronavirus metrics and vaccination drive metrics. Administrator(Central Government) can approve or reject district office applications and track country level coronavirus metrics and vaccination drive metrics.

3.3 General Constraints

The "Co-VMS" should run on all Internet Browsers and all processors which support the said Internet Browser, provided one has a working internet connection.

4. REQUIREMENTS

4.1 FUNCTIONAL REQUIREMENTS

4.1.1 Complete System:

ADMINISTRATOR

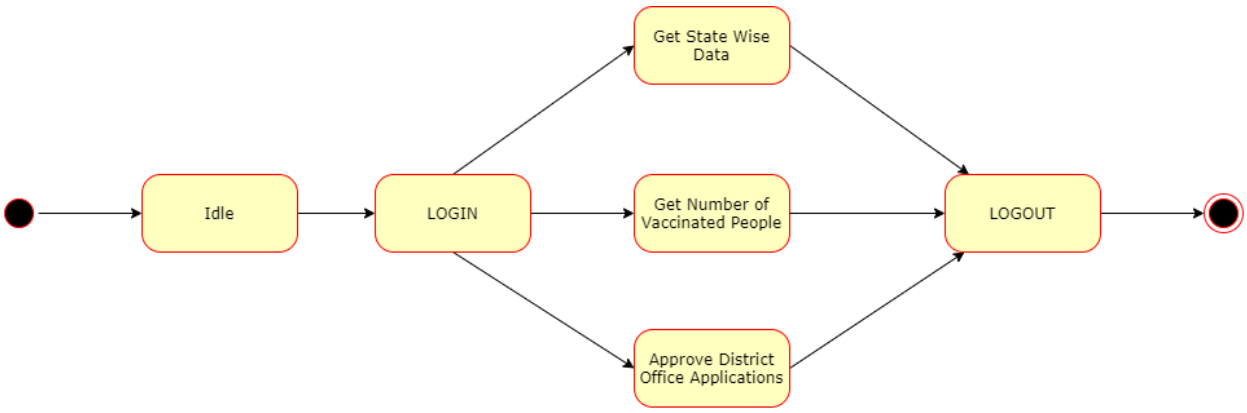


fig 1.1 Admin Activity Diagram

DISTRICT OFFICE

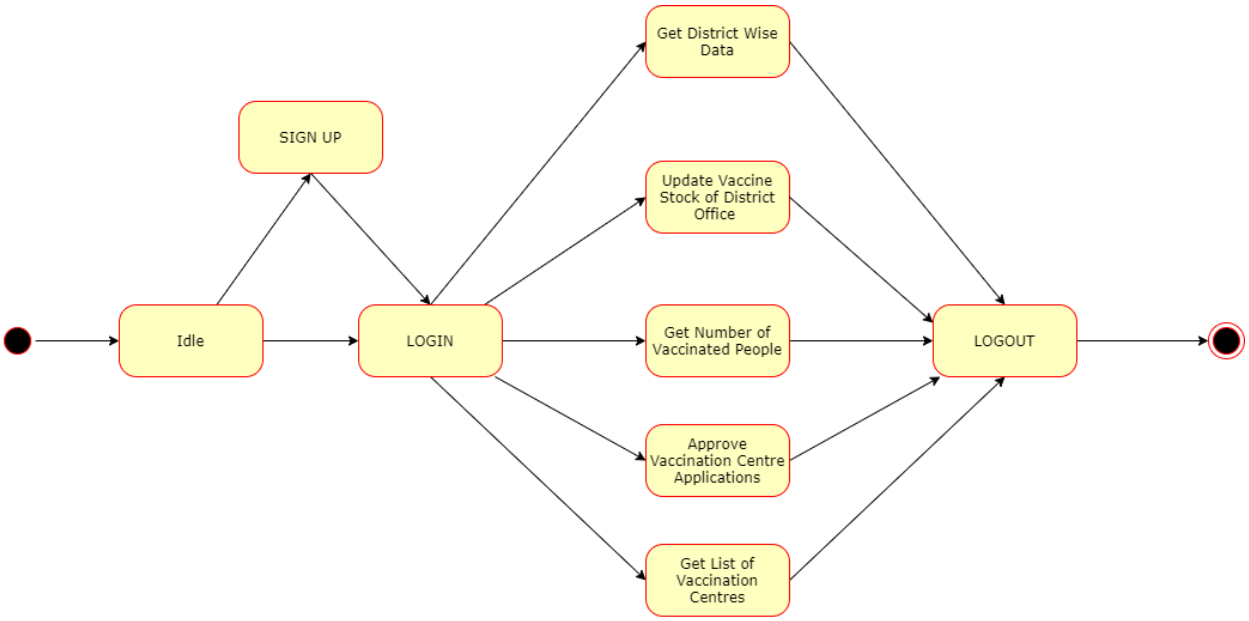


fig 1.2 District Office Activity Diagram

VACCINATION CENTRE

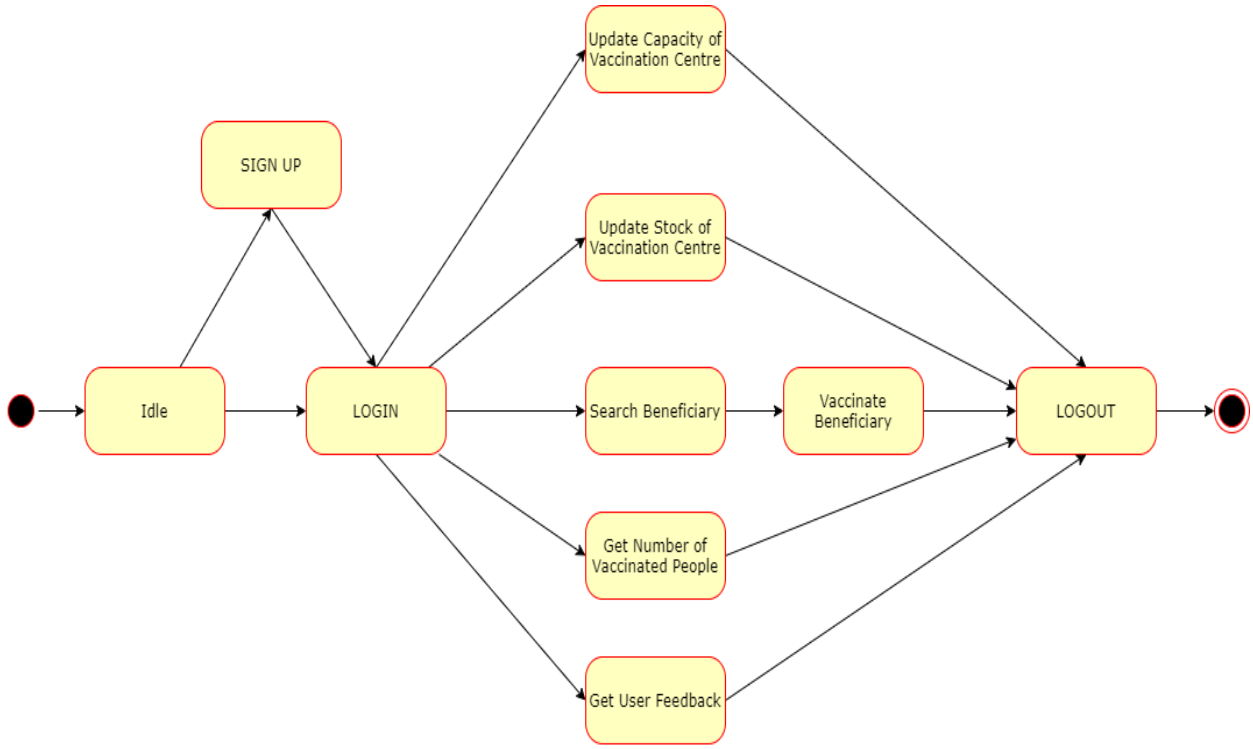


fig 1.3 Vaccination Centre Activity Diagram

BENEFICIARY

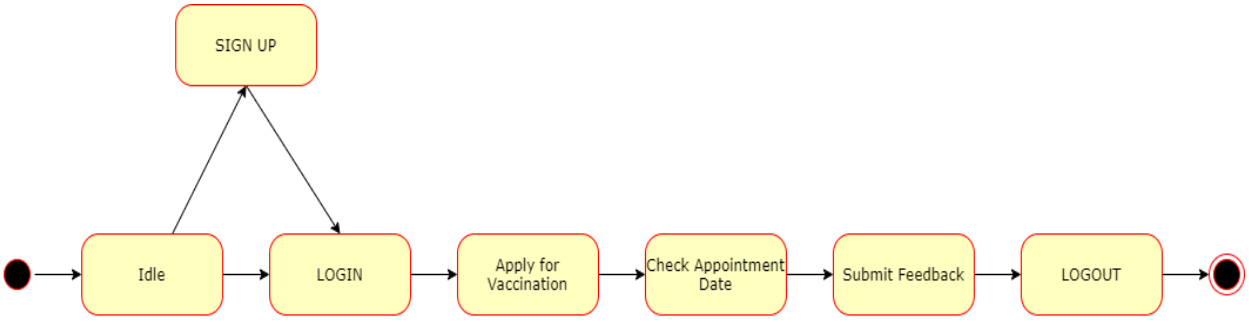


fig 1.4 Beneficiary Activity Diagram

UI : Login

There is an entry interface that is intended to facilitate the user (Administrator, District Office, Vaccination Centre, Beneficiary) to login to the system. If user is not registered before then they can register themselves with system by using a Register link and by filling registration form. User has to enter information(Name, Password, Role, Name, State, District).

Scenario 1: Mainline Sequence
1. Administrator: Enter Admin username and password.
2. System: Display the Admin home page where they can approve or reject district office applications and track country level coronavirus metrics and vaccination drive metrics.

Scenario 2: Mainline Sequence
1. District Office: Enter District Office username and password.
2. System: Display the District Office home page where they they can approve or reject vaccination centre applications, can allot vaccine stock to respective vaccination centres and track district level coronavirus metrics and vaccination drive metrics.

Scenario 3: Mainline Sequence
1. Vaccination Centre: Enter Vaccination Centre username and password.
2. System: Display the Vaccination Centre home page where they can manage vaccine inventory and vaccinate scheduled beneficiaries.

Scenario 4: Mainline Sequence
1. User: Enter username and password.
2. System: Display the User home page where they can apply for vaccination, track appointment schedule and submit feedback.

4.2 Administrator Task:

ADMINISTRATOR

U1: Disaply State-wise Covid Data:

Scenario 1: Mainline Sequence:
1. Admin: Admin logins.
2. System: Opens AdminDashboardComponent Which Shows Covid Data.
3. Admin: Admin Selects State from dropdown.
4. System: REST Call which gets data form Covid19 API.

U2 : Get Report of District-wise Vaccinated Count:

Scenario 1: Mainline Sequence:
1. Admin: Admin Selects State and District from dropdown.
2. System: REST Call which gets data.

U3: Approve District Office Applications:

Scenario 1: Mainline Sequence:
1. Admin: Admin Approves District Office Applications.
2. System: REST Call which sets isApproved status true for district office.

4.3 District Office Task:

DISTRICT OFFICE

U1: Display Respective District Covid Data:

Scenario 1: Mainline Sequence:
1) District Office: District Office logins.
2) System: Opens DistrictOfficeDashboardComponent which shows covid data.

U2 : Update Inventory:

Scenario 1: Mainline Sequence:
1. District Office: District Office Clicks on update.
2. System: REST Call which updates vaccine_inventory.

U3: Get Report of Vaccinated Count:

Scenario 1: Mainline Sequence:
1. District Office: District Office Clicks on update.
2. System: REST Call which gets vaccinated count data.

U4: Approve Vaccination Centre Applications:

Scenario 1: Mainline Sequence:
1. District Office: District Office Approves Vaccination Centre Applications.
2. System: REST Call which sets isApproved status true for vaccination centre.

U5: Get List of Vaccination Centres:

Scenario 1: Mainline Sequence:
1. District Office: District Office Clicks on Centre List.
2. System: REST Call which gets List of Approved Vaccination Centres.

4.4 Vaccination Centre Task:

VACCINATION CENTRE

U1: Display Vaccination Centre Dashboard:

Scenario 1: Mainline Sequence:
1. Vaccination Centre: Vaccination Centre logins.
2. System: Opens VaccinationCentreDashboardComponent which shows vaccination protocol to be followed.

U2: Update Vaccination Capacity and Stock:

Scenario 1: Mainline Sequence:
1. Vaccination Centre: Vaccination Centre updates capacity and stock.
2. System: REST Call which updates vaccine_inventory.

U3: Get List of Scheduled Beneficiaries:

Scenario 1: Mainline Sequence:
1. Vaccination Centre: Vaccination Centre clicks on requests.
2. System: REST Call which fetches list of scheduled beneficiaries.

U4: Get Report of Total Vaccinated Count by Centre:

Scenario 1: Mainline Sequence:
1. Vaccination Centre: Vaccination Centre clicks on reports.
2. System: REST Call which gets vaccinated count data.

U5: View Beneficiary Feedback:

Scenario 1: Mainline Sequence:
1. Vaccination Centre: Vaccination Centre clicks on feedback.
2. System: REST Call which gets vaccinated beneficiaries' feedback.

4.5 Beneficiary Task:

BENEFICIARY

U1: Display Beneficiary Dashboard:

Scenario 1: Mainline Sequence:
1. Beneficiary: Beneficiary logins.
2. System: Opens BeneficiaryDashboardComponent which shows covid instructions.

U2: Apply For Vaccination:

Scenario 1: Mainline Sequence:
1. Beneficiary: Beneficiary clicks on apply.
2. System: REST Call which accepts adhaar number and age and schedules appointment as per availability.

U3: View Appointment:

Scenario 1: Mainline Sequence:
1. Beneficiary: Beneficiary clicks appointment.
2. System: REST Call which fetches scheduled appointment.

U4: Submit Feedback:

Scenario 1: Mainline Sequence:
1. Beneficiary: Beneficiary clicks feedback.
2. System: REST Call which submits the feedback to vaccination centre.

4.6 System Requirement

4.6.1 Hardware Requirement

- ☐ Processor: 2.0+ GHZ Single Core Processor
- RAM: 2 GB
- Operating System: Windows/Mac OS/Linux
- Internet Connection

4.6.2 Software Requirement

- ☐ Internet Browser
- Operating System: Windows/Mac OS/Linux
- JDK 1.8+
- Angular CLI

5.Design:

5.1 Database Design

The following table structures depict the database design.

Table1: user

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null
pk	id	int	10	No
uni	email	varchar	100	Yes
	password	varchar	100	Yes
	name	varchar	50	Yes

Table2: role

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null
pk	id	int	10	No
	role_type	varchar	255	Yes

Table3: user_roles

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null
pk	user_id	int	10	No
mul	role_id	int	10	Yes

Table4: state

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null
pk	id	int	10	No
	state_name	varchar	50	Yes

Table5: district

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null
pk	id	int	10	No
	district_name	varchar	50	Yes
mul	state_id	int	10	Yes

Table6: admin

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null
pk	id	int	10	No
	name	varchar	50	Yes

Table7: district_office

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null
pk	id	int	10	No
	district_inventory	int	10	Yes
	is_approved	bit	1	Yes
	office_name	varchar	50	Yes
mul	district_id	int	10	Yes

Table8: vaccination_centre

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null
pk	id	int	10	No
	centre_name	varchar	50	Yes
	district_id	int	10	Yes
	district_office_id	int	10	Yes

Table9: centre_inventory

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null
pk	id	int	10	No
	capacity	int	10	Yes
	inventory	int	10	Yes
	centre_id	int	10	Yes

Table10: beneficiary

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null
pk	id	int	10	No
	adhaar_no	varchar	12	Yes
	age	int	10	Yes
	is_vaccinated	bit	1	Yes
	name	varchar	50	Yes
mul	district_id	int	10	Yes
mul	centre_id	int	10	Yes

Table11: appointment

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null
pk	id	int	10	No
	appointment_date	date		Yes
	is_active	bit	1	Yes
mul	beneficiary_id	int	10	Yes

Table12: beneficiary_feedback

Key Type/ Constraint	Column Name	Data Type	Length	Allow Null
pk	id	int	10	No
	adverse_effect	varchar	50	Yes
	details	varchar	200	Yes
mul	beneficiary_id	int	10	Yes

5.2 ER Diagram

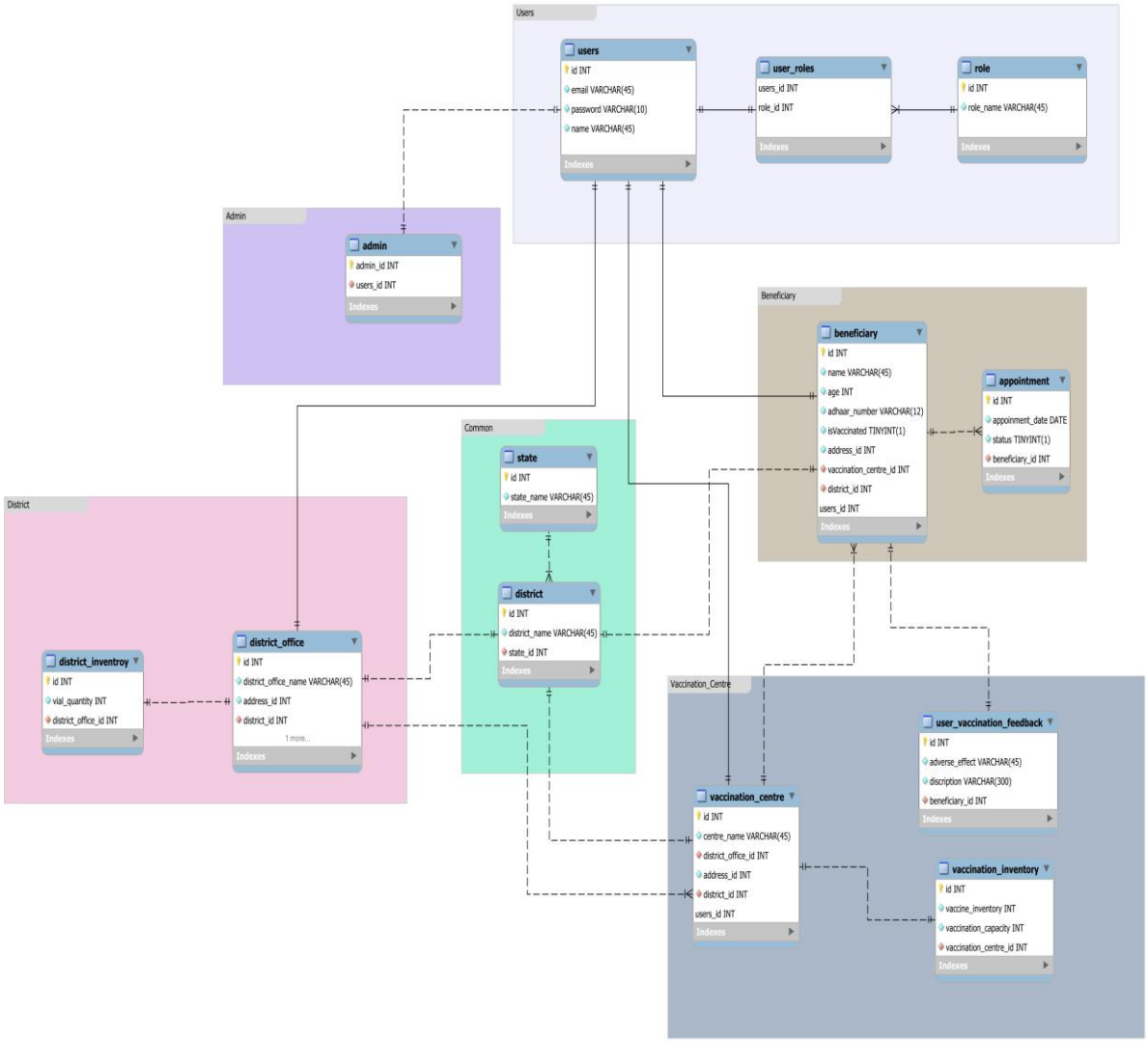


fig 2.1 ER Diagram

5.3 CLASS DIAGRAM

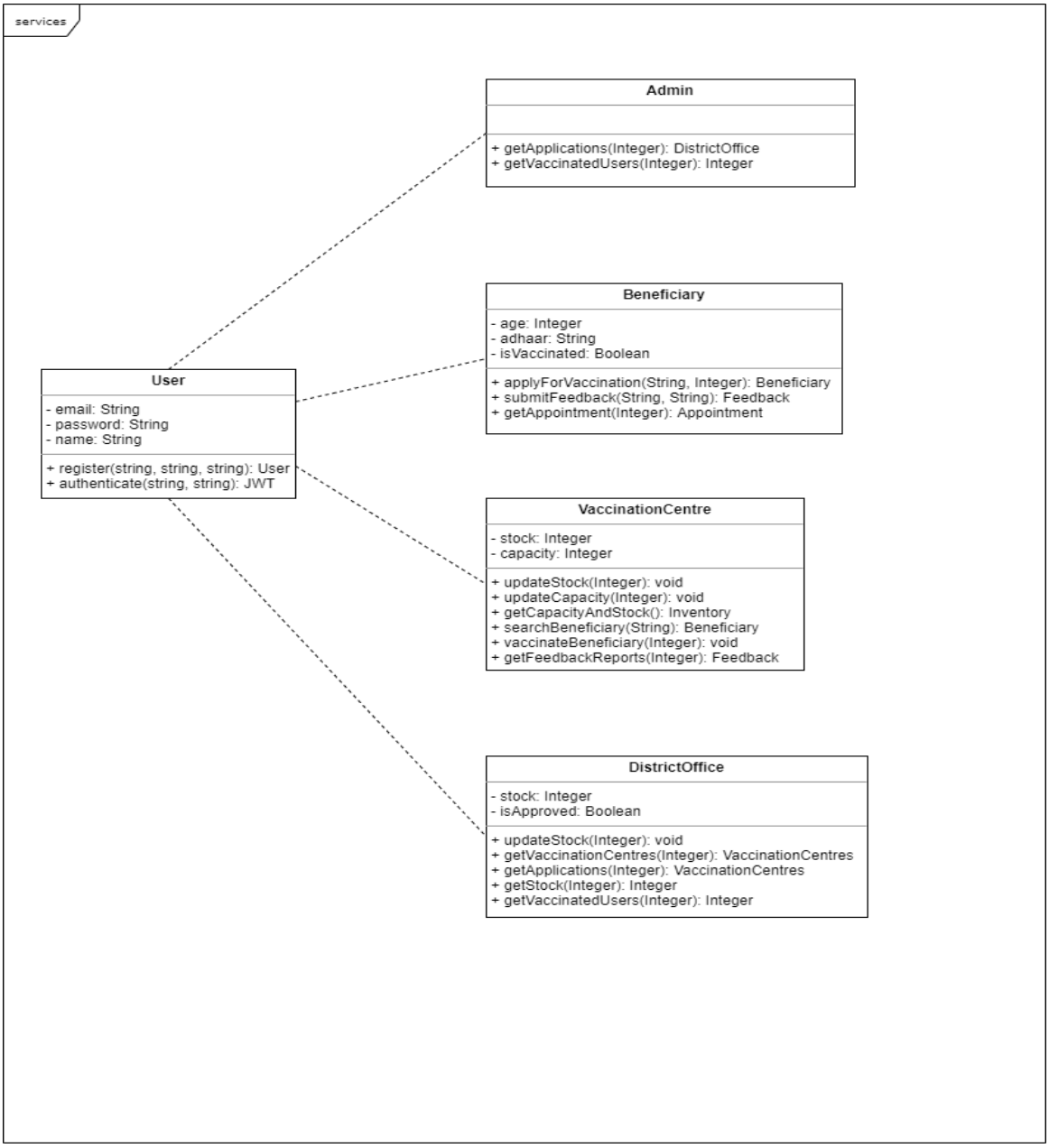
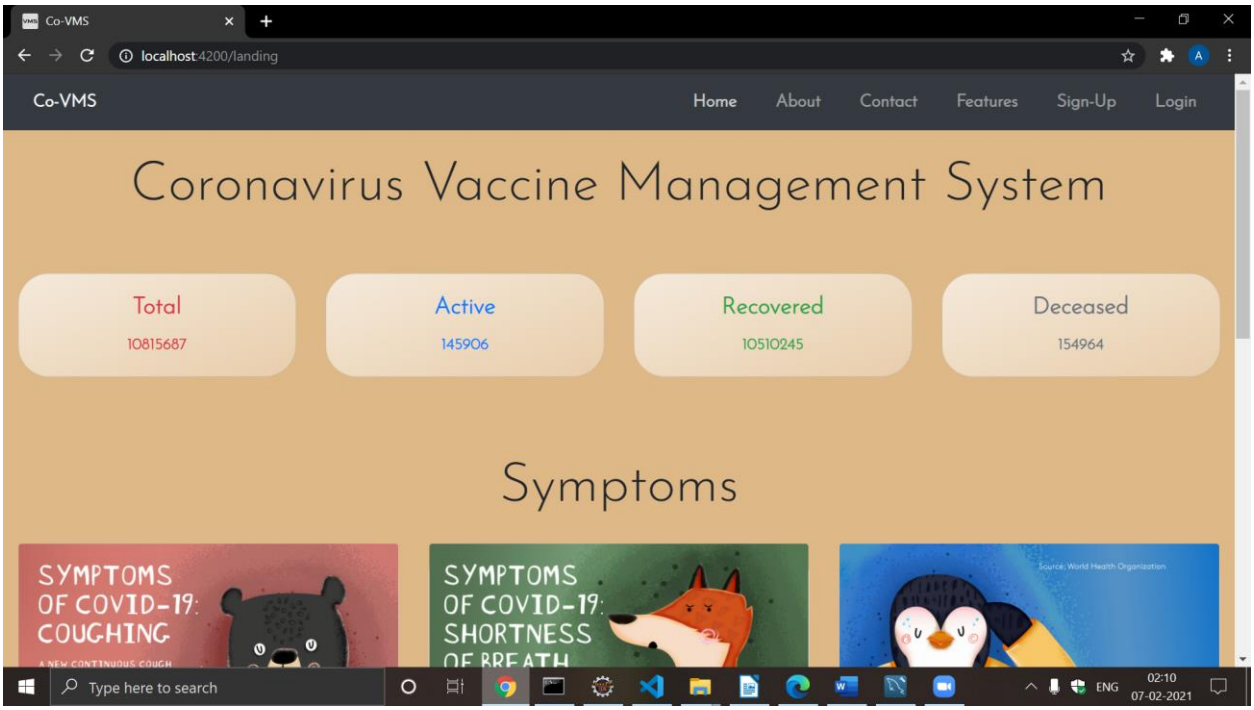


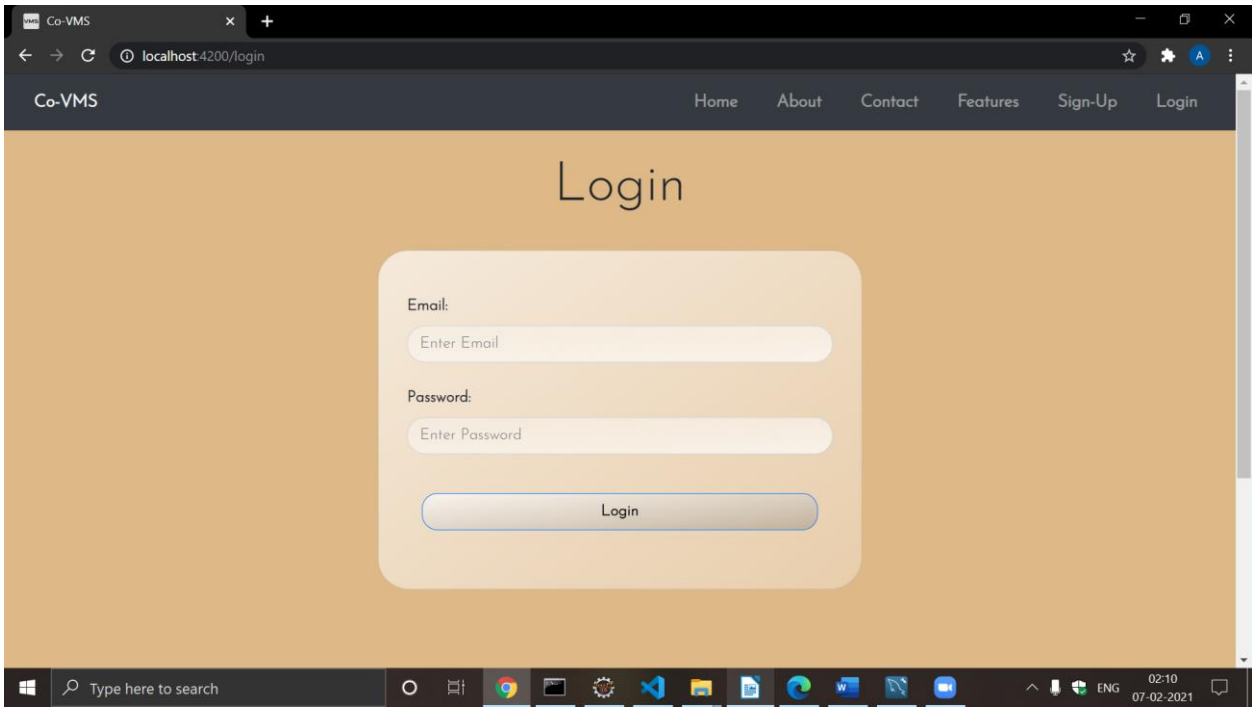
fig 3.1 Class Diagram

6. Interfaces:

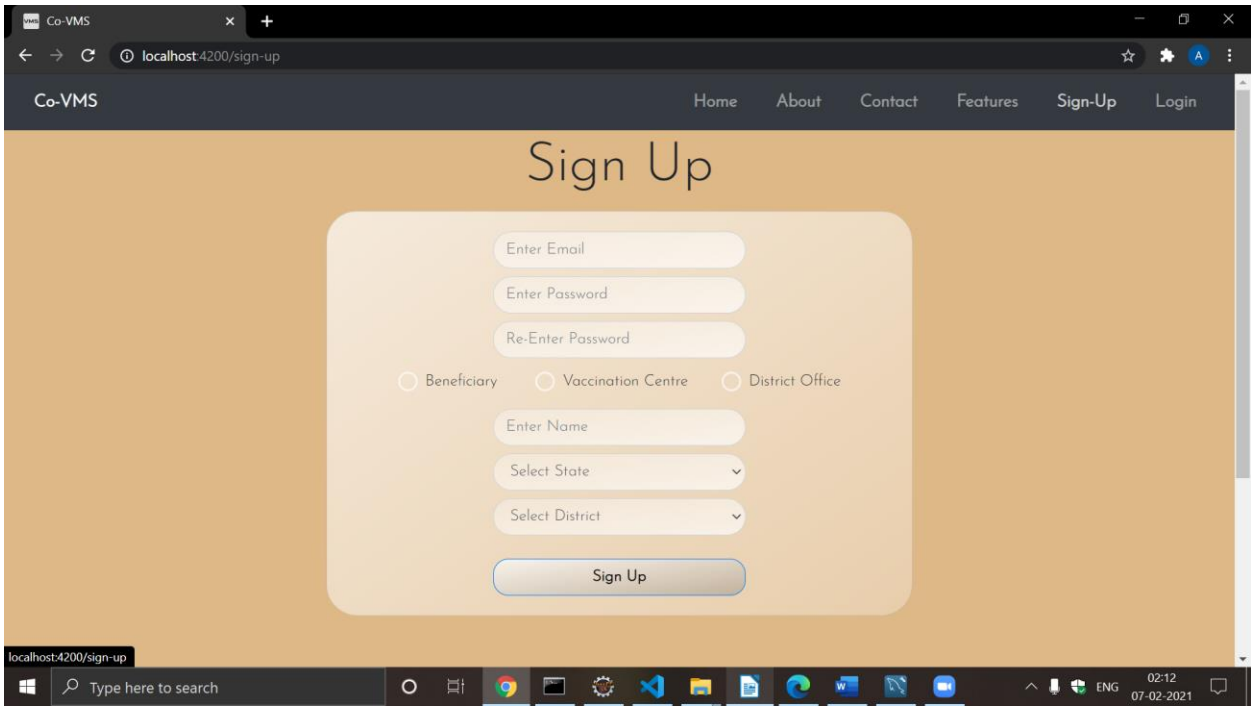
1) Home Page



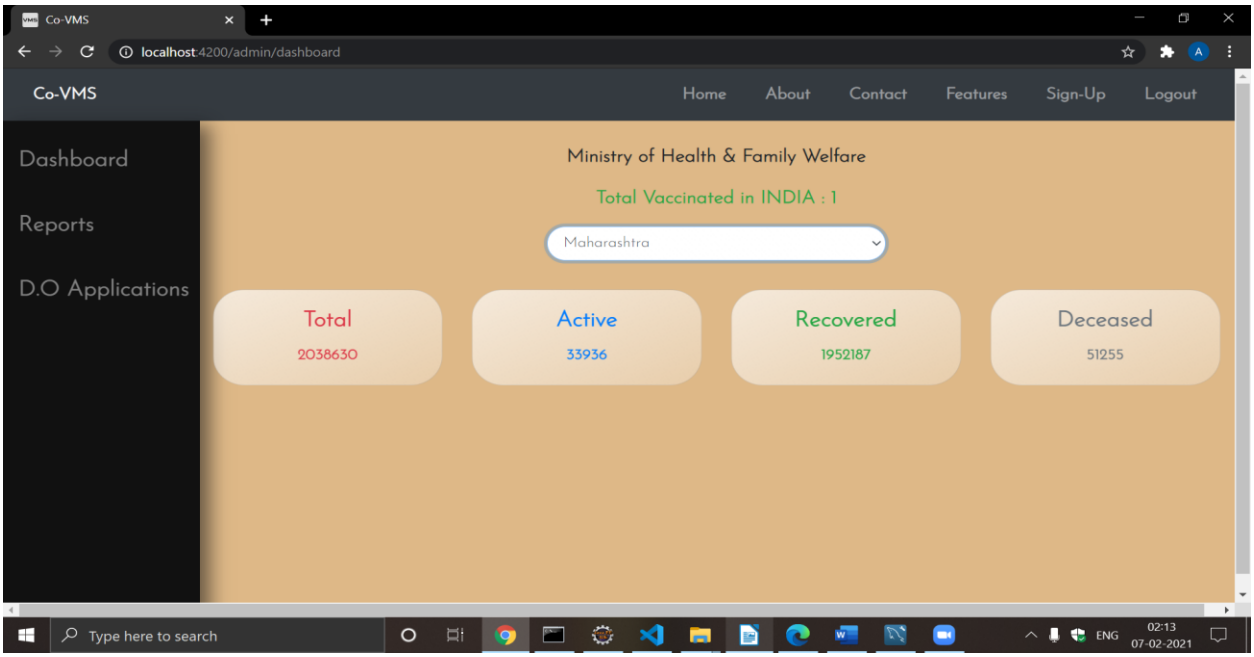
2) Login


Co-VMS
localhost:4200/login
Co-VMS
Home
About
Contact
Features
Sign-Up
Login
Login
Email:
Enter Email
Password:
Enter Password
Login
Type here to search
ENG
02:10
07-02-2021

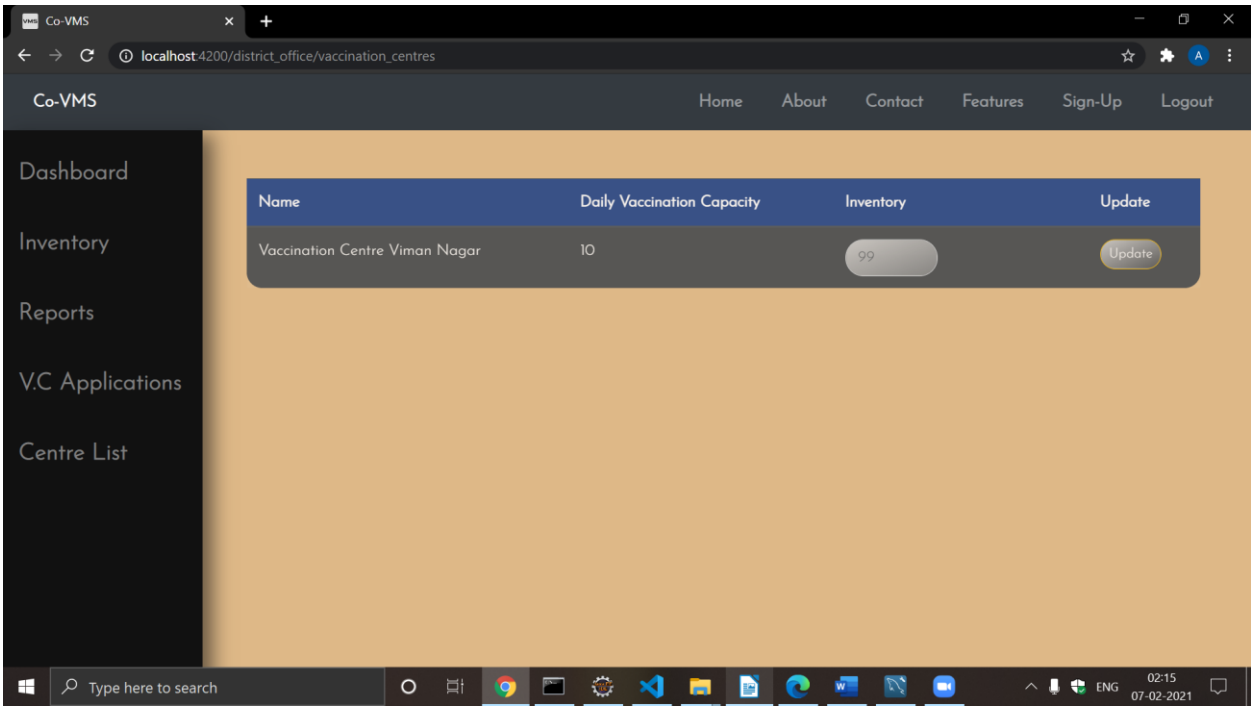
3) Sign Up


Co-VMS
localhost:4200/sign-up
Co-VMS
Home
About
Contact
Features
Sign-Up
Login
Sign Up
Enter Email
Enter Password
Re-Enter Password
Beneficiary
Vaccination Centre
District Office
Enter Name
Select State
Select District
Sign Up
localhost:4200/sign-up
Type here to search
ENG
02:12
07-02-2021

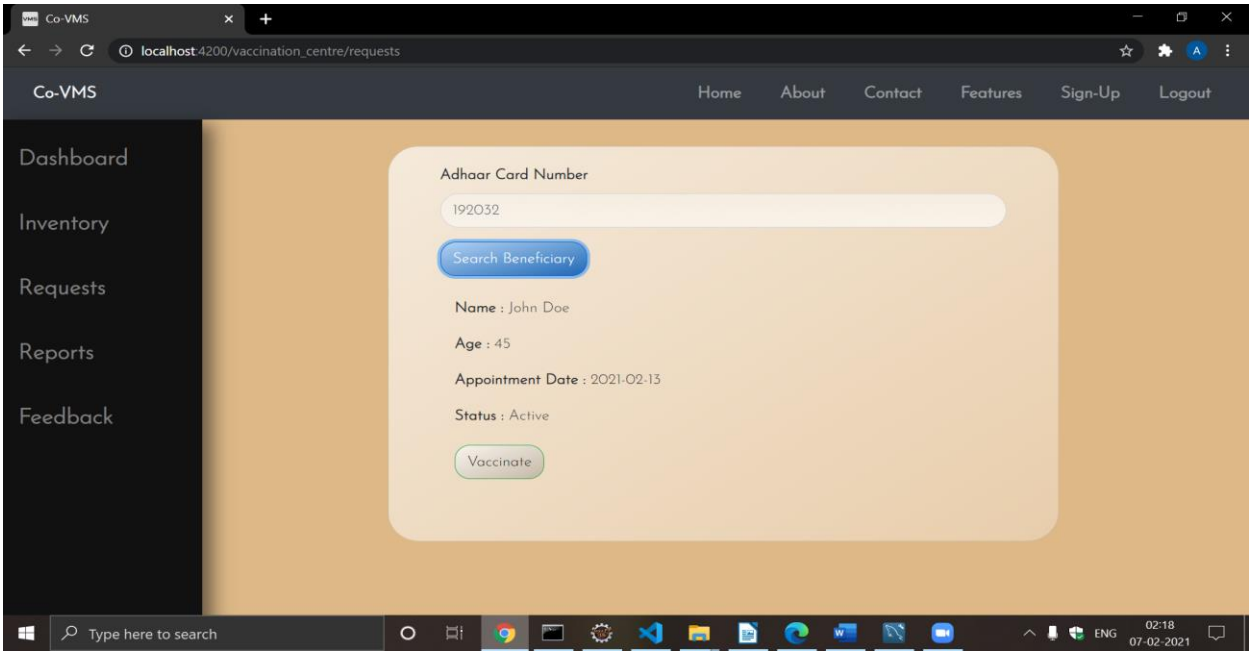
4) Admin Dashboard



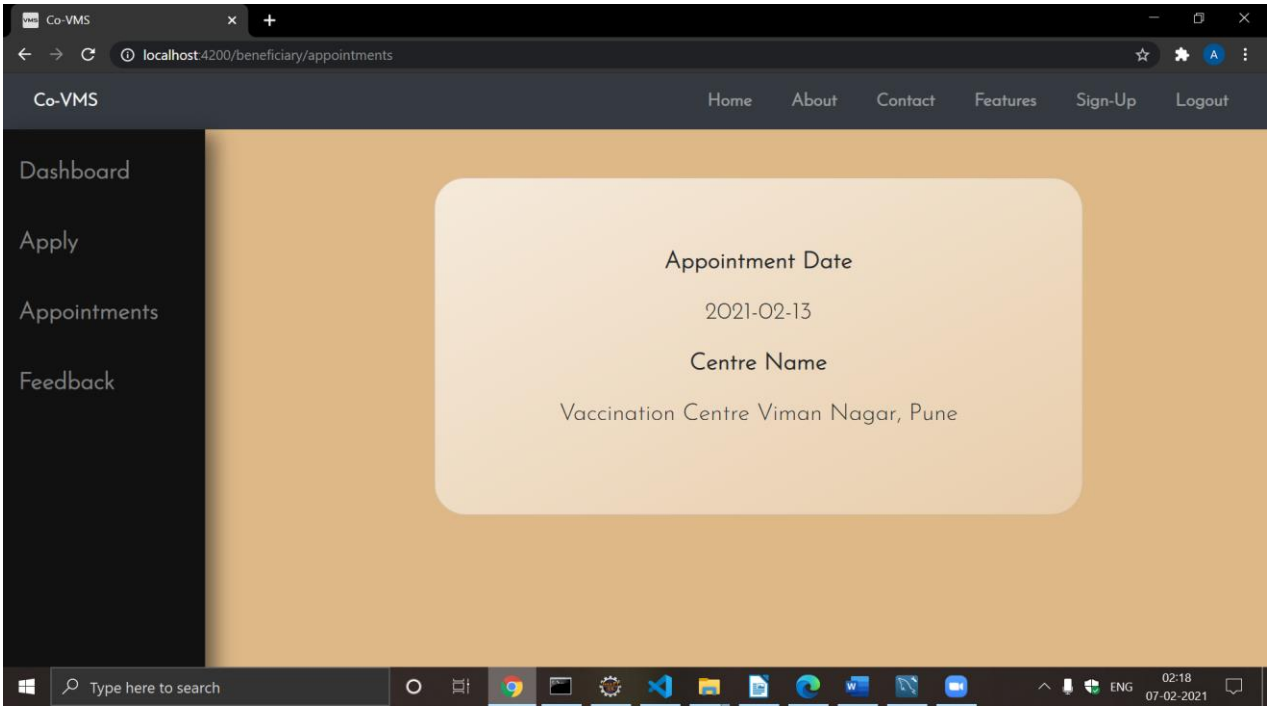
5) Centre List



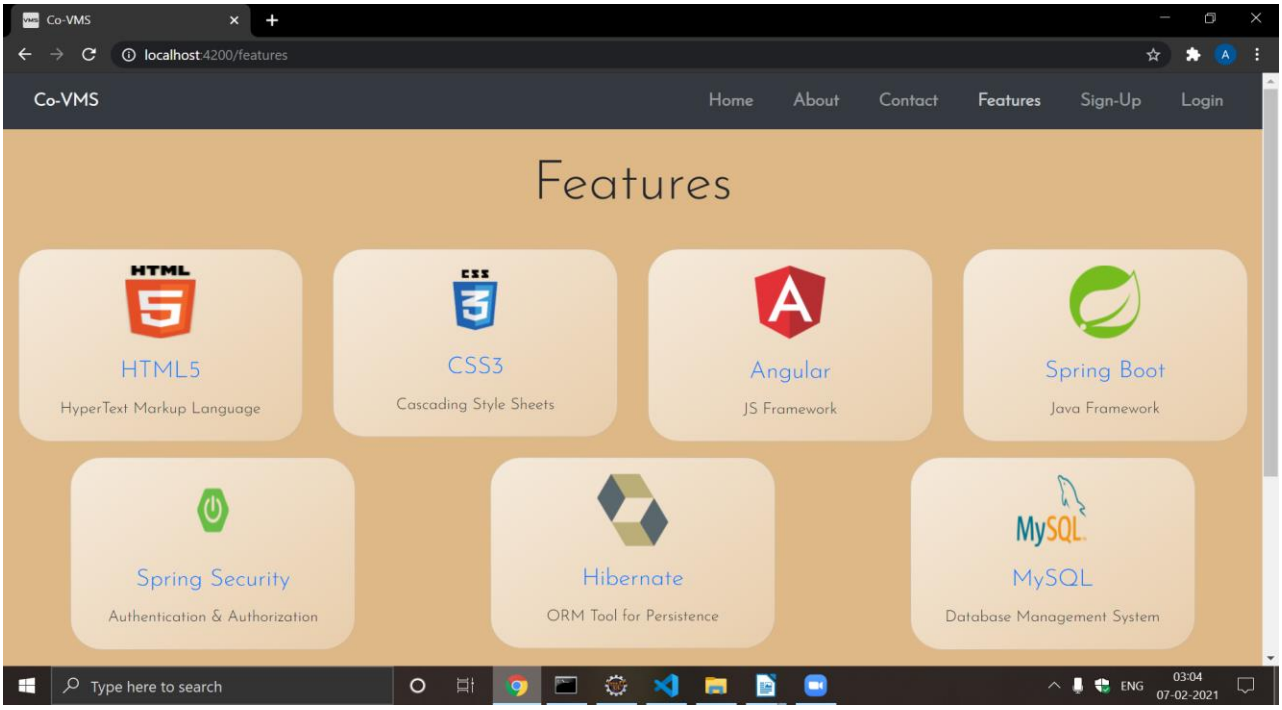
6) Vaccination Request



7)Scheduled Appointment



8) Features


Co-VMS
localhost:4200/features
Co-VMS
Home
About
Contact
Features
Sign-Up
Login
Features
HTML5
HyperText Markup Language
CSS3
Cascading Style Sheets
Angular
JS Framework
Spring Boot
Java Framework
Spring Security
Authentication & Authorization
Hibernate
ORM Tool for Persistence
MySQL
Database Management System
Type here to search
ENG
03:04
07-02-2021

7. TEST REPORT

The report of the testing is given here under.

Project Name :- Co-VMS

Sr. No	Test Title	Description	Expected Outcome	Error Message	Result
1	Login Page-	If User email=user email, Password= user Password	If Validated allow for Role Based Home Page. If not redirect to same page.	Username and password required	Passed
2	Sign Up Page	Should not allow any control to be empty if not null	If validated Allow to go to login page	Suitable Validation Error	Passed
3	Dahsboard page Displayed	Dahsboard display for every successful log in.	Dahsboard Page Displayed	No Error	Passed
4	Show Covid Data	User Can See Covid Data	Covid Data	No Error	Passed
5	Vaccination	Vaccinate beneficiary	If stock is >0 then vaccinate. If not then no vaccination	No Error	Passed
6	Feedback	Beneficiary Feedback	If vaccinated can give feedback. If not feedback not allowed.	No Error	Passed
7	Approve District Office	Approval of District Office	If approved then perform actions. If not no action allowed.	District Office Not Approved	Passed
8	Approve Vaccination Centre	Approval of Vaccination Centre	If approved then perform actions. If not no action allowed.	Vaccination Centre Not Approved	Passed
9	Log out	User logout by using Logout Link	Redirect to Login Page	No Error	Passed

8. FUTURE SCOPE :

1. Provide appointment date for second dose of the vaccine.
2. Optimization of Appointment Scheduling Algorithm to include age-group based priority scheduling.
3. Addition of Supplier Actor to further regulate supply chain.
4. Addition of Centre Actor to include demand planning in order to optimize supply chain management.
5. Integration of Payment Service for Beneficiary.

9. REFERENCES :

https://www.mohfw.gov.in/
https://www.covid19india.org/
https://angular.io/docs
https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/
https://docs.oracle.com/javase/8/docs/api/
https://javaee.github.io/javaee-spec/javadocs/