Project Report

on

CDAC Certificate Verification (CCV)

“Verification platform”

Submitted in partial fulfillment for the award of

DIPLOMA IN ADVANCED COMPUTING(**DAC**)

From

C-DAC, HYDERABAD

Guided by:

Prof. Sadhu Sreenivas

Presented by:

220350320024 Choudhari Riddhi Kishor

220350320025 Dangi Rawalnath Satappa

220350320026 Deodhar Sarang Girish

220350320027 Deshmane Saurabh Amol

220350320028 Devane Dharmveer NivurattiS

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

CERTIFICATE

This is to certify that the project work under the title **'CDAC Certificate Verification (CCV)'** is done by

Choudhari Riddhi Kishor	220350320024
Dangi Rawalnath Satappa	220350320025
Deodhar Sarang Girish	220350320026
Deshmane Saurabh Amol	220350320027
Devane Dharmveer NivurattiS	220350320028

in partial fulfillment of the requirement for award of Diploma in Advanced Computing Course.

Guided by:

Prof. Sadhu Sreenivas

Course Coordinator, E-DAC

Mr. Sharanabassappa

ACKNOWLEDGEMENT

Project "**CDAC Certificate Verification (CCV)**" is the product of our hard teamwork and patience, it brought us a great learning experience. There were many challenges, ups and downs we faced during its development and with immense pleasure we would like to present this work to Centre of Advanced Computing, Hyderabad.

We are highly grateful to our mentor **Prof. Sadhu Sreenivas** for his valuable guidance to work on this project. His guidance and support helped us to overcome various obstacles and intricacies during the course of project work.

Our heartfelt thanks goes to **Mr. Sharanabassappa** (Course Coordinator, E-DAC) who gave all the required support and kind coordination to provide all the necessities and extra hours for course completion and supporting us throughout the course up to the last day here in C-DAC, Hyderabad.

From:

Choudhari Riddhi Kishor

Dangi Rawalnath Satappa

Deodhar Sarang Girish

Deshmane Saurabh Amol

Devane Dharmveer NivurattiS

TABLE OF CONTENTS

1. Introduction

CDAC Certificate Verification (CCV) is a web app having a goal to reduce the time to design and implement an enhanced web-based certificate verification system that will assist co-operate organizations, and college institutes to confirm the originality of students' certificates.

User Interface, developed in React uses user email to authenticate and data is displayed using REST. UI makes secure calls to Spring Boot, we develop this project in microservice architecture. In the backend, JAVA is used to fetch and manipulate the data and used MySQL as database.

2. Overview

2.1 Purpose

Many Institutes like CDAC does not have a common online platform where organization can verify their candidate's documents. Before our proposal to this website all the work are done manually and takes a lot of time. This work was motivated by the level of certificate forgery, time wastage, and the stress encountered while processing this manually.

2.2 Summary

So, basically **CDAC Certificate Verification (CCV)** provides fast verification platform for the organizations. This will reduce the human interaction and time taken by them.

3. Description

3.1 Features

The main features of the website are :

- It provides a registration page for users confirming that organizations to maintain the record.
- A login page for the organization as well as to the admin so that authentication of the users can be done. It also add a perk to security so that keep the data away other peoples.
- A dashboard, it provides an Interface to end users,
  - From organization's dashboard, they can edit their own data (if required).
  - Also, they can verify candidate's information
  - From Admin's dashboard, they can edit, delete and add student's data. For organization records – admin can only delete the records but cannot edit them.
- We have use microservices in our backend module, so we can store every module in separate server as module's requirement.

3.2 Technology Used

- Back-end:
  - Spring Framework(Spring boot)-> Dependencies- Spring Web, Spring Dev-Tools, MySQL driver, JPA Repository, JDBC Drivers
- Front-end:
  - HTML 5, CSS 3, Bootstrap 4, ES6(Vanilla JavaScript), ReactJS Framework.
- Middle-ware:
  - Axios Library.
- Database
  - MySQL 8
- Platform:
  - MySQL Workbench + Client, eclipse, STS, Visual Studio Code, PostMan (for testing).

J2EE Spring Boot –

Spring Boot is an open source Java-based framework used to create a micro Service. It is developed by Pivotal Team and is used to build stand-alone and production ready spring applications.

Spring Boot has been built for Rapid Application Development. The goal of Spring Boot is to provide a way to create Java applications quickly and simply, through an embedded server. By default, it used an embedded version of Tomcat and hence eliminating the need of Java EE containers.

It is a framework to ease the bootstrapping and development of new Spring Applications. Bootstrapping with defaults included in the configuration/ jar-dependencies. Easy to create standalone applications with embedded Tomcat/Jetty/Undertow. It provides defaults for code and annotation configuration to quick start new spring projects within no time. Plenty of Spring Boot Starter to quickly get up and running.

No code generation and no requirement for XML configuration. It reduces lots of development time and increases productivity.

React JS –

React is a JavaScript library for building user interfaces. It has transformed the way we think about front-end development. React.js has clasped the engagement of the open-source community. And its demand is irreversible in the coming future. It is here to stay.

Improved performance: React uses Virtual DOM, thereby creating web applications faster. Virtual DOM compares the components' previous states and updates only the items in the Real DOM that were changed, instead of updating all of the components again, as conventional web applications do.

MySQL –

MySQL is an open-source relational database management system (RDBMS). A list of commonly used MySQL queries to create database, use database, create table, insert record, update record, delete record, select record, truncate table and drop table etc. MySQL is a relational database management system based on SQL – Structured Query Language.

The most common use for MySQL, however, is for the purpose of a web database. It can be used to store anything from a single record of information to an entire inventory of available products for an online store.

3.3 User Classes

There is two type of users which can access this website. One is Organization and the second one is ADMIN. the admin will manage the records of student and organization. And organization can verify their data. To compare the data which we get from organizations we have a student repository (Database) on our server.

4. Requirements

4.1 Functional Requirements

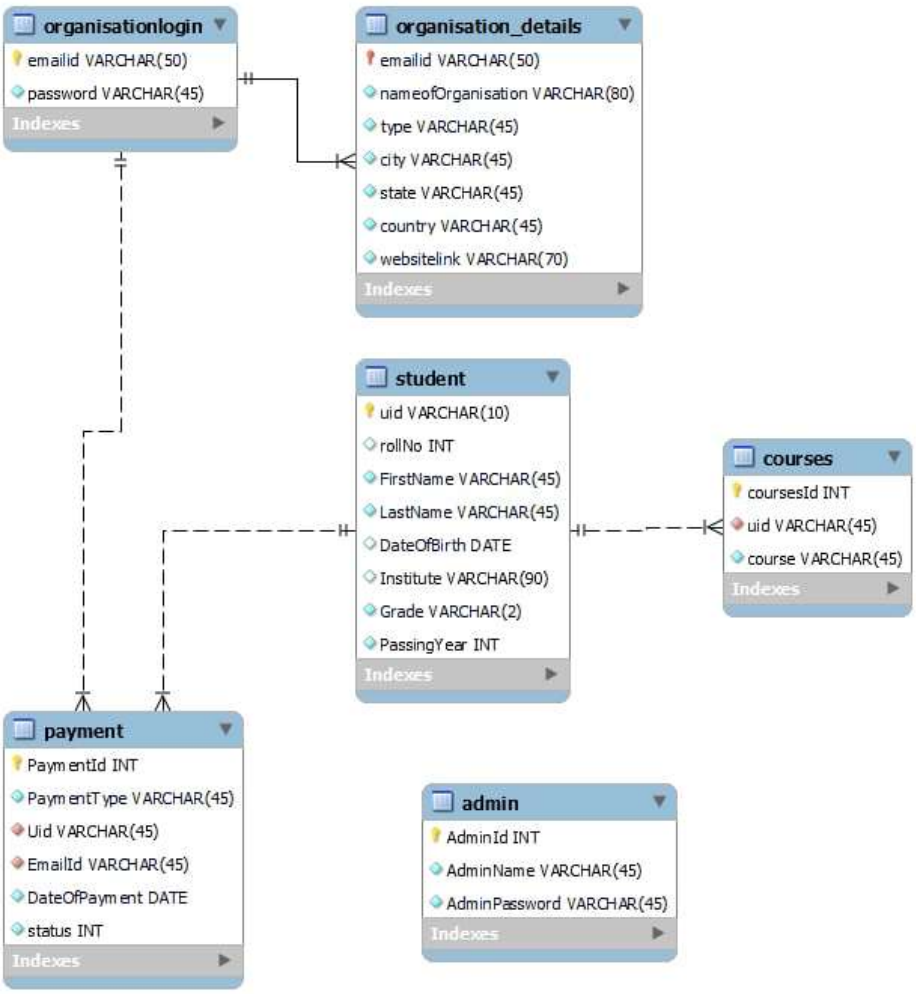
- The client machine must have physical support to JVM.
- A Client must have a working machine through which the website can be accessed.
- The machine must have access to the internet.

4.2 UI Requirements

- The client browser must support HTML 5 and below.
- It must also support ECMAScript version 6 and below, so that JavaScript based code can be run on the browser.
- A client must have a browser installed that passes the above parameters, to get access of the website.

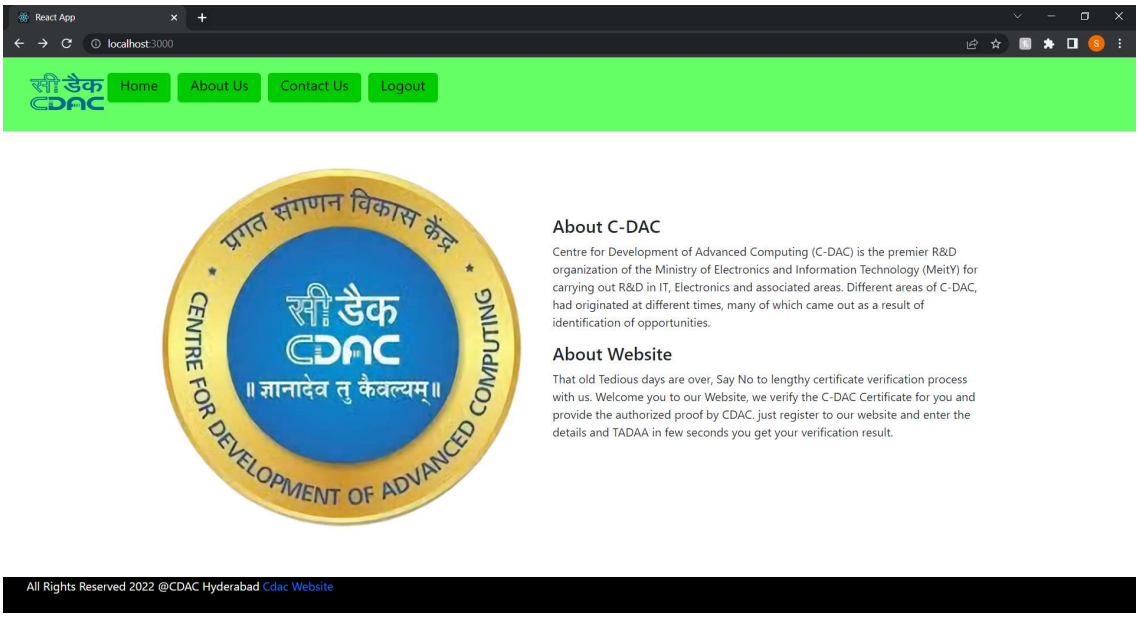
5. Design

- ER Diagram:

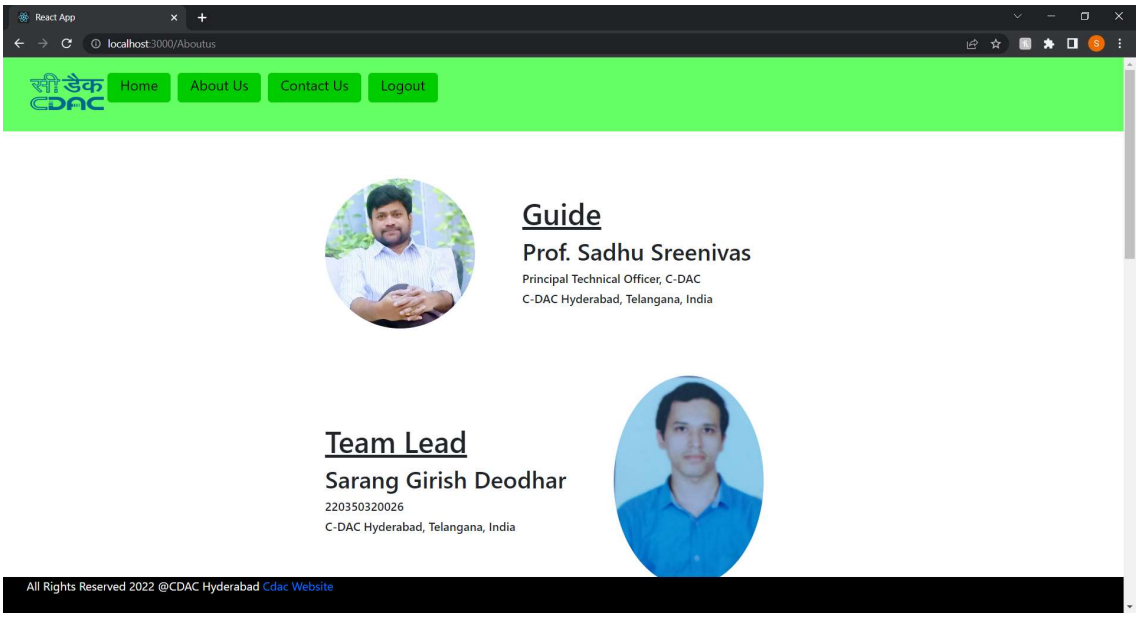


5.1 UI Design

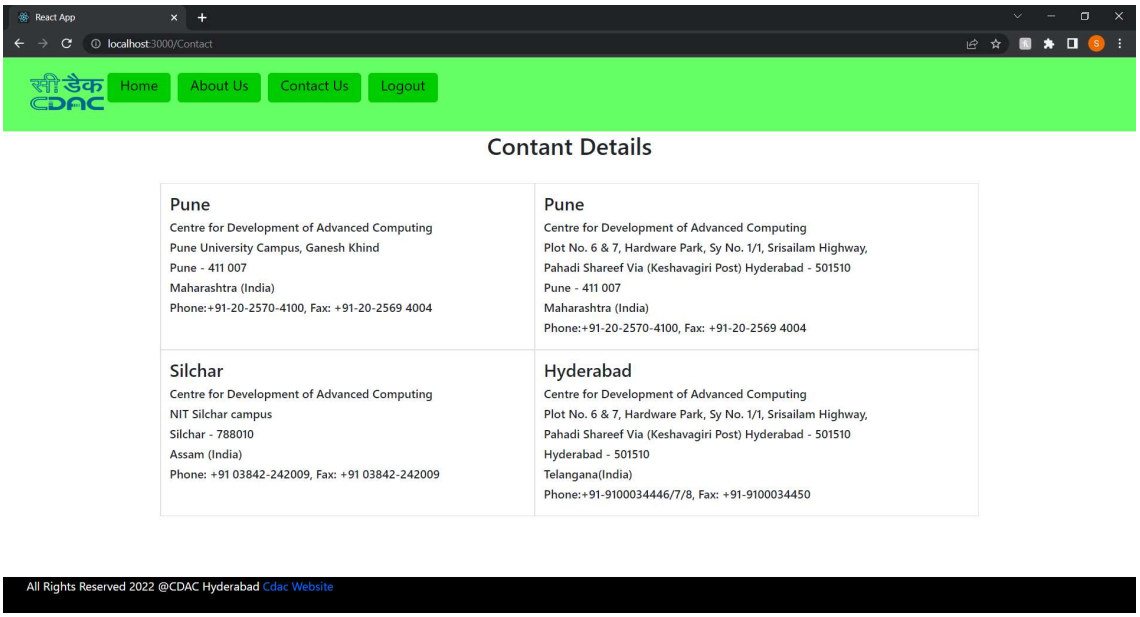
- Home page:



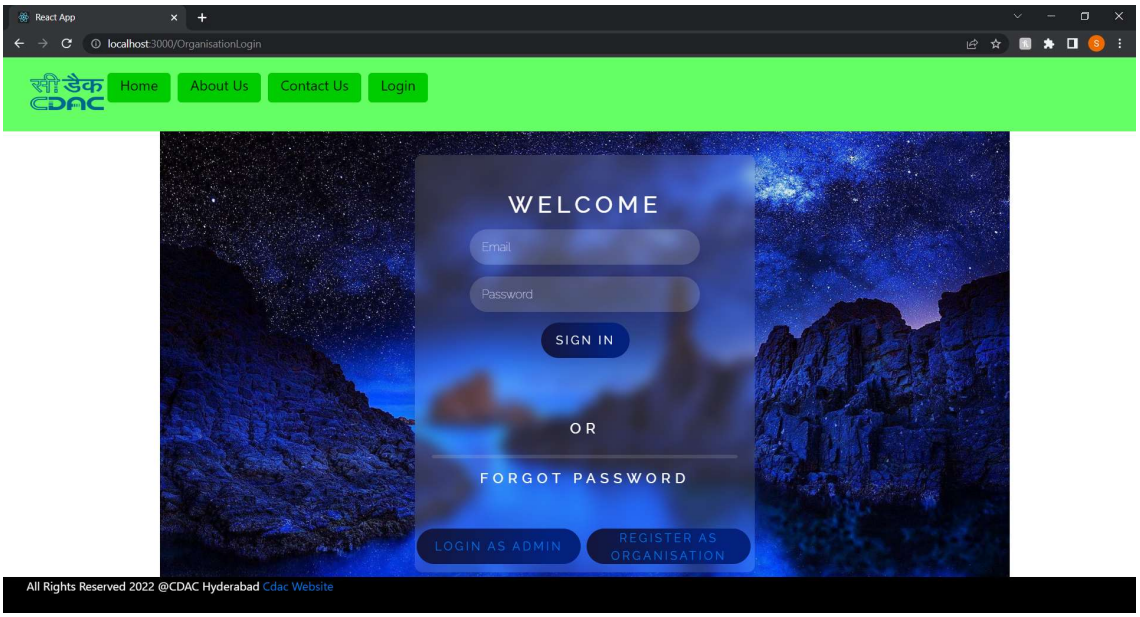
- About Us



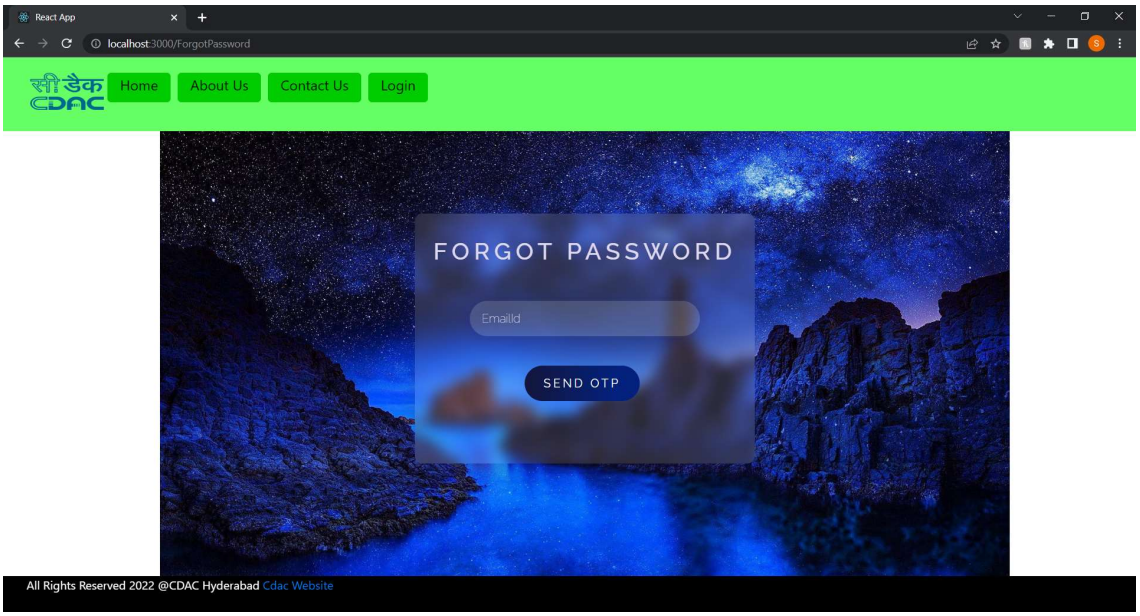
- Contact



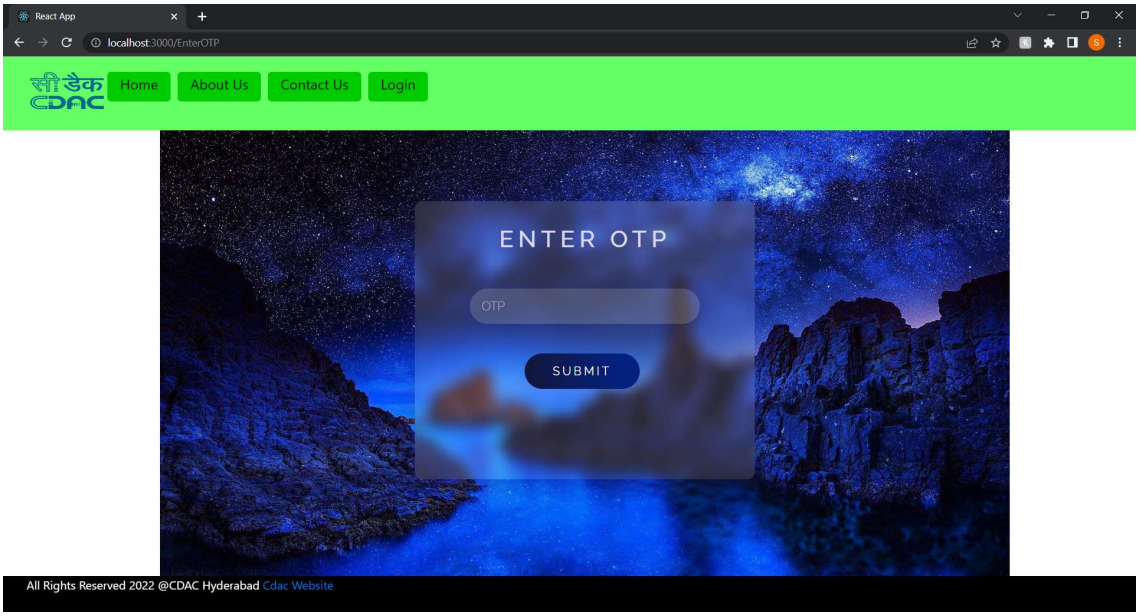
- Login As Organization :



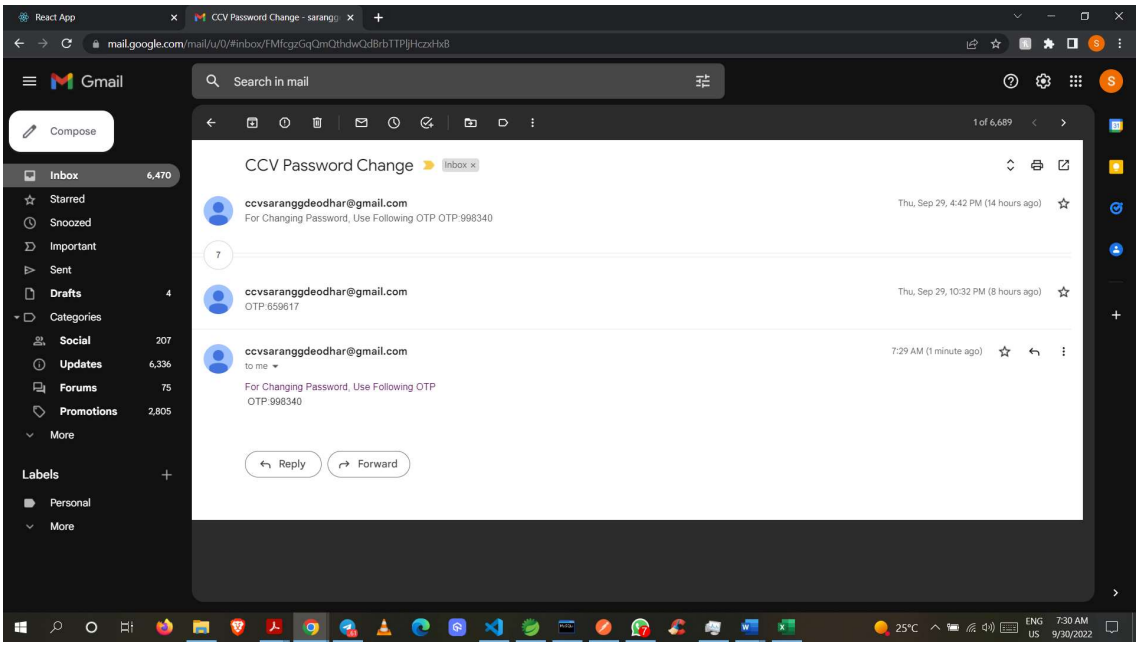
- Organization forgot password



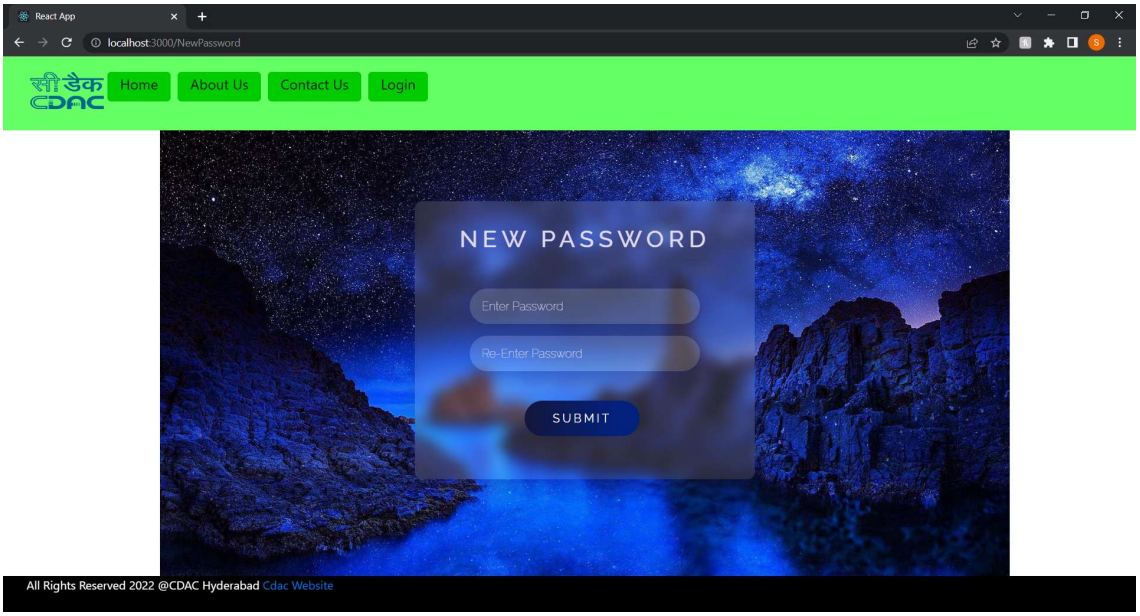
- OTP Screen



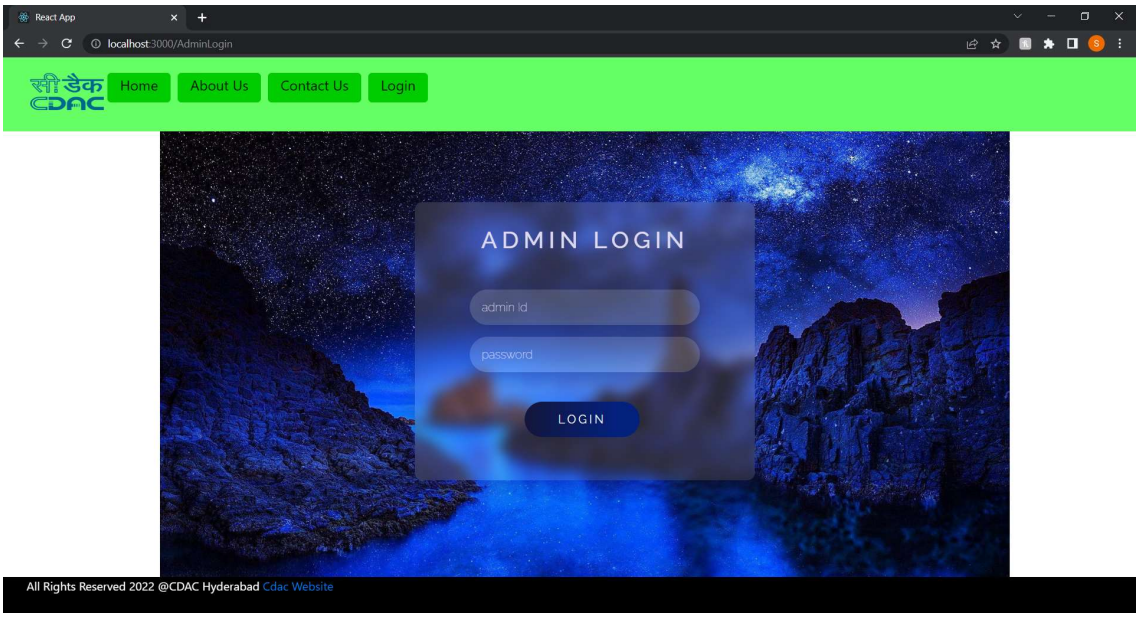
- OTP Through Mail



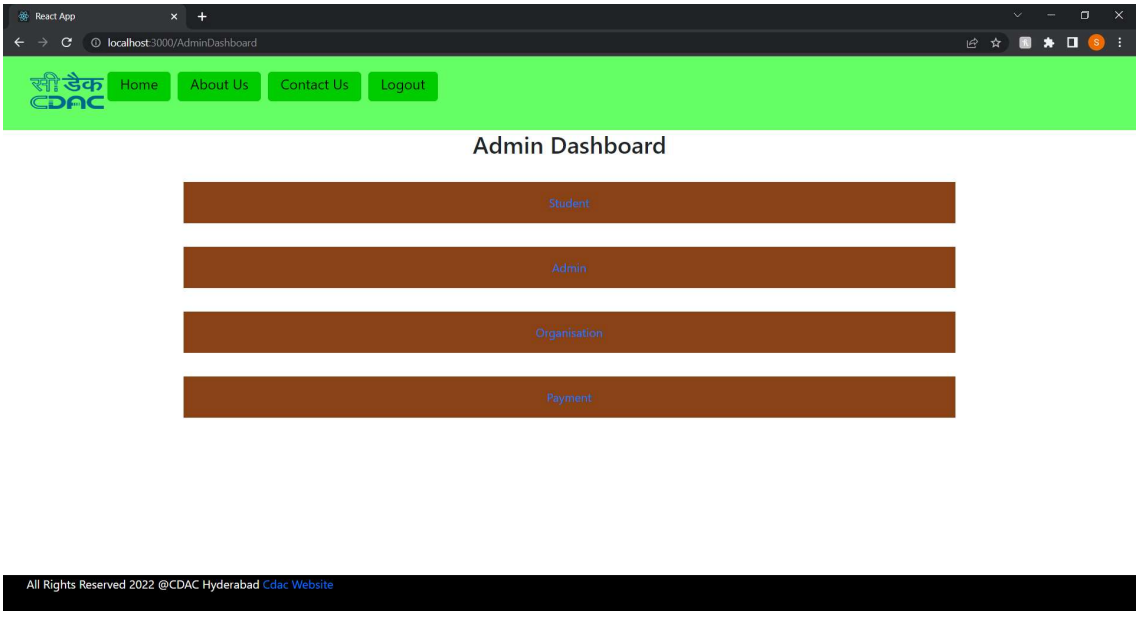
- Change Password



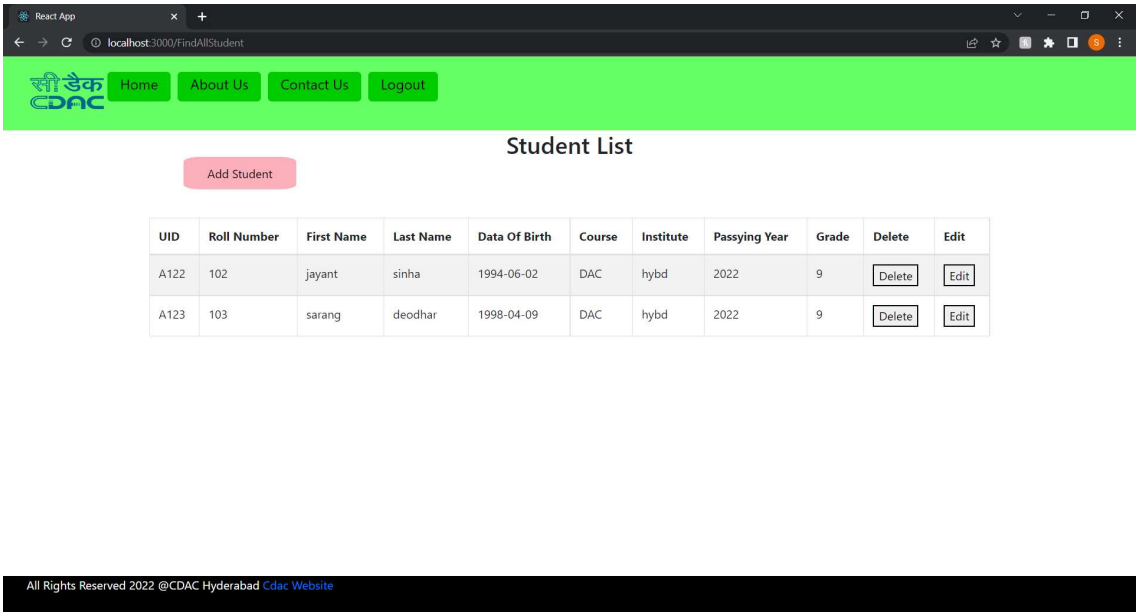
- Login As Admin



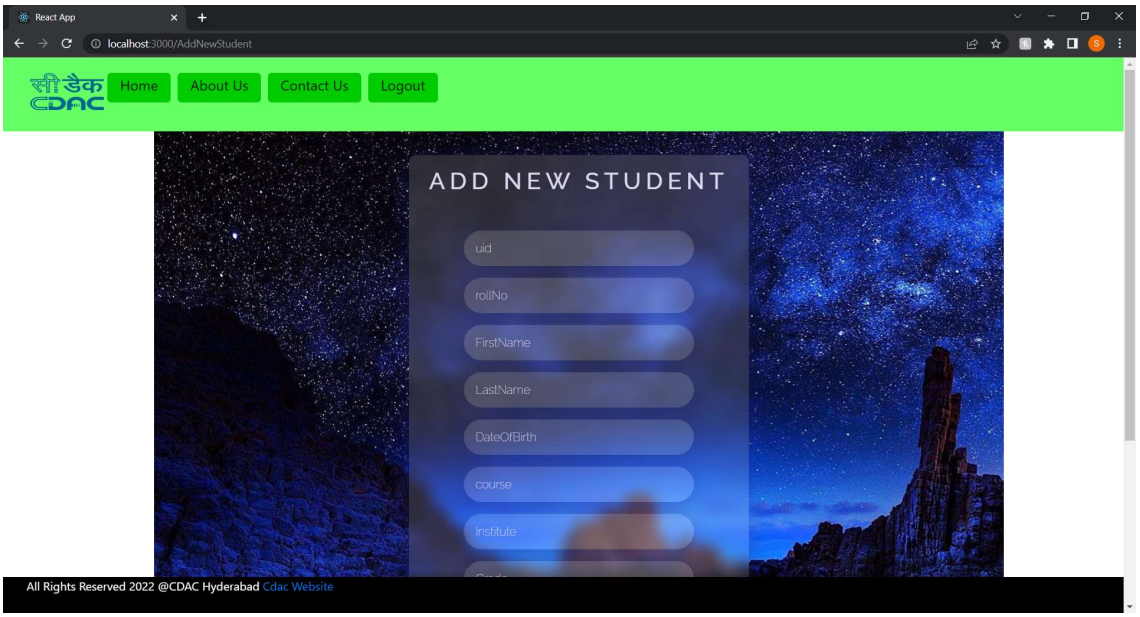
- Admin Dashboard



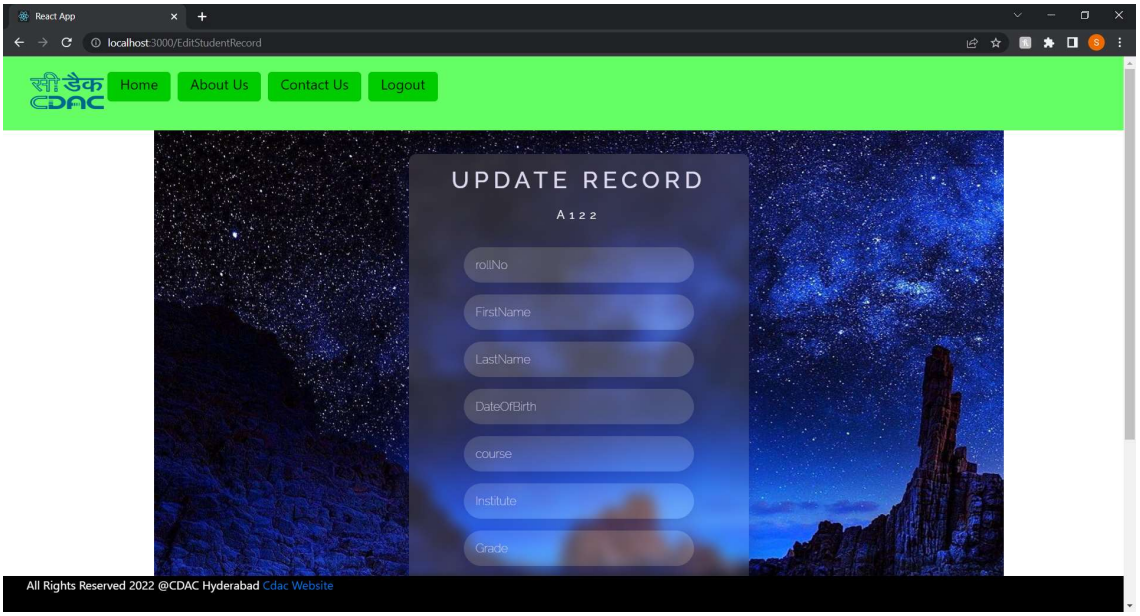
- Admin : Student List



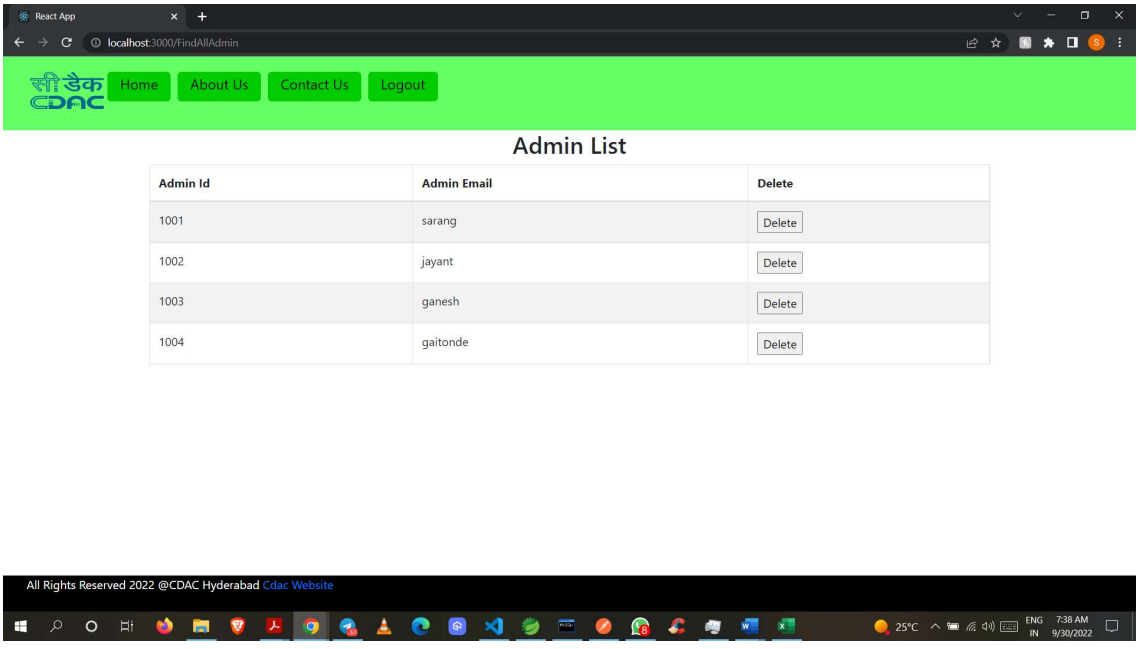
- Admin : Add Student Record



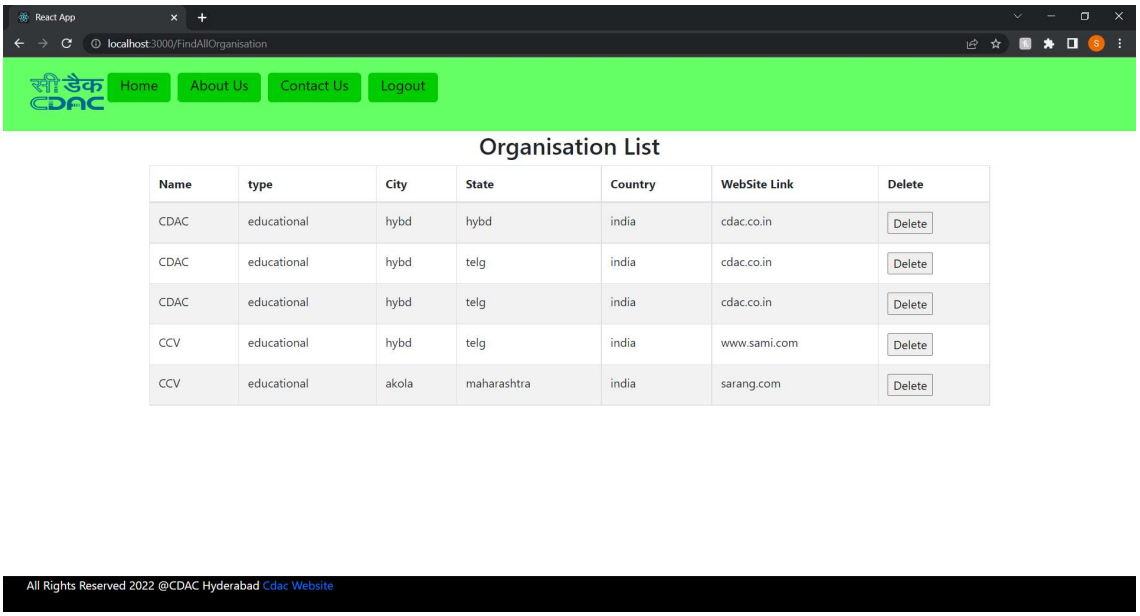
- Admin : update Student Details



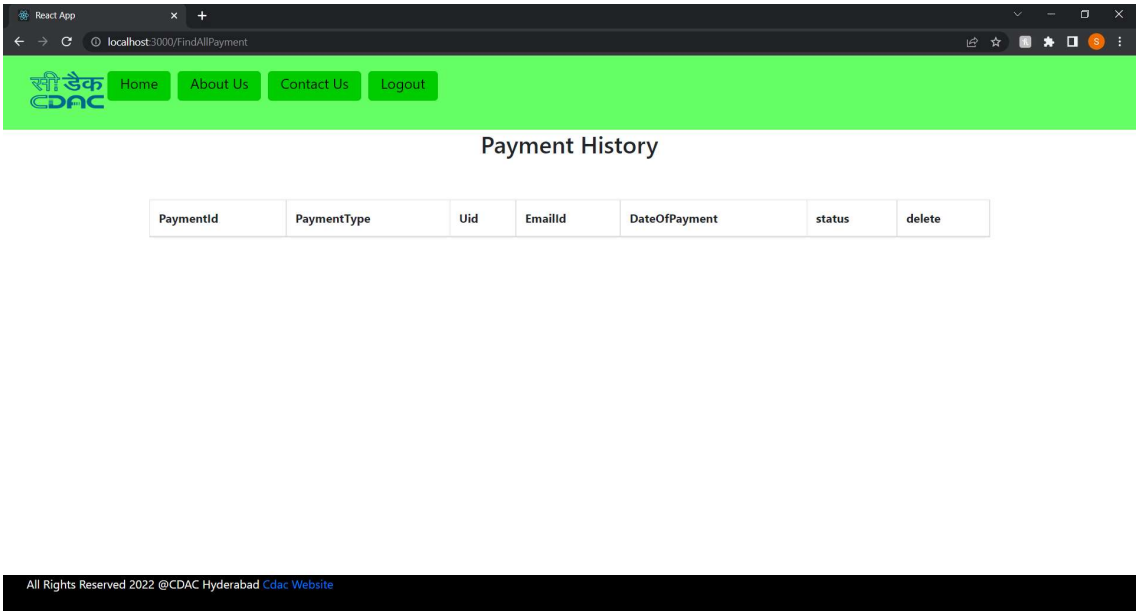
- Admin : Admin List



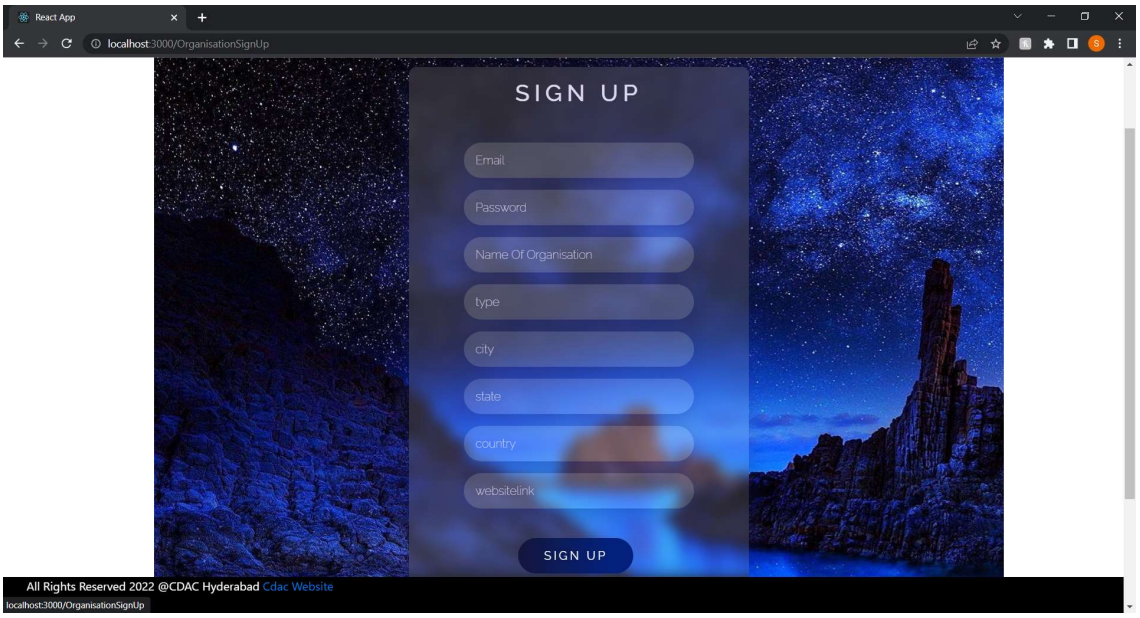
- Admin : Organisation List



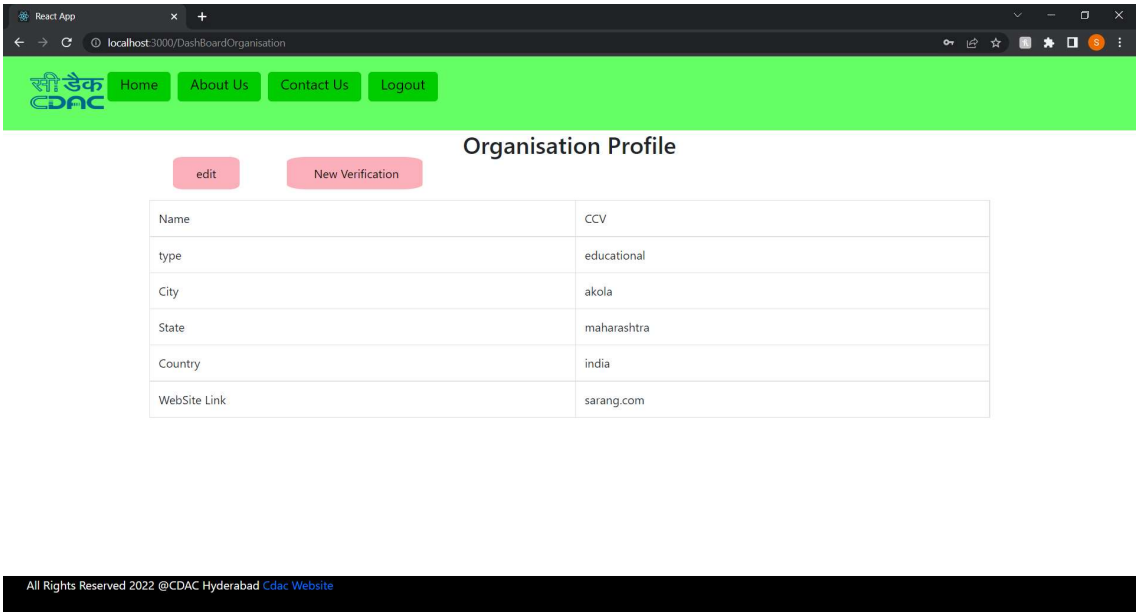
- Admin : Payment History



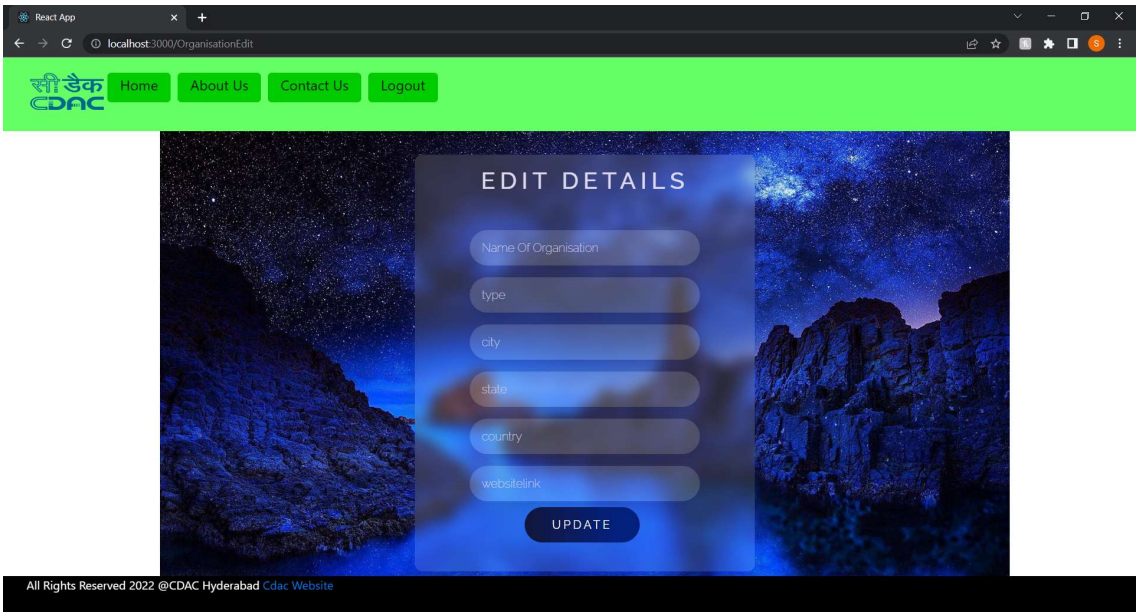
- Organisation Registration Page



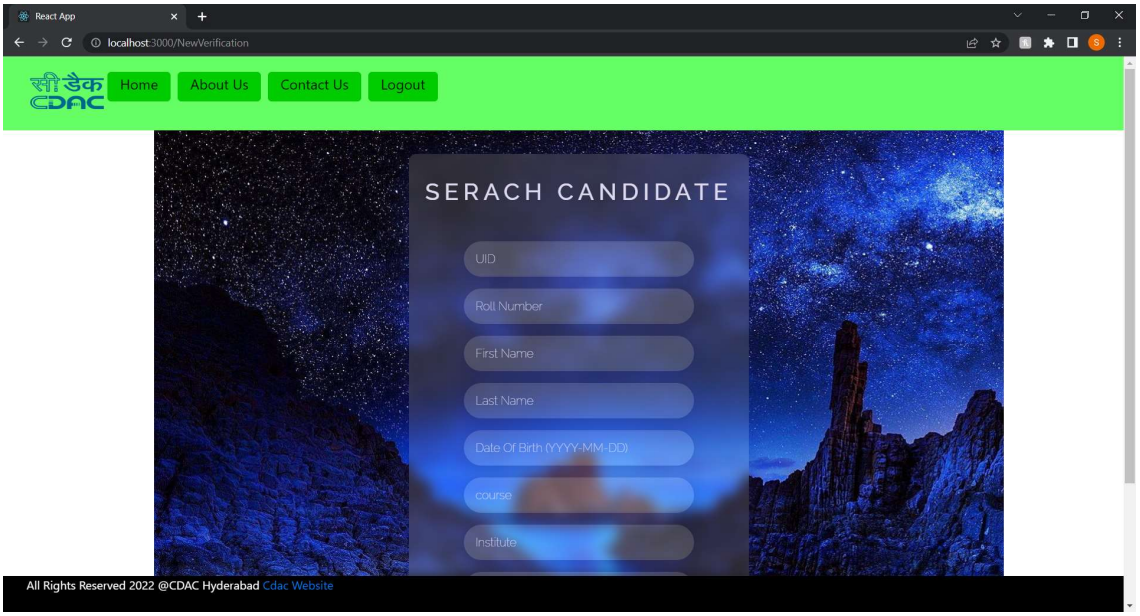
- Organization : Dashboard



- Organization : Edit Own Details



- New Certificate Verification Request



6. Functionalities

6.1 Back-end

- Firstly, backend connects with the database using MySQL/JDBC drivers.

- Secondly backend communicates with the Database using Entity classes, Data Transfer Objects(DTOs) and DAO Repository.

- It provides RESTful Services using APIs and their Implementation. The service implementation is responsible for performing the logics that does CRUD operation on database.

- The flow of data that comes from the front-end is managed by back-end controllers. The controller feed the data to respective service implementation so that further operation can be done.

- Backend controllers are also responsible for sending the response entity to the front-end.

- The back-end also consist configuration files required for the proper functioning of the application.

6.2 Front-end

- First work of front-end is to provide UI to the end users, so that the user can interact with the website using their system browsers.

- The front-end takes data from users using input fields and other input methods and send the request/request body to back-end in

key value paired URL variables or with the request body JSON format.

- The mapped request and path of backend controller on which data is going to be sent is mentioned at the front-end. The data then hits the respective entry point controllers at the backend.

- The response which is sent by back-end is handled by he front-end and required page/data is displayed to the user.

- Front-end is also our first line of defenses from bad data; It provides front-end authentications and validation.

7. Testing and Report

The report of the testing is given here under.

Test Cases

Sr. No	Test Case Title	Description	Expected Outcome	Error Message	Result
1	Sign Up	Should not allow any control to be empty if not null	If validated Allow to go to home page	Validation Error, User Exists Error	Passed
2	Login Page Organization	Organization should be able to login after entering email and password	After successful login user to be directed to dashboard	Invalid Login	Passed
3	Login Page Admin	Organization should be able to login after entering adminId and password	After successful login user to be directed to dashboard	Invalid Login	Passed
4	Organization Dashboard	Organization can see its own data and option to edit it.	Organization Base page	No Error	Passed
5	Admin Dashboard	Admin options to add, delete and update student, admins and organisation records	Options to interact with each entity i.e., admin, student, organization, payment	No Error	Passed
6	LogOut Admin	Admin should be able to logout from the website and neither back space or direct url access could	Admin will logout and routed back to the login page	No Error	Passed

		let the admin see the data.			
7	LogOut Organisation	Organisation should be able to logout from the website and neither back space or direct url access could let the Organisation see the data.	Organisation will logout and routed back to the login page	No Error	Passed
8	Verification Page	can send the verification data to the server and get respond accordingly	If data found or not found an alert should be pop up that show the details are found or not.	No Error	passed

8. Final Review

All the main functionalities of the project were implemented and the project's final outcome were excellent, though the application can be improved in many functionalities and many features can be added, it serves its basic purpose.

There were many challenges while making this project and we overcame a lot of hurdles, improvised a lot of things to make this outcome successful. We deeply feel that there is a lot of room for improvements and many changes to the project can be done to upscale it.

9. Future Scope

- Improve User Interface.
- Add PDF output for organization results.
- Add Search Functionalities more flexible.
- Add cryptographic security like JWT for login.
- Allow more interactive dashboard.
- Feature to add images for the organization and the students.

10. References

- https://developer.mozilla.org/
- https://www.w3schools.com/
- https://stackoverflow.com/
- https://www.baeldung.com/
- https://www.geeksforgeeks.org/
- https://www.javatpoint.com/
- https://www.knowledgehut.com/
- https://www.freecodecamp.org/