Project Report

On

PROJECT MANAGEMENT SYSTEM

Submitted by

K Karthik, S Abdulla, B Prakash, J Ramcharan

Under the guidance of

Ms C.Suneetha

Assistant Professor

Department of Computer Science and Engineering

Rajiv Gandhi University of Knowledge and Technologies R.K.Valley, Kadapa

Andhrapradesh

Rajiv Gandhi University of Knowledge Technologies
RK Valley, Kadapa (Dist), Andhra Pradesh, 516330

DECLARATION

We hereby declare that the report of the B.Tech Mini Project Work entitled "**PROJECT MANAGEMENT SYSTEM**" which is being submitted to Rajiv Gandhi University of Knowledge Technologies, RK Valley, in partial fulfillment of the requirements for the award of Degree of Bachelor of Technology in Computer Science and Engineering, is a bonafide report of the work carried out by me. The material contained in this report has not been submitted to any university or institution for award of any degree.

K Karthik – R170978

S Abdulla – R170518

B Prakash – R170046

J Ramcharan – R170927

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

RGUKT

(A.P.Government Act 18 of 2008)
RGUKT, RK VALLEY
Department of Computer Science and Engineering

CERTIFICATE FOR PROJECT COMPLETION

This is certify that the project entitled "**PROJECT MANAGEMENT SYSTEM**" submitted by **K Karthik (R170978),S Abdulla (R170518),B Prakash (R170046),J Ramcharan (R170927)** under our guidance and supervision for the partial fulfillment for the degree Bachelor of Technology in Computer Science and Engineering during the academic year 2021-2022 at RGUKT, RK VALLEY. To the best of my knowledge, the results embodied in this dissertation work have not been submitted to any University or Institute for the award of any degree or diploma.

Project Internal Guide

Ms C.Suneetha

Project Coordinator,

Assistant Professor,
RGUKT, RK Valley.

Head of the Department

Mr.P.Harinadha

HOD OF CSE,
RGUKT, RKValley.

Acknowledgement

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose constant guidance and encouragement crown all the efforts success. I am extremely grateful to our respected Director,Prof. K. SANDHYA RANI for fostering an excellent academic climate in our institution. I also express my sincere gratitude to our respected Head of the Department Mr P.HARINADHA for their encouragement, overall guidance in viewing this project a good asset and effort in bringing out this project. I would like to convey thanks to our guide at college Ms.C.SUNEETHA for their guidance, encouragement, co-operation and kindness during the entire duration of the course and academics. My sincere thanks to all the members who helped me directly and indirectly in the completion of project work. I express my profound gratitude to all our friends and family members for their encouragement.

Index

Abstract

In the existing Manual System, In which Every Student has to consult project mentor and ask for vacancy. If no vacancy available at that particular faculty ,they have to consult another faculty .Which is highly time taking for both sides.

Online Project Management System is a web based application which can be accessed all over the organization .This application provides easy online registration facility for Students .Faculty can also approve or reject Students registration very easily by reviewing their profile and interest at the same place of registration.

Each Student is provided with unique ID and password for log in to system and send request for project mentor registration .Faculty will look after the registration requests and based vacancies and Student profile ,they will approve or reject the student request.

INTRODUCTION

- The main objective of the proposed system is to decrease the paper work and easier record maintenance and registration process by having a particular website for Project Management.

- This Project Management System reduces the entire process revolved around In manual project registration and maintenance . This will save time for both Faculty and Students.

- This System eases the registration process of students by providing them all needed information about available faculty like their vacancies , Area of Interest etc.

- This System reflects the registration request of student to relevant Faculty for approval , based on vacancies ,

Student Profile and Area of Interest Faculty can either approve or reject the request .

- The approval response will be shown to the Student.

Purpose

The purpose of this Project Management System is to change Already Existing Manual Project Registration and maintaining System to online Project Management System to reduce Paper Work, and easier the registration process, to avoid the loss of data and Making the Project Management easy unlike Traditional Manual Project Management System.

Product Vision :

The main Vision of this project is to make manual project registration and maintenance System to a Complete Online web application for ease of Usage.

- **Existing System :**

In existing System our college follows manual project registration procedure in which every student has to consult project mentor for knowing mentor Area of Interests and Vacancies.If there are no vacancies available under corresponding mentor , Student have to consult another mentor for project registration.

- **Proposed System**

In Proposed System , Student don’t necessarily need to consult project Mentor for project registration . Students can registered through the online provided website . This will reduce work for students by providing easy availability and online registration feature. It eases the mentor approval process also.

Requirement Specification

Hardware Configuration:

Client side:

Ram	512 MB
Hard disk	10GB
Processor	1.0 GHz

Server side:

Ram	1GB
Hard disk	20GB
Processor	2.0GHz

Software Requirements:

Front end	Html,Css,javascript
Server side Language	Node JS
Database Server	Mongo db
Web Browser	Firefox,Chrome or any compatible Browser

Operating System	Ubuntu,Windows or any Compatible Browser
Software	EPASS

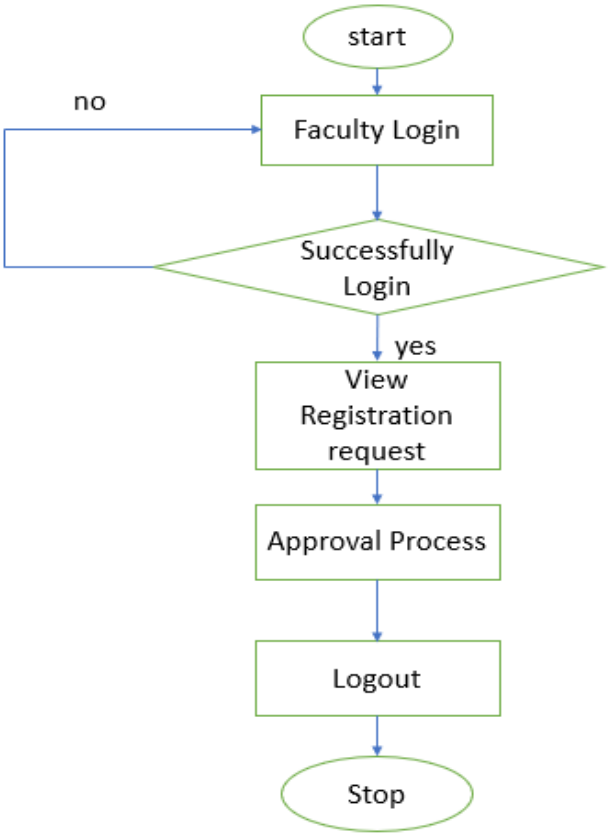
MODULES

- Project Guide
- Student
- HOD
- Project Coardinator

Modules Description

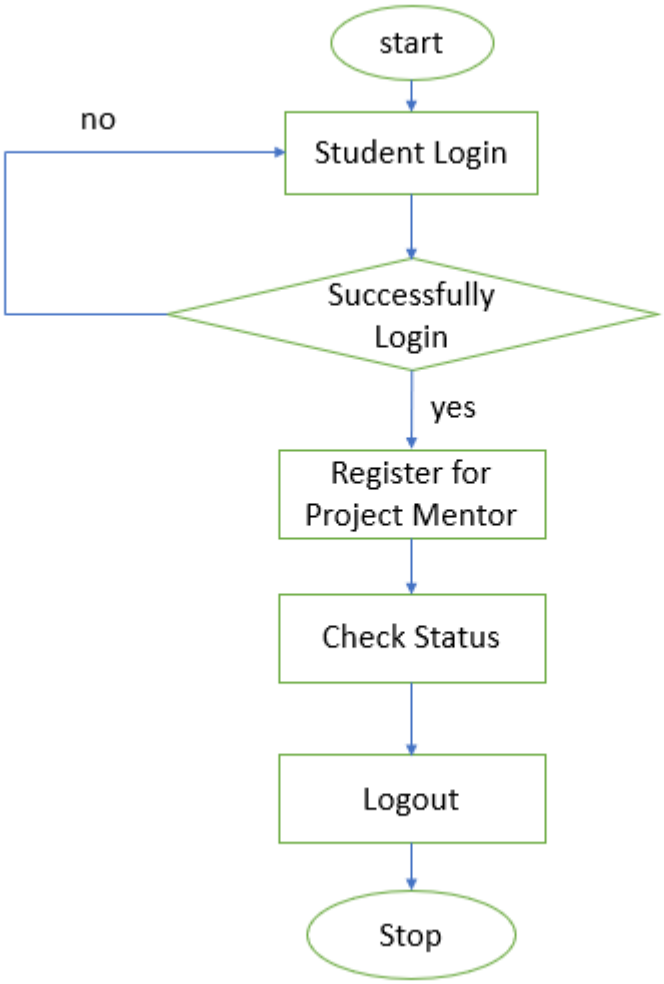
Project Guide

Project Mentor will have the permissions to look after data of every student of the Branch. Project Mentor can view the student details. Project Mentor can approve registration through this application and he can view Area of interest in Project of every individual student. Project Management System application accepted/rejected details are available to the Students through the application.


start
no
Faculty Login
Successfully Login
yes
View Registration request
Approval Process
Logout
Stop

Student Module

Each student will have unique id for login into the system, login username and password is valid means he/she continue to view all the details of project Mentors. The students can apply for their project mentor thorgh registration process. This record will automatically send to the Project Mentors. Project Mentors will view all the records and send the confirmation to the particular student through application . and finally Students can view the approval status in their respective login in this site.



- **HOD** :

This module is very important module in this project. The main aim of this module is to coordinate the faculties for different kind of projects based on available domains.This module has access for adding or deleting the faculty profiles from the faculty database.The module can able to view and control the access of all users.

- **Project Coordinator:**

In this module all the students can view the registration approval details and their Status like accepted, rejected or pending.

Advantages of Proposed Methodology

- The Online Project Management System will Reduce the Paper work.
- The Online Project Management System will reduce the time.
- It makes the Maintenance easy.
- Project registration for students and approval for Mentors will be easy because of the online website availability and easy steps involved.
- Data loss won't happen, as the Data will be stored in the Data Bases.
- User friendly Web Application.

Technologies Used

- FRONTEND : HTML ,CSS , JavaScript , Boot Strap, React JS .
- BACKEND : NODEJS
- DATA BASE: MONGO DB
- ENVIRONMENT : Visual Studio Code.

MongoDB

MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas. MongoDB is developed by MongoDB Inc. and licensed under the Server Side Public License which is deemed non-free by several distributions.

Node Js

Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on a JavaScript Engine and executes JavaScript code outside a web browser, which was designed to build scalable network applications.

HTML

HTML, or Hypertext Markup Language, is a markup language for the web that defines the structure of web pages.

Hypertext: text (often with embeds such as images, too) that is organized in order to connect related items
Markup: a style guide for typesetting anything to be printed in hardcopy or soft copy format
Language: a language that a computer system understands and uses to interpret commands.

CSS

Cascading Style Sheets (**CSS**) is a Style sheet Language used for describing the Presentation of a document written in a markup language such as HTML or XML (including XML dialects such as SVG, MathMl or Xhtml). CSS is a cornerstone technology of the World wide web, alongside HTML and Javascript(js).

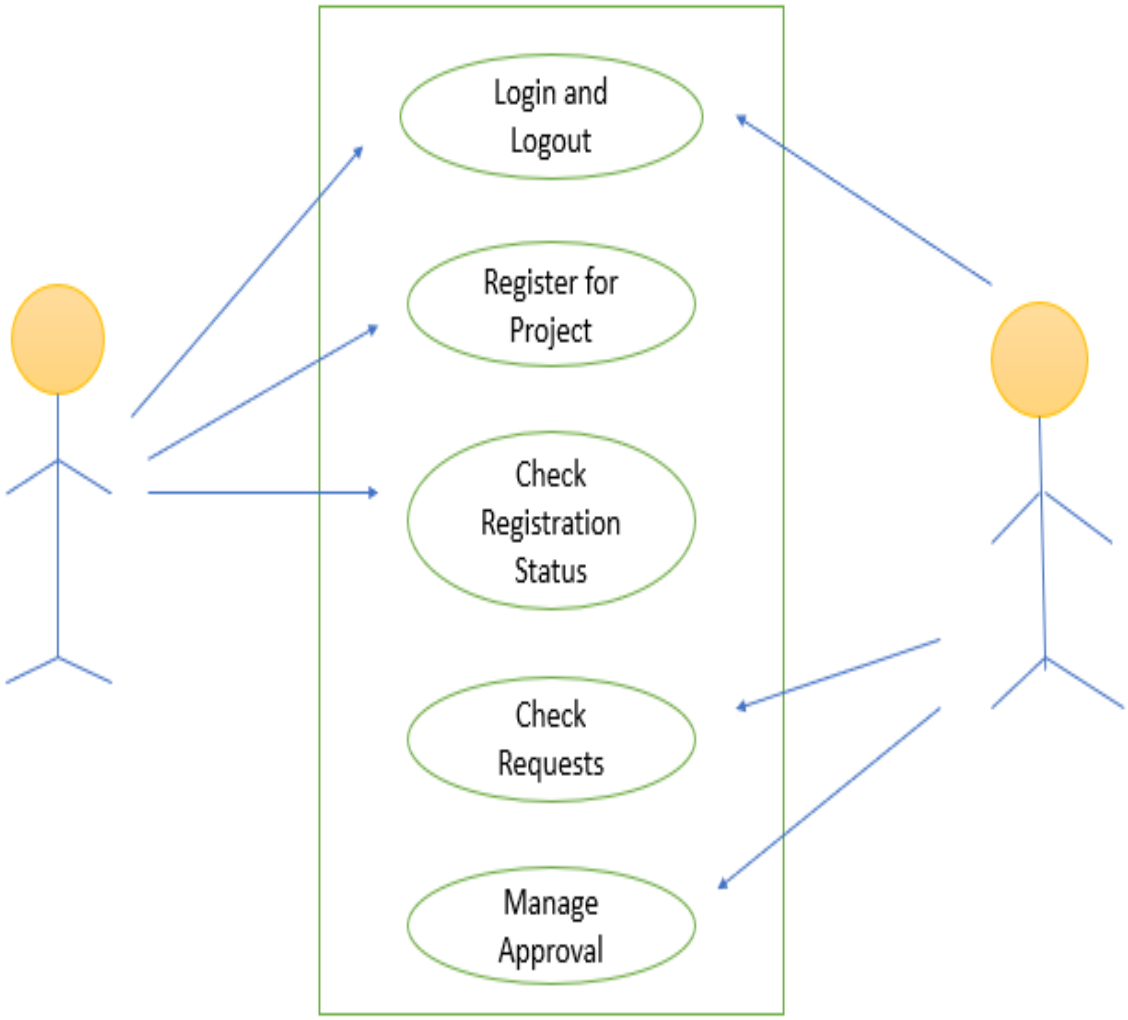
Javascript

JavaScript is **a dynamic programming language that's used for web development, in web applications, for game development, and lots more**. It allows you to implement dynamic features on web pages that cannot be done with only HTML and CSS.

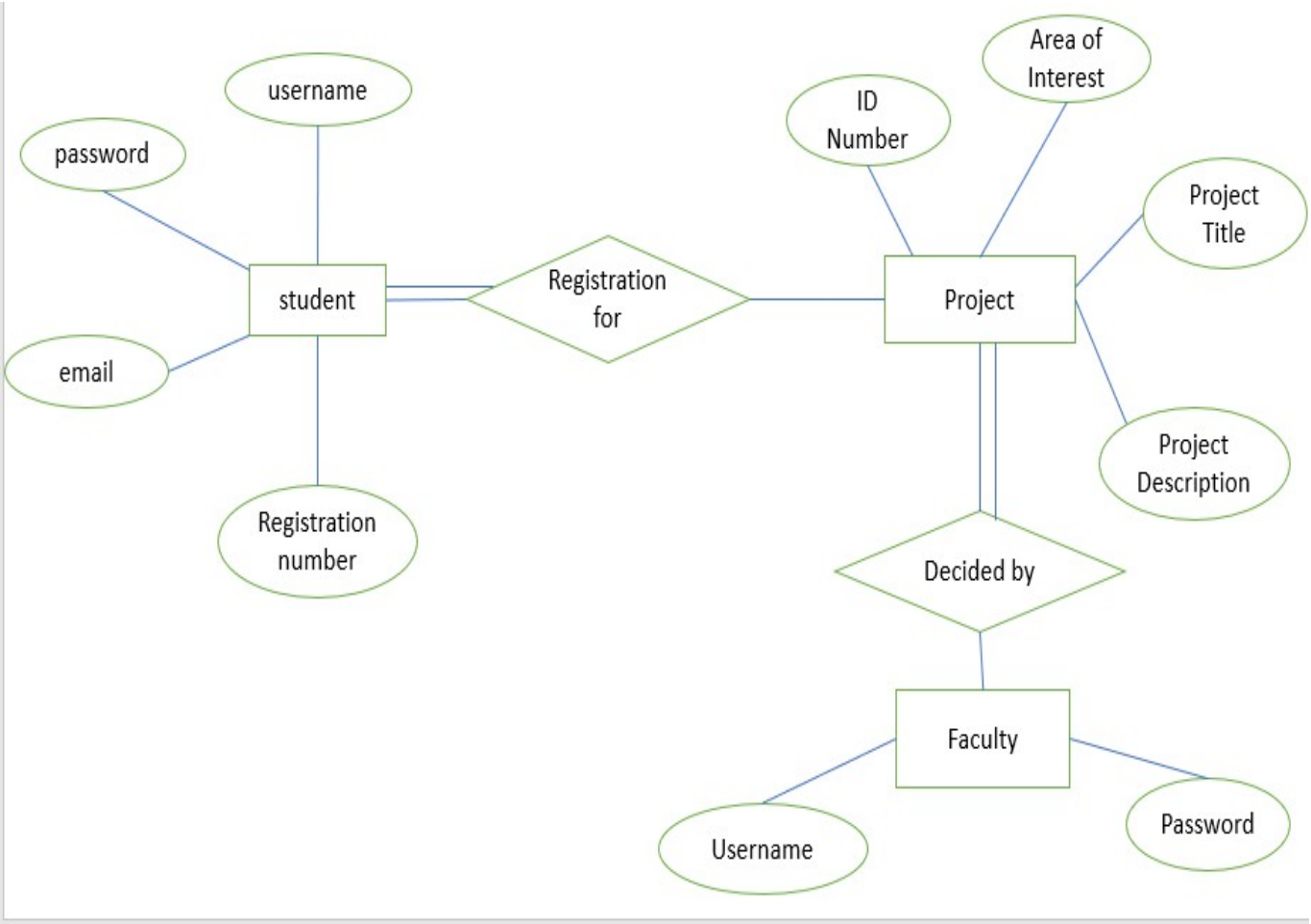
React Js

ReactJS is a declarative, efficient, and flexible JavaScript library for building reusable UI components. It is an open-source, component-based front end library responsible only for the view layer of the application. It was created by **Jordan Walke,** who was a software engineer at **Facebook.** It was initially developed and maintained by Facebook and was later used in its products like **WhatsApp** & **Instagram.**

USE CASE DIAGRAM



ER Diagram



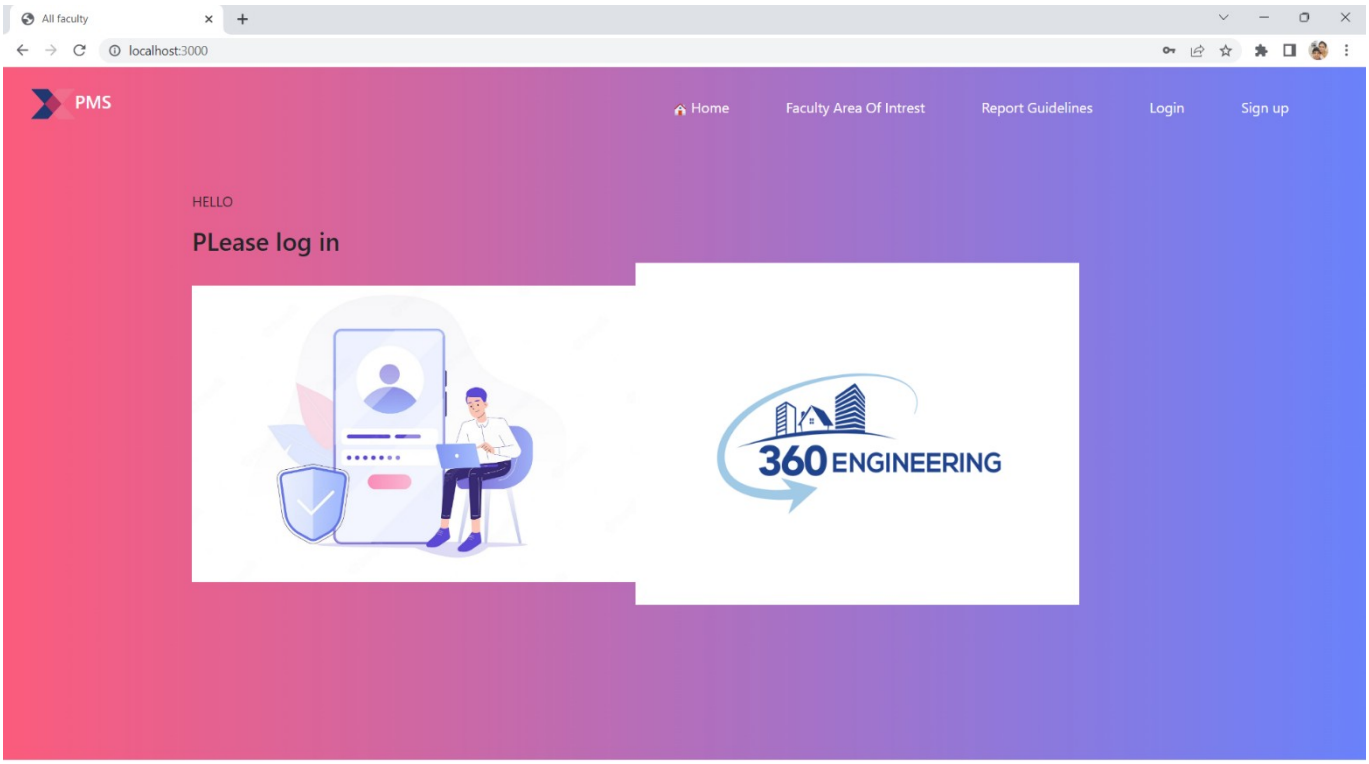
SOURCE CODE

Home.html

```
import React,{useState,useEffect} from 'react';
import login from '../images/login.jpg'
import engg from '../images/engg.png'
import welcome from '../images/welcome.png'
import './home.css'

const Home = () => {

  const [products,setProducts] = useState([]);
  const [userName,setUserName] = useState('');
  const [show ,setShow] = useState(false);
  const userHomePage = async () =>{
    try{
      const res =await fetch('/getdata',{
          method:"GET",
          headers:{
            "Content-Type":"application/json"
          },
      });
      const data = await res.json();
      console.log(data);
      setUserName(data.name);
      setShow(true);

      if(!res.status===200){
        const error = new Error (res.error);
        throw error;
      }
    }catch(err){
      console.log(err);


    }
  }
useEffect(() => {
  userHomePage();

}, [])



  return (
```

```
    <>
    {/* <Homeproducts /> */}
    <div className='background'>
    <p className='pt-5'>HELLO</p>
    <h3>{show ? `Welcome to MINIPROJECT ${userName}`:'PLease log in '}</h3>
    {
      !show &&
      <>
      <img width="500px"  src={login}/>
      <img width="500px"  src={engg}/>
      </>
    }
    {
      show && <img src={welcome} />
    }
    </div>
    </>
  )
}

export default Home
```

OUTPUT



Login.html

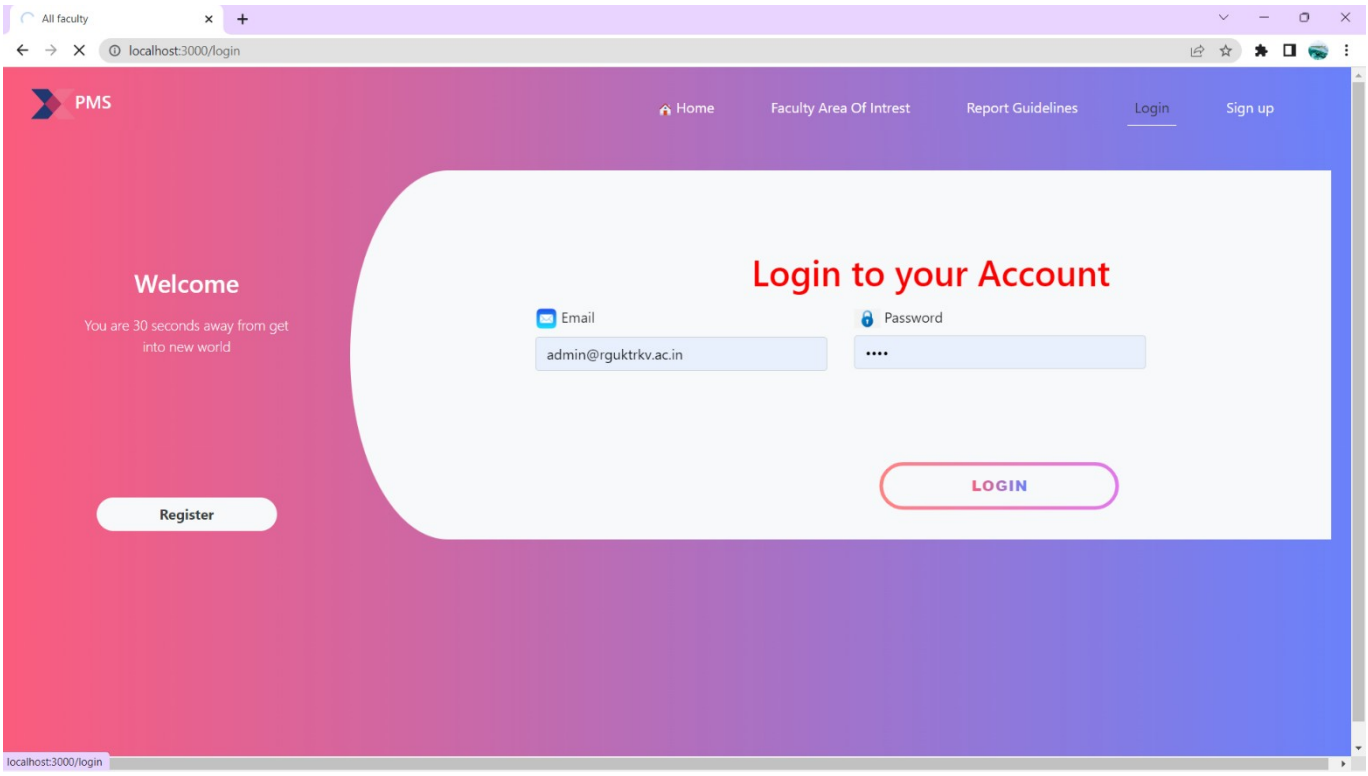
```
import { useContext } from "react";
import React,{  useState } from "react";
import { Link, useHistory } from "react-router-dom";
import { userContext } from "../../App";
import slido from '../images/slido.png';
import mail from '../images/mail.webp';
import password1 from '../images/password1.png'
import './login.scss'
const Login = () => {
  const { state, dispatch } = useContext(userContext);
  const history = useHistory();
  const [email, setEmail] = useState("");
  const [password, setPassword] = useState("");
  const loginUser = async (e) => {
    e.preventDefault();
```

```
  const res = await fetch("/signin", {
    method: "POST",
    headers: {
      "Content-Type": "application/json",
    },
    body: JSON.stringify({
      email,
      password,
    }),
  });
  const data = await res.json();
  if (res.status === 400 || !data) {
    window.alert("Invalid Credentials");
  } else {
    dispatch({ type: "USER", payload: true });
    history.push("/");
  }
};

return (
  <>

    <div class="user-ragistration main">
      <div class="main register">
        <div class="row span">
          <div class="col-md-3 register-left">
            <img src={slido} alt="" />
            <h3>Welcome</h3>
            <p>You are 30 seconds away from get into new world</p>
            <Link to="/register">
              <input type="submit" name="" value="Register" />{" "}
            </Link>
          </div>
          <div class="col-md-9 register-right">
            <div class="tab-content" id="myTabContent">
              <div
                class="tab-pane fade show active"
                id="home"
                role="tabpanel"
                aria-labelledby="home-tab"
              >
              <form method="POST">
              <h1 className="heading">Login to your Account</h1>
                <div class="row register-form">
                  <div class="col-md-6 cont2">
```

```
                    <div class="form-group">
                    <label> <img src={mail} style={{width:'25px'}} /> Email
</label>
                      <input type="email" className="form-control" id="email"
name="email" autoComplete="off" value={email}
           onChange={(e)=>setEmail(e.target.value)} placeholder="Enter email"
aria-describedby="emailHelp" />
                      </div>
                    </div>
                    <div class="col-md-6 cont2">
                    <div class="form-group">
                    <label><img src={password1} style={{width:'30px'}} />
Password</label>
                      <input type="password" name="password" className="form-
control" id="password" value={password}
          onChange={(e)=>setPassword(e.target.value)} placeholder="Enter
password"/>
                      </div>
                      {/*button  */}
                      <div class="container1">
          <div class="btn1">
         <span type="submit"
                              name="signin"
                              id="signin"
                              value="login"
                              onClick={loginUser}
                              className="rebutton">
            <svg width="277" height="62">
              <defs>
                  <linearGradient id="grad1">
                      <stop offset="0%" stop-color="#FF8282"/>
                      <stop offset="100%" stop-color="#E178ED" />
                  </linearGradient>
              </defs>
              <rect x="5" y="5" rx="25" fill="none" stroke="url(#grad1)"
width="266" height="50"></rect>
            </svg>
              Login</span>
          </div>
          </div>
                    </div>
                  </div>
                </form>
              </div>
            </div>
```

```
            </div>
          </div>
        </div>
      </div>
    </>
  );
};

export default Login;
```

OUTPUT



regis.js

```
import React,{  useState } from "react";
import registerlogo from "../images/registerlogo.jpg";
import "./style.scss";
import slido from '../images/slido.png';
import mail from '../images/mail.webp'
import name1 from '../images/name1.png';
import profession from '../images/profession.jpg'
import Phone_icon from '../images/Phone_icon.png'
import password1 from '../images/password1.png'
import { Link, useHistory } from "react-router-dom";
const Register = () => {
  const history = useHistory();

  const [details, setDetails] = useState({
```

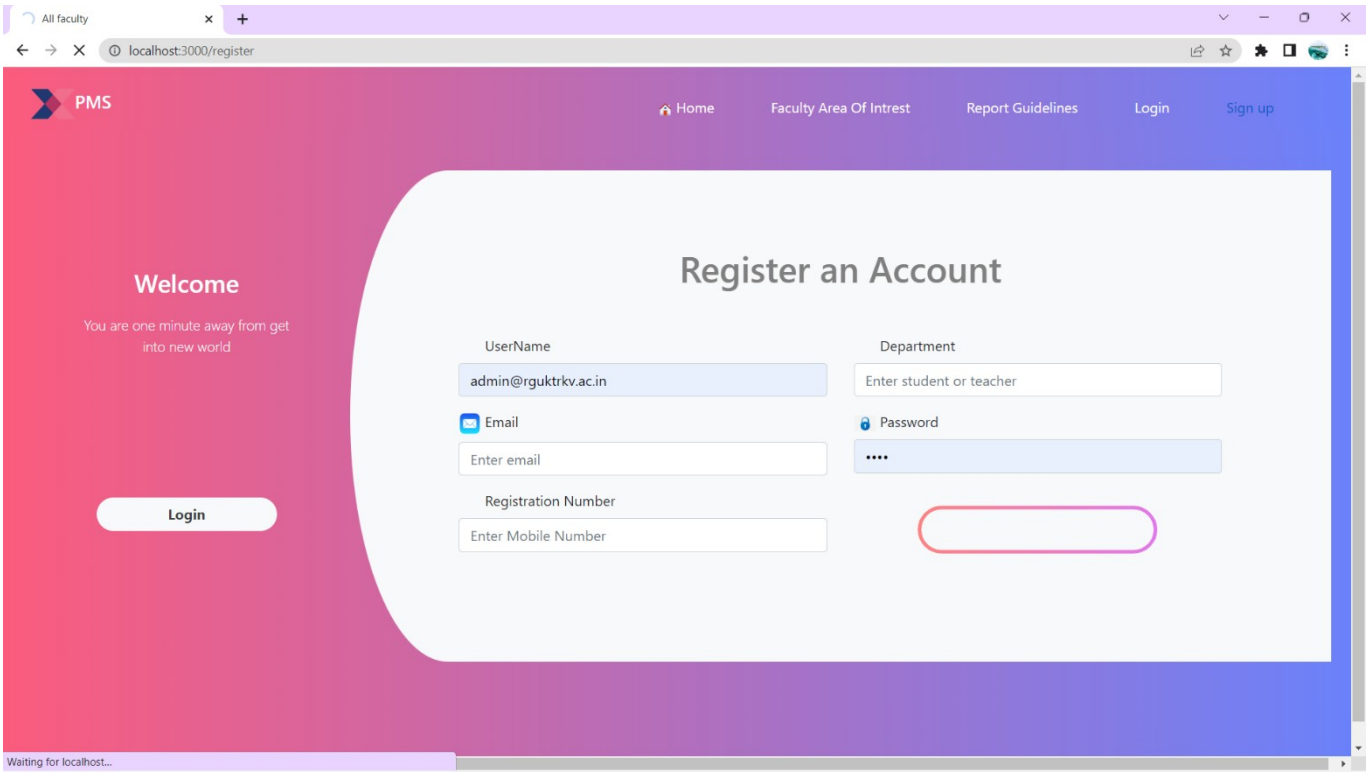
```
  name: "",
  email: "",
  regno: "",
  department: "",
  password: "",

});
// handle inputs\
let name, value;
const handleInputs = (e) => {
  console.log(e);
  name = e.target.name;
  value = e.target.value;

  setDetails({ ...details, [name]: value });
};

const PostData = async (e) => {
  e.preventDefault();

  const { name, email, regno, department, password } = details;

  const res = await fetch("/register", {
    method: "POST",
    headers: {
      "Content-Type": "application/json",
    },
    body: JSON.stringify({
      name,
      email,
      regno,
      department,
      password,

    }),
  });
  const data = await res.json();
  if (res.status === 422 || !data) {
    window.alert("Failed");
    console.log("Failed");
  } else {
    <div class="alert alert-success" role="alert">
      Sign in
    </div>;
    console.log("success");
```

```
      history.push("/login");
    }
  };


  return (


      <div class="user-ragistration main">
        <div class="main register">
          <div class="row span">
            <div class="col-md-3 register-left">
              <img src={slido} alt="" />
              <h3>Welcome</h3>
              <p>You are one minute away from get into new world</p>
              <Link to="/login">
                <input type="submit" name="" value="Login" />{" "}
              </Link>
              <br />
            </div>
            <div class="col-md-9 register-right">
              <div class="tab-content" id="myTabContent">
                <div
                  class="tab-pane fade show active"
                  id="home"
                  role="tabpanel"
                  aria-labelledby="home-tab"
                >
                  <h1 class="register-heading">Register an Account</h1>
                  <div class="row register-form">
                    <div class="col-md-6 cont1">
                      <div class="form-group">
                      <label> <img src={name1} style={{width:'25px'}} />
UserName </label>
                      <input type="text" className="form-control" id="name"
name="name" autoComplete="off"
                      value={details.name}
                      onChange={handleInputs}
                      placeholder="Enter Your Name" aria-describedby="emailHelp"
/>
                      </div>
                      <div class="form-group">
                      <label> <img src={mail} style={{width:'25px'}} /> Email
</label>
```

```
                    <input type="email" className="form-control" id="email"
name="email" autoComplete="off"
                    value={details.email}
                    onChange={handleInputs}
                    placeholder="Enter email " aria-describedby="emailHelp" />
                    </div>
                    <div class="form-group">
                    <label> <img src={Phone_icon} style={{width:'25px'}} />
Registration Number </label>
                    <input type="number" className="form-control" id="regno"
name="regno" autoComplete="off"
                    value={details.regno}
                    onChange={handleInputs}
                     placeholder="Enter Mobile Number " aria-
describedby="emailHelp" />
                    </div>

                    <div class="form-group">
                    </div>
                  </div>
                  <div class="col-md-6 cont1">


                  <div class="form-group">
                  <label> <img src={profession} style={{width:'25px'}} />
Department  </label>
                    <input type="text" className="form-control"
id="profession" name="department" autoComplete="off"
                    value={details.department}
                    onChange={handleInputs}
                    placeholder="Enter student or teacher" aria-
describedby="emailHelp" />
                    </div>

                    <div class="form-group">
                    <label> <img src={password1} style={{width:'25px'}} />
Password </label>
                    <input type="password" name="password" className="form-
control" id="password"
                    value={details.password}
                    onChange={handleInputs}
                    placeholder="Enter password"/>
                    </div>
```

```
                    <div class="container1 container2">
          <div class="btn1">
         <span type="submit" name="signup" id="signup"  value="register"
onClick={PostData} >
  <svg width="277" height="62">
    <defs>
        <linearGradient id="grad1">
            <stop offset="0%" stop-color="#FF8282"/>
            <stop offset="100%" stop-color="#E178ED" />
        </linearGradient>
    </defs>
     <rect x="5" y="5" rx="25" fill="none" stroke="url(#grad1)" width="266"
height="50"></rect>
  </svg>
    Register</span>
</div>
</div>
                  </div>
                </div>
              </div>
               </div>
            </div>
          </div>
        </div>
      </div>
    </>
  );
};
export default Register;
```

OUTPUT



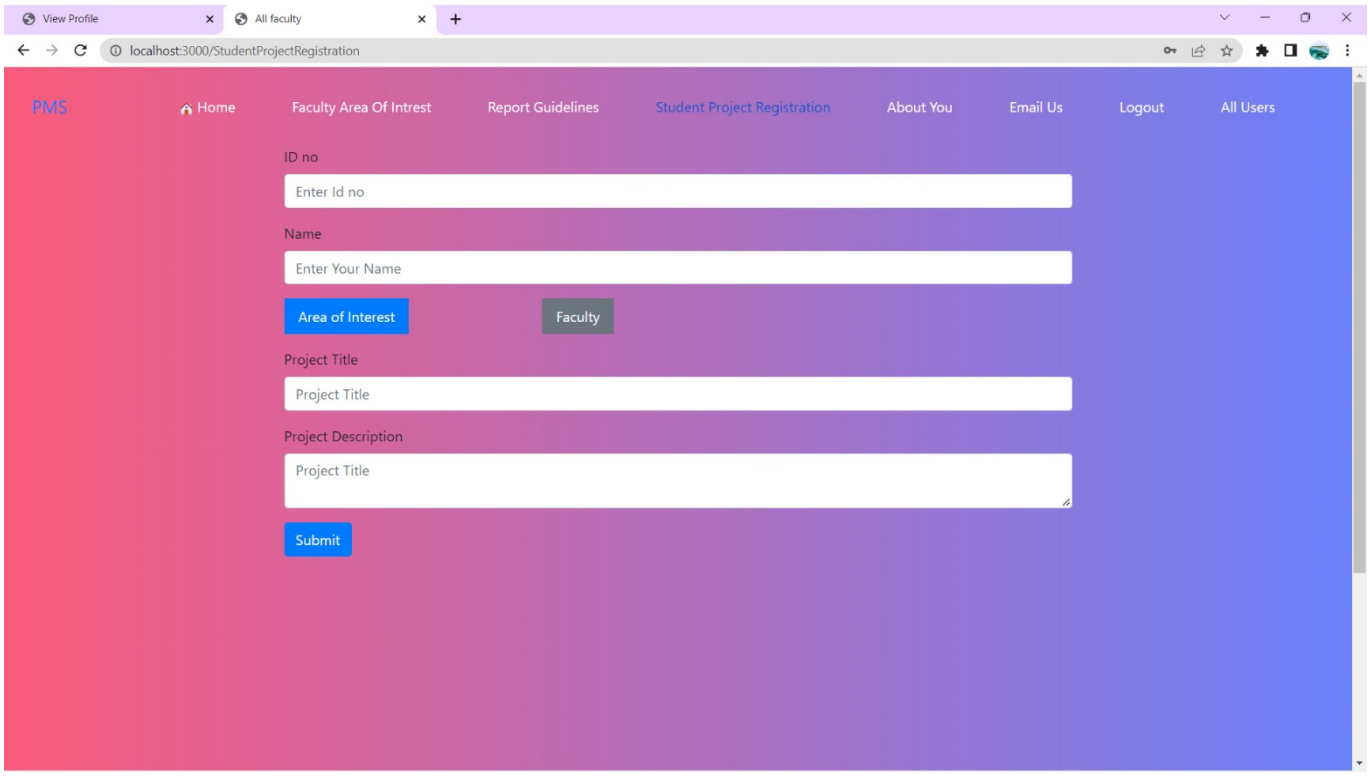
Student-Project-Registration.js

```
import React from 'react';
import './StudentProjectRegistration.css'

const StudentProjectRegistration = () => {
  return (
   <>
        <div className='StudentProjectRegistration'  style={{height:"1000px"}}>
        <form>
  <div class="form-group">
    <label for="exampleInputEmail1">ID no</label>
    <input type="text" class="form-control" id="exampleInputEmail1" aria-
describedby="emailHelp" placeholder="Enter Id no" />
  </div>
  <div class="form-group">
    <label for="exampleInputPassword1">Name</label>
```

```
    <input type="text" class="form-control" id="exampleInputPassword1"
placeholder="Enter Your Name" />
  </div>
  <div className='form-group'>


  <div className='dropdown bg-primary '>
      <a className='nav-link collapsed'
      role='button'
      id='dropdownMenuLink'
      data-bs-toggle='dropdown'
      aria-expanded='false'
      >
        Area of Interest
      </a>
      <ul className='dropdown-menu collapse ms-2' id='collpasePages' aria-
labelledby='dropdownMenuLink'  data-parent='#accordionSidebar'>
        <li className='dropdown-item'>Web Development</li>
        <li className='dropdown-item'>Cryptography and Network Security </li>
        <li className='dropdown-item'>Machine Learning </li>
        <li className='dropdown-item'>DevOps</li>
        <li className='dropdown-item'>MObile Application </li>
        <li className='dropdown-item'>Image processing </li>
        <li className='dropdown-item'>Cloud Computing </li>
        <li className='dropdown-item'>Data Science </li>
      </ul>
    </div>
    <div className='dropdown bg-secondary faculty'>
      <a className='nav-link collapsed'
      role='button'
      id='dropdownMenuLink'
      data-bs-toggle='dropdown'
      aria-expanded='false'
      >
        Faculty
      </a>
      <ul className='dropdown-menu collapse ms-2' id='collpasePages' aria-
labelledby='dropdownMenuLink'  data-parent='#accordionSidebar'>
        <li className='dropdown-item'>Vinod Kumar</li>
        <li className='dropdown-item'>ChandraShekar</li>
        <li className='dropdown-item'>Ravi Kumar </li>
        <li className='dropdown-item'>Ratna Kumari</li>
        <li className='dropdown-item'>Suneetha </li>
        <li className='dropdown-item'>Harinadha </li>
        <li className='dropdown-item'>Sandeep Kumar Reddy </li>
```

```
        <li className='dropdown-item'>satyanandaram</li>
        <li className='dropdown-item'>Muni Babu</li>
        <li className='dropdown-item'>Mahendra </li>
        <li className='dropdown-item'>Shalima Sulthana </li>
        <li className='dropdown-item'>Susmitha </li>
        <li className='dropdown-item'>HimaBindhu </li>
        <li className='dropdown-item'>SivaLakshmi </li>
        <li className='dropdown-item'>Udayasree </li>
        <li className='dropdown-item'>Shabana </li>
        <li className='dropdown-item'>Santosh Kumar</li>
        <li className='dropdown-item'>Linga Murthy </li>
      </ul>
    </div>
  </div>
  <div class="form-group">
  <label for="exampleInputPassword1">Project Title</label>
    <input type="text" class="form-control" id="exampleInputPassword1"
placeholder="Project Title" />
  </div>
  <div class="form-group">
  <label for="exampleInputPassword1">Project Description</label>
    <textarea type="text" class="form-control" id="exampleInputPassword1"
placeholder="Project Title" />
  </div>
  <button type="submit" class="btn btn-primary">Submit</button>
</form>
        </div>
   </>
  )
}

export default StudentProjectRegistration
```

OUTPUT



FacultyAreaofintrest.js

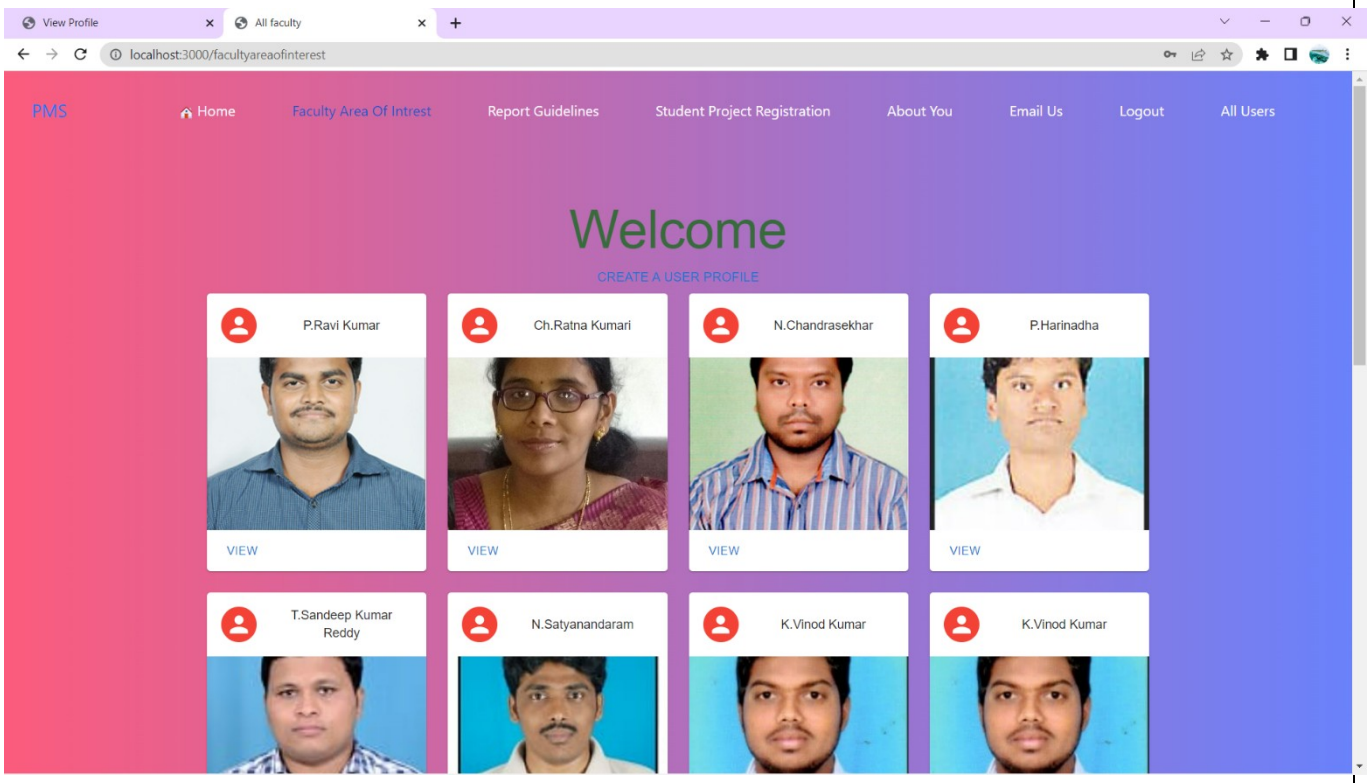
```
// import React from 'react'
// import Card from './Card'
// import facultydata from './faculty';
// import './faculty.css'

// const FacultyAreaOfIntrest = () => {
//     console.log(facultydata)
//   return (
//     <>
//     <div className='grid'>
//         {
//             facultydata.map((item)=>{
//                return <>
//                 <Card name={item.name} dept={item.dept}
designation={item.designation} education={item.education} gmail={item.gmail}
phone={item.phone} img={item.img}/>
//                 </>
//             })
//         }
//         </div>
```

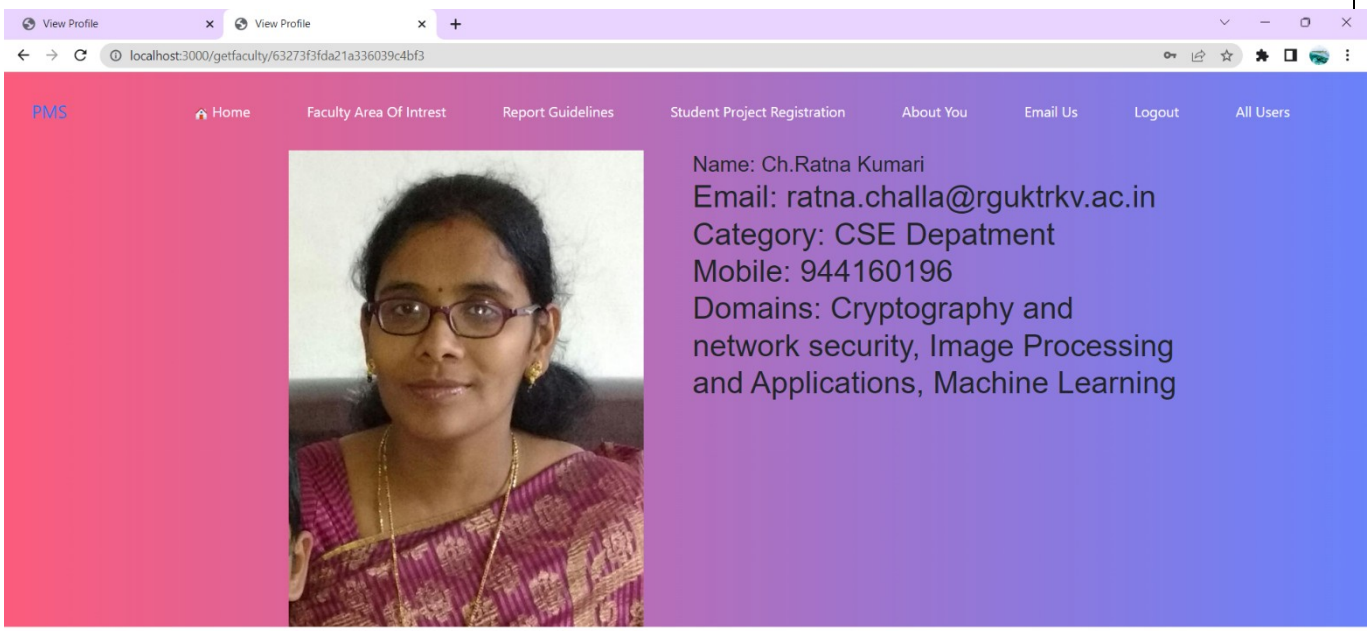
```
//     </>
//   )
// }

// export default FacultyAreaOfIntrest
import React,{useEffect, useState,useContext} from 'react'
import {Link,useHistory} from 'react-router-dom'
import { Container, Grid, Button, Typography } from '@mui/material';
import {ImageListItem} from '@mui/material'
import ViewPage from './ViewPage';
import PostCard from './PostCard';







const FacultyAreaOfIntrest = () => {
    // const [localData,setLocalData] = useState()
    // data stroring
    const [data,setData] = useState([])
    // loading animatin
    const [loading, setLoading] = useState(true);
    // navigating to specific id
    const navigate = useHistory()
    // fetching data
    const getData = ()=>{
        try {
             fetch('http://localhost:5000/getfaculty',{

             })
            .then((res)=>res.json())
            .then(response=>{
                setData(response)
                localStorage.setItem('posts',JSON.stringify(response));
                setLoading(false)
                })
        } catch(err){console.log(err)}
    }
```

```
    const [userData,setUserData] = useState({});
    const callAboutPage = async () =>{
      try{
        const res =await fetch('/about',{
            method:"GET",
            headers:{
              Accept:"appllication/json",
              "Content-Type":"application/json"
            },
            credentials:"include"
        });
        const data = await res.json();
        console.log(data);
        setUserData(data);

        if(!res.status===200){
          const error = new Error (res.error);
          throw error;
        }
      }catch(err){
        console.log(err);


      }
    }

    useEffect(()=>{

      getData()
      callAboutPage()
      document.title = "All faculty"

   },[])
   console.log(userData)

  return (
    <>
    <Container style={{"marginTop":"70px","textAlign":"center"}}>

                    <Typography variant='h2'>Welcome </Typography>
                    {
                     userData.email === 'admin@rguktrkv.ac.in' ? <Link
to="/create"><Button color='primary'>Create a User profile</Button></Link>  :""
                    }
```

```
                        <Grid container spacing={3}>

                       {
                        data!==undefined ?<>
                        {
                       data.map((item)=>{
                        return <>
                          <Grid item xs={3}>
                            <PostCard  imgdata={item}/>
                        </Grid>
                        </>
                      })
                    }
                        </>

                    :<h1>Loading</h1>
                       }

                          </Grid>
          </Container>
      </>
    )
}

export default FacultyAreaOfIntrest
```

OUTPUT


View Profile
All faculty
localhost:3000/facultyareaofinterest
PMS
Home
Faculty Area Of Intrest
Report Guidelines
Student Project Registration
About You
Email Us
Logout
All Users
Welcome
CREATE A USER PROFILE
P.Ravi Kumar
Ch.Ratna Kumari
N.Chandrasekhar
P.Harinadha
VIEW
VIEW
VIEW
VIEW
T.Sandeep Kumar Reddy
N.Satyanandaram
K.Vinod Kumar
K.Vinod Kumar


View Profile
View Profile
localhost:3000/getfaculty/63273f3fda21a336039c4bf3
PMS
Home
Faculty Area Of Intrest
Report Guidelines
Student Project Registration
About You
Email Us
Logout
All Users
Name: Ch.Ratna Kumari
Email: ratna.challa@rguktrkv.ac.in
Category: CSE Depatment
Mobile: 944160196
Domains: Cryptography and network security, Image Processing and Applications, Machine Learning

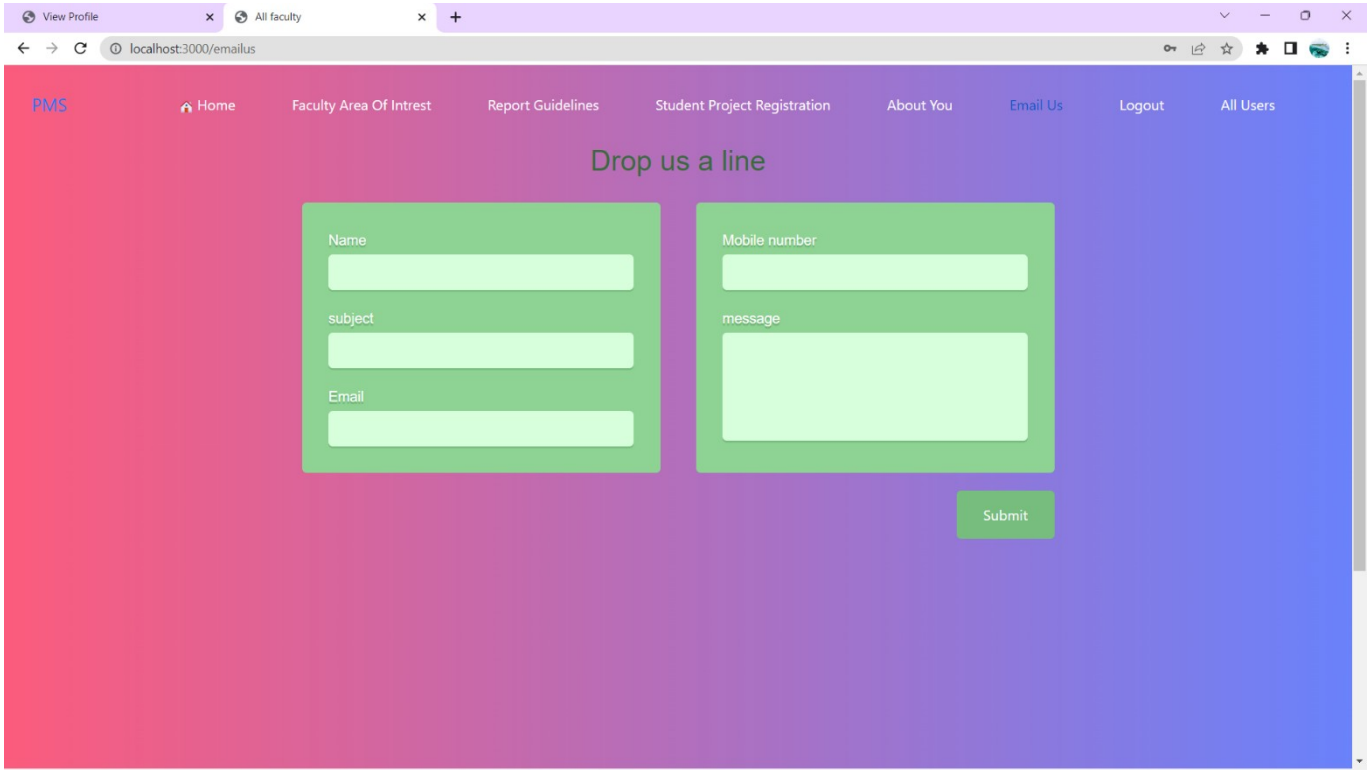
Emailus.js

```
import React from 'react';
import { useState,useEffect } from 'react';
import emailjs from 'emailjs-com';
import './styles.css';
import ProgressBar from 'react-progressbar-on-scroll'



const Email = () => {
const [userData,setUserData] = useState({});

const userDetails = async () =>{
    try{
      const res =await fetch('/getdata',{
          method:"GET",
          headers:{
            "Content-Type":"application/json"
          },
      });
      const data = await res.json();
      console.log(data);
      setUserData(data);

      if(!res.status===200){
        const error = new Error (res.error);
        throw error;
      }
    }catch(err){
      console.log(err);


    }
  }
useEffect(() => {
  userDetails();
}, [])



    const [data, setData] = useState({
        userName:'',
        subject: '',
        email: '',
        phoneNumber: '',
```

```
        message: '',

    })


    const {userName,subject,email,phoneNumber,message} = data;
    const changeHandler = (e) => {
        setData({...data, [e.target.name]: e.target.value})
    }
    const submitHandler = (e) => {
        e.preventDefault();
        console.log(data);

emailjs.sendForm('service_uvzebja','template_nbg3an5',e.target,'eEJip54KqTS3D2aZ
a')
        .then((result) => {
            console.log(result.text);
        }, (error) => {
            console.log(error.text);
        });
        e.target.reset();
    }

  return (
    <>
<div style={{height:"1000px"}}>
        <ProgressBar
            color="#1515c4"
        height={5}
        gradient={true}
        gradientColor="#00ffff"
        />
        <h2>Drop us a line</h2>
<form onSubmit={submitHandler}>
  <section class="left">
    <div class="input-container">
      <label for="name" >Name</label>
      <input type="text" name="userName" value={userData.name} />
    </div>
    <div class="input-container">
      <label for="subject">subject</label>
      <input type="text" name="subject"/>
    </div>
    <div class="input-container">
      <label for="email">Email</label>
```

```
      <input type="email" value={userData.email}  name="email"/>
    </div>
  </section>
  <section class="right">
    <div class="input-container">
      <label for="number">Mobile number</label>
      <input type="number" name="phoneNumber" />
    </div>

    <div class="input-container">
      <label for="comments">message</label>
      <textarea name="message" id=""></textarea>
    </div>
  </section>
  <div class="send-container">
   <input type="submit"  />
   </div>
</form>
</div>

    </>
  )
}

export default Email
```

OUTPUT



About.js

```
import React,{useState,useEffect} from 'react'

const About = () => {
  const [userData,setUserData] = useState({});


  const callAboutPage = async () =>{
    try{
      const res =await fetch('/about',{
          method:"GET",
          headers:{
            Accept:"appllication/json",
            "Content-Type":"application/json"
          },
          credentials:"include"
      });
      const data = await res.json();
      console.log(data);
      setUserData(data);
```

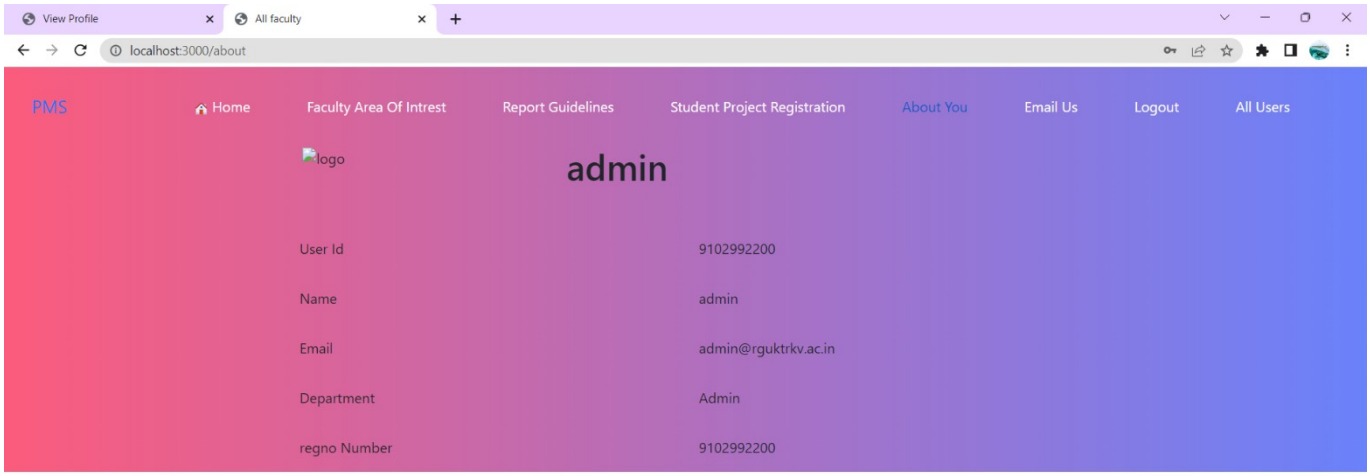
```
      if(!res.status===200){
        const error = new Error (res.error);
        throw error;
      }
    }catch(err){
      console.log(err);


    }
  }
useEffect(() => {
  callAboutPage();
}, [])



  return (
   <>
   <div className="container">
     <form method="GET">
        <div className="row">
          {/* profile pic */}
          <div className="col-md-4">
            <img src={userData.image} alt="logo" style={{width:"250px"}} />
          </div>

          <div className="col-md-6">
            <div className="profileHead">
                <h1>{ userData.name}</h1>
            </div>
          </div>
            <div className="contain">
              <div className="tab-content" id='mytabcontent'>
                <div className="tab-panel fade show active" id="home "
role="tabpanel" aria-labelledby="home" >
                  <div className="row mt-5">
                    <div className="col-md-6 ">
                      <label htmlFor="">User Id</label>
                    </div>
                    <div className="col-md-6">
                      <p>{userData.regno}</p>
                    </div>
                  </div>
```

```
                <div className="row mt-3">
                  <div className="col-md-6 ">
                    <label htmlFor="">Name</label>
                  </div>
                  <div className="col-md-6">
                    <p>{userData.name}</p>
                  </div>
                </div>

                <div className="row mt-3">
                  <div className="col-md-6 ">
                    <label htmlFor="">Email</label>
                  </div>
                  <div className="col-md-6">
                    <p>{userData.email}</p>
                  </div>
                </div>

                <div className="row mt-3">
                  <div className="col-md-6 ">
                    <label htmlFor="">Department</label>
                  </div>
                  <div className="col-md-6">
                    <p>{userData.department}</p>
                  </div>
                </div>

                <div className="row mt-3">
                  <div className="col-md-6 ">
                    <label htmlFor="">regno Number</label>
                  </div>
                  <div className="col-md-6">
                    <p>{userData.regno}</p>
                  </div>
                </div>

              </div>
            </div>
          </div>

      </div>
    </form>
  </div>

  </>
```

```
  )
}

export default About
```

OUTPUT

+++



Auth.js

```
const express = require("express");
const router = express.Router();
require('../db/conn');
const User = require("../model/userSchema");
const bcrypt = require("bcryptjs");
const jwt = require("jsonwebtoken");
const authenticate = require("../middleware/Authenticate");


router.get('/',async (req,res)=>{
    res.send(`Hello backend from router`);
});
```

```
router.post('/register',async (req,res)=>{

    const {name,email,regno,department,password} = req.body;
    if(!name || !email || !regno || !department || !password ){
        return res.status(422).json({error:"Fill all fields"})
    // return res.json({error:"Fill all fields"});
    }
    try{
        const userExist = await User.findOne({email:email})
        if(userExist){
            return res.status(422).json({error:"User already exist"})
        }else{
            const user = new User({name,email,regno,department,password});

            // hashning password
            await user.save();
            res.status(201).json({message:"User created successfully"});
        }

    }catch(err){
        console.log(err);

    }

    // console.log(name)
    // console.log(req.body);
    // res.json({message:req.body});

   });


//    login route
router.post('/signin',async (req,res)=>{
    // console.log(req.body);
    // res.json({message:"departmenting"});
    // const {email,password} = req.body;

    try{
       const {email,password} = req.body;
       if(!email || !password){
           return res.status(400).json({error:"Fill all fields"})
       }
         const userLogin = await User.findOne({email:email});
            // console.log(userLogin);
            if(userLogin){
```

```
                const isMatch = await
bcrypt.compare(password,userLogin.password);

                const token =await userLogin.generateAuthToken();
                // console.log(token);
                res.cookie('jwtoken',token,{
                    httpOnly:true,
                    expires:new Date(Date.now() + 3600000)
                });
                    if(!isMatch){
                        res.status(400).json({error:"Invalid credentials"});
                    }else{
                        res.status(200).json({message:"signin success"});
            }

            }else{
                res.status(400).json({error:"Invalid credentials"});
            }


    }catch(err){
        console.log(err);
    }
});



// about user
router.get('/about',authenticate,(req,res)=>{
    console.log(`about`);
    // res.send(`Hello about`);
    res.send(req.rootUser);
});

// get userdata
router.get('/getdata',authenticate,(req,res)=>{
    console.log(`Contact`);
    // res.send(`Hello about`);
    res.send(req.rootUser);
})
// contact us
router.post('/contact',authenticate, async(req,res)=>{
    try{
        const {name,email,regno,message} =req.body;
        if(!name || !email || !regno || !message){
```

```
            console.log("error in contact");
            return res.status(422).json({error:"Fill all contact fields"})
        }
        const userContact = await User.findOne({_id:req.rootUserId});

        if(userContact){
            const userMessage =await
userContact.addMessage(name,email,regno,message);

            await userContact.save();
            res.status(201).json({message:"Message sent successfully"});
        }

    }catch(error){
        console.log(error);
    }
});



router.post('/faculty',authenticate,async (req,res)=>{
    const {name,img,email,category,mobile,domains} = req.body;
    try{
        const newData = new faculty({name,img,email,category,mobile,domains});
        await  newData.save();
        console.log("new post added")
        return res.json(await faculty.find())
    }
    catch(err){
        console.log(err.message)
        console.log("Error in adding posts")

    }
})


// logout

router.get('/logout',(req,res)=>{
    console.log(`logout`);
    // res.send(`Hello about`);
    res.clearCookie('jwtoken',{path:'/'})
    res.status(200).send(`userLogout`);
});
module.exports = router;
```

Logout.js

```
import React from 'react'
import { useEffect ,useState,useContext} from 'react'
import { useHistory } from 'react-router-dom';
import {userContext} from "../../App"

const Logout = () => {

    const {state,dispatch} = useContext(userContext);

    const history = useHistory();
    useEffect(()=>{
        fetch('/logout',{
            method:'GET',
            headers:{
                Accept:'appllication/json',
                'Content-Type':'application/json'
            },
            credentials:'include'

        }).then((res)=>{
            dispatch({type:"USER",payload:false});
            history.push('/login',{replace:true});
            if(res.status !== 200){
                const error = new Error(res.error);
                throw error;
            }
        }).catch((err)=>{
            console.log(err);
        });
    });
  return (
    <>
    <h1>Logout</h1>

    </>
  )
}
export default Logout
```

Conclusion

This web application provides Faculty to maintain online project management system. It saves time as it allows the students in the University can register for their project in their project web portal. It is automatically manipulated by the server. Project registration Management System, being web-based, needs to be throughly tested before implementation to find any security gaps.

References

- https://reactjs.org/
- https://www.javatpoint.com/nodejs
- https://www.javatpoint.com/