CAMPUS CONNECTION

PROJECT REPORT

Submitted by

IFRA – B191769

SHAIK ARIEF AHMED – B192224

BANOTHU PAVAN KALYAN- B191200

Of

Bachleor of Technology

Under the guidance of

Mrs. Latha

Asst.Prof.CSE, RGUKT BASAR

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

BASAR, NIRMAL (DIST.),

TELANGANA – 504107

CAMPUS CONNECTION

Project Report submitted to

Rajiv Gandhi University of Knowledge Technologies,
Basar

or the partial fulfillment of the

requirements for the award of the

degree of

Bachelor of Technology

in
Computer Science & Engineering

by

IFRA - B191769

SHAIK ARIEF AHMED - B192224

BANOTHU PAVAN KALYAN- B191200

Under the Guidance

of Mrs. Latha

Asst. Prof. CSE, RGUKT BASAR

DEPARTMENT OF COMPUTER SCIENCE AND
ENGINEERING
RAJIV GANDHI UNIVERSITY OF KNOWLEDGE
TECHNOLOGIES

BASAR, NIRMAL(DIST.),
TELANGANA - 504107
JUNE - 2024

DEPARTMENT OF COMPUTER SCIENCE
AND ENGINEERING

RAJIV GANDHI UNIVERSITY OF
KNOWLEDGE TECHNOLOGIES,
Basar

CERTIFICATE

This is to certify that the Project Report entitled ‘CAMPUS CONNECTION’ submitted by Ifra – B191769, Shaik Arief Ahmed– B192224,Banothu Pavan Kalyan-B191200 Department of Computer Science and Engineering, Rajiv Gandhi University Of Knowledge Technologies, Basar, for partial fulfillment of the requirements for the degree of Bachelor of Technology in Computer Science and Engineering; is a bonafide record of the work.

PROJECT SUPERVISOR :	**HEAD OF DEPARTMENT :**
Mrs.LATHA	**Mr.REVYA NAIK**
Assistant Professor	**Assistant Professor**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES, Basar

DECLARATION

We hereby declare that the work which is being presented in this project entitled, ”CAMPUSCONNECTION” submitted to RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES, BASAR in the partial fulfillment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE AND ENGINEERING, is an authentic record of our own work carried out under the supervision of “Mrs. Latha”, Assistant Professor in Department of Computer Science And Engineering, RGUKT, Basar. The matter embodied in this project report has not been submitted by us for the award of any other degree.

Place: Basar
Date: 18-07-24

IFRA - B191769
SK ARIEF AHMED - B192224
B. PAVAN KALYAN - B191200

ACKNOWLEDGMENT

We would like to express our deep gratitude to our project guide Mrs.Sindhuja, Assistant Professor, Department of Computer Science and Engineering, RGUKT BASAR, for his guidance with unsurpassed knowledge and immense encouragement.

We are grateful to Mr. REVYA NAIK, Head of the Department, Computer Science and Engineering, for providing us with the required facilities for the completion of the project work.

We are very much thankful to the Director and Administration, RGUKT, BASAR for their encouragement and cooperation to carry out this work.

We express our thanks to all teaching faculty of Department of CSE, whose suggestions during reviews helped us in accomplishment of our project.

We would like to thank all non-teaching staff of the Department of CSE, RGUKT BASAR for providing great assistance in accomplishment of our project.

We would like to thank our parents, friends, and classmates for their encouragement throughout our project period. At last but not the least, we thank everyone for supporting us directly or indirectly in completing this project successfully.

PROJECT STUDENTS :

IFRA – B191769

SK ARIEF AHMED – B192224

B. PAVAN KALYAN - B191200

ABSTRACT

"Building Bridges: Enhancing Campus Connection through Collaboration and Engagement".In today's dynamic educational landscape, fostering strong connections within campus communities is pivotal for student success and institutional growth. This abstract explores strategies and initiatives aimed at promoting collaboration, communication, and active engagement among students, faculty and staff. By cultivating a supportive and inclusive environment, campuses can nurture a sense of belonging and collective responsibility, ultimately enriching the overall educational experience and preparing students for future challenges.

Effective campus connection is integral to fostering a supportive and inclusive environment where students can thrive academically, socially, and personally. This abstract explores various strategies and initiatives aimed at enhancing connectivity among students, faculty, and staff. By emphasizing collaboration, communication, and shared goals, campuses can promote holistic student development, ensuring that every member of the community feels valued and empowered. Through these efforts, we aim to create a vibrant and cohesive campus culture that prepares students for success beyond graduation.

This abstract encapsulates the essence of bringing together students, faculty, and staff in a cohesive and collaborative environment. It highlights the importance of connectivity in academia, emphasizing mutual support, shared learning, and personal development within the campus community.

Contents

CHAPTER 1

INTRODUCTION

1.1 PROBLEM OBJECTIVE

To establish cohesive communication channels and foster inclusive engagement among students, faculty, and staff, enhancing community collaboration and enriching the overall campus experience.

1. **Enhance Communication Channels:**
Establish clear and effective communication channels that facilitate seamless information dissemination and interaction among students, faculty, and staff.

2.**Strengthen Community Collaboration:**

Encourage collaboration and teamwork across diverse groups within the campus community to foster a sense of unity and collective responsibility.

3.**Utilize Technology Effectively:**
Leverage technological tools and platforms to enhance virtual connectivity and facilitate remote engagement, especially in hybrid learning environments.

4.**Evaluate and Improve Initiatives:**
Continuously assess the effectiveness of campus connection initiatives through feedback and data analysis, adapting strategies as needed to meet evolving community needs.

1.2 BACKGROUND

The Rajiv Gandhi University of Knowledge Technologies (RGUKT) is a prestigious university that aims to provide quality education to students from rural

background.One of the key challenges faced students is lack of communication on a single platform and miscommunication in the premises, which can lead misunderstanding
of messages.

Effective communication and collaboration are essential pillars of a thriving academic community today's interconnected world, universities and colleges recognize the significance of fostering strong connections among students, faculty, and staff. These connections not only enhance the educational experience but also contribute to a supportive and inclusive campus environment.

In higher education, "Campus Connection" refers to the strategic efforts and initiatives aimed at fostering strong interpersonal relationships, collaboration, and community engagement within academic institutions. These efforts are integral to creating a vibrant and supportive campus environment where students, faculty, and staff can thrive academically, professionally, and personally.

1.3 PROBLEM STATEMENT

Many universities face challenges in fostering cohesive communication and community engagement among students, faculty, and staff, impacting overall campus connectivity and collaboration.

Inadequate communication channels and disconnected engagement efforts hinder the formation of a cohesive campus community, impacting student integration, academic success, and overall campus satisfaction.

Here's an example of a problem statement for a "Campus Connect" project:

In many academic institutions, there exists a significant gap in fostering meaningful connections and collaboration among students, faculty, and staff. Despite advancements in technology and various initiatives aimed at promoting campus engagement, the lack of cohesive communication strategies and inclusive community-building efforts often results in fragmented campus relationships and missed opportunities for holistic student development and institutional cohesion.

Thus resulting in a disjointed campus experience that undermines community cohesion, academic success, and overall institutional effectiveness.

1.4 AIM and OBJECTIVE

The aim of the Campus Connect project is to enhance communication, collaboration, and community engagement within the academic institution, fostering a cohesive and inclusive campus environment.

Objectives:

1. Establish Effective Communication Channels:

- Implement robust communication platforms and strategies to facilitate seamless information sharing among students, faculty, and staff.

2. Promote Interdisciplinary Collaboration:

- Encourage collaboration across departments and disciplines to promote innovative research, teaching methodologies, and student initiatives.

3. Enhance Student Engagement:

- Develop programs and activities that actively involve students in campus life, fostering leadership skills, teamwork, and personal development.

4. Support Faculty and Staff Development:

- Provide professional development opportunities and resources to enhance teaching effectiveness, job satisfaction, and institutional commitment.

5. Cultivate an Inclusive Campus Culture:

- Promote diversity, equity, and inclusion through initiatives that celebrate cultural differences, address systemic barriers, and ensure equitable access to resources.

By achieving these objectives, the Campus Connect project aims to create a supportive and dynamic campus community where all members—students, faculty, and staff—feel connected, valued, and empowered to contribute to the overall success and vibrancy of the institution.

1.5 SCOPE OF THE PROJECT

Here are some potential scope points for the Campus Connect project:

1. Communication Infrastructure Enhancement:

- Evaluate and upgrade existing communication channels (e.g., email systems, intranet, mobile apps) to ensure they support seamless and efficient information dissemination.

2. Virtual Collaboration Tools:

- Implement and optimize virtual meeting platforms, collaboration tools (e.g., Microsoft Teams, Zoom), and project management software to facilitate remote teamwork and communication.

3. Community Engagement Initiatives:

- Develop and execute community-building activities such as social events, workshops, and forums that encourage interaction and networking among students, faculty, and staff.

4.Interdisciplinary Projects and Initiatives:

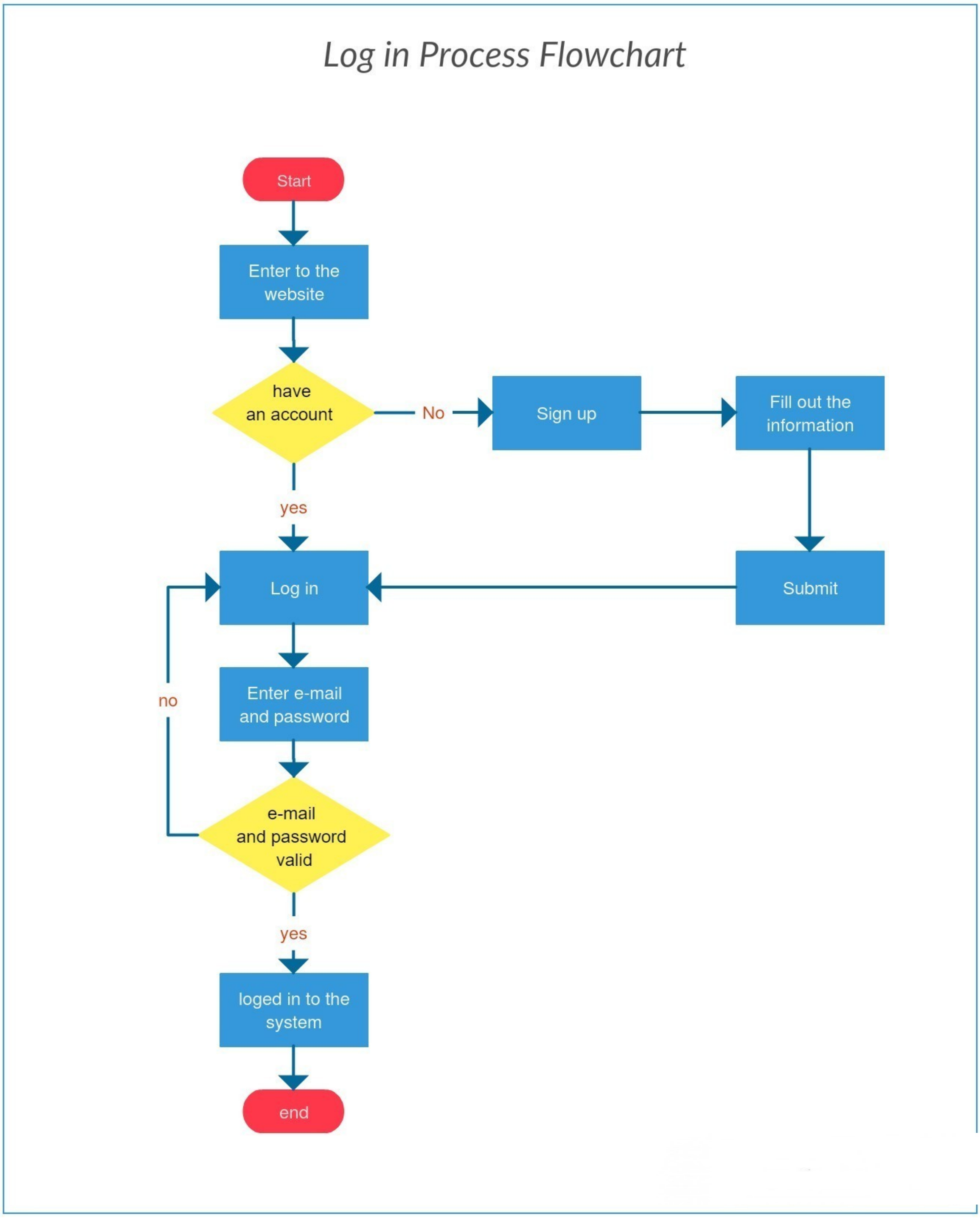
- Foster interdisciplinary collaborations through joint research projects, cross-departmental courses, and shared academic initiatives that leverage diverse expertise and perspectives.

5. Evaluation and Feedback Mechanisms:

- Implement regular surveys, focus groups, and performance metrics to assess the effectiveness of campus connection initiatives and gather feedback for continuous improvement.

These scope points aim to enhance connectivity, collaboration, and community engagement within the academic institution, contributing to a vibrant and supportive campus environment.

1.6 FLOW CHART



CHAPTER 2

Literature review

2.1 Review of Literature

Introduction to Campus Connectivity:

- Define the concept of "Campus Connect" and its importance in enhancing the overall educational experience.
- Discuss the impact of effective campus connectivity on student engagement, retention, and academic success.

Theoretical Frameworks:

- Review theoretical frameworks (e.g., social capital theory, community of practice) that underpin the importance of social connections and collaboration in educational settings.
- Explore how these theories are applied to understanding and improving campus connectivity.

Communication Strategies and

- Examine studies on effective communication channels and strategies used in higher education institutions to foster connections among students, faculty, and staff.
- Discuss the role of technology (e.g., social media, learning management systems) in facilitating communication and collaboration on campuses.

Future Directions and Innovations:

- Highlight emerging trends and innovations in campus connectivity, such as virtual engagement tools, hybrid learning models, and global collaborations.
- Discuss potential areas for future research and development in enhancing campus connectivity and community

\

This literature review would provide a comprehensive overview of the current state of knowledge and research on “Campus Connect” offering insights into effective strategies, challenges,and opportunities for enhancing connectivity within academic communitites.

2.2 Advantages

advantages of a Campus Connect project can be substantial, impacting various aspects of academic life and institutional effectiveness. Here are some key advantages:

1. Enhanced Communication:

- Facilitates seamless and efficient communication among students, faculty, and staff, fostering collaboration and knowledge-sharing.

2. Improved Community Engagement:

- Promotes a sense of belonging and community among campus members, enhancing morale, satisfaction, and overall campus culture.

3. Increased Collaboration:

- Encourages interdisciplinary collaboration and teamwork across departments, leading to innovative research, teaching methodologies, and student initiatives.

4.Support for Student Success:

- Provides resources, mentorship, and support services that contribute to academic achievement, personal development, and career readiness.

5. Technological Integration:

- Utilizes technology to enhance virtual connectivity, facilitate remote learning, and provide access to educational resources and services.

6.Student Leadership Opportunities:

- Engages students in leadership roles and peer mentorship programs, promoting leadership skills, teamwork, and campus involvement.

7. Evaluation and Continuous Improvement:

- Establishes mechanisms for evaluating the effectiveness of campus connection initiatives, allowing for continuous improvement based on feedback and data analysis.

Overall, a Campus Connect project contributes to creating a vibrant and cohesive campus community where students, faculty, and staff can thrive academically, professionally, and personally. It supports holistic development, innovation, and collaborative excellence within the institution, preparing students for success in a globalized and interconnected world.

CHAPTER 3
METHODOLOGY

The Tech Stack we used contains React.js, Node.js, Express.js, MongoDB and CSS. As we used react for frontend, everything will be as components. Typically we created 4 components. They are:

1.Home Page

2.Login Page

3.Sign Up page

4.Personal Messsage Chatbox Page

3.1 App Page

The home page shows the interface of the application along with the login and sign up options.

This home page depicts the outer interface of the developed application.

3.2 Login Page

The Login Page asks for username and password of the user. The credentials we have added an eye icon for the password to check whether the user is entering the correct password or not. The Login page connects with the login api and fetch the response from backend. If that is a successful Login , the user is navigated to their corresponding Post Login pages.

3.3 Sign Up Page

The Sign up deals with the user who had not registered.Page asks for username email and password of the user. The credentials we have added an eye icon for the password to check whether the user is entering the correct password or not.thus the user get registered and can access the application.

3.4 Post Page

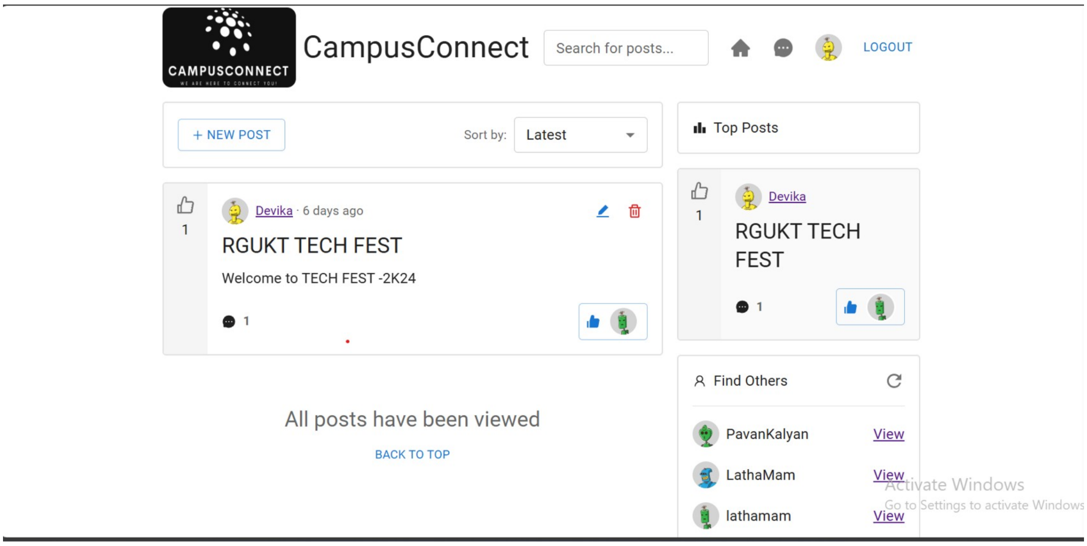
The registered user is allowed to post the new posts that he/she wish to post, which later allowed to by viewed by other users and respond towards it.

3.5 Message box

The registered user is allowed to send messages to the users of the group personally to whom they wish to send.

CHAPTER 4
CODE IMPLEMETATION

4.1 App.js



RGUKT TECH FEST

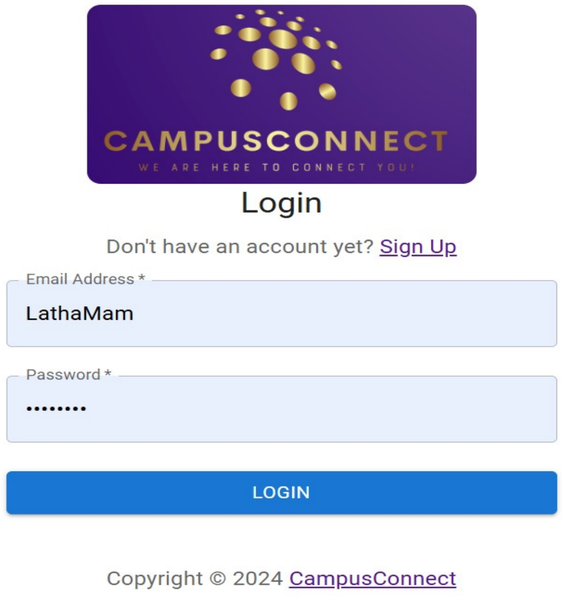
Welcome to TECH FEST -2K24

RGUKT TECH FEST

All posts have been viewed

```
import "@mui/material";
 import "react-icons";
 import "react-icons/bi";
 import "react-icons/md";
 import "react-icons/bs";
 import "react-router-dom";
 import { CssBaseline } from "@mui/material";
 import { ThemeProvider } from "@mui/material/styles";

 import {
   BrowserRouter,
   Route,
   Routes,
   useParams,
   useSearchParams,
 } from "react-router-dom";
 import theme from "./theme";

 import PostView from "./components/views/PostView";
 import CreatePostView from "./components/views/CreatePostView";
 import ProfileView from "./components/views/ProfileView";
 import LoginView from "./components/views/LoginView";
 import SignupView from "./components/views/SignupView";
 import ExploreView from "./components/views/ExploreView";
import PrivateRoute from "./components/PrivateRoute";
import SearchView from "./components/views/SearchView";
import MessengerView from "./components/views/MessengerView";
import { initiateSocketConnection, socket } from "./helpers/socketHelper";


function App() {
  initiateSocketConnection();

  return (
    <ThemeProvider theme={theme}>
      <BrowserRouter>
        <CssBaseline />
        <Routes>
          <Route path="/" element={<ExploreView />} />
          <Route path="/posts/:id" element={<PostView />} />
          <Route
            path="/posts/create"
            element={
              <PrivateRoute>
                <CreatePostView />
              </PrivateRoute>
            }
          />
          <Route
            path="/messenger"
            element={
              <PrivateRoute>
                <MessengerView />
              </PrivateRoute>
            }
          />
          <Route path="/search" element={<SearchView />} />
          <Route path="/users/:id" element={<ProfileView />} />
          <Route path="/login" element={<LoginView />} />
          <Route path="/signup" element={<SignupView />} />
        </Routes>
      </BrowserRouter>
    </ThemeProvider>
  );
}
export default App;
```

4.2 Loginview.js



Login

Don't have an account yet? Sign Up

Email Address *

LathaMam

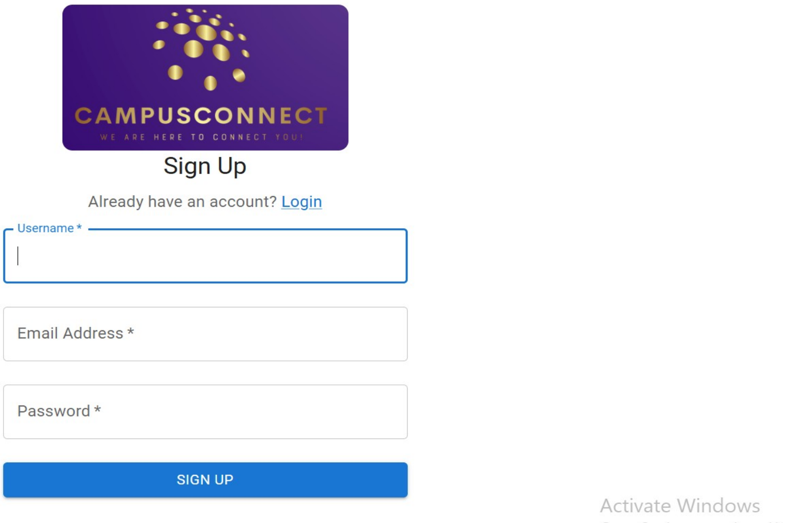
Password *

••••••••

Copyright © 2024 CampusConnect

```
 import {
  Alert,
  Button,
  Checkbox,
  Container,
  FormControlLabel,
  Stack,
  TextField,
  Typography,
} from "@mui/material";
import { Box } from "@mui/system";
import React, { useState } from "react";
import { Link, useNavigate } from "react-router-dom";
import { login } from "../../api/users";
import ErrorAlert from "../ErrorAlert";
import { loginUser } from "../../helpers/authHelper";
import Copyright from "../Copyright";
import logo from '../../images/blueSmallLogo.png'

const LoginView = () => {
  const navigate = useNavigate();

  const [formData, setFormData] = useState({
    email: "",
    password: "",
  });
  const [serverError, setServerError] = useState("");

  const handleChange = (e) => {
    setFormData({ ...formData, [e.target.name]: e.target.value });
  };

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

    const data = await login(formData);
    if (data.error) {
      setServerError(data.error);
    } else {
      loginUser(data);
      navigate("/");
    }
  };

  return (
    <Container maxWidth={"xs"} sx={{ mt: 6 }}>
      <Stack alignItems="center">

          <div style={{display:'flex' , flexDirection:'column', justifyContent:'center', alignItems:'center'}}>
            <img src={logo} style={{height: '150px' , width: '280px', borderRadius: '10px'}}></img>
          </div>



        <Typography variant="h5" gutterBottom>
          Login
        </Typography>
        <Typography color="text.secondary">
          Don't have an account yet? <Link to="/signup">Sign Up</Link>
        </Typography>
        <Box component="form" onSubmit={handleSubmit}>
          <TextField
            label="Email Address"
            fullWidth
            margin="normal"
            autoComplete="email"
            autoFocus
            required
            id="email"
            name="email"
            onChange={handleChange}
          />
          <TextField
            label="Password"
            fullWidth
            required
            margin="normal"
            id="password  "
            name="password"
            onChange={handleChange}
            type="password"
          />

          <ErrorAlert error={serverError} />
          <Button type="submit" fullWidth variant="contained" sx={{ my: 2 }}>
            Login
          </Button>
        </Box>
        <Box sx={{ mt: 3 }}>
          <Copyright />
        </Box>
      </Stack>
    </Container>
  );
};
export default LoginView;
```

4.3 SIGN UP

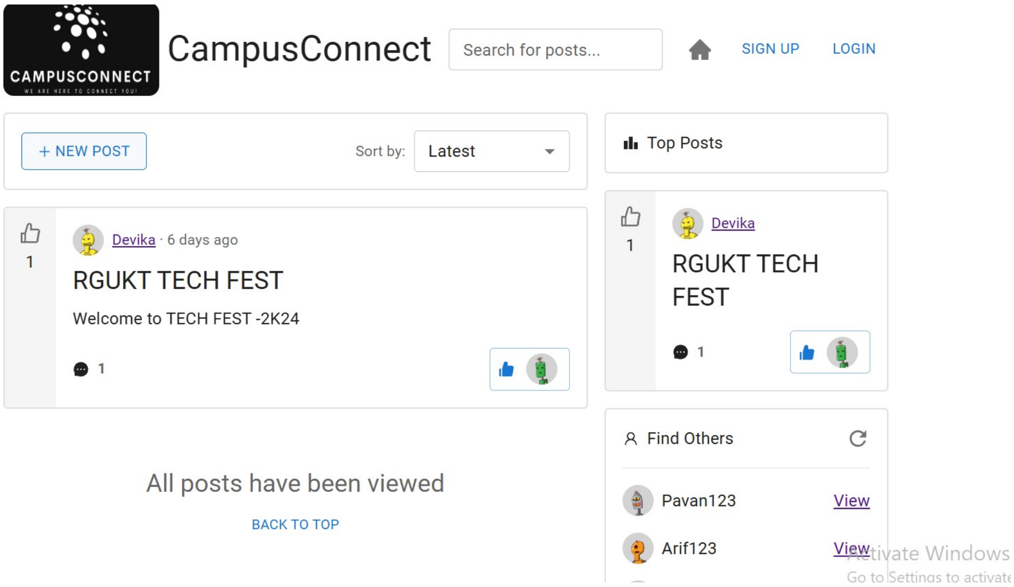


Sign Up

Already have an account? Login

```
import {
  Button,
  Container,
  Stack,
  TextField,
  Typography,
  Link,
  Alert,
} from "@mui/material";
import { Box } from "@mui/system";
import React, { useState } from "react";
import { signup } from "../../api/users";
import { loginUser } from "../../helpers/authHelper";
import { useNavigate } from "react-router-dom";
import Copyright from "../Copyright";
import ErrorAlert from "../ErrorAlert";
import { isLength, isEmail, contains } from "validator";
import logo from '../../images/blueSmallLogo.png';

const SignupView = () => {
  const navigate = useNavigate();
  const [serverError, setServerError] = useState("");
  const [errors, setErrors] = useState({});

  const [formData, setFormData] = useState({
    username: "",
    email: "",
    password: "",
  });

  const handleChange = (e) => {
    setFormData({ ...formData, [e.target.name]: e.target.value });
  };

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

    const errors = validate();
    if (Object.keys(errors).length !== 0) return;

    const data = await signup(formData);

    if (data.error) {
      setServerError(data.error);
    } else {
      loginUser(data);
      navigate("/");
    }
  };

  const validate = () => {
    const errors = {};

    if (!isLength(formData.username, { min: 6, max: 30 })) {
      errors.username = "Must be between 6 and 30 characters long";
    }

    if (contains(formData.username, " ")) {
      errors.username = "Must contain only valid characters";
    }

    if (!isLength(formData.password, { min: 8 })) {
      errors.password = "Must be at least 8 characters long";
    }

    if (!isEmail(formData.email)) {
      errors.email = "Must be a valid email address";
    }

    setErrors(errors);

    return errors;
  };

  return (
    <Container maxWidth={"xs"} sx={{ mt: { xs: 2, md: 6 } }}>
      <Stack alignItems="center">
      <div style={{display:'flex' , flexDirection:'column', justifyContent:'center',
alignItems:'center'}}>
            <img src={logo} style={{height: '150px' , width: '280px', borderRadius:
'10px'}}></img>

          </div>
        <Typography variant="h5" gutterBottom>
          Sign Up
        </Typography>
        <Typography color="text.secondary">
          Already have an account? <Link to="/login">Login</Link>
        </Typography>
        <Box component="form" onSubmit={handleSubmit}>
          <TextField
            label="Username"
            fullWidth
            margin="normal"
            autoFocus
            required
            id="username"
            name="username"
            onChange={handleChange}
            error={errors.username !== undefined}
            helperText={errors.username}
          />
          <TextField
            label="Email Address"
            fullWidth
            margin="normal"
            autoComplete="email"
            required
            id="email"
            name="email"
            onChange={handleChange}
            error={errors.email !== undefined}
            helperText={errors.email}
          />
          <TextField
            label="Password"
            fullWidth
            required
            margin="normal"
            autoComplete="password"
            id="password"
            name="password"
            type="password"
            onChange={handleChange}
            error={errors.password !== undefined}
            helperText={errors.password}
          />
          <ErrorAlert error={serverError} />
          <Button type="submit" fullWidth variant="contained" sx={{ my: 2 }}>
            Sign Up
          </Button>
        </Box>
        <Box sx={{ mt: 3 }}>
          <Copyright />
        </Box>
      </Stack>
    </Container>
  );
};

export default SignupView;
```

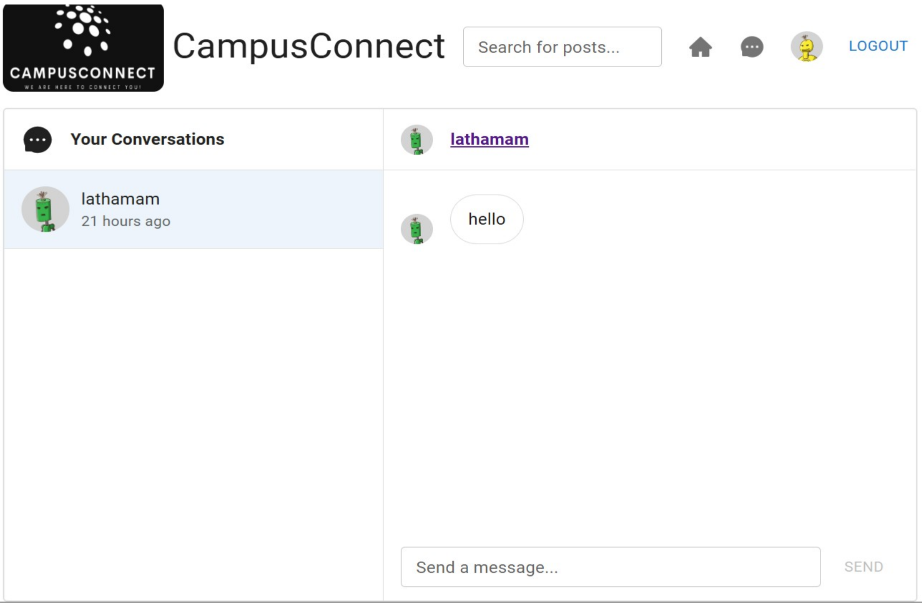
4.4 Post.js


CAMPUSCONNECT
WE ARE HERE TO CONNECT YOU!
CampusConnect
Search for posts...
SIGN UP
LOGIN
+ NEW POST
Sort by: Latest
Top Posts
1
Devika · 6 days ago
RGUKT TECH FEST
Welcome to TECH FEST -2K24
1
All posts have been viewed
BACK TO TOP
1
Devika
RGUKT TECH FEST
1
Find Others
Pavan123 View
Arif123 View

```
import { BASE_URL } from "../config";
const getUserLikedPosts = async (likerId, token, query) => {
  try {
    const res = await fetch(
      BASE_URL +
        "api/posts/liked/" +
        likerId +
        "?" +
        new URLSearchParams(query),
      {
        headers: {
          "x-access-token": token,
        },
      }
    );
    return await res.json();
  } catch (err) {
    console.log(err);
  }
};

const getPosts = async (token, query) => {
  try {
    const res = await fetch(
      BASE_URL + "api/posts?" + new URLSearchParams(query),
      {
        headers: {
          "x-access-token": token,
        },
      }
    );
    return await res.json();
  } catch (err) {
    console.log(err);
  }
};

const getPost = async (postId, token) => {
  try {
    const res = await fetch(BASE_URL + "api/posts/" + postId, {
      headers: {
        "x-access-token": token,
      },
    });
    return await res.json();
  } catch (err) {
    console.log(err);
  }
};

const getUserLikes = async (postId, anchor) => {
  try {
    const res = await fetch(
      BASE_URL +
        "api/posts/like/" +
        postId +
        "/users?" +
        new URLSearchParams({
          anchor,
        })
    );

    return await res.json();
  } catch (err) {
    console.log(err);
  }
};

const createPost = async (post, user) => {
  try {
    const res = await fetch(BASE_URL + "api/posts", {
      method: "POST",
      headers: {
        Accept: "application/json",
        "Content-Type": "application/json",
        "x-access-token": user.token,
      },
      body: JSON.stringify(post),
    });
    return await res.json();
  } catch (err) {
    console.log(err);
  }
};

const updatePost = async (postId, user, data) => {
  try {
    const res = await fetch(BASE_URL + "api/posts/" + postId, {
      method: "PATCH",
      headers: {
        Accept: "application/json",
        "Content-Type": "application/json",
        "x-access-token": user.token,
      },
      body: JSON.stringify(data),
    });
    return res.json();
  } catch (err) {
    console.log(err);
  }
};

const deletePost = async (postId, user) => {
  try {
    const res = await fetch(BASE_URL + "api/posts/" + postId, {
      method: "DELETE",
      headers: {
        "x-access-token": user.token,
      },
    });
    return res.json();
  } catch (err) {
    console.log(err);
  }
};

const getComments = async (params) => {
  try {
    const { id } = params;
    const res = await fetch(BASE_URL + "api/comments/post/" + id);
    return res.json();
  } catch (err) {
    console.log(err);
  }
};

const getUserComments = async (params) => {
  try {
    const { id, query } = params;
    const res = await fetch(
      BASE_URL + "api/comments/user/" + id + "?" + new
URLSearchParams(query)
    );
    return res.json();
  } catch (err) {
    console.log(err);
  }
};

const createComment = async (comment, params, user) => {
  try {
    const { id } = params;
    const res = await fetch(BASE_URL + "api/comments/" + id, {
      method: "POST",
      headers: {
        Accept: "application/json",
        "Content-Type": "application/json",
        "x-access-token": user.token,
      },
      body: JSON.stringify(comment),
    });
    return res.json();
  } catch (err) {
    console.log(err);
  }
};

const updateComment = async (commentId, user, data) => {
  try {
    const res = await fetch(BASE_URL + "api/comments/" + commentId, {
      method: "PATCH",
      headers: {
        Accept: "application/json",
        "Content-Type": "application/json",
        "x-access-token": user.token,
      },
      body: JSON.stringify(data),
    });
    return res.json();
  } catch (err) {
    console.log(err);
  }
};

const deleteComment = async (commentId, user) => {
  try {
    const res = await fetch(BASE_URL + "api/comments/" + commentId, {
      method: "DELETE",
      headers: {
        "x-access-token": user.token,
      },
    });
    return res.json();
  } catch (err) {
    console.log(err);
  }
};

const likePost = async (postId, user) => {
  try {
    const res = await fetch(BASE_URL + "api/posts/like/" + postId, {
      method: "POST",
      headers: {
        "x-access-token": user.token,
      },
    });
    return res.json();
  } catch (err) {
    console.log(err);
  }
};

const unlikePost = async (postId, user) => {
  try {
    const res = await fetch(BASE_URL + "api/posts/like/" + postId, {
      method: "DELETE",
      headers: {
        "x-access-token": user.token,
      },
    });
    return res.json();
  } catch (err) {
    console.log(err);
  }
};
```

```
export {
  getPost,
  createPost,
  updatePost,
  deletePost,
  getPosts,
  getUserComments,
  getUserLikedPosts,
  getComments,
  createComment,
  deleteComment,
  updateComment,
  likePost,
  unlikePost,
  getUserLikes,
};
```

4.5 Message Box

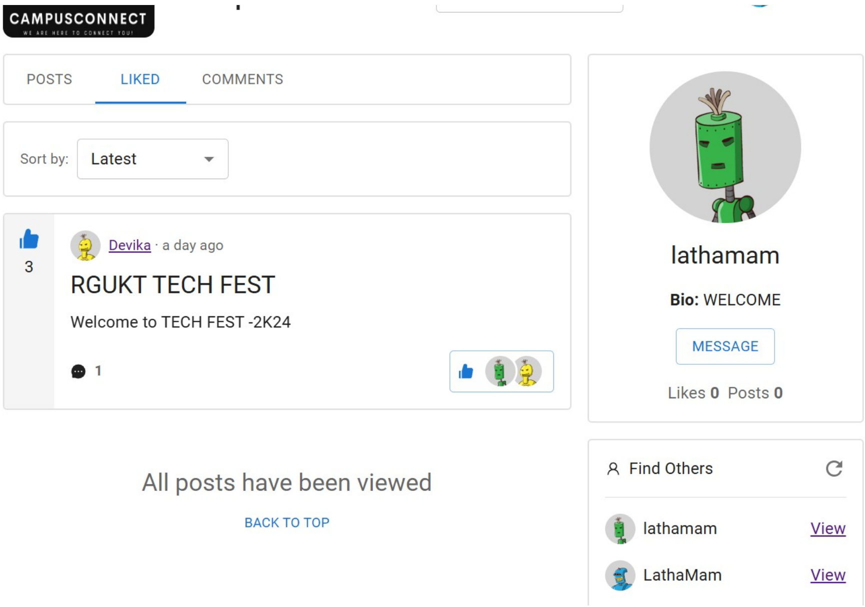


hello

```
import { BASE_URL } from "../config";
const getConversations = async (user) => {
  try {
    const res = await fetch(BASE_URL + "api/messages", {
      headers: {
        "x-access-token": user.token,
      },
    });
    return await res.json();
  } catch (err) {
    console.log(err);
  }
};

const getMessages = async (user, conversationId) => {
  try {
    const res = await fetch(BASE_URL + "api/messages/" + conversationId, {
      headers: {
        "x-access-token": user.token,
      },
    });
    return await res.json();
  } catch (err) {
    console.log(err);
  }
};

const sendMessage = async (user, message, recipientId) => {
  try {
    const res = await fetch(BASE_URL + "api/messages/" + recipientId, {
      method: "POST",
      headers: {
        Accept: "application/json",
        "Content-Type": "application/json",
        "x-access-token": user.token,
      },
      body: JSON.stringify(message),
    });
    return await res.json();
  } catch (err) {
    console.log(err);
  }
};

export { getConversations, getMessages, sendMessage };
```

CHAPTER 5

RESULTS



Devika · a day ago

RGUKT TECH FEST

Welcome to TECH FEST -2K24

All posts have been viewed

lathamam

Bio: WELCOME

Likes **0** Posts **0**

lathamam

LathaMam

→ Thus this CAMPUS CONNECTION platform enables users to Communicate and interconnect with other.
→ This helps in faster communication and enables users to share their thoughts and their views on the particular posted post.

CHAPTER 6
CONCLUSION

In conclusion, Campus Connection has successfully addressed the need for a unified platform that enhances communication and collaboration within our campus community. By integrating features such as [list key features], the platform promotes inclusivity, efficiency, and engagement among students, faculty, and staff alike. Moving forward, ongoing user feedback and iterative improvements will be essential to ensure Campus Connection remains a vital tool for enhancing campus life and fostering a sense of belonging and connectivity among all stakeholders.

Campus Connection represents a pivotal advancement in enhancing the interconnectedness and communication within our campus community. By providing a centralized platform for students, faculty, and staff to share information, collaborate on projects, and engage in meaningful discussions, Campus Connection fosters a vibrant and inclusive campus environment. Through its user-friendly interface and robust features such as [mention specific features], the platform promotes efficiency, transparency, and a sense of belonging among all stakeholders. As we continue to refine and expand Campus Connection based on user feedback and technological advancements, we are confident that it will remain a cornerstone of our campus culture, empowering individuals and groups to thrive academically, socially, and professionally.

This conclusion highlights the platform's benefits, its impact on campus life, and its potential for future development. Adjust the specific features and benefits based on the details of your project.