



RAJIV GANDHI
INSTITUTE OF TECHNOLOGY

EDUCONNECT: A STUDENT COMMUNITY PLATFORM

PRESENTED BY :

RAVI DINNI – 1RG23CS051

SOPNIL PATIL – 1RG23CS069

SAMPATAKUMAR V – 1RG23CS056

MAHESH – 1RG23CS029

SUB CODE :- BCS586

UNDER THE GUIDANCE OF PROF. SONIYA KOMAL V OF CSE

CONTENTS

- **INTRODUCTION**
- **EXISTING SYSTEMS**
- **DISADVANTAGES OF EXISTING SYSTEM**
- **PROPOSED SYSTEM**
- **ADVANTAGES OF PROPOSED SYSTEM**
- **SYSTEM ANALYSIS**
- **DESIGN**
- **GUI LAYOUTS**
- **IMPLEMENTATION**
- **DEMO**
- **CONCLUSION**
- **REFERENCE**

INTRODUCTION

Educonnect is a responsive web application designed to connect students across colleges and branches. It enables academic collaboration through community groups, resource sharing, and real-time chat. The platform promotes inclusive learning and simplifies peer-to-peer interaction.

EXISTING SYSTEMS

- ◆ **Whatsapp communities**

- Used for informal student groups and sharing notes.



- ◆ **Vtu connect**

- Provides VTU results, syllabus, and notifications.



- ◆ **Vtu circle**

- Offers vtu-related updates and study materials.



- ◆ **Telegram groups**

- Used for sharing pdfs and updates.



- ◆ **Google drive sharing**

- Students share folders of notes and assignments.



DISADVANTAGES OF EXISTING SYSTEMS

-  No student directory or profile search
-  No personalized dashboard or login
-  Unorganized file sharing
-  No real-time chat or collaboration
-  Privacy concerns (e.g., exposed phone numbers)
-  No academic structure or subject tagging

PROPOSED SYSTEM – EDUCONNECT

EduConnect is a responsive web application built using React that enables students to connect, collaborate, and share academic resources across colleges and branches.

- ◆ **Core Modules:**

- Login/Register with JWT
- Dashboard with community feed
- Upload notes with subject tags
- Real-time chat (group/private)
- Searchable student directory

ADVANTAGES OF PROPOSED SYSTEM

- ✓ Academic-first design
- ✓ Secure login and personalized dashboard
- ✓ Structured resource sharing with filters
- ✓ Real-time communication
- ✓ Student directory with search and filters
- ✓ Responsive UI for desktop and android

SYSTEM ANALYSIS

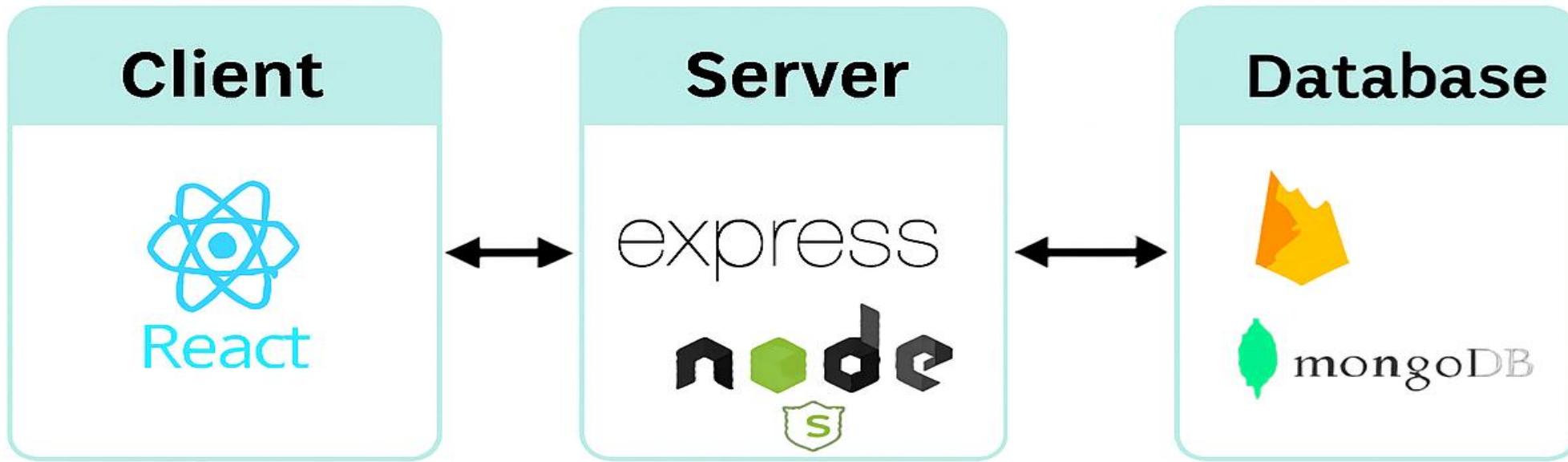
Functional Requirements:

- User Registration
- Authentication
- Admin Login
- User Login

Non-functional Requirements:

- Performance
- Security
- Reliability & Availability
- Usability
- Maintainability
- Interoperability
- Legal & Ethical

DESIGN



React
(Frontend)

Express +
Node.js
(Backend)

Firebase +
MongoDB
(Database)

GUI LAYOUT



Login

Email

Password

[Login](#)

[Forgot Password?](#)

---OR---

[Login with Google](#)

[Don't have an account? Register here](#)

Register

First Name

Last Name

USN

College

Branch

Semester

Email

Password

Confirm Password

[Register](#)

---OR---

[Register with Google](#)

[Already have an account? Login here](#)

StudentConnect  [Logout](#)

Home Resources Chatroom

Hey, Guest

New User 
USN: N/A
Add your bio here!

College:
Branch:
Course:
Semester:

[Edit Profile](#)

Your Uploaded Notes (0)
You haven't uploaded any notes yet. Be the first!
[Upload New Note](#)

IMPLEMENTATIONS

TECHNOLOGIES USED:

- **Frontend (React)**
- **Authentication (FireBase)**
- **Backend (Express+Node.js)**
- **DataBase (Mongo DB)**
- **Real Time Chat (Socket.io)**

TESTING

- Unit testing (API endpoints)
- Integration testing (frontend ↔ backend)
- User testing (login, upload, chat)
- Error handling (invalid inputs, unauthorized access)

DEMO

STEPS :

- Welcome Page
- Login page
- Registration Page
- Dashboard
 - 1.Resources
 - 2.Chatroom
 - 3.Coding platform
 - 4.Notifications

CONCLUSION

The student community platform in react provides an efficient solution for enhancing communication among students. It overcomes the limitations of existing systems by offering a centralized, interactive, and user-friendly platform. This project demonstrates the potential of react.Js in building scalable community platforms and can be further expanded into a full-fledged social networking application for educational institutions.

REFERENCES

1. *Dan Abramov and Andrew Clark, React Documentation, Meta Platforms, Inc.*
<https://react.dev/>
2. *Mozilla Developer Network (MDN), JavaScript Documentation,*
<https://developer.mozilla.org/>
3. *W3Schools, React Tutorial,* <https://www.w3schools.com/react/>
4. *GeeksforGeeks, React.js Articles and Tutorials,*
<https://www.geeksforgeeks.org/reactjs/>
5. *FreeCodeCamp, Building Modern Web Apps with React,*
<https://www.freecodecamp.org/>
6. *Firebase Documentation, Realtime Database and Authentication,*
<https://firebase.google.com/docs>
7. *Robin Wieruch, The Road to React, Leanpub, 2021.*
8. *Stack Overflow, Community Discussions on React Development,*
<https://stackoverflow.com/questions/tagged/reactjs>

THANK YOU