(CUBE)

Submitted in partial fulfilment of the requirements of the degree of

BACHELOR OF COMPUTER ENGINEERING

by

(Kunj Patel, 20102149)

(Prince Pal, 20102147)

(Avadh Pandey,20102171)

(Saket Nigam, 20102122)

Guide:

(Prof. Deepak Kachane)

Department of Computer

Engineering

A. P. SHAH INSTITUTE OF TECHNOLOGY, THANE(W)
Mumbai-400615
UNIVERSITY OF MUMBAI

(2022-2023)

A. P. SHAH INSTITUTE OF TECHNOLOGY, THANE

CERTIFICATE

This is to certify that the project entitled "CUBE" is a bonafide work of **"Kunj Patel" (20102149), "Prince Pal"(20102149), "Avadh Pandey"(20102171), "Saket Nigam"(20102122)** submitted to the University of Mumbai in fulfilment of the requirement for the Mini Project 2A of **Bachelor of Engineering** in **Computer Engineering**

_____

Prof.Deepak Kachane

_____ _____

Prof. Sachin Malave

Head of Department

Principal

A. P. SHAH INSTITUTE OF TECHNOLOGY, THANE

Project Report Approval for Mini Project-2A

This project report entitled ***(CUBE)*** by ***Kunj Patel(20102149), Prince Pal(20102147), Avadh Pandey(20102171), Saket Nigam(20102122)*** is approved for the degree of ***Bachelor of Engineering*** in ***Computer Engineering***, ***2022-23***.

Examiner Name	Signature
1._____	_____
2._____	_____

Date:

Place:

Declaration

We declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Kunj Patel (20102067)

Prince Pal (20102147)

Avadh Pandey (20102171)

Saket Nigam (20102122)

Date:

Abstract

As we are living in 21st Century, the internet has created a lot of opportunities that weren't there before. One of these is communication through various types of chat application for instant communication with people in a web-based domain. All of these works well for simple two-way communication where users can simply send text, pictures, audio, etc for each other. But the chat application we intend to do will have an admin administration who will have the authority to choose among the members whether they have access to the particular veins or not, so only one admin is needed irrespective of the member of subgroups.

CONTENTS

1. Introduction

We have used different types of chat application, all these chat application support instant messaging services. But as soon as the number of contacts and groups increases in the application it becomes difficult for the user to navigate over the chats. A teacher/professor from an educational institute teaches multiple classes and hence for better communication with the students they create chat groups for individual classes and thus has to manage multiple chat groups so the student after each passing year encounters a different class thus a new group. The notices of exams, events, study materials, timetables, forms, etc these messages are quite frequent and at times the important ones are lost in this clutter of different messages and its difficult to find them via scrolling through those useless messages. This is where cubes comes into picture, which helps to declutter these messages by adding a separate space for each genre of messages so its easier to navigate to the messages we require at that time without the need to scroll through those endless messages which we do not require at the particular moment. The application will be more effective by modelling it after a tree with branches & sub-branches which helps the user to navigate even more effectively.

Problem Statement

Communication between individuals or groups has grown from Emails to chat rooms and chat applications, but the number of contacts and groups a person has also increased significantly. A teacher from an educational institute teaches multiple classes and subjects and for better communication with students, they create chat groups for each individual class and thus has to manage multiple chat groups and student after each passing year encounters a different class thus a new group. The notices of exams, events or holidays, pdfs of study material timetables, forms, doubts of students etc. these msgs are quite frequent and at times the important ones are lost in this clutter of different messages and its difficult to find them via scrolling through those useless messages. There are also similar problems with other types of groups as well not just educational groups

Scope:

-Sub-group for each genre of the message.
-Ability to send real-time messages over the internet.
-Ability to send pictures, videos and documents.
-Assigning roles so that only a certain few with special permission can access several veins.
-Each student is an ID which is unique for everyone and it’s derived from their admission ID

Objective:

-The objective of Cube is to create a web-based chat application which helps to declutter these messages by adding a separate space for each genre of messages so its easier to navigate to the messages we require at that time without the need to scroll through those endless messages which we do not require at the moment.

-To make the existing system of chat application more effective by modelling it after a tree with branches and sub-branches which will help the user to easily navigate through different types of messages.

-To urge the user to make subgroups according to their need and assign a name befitting the task which it will serve and use it for its messages of the same genre

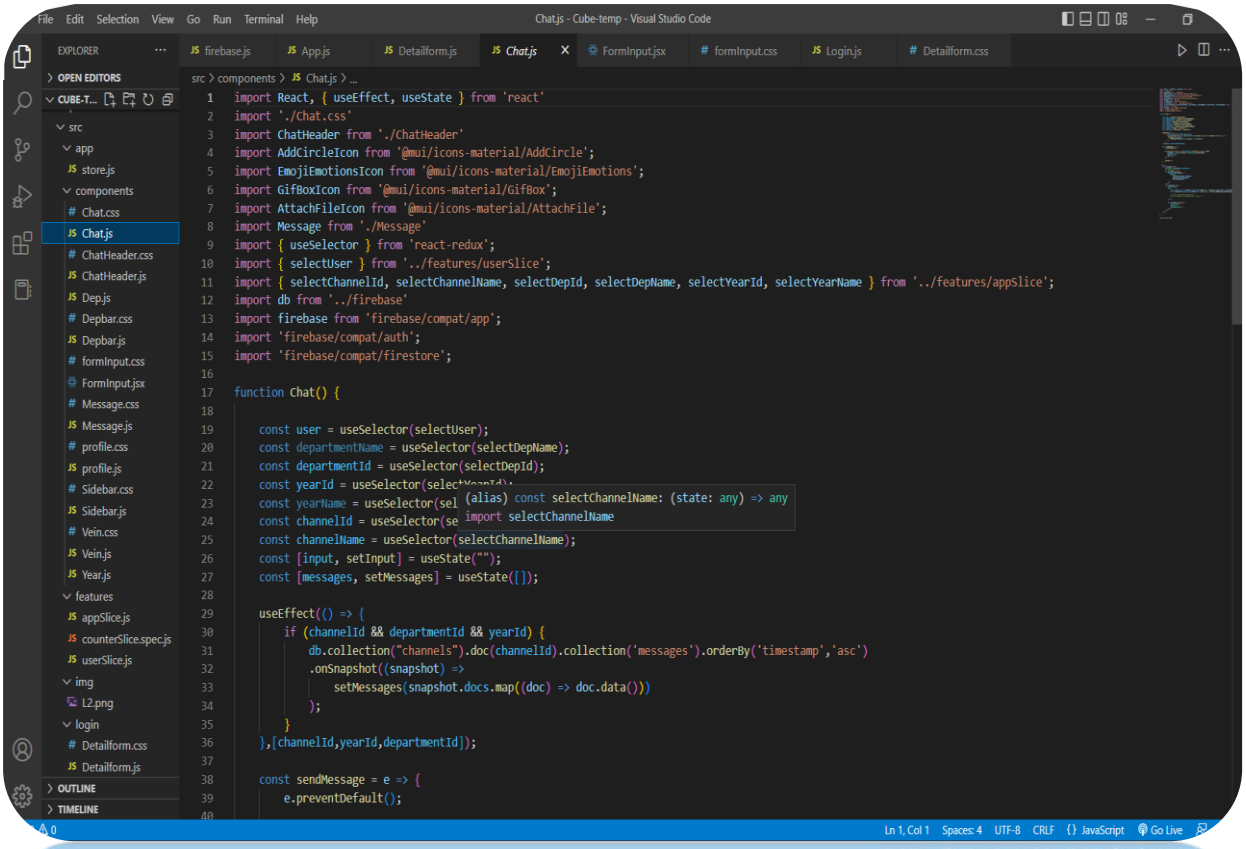
Chapter 3

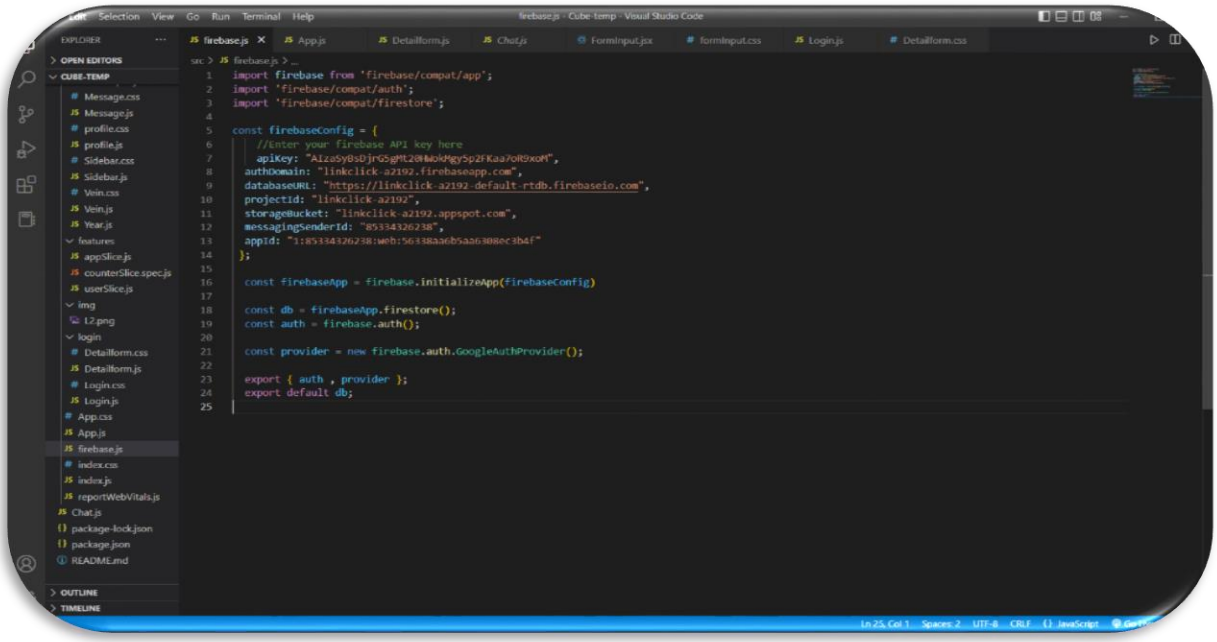
Cloud Platform

The CUBE project is hosted on the AWS cloud platform. AWS is a popular cloud computing platform that provides a wide range of services, including hosting, storage, and computing resources. The use of AWS allows the project to be accessed from anywhere with an internet connection, making it accessible to a wide range of users.
The AWS platform provides a scalable and reliable hosting solution for the project. This means that the project can handle large numbers of users and traffic without experiencing performance issues or downtime. AWS also provides security features to protect the project from cyber threats and data breaches.

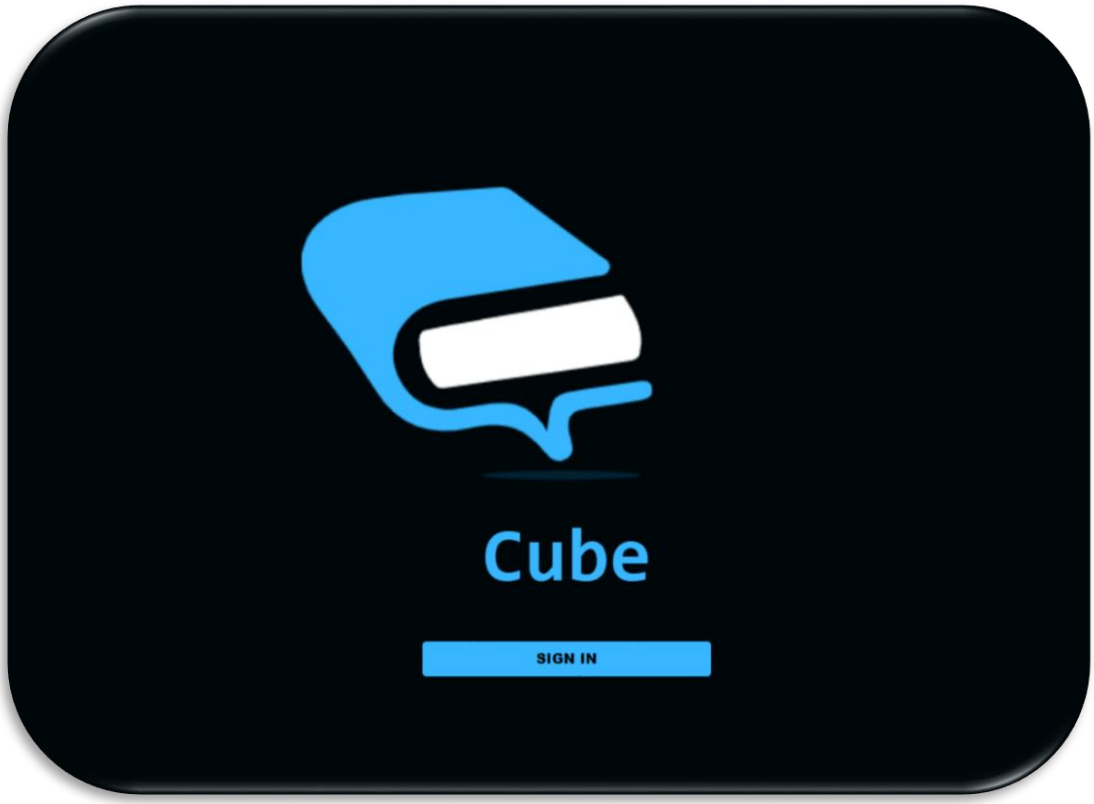
The CUBE project is hosted on the AWS cloud platform, which provides a scalable and reliable hosting solution for the project. The platform uses a range of AWS services to ensure that the project is secure, performant, and available to users.

Amazon S3 (Simple Storage Service) is a highly scalable object storage service provided by Amazon Web Services (AWS). It allows users to store and retrieve data from virtually anywhere on the web. With Amazon S3, users can easily and securely store and access any amount of data in a highly available and durable manner. S3 buckets provide a simple and cost-effective way to store and share data across different applications and users, and can be used for a variety of use cases including backup and restore, big data analytics, and content distribution. S3 also supports a range of data management features, such as versioning, lifecycle policies, and encryption, to ensure the integrity and security of the data stored in the bucket.


File Edit Selection View Go Run Terminal Help
Chat.js - Cube-temp - Visual Studio Code
EXPLORER
OPEN EDITORS
CUBE-T...
src
app
store.js
components
Chat.css
Chat.js
ChatHeader.css
ChatHeader.js
Dep.js
Depbar.css
Depbar.js
formInput.css
FormInput.jsx
Message.css
Message.js
profile.css
profile.js
Sidebar.css
Sidebar.js
Vein.css
Vein.js
Year.js
features
appSlice.js
counterSlice.spec.js
userSlice.js
img
L2.png
login
Detailform.css
Detailform.js
OUTLINE
TIMELINE
firebase.js
App.js
Detailform.js
Chat.js
FormInput.jsx
formInput.css
Login.js
Detailform.css
src > components > Chat.js > ...
import React, { useEffect, useState } from 'react'
import './Chat.css'
import ChatHeader from './ChatHeader'
import AddCircleIcon from '@mui/icons-material/AddCircle';
import EmojiEmotionsIcon from '@mui/icons-material/EmojiEmotions';
import GifBoxIcon from '@mui/icons-material/GifBox';
import AttachFileIcon from '@mui/icons-material/AttachFile';
import Message from './Message'
import { useSelector } from 'react-redux';
import { selectUser } from '../features/userSlice';
import { selectChannelId, selectChannelName, selectDepId, selectDepName, selectYearId, selectYearName } from '../features/appSlice';
import db from '../firebase'
import firebase from 'firebase/compat/app';
import 'firebase/compat/auth';
import 'firebase/compat/firestore';
function Chat() {
const user = useSelector(selectUser);
const departmentName = useSelector(selectDepName);
const departmentId = useSelector(selectDepId);
const yearId = useSelector(selec
const yearName = useSelector(sel
(alias) const selectChannelName: (state: any) => any
import selectChannelName
const channelId = useSelector(se
const channelName = useSelector(selectChannelName);
const [input, setInput] = useState("");
const [messages, setMessages] = useState([]);
useEffect(() => {
if (channelId && departmentId && yearId) {
db.collection("channels").doc(channelId).collection('messages').orderBy('timestamp','asc')
.onSnapshot((snapshot) =>
setMessages(snapshot.docs.map((doc) => doc.data()))
);
}
},[channelId,yearId,departmentId]);
const sendMessage = e => {
e.preventDefault();
Ln 1, Col 1
Spaces: 4
UTF-8
CRLF
{} JavaScript
Go Live


firebase.js - Cube-temp - Visual Studio Code
CUBE-TEMP
import firebase from 'firebase/compat/app';
import 'firebase/compat/auth';
import 'firebase/compat/firestore';
const firebaseConfig = {
//Enter your firebase API key here
apiKey: "AIzaSyBsDjrG5gMt20MokMgySp2FKaa7oR9xoM",
authDomain: "linkclick-a2192.firebaseapp.com",
databaseURL: "https://linkclick-a2192-default-rtdb.firebaseio.com",
projectId: "linkclick-a2192",
storageBucket: "linkclick-a2192.appspot.com",
messagingSenderId: "85334326238",
appId: "1:85334326238:web:5638aa6b5aa6308ec3b4f"
};
const firebaseApp = firebase.initializeApp(firebaseConfig)
const db = firebaseApp.firestore();
const auth = firebase.auth();
const provider = new firebase.auth.GoogleAuthProvider();
export { auth , provider };
export default db;

5 Result


Cube
SIGN IN


TE
BE
FE
SE
TE
EXAM TIME TABLE
SPORTS
ASSIGNMENTS
COURSES
AVADH PANDEY
SE
AVADH PANDEY
TCS
COMPUTER
AVADH PANDEY
Message in ASSIGNMENTS

Chapter 5

Conclusion

The CUBE is an innovative web-based application that provides a user-friendly platform for sharing information online. The project is built using React with a Firebase database and is hosted on the AWS cloud platform. The use of AWS services such as S3 bucket and Route53 ensures that the project is secure, performant, and available to users. The project provides a range of features that make it easy for users to create, manage, and share notices, making it a great tool for information sharing in today's digital age

Chapter 6

Demonstration and Code Link

https://drive.google.com/drive/folders/1rXQPz-vYIssyuTCljL8gXET_0jvBci5q?usp=share_link

https://github.com/Kunj-Pate1/Cube