Firebase Authentication:

After creating project, in firebase console sidebar, click on authentication

Under sign in method, select gmail or any of needed and click on enable and save

In firebase.js:

Const auth = firebase.auth()

ADDIND DATA TO FIRESTORE:

     //Adding data to firestore

      db.collection('emails').add({

          to:FormData.to,

          subject: FormData.subject,

          message: FormData.message,

          timestamp: firebase.firestore.FieldValue.serverTimestamp()

      })

GETTING DATA:

    //QUERING DATA

    useEffect(() => {

         db.collection('emails').orderBy('timestamp', 'desc').onSnapshot(snapshot=> setEmails(snapshot.docs.map((doc) => ({

             id: doc.id,

             data: doc.data,

         }))))

    }, [])

 {emails.map(({id, data: {to, subject, message, timestamp}}) => (

                <EmailRow key={id} title={to} subject={subject} description={message} time={new Date(timestamp?.seconds \* 1000).toUTCString()}/>

AUTHENTICATION:

Create userslice.js file in features folder using redux

Npx create-react-app <appname> --template redux

Paste this code their

import { createSlice } from '@reduxjs/toolkit';

export const userSlice = createSlice({

  name: 'user',

  initialState: {

    user:null,

  },

  reducers: {

    login: (state, action) => {

      state.user = action.payload;

    },

    logout: (state) => {

      state.user = null;

    },

  }

})

export const { login, logout } = userSlice.actions;

export const selectUser = (state) => state.user.user;

export default userSlice.reducer;

update store.js by using userReducer their. Like this:

import { configureStore } from '@reduxjs/toolkit';

import mailReducer from '../features/mailSlice';

import userReducer from '../features/userSlice'

export default configureStore({

  reducer: {

    mail: mailReducer,

    user: userReducer,

  },

});

Now go to app.js

const user = useSelector(selectUser)

and wrap whole app around this user, like if user not logged in, then show him login page otherwise whole app

function App() {

  //Authentication

import React, { useEffect } from 'react';

import './App.css';

import Header from './components/Header';

import Sidebar from './components/Sidebar';

import {BrowserRouter as Router, Switch, Route, Link} from 'react-router-dom';

import Mail from './components/Mail';

import EmailList from './components/EmailList';

import SendMail from './components/SendMail';

import { useDispatch, useSelector } from 'react-redux';

import { selectSendMessageIsOpen } from './features/mailSlice'

import { login, selectUser } from './features/userSlice';

import Login from './components/Login';

import { auth } from './firebase';

function App() {

  //Authentication

  const user = useSelector(selectUser);

  const dispatch = useDispatch();

  useEffect(()=>{

    auth.onAuthStateChanged(user => {

      //if user is logged in

      if (user){

        dispatch(login({

          displayName:user.displayName,

          email:user.email,

          photoUrl:user.photoURL,

        }))

      }

    })

  }, [])

  //Importing sendmessage state from redux using selector and passing const value to it which is by default false

  const sendMessageIsOpen = useSelector(selectSendMessageIsOpen);

  return (

    <Router>

      {!user ? (

      <Login/>

      ):(

      <div className="app">

        <Header />

        <div className="app\_\_body">

          <Sidebar />

          <Switch>

            <Route path="/mail">

              <Mail/>

            </Route>

            <Route path="/">

              <EmailList/>

            </Route>

          </Switch>

        </div>

        {/\*if selectsendmessageisopen is true then compose part open otherwise not\*/}

        {sendMessageIsOpen && <SendMail/>}

      </div>

      )}

    </Router>

  );

}

export default App;

Then for login, create Login component as we have used above and us auth in it from firebase when use click on login

import { Button } from '@material-ui/core'

import React from 'react'

import { useDispatch } from 'react-redux'

import { login } from '../features/userSlice'

import { auth, provider } from '../firebase'

import './Login.css'

function Login() {

    const dispatch = useDispatch();

    const signIn = () => (

        auth.signInWithPopup(provider)

        .then(({user}) => {

            dispatch(

                login({

                    displayName:user.displayName,

                    email:user.email,

                    photoUrl:user.photoURL,

                })

            )

        })

        .catch(error => alert(error.message))

    )

    return (

        <div className="login">

            <div className="login\_\_container">

                <img src="https://static.dezeen.com/uploads/2020/10/gmail-google-logo-rebrand-workspace-design\_dezeen\_2364\_col\_0.jpg" alt=""/>

                <Button variant="contained" color="primary" onClick={signIn}>Login</Button>

            </div>

        </div>

    )

}

export default Login

Now to make signout using avatar and show img on avatar, we do changes in header.js

const user = useSelector(selectUser);

    const dispatch = useDispatch();

    const signOut = () => {

        auth.signOut().then(()=>{

            dispatch(logout())

        })

    };

 <Avatar onClick={signOut} src={user?.photoUrl}/>