Creating Project:

Npx create-react-app linkedin –template redux

Dependencies:

Yarn add @material-ui/core

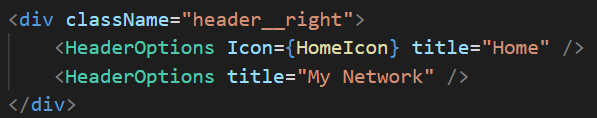
Yarn add @material-ui/icons

Yarn add [firebase@^8.10.0](mailto:firebase@%5e8.10.0)

Yarn add react-flip-move (to animate post)

To pass Icon as props or component:

Make Icon (i) capital as we are passing material-ui Icon



import React from 'react'

import './HeaderOptions.css';

function HeaderOptions({Icon, title}) {

    return (

        <div className="headerOption">

            {Icon && <Icon className="headerOption\_\_icon"/>}

            <h3>{title}</h3>

        </div>

    )

}

export default HeaderOptions

align items work with flexbox, flex direction row or colum

.headerOption{

    display: flex;

    flex-direction: column;

    align-items: center;

    margin-right: 20px;

    color: gray;

    cursor: pointer;

}

Function inside function declare and calling

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

import React from 'react'

import './Sidebar.css'

function Sidebar() {

    const recentItem = (topic) => (

        <div className="sidebar\_\_recentItem">

            <span className="sidebar\_\_hash">#</span>

            <p>{topic}</p>

        </div>

    )

    return (

        <div className="sidebar">

            <div className="sidebar\_\_bottom">

                <p>Recent</p>

                {recentItem('reactjs')}

                {recentItem('programming')}

                {recentItem('fullstack')}

                {recentItem('design')}

                {recentItem('developer')}

            </div>

        </div>

    )

}

export default Sidebar

To limit text with in area or div or body:

Used overflow-wrap: anywhere;

            <div className="post\_\_body">

                <p>{message}</p>

            </div>

.post\_\_body{

    overflow-wrap: anywhere;

}

Firebase for authentication and db:

import firebase from 'firebase';

const firebaseConfig = {

  apiKey: "AIzaSyDUdfLCHFAHvs9sc5h7O-jJfJ0P8n1U5BY",

  authDomain: "linkedin-311e5.firebaseapp.com",

  projectId: "linkedin-311e5",

  storageBucket: "linkedin-311e5.appspot.com",

  messagingSenderId: "650316104831",

  appId: "1:650316104831:web:55793589eabf36d1b16b4a"

};

// Initialize Firebase

const firebaseApp = firebase.initializeApp(firebaseConfig);

const db = firebaseApp.firestore();

const auth = firebase.auth();

export {db, auth};

Firebase update and collection:

Adding data to collection

    const sendPost = (e) => {

        e.preventDefault()

        db.collection('posts').add({

            name: "Muneeb",

            description: "test",

            message: input,

            photourl: "",

            timestamp: firebase.firestore.FieldValue.serverTimestamp(),

        })

    }

Getting and updating data

    useEffect(()=>{

        db.collection('posts').onSnapshot(snapshot=> {

            setPosts(snapshot.docs.map(doc=>(

                {

                    id: doc.id,

                    data: doc.data()

                }

            )))

        })

    }, [])

Now using data from collection

            {posts.map( ({ id, data: { name, description, message, photourl }} )=>{

                <Post

                    key= {id}

                    name={name}

                    description={description}

                    message={message}

                    photourl={photourl}

                />

            })}

Redux:

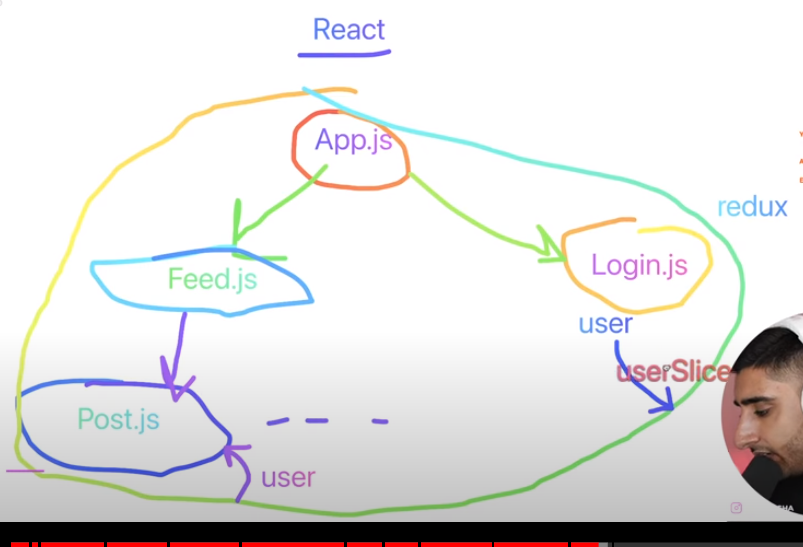


Image css:

.login > img {

    object-fit: contain;

    height: 70px;

    margin-top: 20px;

    margin-bottom: 20px;

}

Signup or Register PAGE:

Repo URL: https://github.com/MUHAMMAD-MUNEEB-WASEEM/linkedin-Social-media.git

REDUX:

Store.js

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

import userReducer from './userSlice';

export const store = configureStore({

  reducer: {

    user: userReducer,

  },

});

userSlice.js

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 selectCount = (state) => state.user.user;

export default userSlice.reducer;

firebase.js:

import firebase from 'firebase';

const firebaseConfig = {

    apiKey: "AIzaSyDUdfLCHFAHvs9sc5h7O-jJfJ0P8n1U5BY",

    authDomain: "linkedin-311e5.firebaseapp.com",

    projectId: "linkedin-311e5",

    storageBucket: "linkedin-311e5.appspot.com",

    messagingSenderId: "650316104831",

    appId: "1:650316104831:web:55793589eabf36d1b16b4a"

  };

const firebaseApp = firebase.initializeApp(firebaseConfig);

const db = firebaseApp.firestore();

const auth = firebase.auth();

export {auth, db}

App.js

import React, { useEffect } from 'react';

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

import './App.css';

import Feed from './components/Feed/Feed';

import Header from './components/Header/Header';

import Login from './components/Login/Login';

import Sidebar from './components/Sidebar/Sidebar';

import { auth } from './firebase';

import { login, logout } from './redux/userSlice';

function App() {

  const dispatch = useDispatch()

  const user = useSelector((state) => state.user.user)

  //to maintain login functionality on page refresh

  useEffect(()=>{

    auth.onAuthStateChanged(userAuth=>{

      if (userAuth){

        //user logged in

        dispatch(login({

          email: userAuth.email,

          uid: userAuth.uid,

          displayName: userAuth.displayName,

          photourl: userAuth.photoURL

        }))

      } else {

        //user is logged out

        dispatch(logout());

      }

    })

  },[])

  return (

    <div className="app">

      <Header/>

      {/\*if user is not present, then show login page8\*/}

      {!user ? (

        <Login/>

      ):(

      <div className="app\_\_body">

        <Sidebar/>

        <Feed/>

      </div>

      )}

    </div>

  );

}

export default App;

Login component:

Login.js

import React, {useState} from 'react'

import { useDispatch } from 'react-redux'

import { auth } from '../../firebase'

import { login } from '../../redux/userSlice';

import './Login.css'

function Login() {

    const dispatch = useDispatch();

     const [email, setEmail] = useState('')

     const [password, setPassword] = useState('')

     const [name, setName] = useState('')

     const [profilePic, setProfilePic] = useState('')

    const register = (e) => {

        if (!name){

            return alert('Please enter a full name')

        }

        auth.createUserWithEmailAndPassword(email, password)

            .then((userAuth)=>{

                userAuth.user.updateProfile({

                    displayName: name,

                    photoURL: profilePic

                })

                .then(()=>{

                    dispatch(login({

                        email: userAuth.user.email,

                        uid: userAuth.user.uid,

                        displayName: name,

                        photourl: profilePic

                    }))

                })

            })

            .catch(error =>{

                alert(error)

            })

    }

    //values taken from redux stored from signup

    const loginToApp = (e) => {

        e.preventDefault();

        auth.signInWithEmailAndPassword(email, password)

            .then(userAuth => {

                dispatch(login({

                    email: userAuth.user.email,

                    uid: userAuth.user.uid,

                    displayName: userAuth.user.displayName,

                    photourl: userAuth.user.photoURL

                }))

            }).catch(error=>{

                alert(error)

            })

    }

    return (

        <div className="login">

            <img src="https://www.seekpng.com/png/full/802-8024287\_advertise-on-linkedin-transparent-linkedin-logo-vector.png" alt="login"/>

            <form>

                <input value={name} onChange={(e)=>setName(e.target.value)} type="text" placeholder="Full name (required if registering)"/>

                <input value={profilePic} onChange={(e)=>setProfilePic(e.target.value)} type="text" placeholder="Profile pic URL (optional)"/>

                <input value={email} onChange={(e)=>setEmail(e.target.value)} type="email" placeholder="Email"/>

                <input value={password} onChange={(e)=>setPassword(e.target.value)} type="password" placeholder="Password"/>

                <button type="submit" onClick={loginToApp}>Sign In</button>

            </form>

            <p>Not a member?{" "}

                <span className="login\_\_register" onClick={register}>Register Now</span>

            </p>

        </div>

    )

}

export default Login

We can find all user in firebase authentication users section

Login.css:

.login{

    display: grid;

    place-items: center;

    padding-top: 10px;

    padding-bottom: 100px;

    margin-left: auto;

    margin-right: auto;

}

.login > img {

    object-fit: contain;

    height: 70px;

    margin-top: 20px;

    margin-bottom: 20px;

}

.login > form {

    display: flex;

    flex-direction: column;

}

.login > form > input {

    width: 350px;

    height: 50px;

    font-size: 20px;

    padding-left: 10px;

    margin-bottom: 10px;

    border-radius: 5px;

}

.login > form > button {

    width: 365px;

    height: 50px;

    font-size: large;

    background-color: #0074b1;

    border-radius: 5px;

    color: #ffffff;

}

.login\_\_register {

    color: #0177b7;

    cursor: pointer;

}

For logout:

import React from 'react'

import './Header.css'

import logo from '../images/linkedin.png'

//material ui

import SearchIcon from '@material-ui/icons/Search';

import HeaderOptions from './HeaderOptions/HeaderOptions';

import HomeIcon from '@material-ui/icons/Home';

import SupervisorAccountIcon from '@material-ui/icons/SupervisorAccount';

import BusinessCenterIcon from '@material-ui/icons/BusinessCenter';

import ChatIcon from '@material-ui/icons/Chat';

import NotificationsIcon from '@material-ui/icons/Notifications';

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

import { logout } from '../../redux/userSlice';

import { auth } from '../../firebase';

function Header() {

    const dispatch = useDispatch();

    const logoutToApp = ()=>{

        dispatch(logout())

        auth.signOut()

    }

    return (

        <div className="header">

            <div className="header\_\_left">

                <img src={logo} alt=""/>

                <div className="header\_\_search">

                    <SearchIcon/>

                    <input placeholder="Search" type="text"/>

                </div>

            </div>

            <div className="header\_\_right">

                <HeaderOptions Icon={HomeIcon} title="Home" />

                <HeaderOptions Icon={SupervisorAccountIcon} title="My Network" />

                <HeaderOptions Icon={BusinessCenterIcon} title="Jobs" />

                <HeaderOptions Icon={ChatIcon} title="Messaging" />

                <HeaderOptions Icon={NotificationsIcon} title="Notification" />

                <HeaderOptions

                    avatar={true}

                    onClick = {logoutToApp}

                    title="me"/>

            </div>

        </div>

    )

}

export default Header

HeaderOptions.js:

import React from 'react'

import './HeaderOptions.css';

import { Avatar } from '@material-ui/core';

import { useSelector } from 'react-redux';

function HeaderOptions({Icon, title, avatar, onClick}) {

    const user = useSelector(state => state.user.user);

    return (

        <div onClick={onClick} className="headerOption">

            {Icon && <Icon className="headerOption\_\_icon"/>}

            {avatar &&

                (user?.photourl ?

                 (<Avatar className="headerOption\_\_icon" src={user?.photourl}/>)

                 :

                 (<Avatar className="headerOption\_\_icon" >{user?.email[0]}</Avatar>)

                 )}

            <h3 className="headerOption\_\_title">{title}</h3>

        </div>

    )

}

export default HeaderOptions

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Login and logout part ends here \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

User true false parameters, avatar, image:

First method

<HeaderOptions

                    avatar={true}

                    onClick = {logoutToApp}

                    title="me"/>

import React from 'react'

import './HeaderOptions.css';

import { Avatar } from '@material-ui/core';

import { useSelector } from 'react-redux';

function HeaderOptions({Icon, title, avatar, onClick}) {

    const user = useSelector(state => state.user.user);

    return (

        <div onClick={onClick} className="headerOption">

            {Icon && <Icon className="headerOption\_\_icon"/>}

            {avatar &&

                (user?.photourl ?

                 (<Avatar className="headerOption\_\_icon" src={user?.photourl}/>)

                 :

                 (<Avatar className="headerOption\_\_icon" >{user?.email[0]}</Avatar>)

                 )}

            <h3 className="headerOption\_\_title">{title}</h3>

        </div>

    )

}

export default HeaderOptions

Another Example:

           <div className="sidebar\_\_top">

                <img src="https://images.unsplash.com/photo-1560345573-9f453083c335?ixlib=rb-1.2.1&ixid=MXwxMjA3fDB8MHxleHBsb3JlLWZlZWR8NjR8fHxlbnwwfHx8&auto=format&fit=crop&w=500&q=60" alt=""/>

                {user?.photourl ?

                 (<Avatar className="headerOption\_\_icon" src={user?.photourl}/>)

                 :

                 (<Avatar className="headerOption\_\_icon" >{user?.email[0]}</Avatar>)

                 }

                <h2>{user.displayName}</h2>

                <h4>{user.email}</h4>

            </div>

Second Method:

 <Avatar src={photourl} alt="photo">{name[0]}</Avatar>

If photourl exist then show that other wise show first letter of name