BACKEND -> index.js

const express = require('express');

const cors = require('cors');

const cookieParser = require('cookie-parser');

const connectDb = require('./config/db')

require('dotenv').config()

  const userAuthRoute = require('./routes/userAuthRoute')

const app = express();

app.use(express.json());

app.use(cookieParser());

const corsOptions = {

    origin:process.env.FRONTEND\_URL,

    credentials:true,

}

app.use(cors(corsOptions))

connectDb();

 app.use('/userAuth', userAuthRoute);

const PORT = process.env.PORT || 8000;

app.listen(PORT, ()=>console.log(`App is running on ${PORT}` ));

BACKEND -> routes -> userAuthRoute.js

const {signUp, signIn, signOut ,getUsers ,getUserByUserId} = require('../controllers/userAuthController')

const express = require('express');

const router = express.Router();

router.post('/signUp', signUp);

router.post('/signIn',signIn);

router.get('/signOut',signOut);

router.get('/getUsers',getUsers )

// delete 10 1

router.post('/getUserById/user/:userName',getUserByUserId)

// delete 10 1

module.exports = router;

BACKEND -> controllers -> userAuthController.js

const getUserByUserId = async(req, res) => {

    const {userName} = req.params;

    console.log("username   : ", userName)

    try {

        if(!userName){

            res.json({message:'UserId is require to get user post'})

        }

        const user = await BaraiUser.findOne({username:userName})

        console.log("user  : ",user)

         user.friendsCount+=1;

         await user.save();

         res.json( {message:'Get user successfully', user})

    } catch (error) {

        console.log('error getting posts',error)

        res.json({message:"wrong"})

    }

}

FRONTEND -> service -> userService.js

import axiosInstance from "./url.service";

export const peopleData =async ()=>{

    const result =  await axiosInstance.get('/userAuth/getUsers') ;

     console.log(" axiosInstance  people : ", result.data.users)

    return result.data.users;

}

// delete 10 7

export const likePost = async(userName)=>{  // C

    try {

        const result = await axiosInstance.post(`/userAuth/getUserById/user/${userName}`)  // CONTROLLER getUserById

        return result?.data?.data;

    } catch (error) {

        console.error('failed in servic.js')

    }

}

FRONTEND -> store -> usersStore.js

import { peopleData } from '@/service/user.service'

import {create} from 'zustand'

import {likePost} from '@/service/user.service'

export const usersStore = create((set)=>({

    users:[],

    fetchUsers: async()=> {

        const users = await peopleData();

        set({users})

    },

    handleLikePost: async(userName)=>{

        try {

            await likePost(userName)

        } catch (error) {

            console.error('failed')

        }

     }

  })

)