CODEJUDGE SERVER API

App.js

var createError = require('http-errors');

var express = require('express');

var path = require('path');

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

var logger = require('morgan');

require("dotenv").config()

const mongoose=require("mongoose")

var indexRouter = require('./routes/index');

var usersRouter = require('./routes/users');

var app = express();

const db=process.env.MongoURI

if(db){

  mongoose.connect(db,{UseNewUrlParser:true}).then(()=>{

    console.log("connected successfully!")

  }).catch((err)=>{

    console.log(err)

    console.log("Something went wrong")

  })

}

else{

  console.log("no uri")

}

// view engine setup

app.set('views', path.join(\_\_dirname, 'views'));

app.set('view engine', 'jade');

app.use(logger('dev'));

app.use(express.json());

app.use(express.urlencoded({ extended: false }));

app.use(cookieParser());

app.use(express.static(path.join(\_\_dirname, 'public')));

app.use('/', indexRouter);

app.use('/', usersRouter);

// catch 404 and forward to error handler

app.use(function(req, res, next) {

  next(createError(404));

});

// error handler

app.use(function(err, req, res, next) {

  // set locals, only providing error in development

  res.locals.message = err.message;

  res.locals.error = req.app.get('env') === 'development' ? err : {};

  // render the error page

  res.status(err.status || 500);

  res.render('error');

});

module.exports = app;

userController

const Users= require("../models/Users")

const getUserByUSername=async(req,res,err)=>{

    try{

        const user=await Users.findOne({username:req.params.username})

        if(!user){

            return res.status(404).json({})

        }

        return res.status(200).json(user)

    }

    catch(err){

        return res.status(500).json({

            // success:false

        })

    }

}

const createUser=async(req,res,err)=>{

    try{

        const users =await Users.findOne({username:req.body.username})

        if(users){

            return res.status(400).json({

                "status":"failure",

                "reason":"user already exists"

            })

        }

        const response =await Users.create({username:req.body.username})

        console.log(response)

        if(response){

            return res.status(201).json({

                username:response.username

            })

        }

        else{

            return res.status(404).json({

                "status":"failure",

                "reason":"something went wrong"

            })

        }

    }

    catch(err){

        return res.status(404).json({

            "status":"failure",

            "reason":"something went wrong"

        })

    }

}

module.exports={

    getUserByUSername,

    createUser

}

Usermodel

const mongoose=require("mongoose")

const UserSchema=new mongoose.Schema({

    username:{

        type:String,

        required:true

    },

    followers:[String],

    following:[String],

    posts:[

        {

            postId:Number,

            imageUrl:String,

            caption:String,

            upvotes:Number

        }

    ],

})

const Users=mongoose.model("User",UserSchema)

module.exports=Users

router/user.js

var express = require('express');

const { getUserByUSername,createUser } = require('../controllers/userController');

var router = express.Router();

/\* GET users listing. \*/

router.get('/api/get-user/:username', getUserByUSername);

router.get('./api/create-user/',createUser)

module.exports = router;

.env

MongoURI="mongodb://localhost:27017/db"