'use strict';

// External Modules -----------------------------------------------------

const express = require('express');

const passport = require('passport');

const jwt = require('jsonwebtoken');

// My Modules -----------------------------------------------------------

const User = require('../database/users/user.model');

const Project = require('../database/projects/project.model');

const config = require('../config/database');

const emailNotifier = require('../services/email.service');

const dashboardMessage = require('../config/email').dashboardMessage;

const notificationMessage = require('../config/email').notificationMessage;

// Definitions ---------------------------------------------------------

const router = express.Router();

// APIs ----------------------------------------------------------------

// User Register -------------------------------------------------------

router.post('/register', (req, res, next) => {

User.addUser(req.body, (err, user) => {

if(err) {

res.json({success: false, msg: 'Falla al registrar usuario'});

}

else {

// Send emails

dashboardMessage.text = 'Nombre: '+req.body.name+'\rEmail: '+req.body.email;

emailNotifier.send(dashboardMessage, (err, res) => {

if(err)

console.log(err);

});

notificationMessage.to = req.body.email;

emailNotifier.send(notificationMessage, (err, res) => {

if(err)

console.log(err);

});

// Send response

res.json({success: true, msg: 'Usuario registrado correctamente'});

}

});

});

// User Login -----------------------------------------------------------

router.post('/authenticate', (req, res, next) => {

const email = req.body.email;

const password = req.body.password;

// Check if user exists

User.getUserByEmail(email, (err, user) => {

if(err)

throw err;

if(!user){

return res.json({success: false, msg: 'Usuario no encontrado'});

}

// Check if password matches

User.comparePassword(password, user.password, (err, isMatch) => {

if(err)

throw err;

if(isMatch){

const token = jwt.sign(user, config.secret, {

expiresIn: 86400 // 1 day

});

// Get projects of this user

Project.getProjects(user.\_id, (err, projects) => {

if(err)

throw err;

res.json({

success: true,

token: 'JWT '+token,

user: {

id: user.\_id,

name: user.name,

email: user.email

},

projects: projects

});

});

}

else

res.json({success: false, msg: 'Password incorrecto'});

});

});

});

// Search user for evaluation

router.post('/search', (req, res, next) => {

const email = req.body.email;

// Check if user exists

User.getUserByEmail(email, (err, user) => {

if(err)

throw err;

if(!user){

return res.json({success: false, msg: 'Usuario no encontrado'});

}

res.json({

success: true,

user: {

id: user.\_id,

name: user.name,

email: user.email

}

});

});

});

router.post('/getdata', (req, res, next) =>{

const id = req.body.\_id;

User.getUserById(id, (err, user) => {

if(err)

throw err;

if(!user){

return res.json({success: false, msg: 'Usuario no encontrado'});

}

res.json({

success: true,

user: {

id: user.\_id,

name: user.name,

email: user.email

}

});

});

});

router.get('/profile', passport.authenticate(

'jwt',

{session:false}),

(req, res, next) => res.json({user: req.user})

);

module.exports = router;