const session = require('express-session'); //complementos para mejorar el nivel de seguridad a partir de esta linea

const passport = require("passport");

const passportLocalMongoose = require("passport-local-mongoose");

const app = express();

app.use(express.static("public"));

app.set("view engine", "ejs");

app.use(bodyParser.urlencoded({ extended: true }));

app.use(session({

    secret: "Our little secret.",

    resave: false,

    saveUninitialized: false

}));

app.use(passport.initialize()); //inicializamos complemento passport

app.use(passport.session()); //inicializamos la session

//conexion con el servidor de mongoDB

mongoose.connect("mongodb://localhost:27017/userDB", { useNewUrlParser: true, useUnifiedTopology: true});

mongoose.set("useCreateIndex", true);

//creacion de esquema para cada documento (conjunto de datos)

const userSchema = new mongoose.Schema ({ //esquema creado por medio de la clase "mongoose.Schema"

    email: String,

    password: String

});

userSchema.plugin(passportLocalMongoose); //usamos el complemento de passport-local-mongoose

//creacion de modelo de tabla con base en el esquema (la cual mongoDB pluraliza posteriormente)

const User = new mongoose.model("User", userSchema);

//las tres siguientes lienas se usan para evitar un warning que se genera al iniciar el servidor

passport.use(User.createStrategy());

passport.serializeUser(User.serializeUser());

passport.deserializeUser(User.deserializeUser());

app.get("/", function(req, res) {

    res.render("home");

})

//VENTANA LOGIN

app.get("/login", function(req, res) {

    res.render("login");

});

app.post("/login", function(req, res){

    const user = new User({

        username: req.body.username,

        password: req.body.password

    });

    req.login(user, function(err){

        if(err) {

            console.log(err);

        } else {

            passport.authenticate("local")(req, res, function() {

                res.redirect("/secrets");

            });

        }

    })

});

//VENTANA REGISTER

app.get("/register", function(req, res) {

    res.render("register");

});

app.get("/secrets", function(req, res){

    if(req.isAuthenticated()){ //con esto verificamos si el usuario realmente ya ha iniciado sesion con su usuario o no

        res.render("secrets");

    } else {

        res.redirect("/login");

    }

});

app.get("/logout", function(req, res){

    req.logout();

    res.redirect("/");

});

app.post("/register", function(req, res) {

    User.register({username: req.body.username}, req.body.password, function(err, user){

        if(err){

            console.log(err);

            res.redirect("/register");

        } else {

            passport.authenticate("local")(req, res, function() {

                res.redirect("/secrets");

            });

        }

    });

});

NIVEL 6 DE SEGURIDAD EN LA ENCRIPTACION DEL PASSWORD