

Bài 5. Authorization Với Passport JWT 🎉

1. File package.json 🍌

npm i passport-jwt

```
{
  "name": "training",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "start": "node server.js",
    "dev": "nodemon server.js"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "dependencies": {
    "bcrypt": "^5.0.0",
    "express": "^4.17.1",
    "mongoose": "^5.9.28",
    "nodemon": "^2.0.4",
    "passport": "^0.4.1",
    "passport-jwt": "^4.0.0",
    "passport-local": "^1.0.0"
  }
}
```

2. File passport.js 🍌 🍌

```
const passport = require("passport");
const LocalStrategy = require("passport-local").Strategy;
const JwtStrategy = require("passport-jwt").Strategy;
const User = require("../model/User");

//authentication with username,password
passport.use(
  new LocalStrategy((username, password, done) => {
    User.findOne({ username }, (err, user) => {
      if (err) return done(err);
      if (!user) return done(null, false);
      user.comparePassword(password, done);
    });
  })
);

const cookiesExtractor = (req) => {
  let token = null;
  if (req && req.cookies) {
    token = req.cookies["access_token"];
  }
  return token;
};

passport.use(
  new JwtStrategy(
    {
      jwtFromRequest: cookiesExtractor,
      secretOrKey: "QuocLiem",
    },
    (payload, done) => {
      User.findById({ _id: payload.sub }, (err, user) => {
        if (err) return done(err, false);
        if (user) return done(null, user);
        else return done(null, false);
      });
    }
  )
);
```

The End

