

Copilot

Dưới đây là một cấu trúc folder và file điển hình cho một dự án Node.js. Cấu trúc này giúp bạn tổ chức mã nguồn một cách khoa học và dễ quản lý:

Cấu trúc folder và file điển hình

```
my-node-app/
├── src/
│   ├── controllers/
│   │   └── userController.js
│   ├── models/
│   │   └── userModel.js
│   ├── routes/
│   │   └── userRoutes.js
│   ├── services/
│   │   └── userService.js
│   └── utils/
│       └── helper.js
├── config/
│   ├── database.js
│   └── server.js
├── public/
│   ├── images/
│   ├── css/
│   └── js/
├── tests/
│   ├── unit/
│   │   └── userController.test.js
│   └── integration/
│       └── userRoutes.test.js
├── logs/
│   ├── app.log
│   └── error.log
├── node_modules/
├── .gitignore
├── package.json
├── README.md
└── server.js
```

Giải thích các folder và file

- **src/**: Chứa mã nguồn chính của ứng dụng.
 - **controllers/**: Chứa các file điều khiển, xử lý logic cho các yêu cầu HTTP.
 - **models/**: Chứa các file mô hình, định nghĩa cấu trúc dữ liệu và tương tác với cơ sở dữ liệu.
 - **routes/**: Chứa các file định tuyến, xác định các endpoint của API.
 - **services/**: Chứa các file dịch vụ, xử lý logic nghiệp vụ.
 - **utils/**: Chứa các file tiện ích, các hàm dùng chung trong dự án.
- **config/**: Chứa các file cấu hình cho ứng dụng, như cấu hình cơ sở dữ liệu và server.
- **public/**: Chứa các file tĩnh như hình ảnh, CSS, JavaScript.
- **tests/**: Chứa các file kiểm thử.
 - **unit/**: Chứa các file kiểm thử đơn vị.
 - **integration/**: Chứa các file kiểm thử tích hợp.
- **logs/**: Chứa các file log của ứng dụng.

- **node_modules/**: Chứa các module và thư viện được cài đặt thông qua npm.
- **.gitignore**: File cấu hình để bỏ qua các file và folder không cần thiết khi commit vào Git.
- **package.json**: File cấu hình của dự án, chứa thông tin về dự án và các dependencies.
- **README.md**: File tài liệu hướng dẫn và thông tin về dự án.
- **server.js**: File chính để khởi động server của ứng dụng.

Ví dụ cụ thể

Dưới đây là một ví dụ cụ thể về nội dung của một số file trong cấu trúc trên:

src/controllers/userController.js

```
const User = require('../models/userModel');

exports.getUser = (req, res) => {
  // Logic để lấy thông tin người dùng
};
```

src/models/userModel.js

```
const mongoose = require('mongoose');

const userSchema = new mongoose.Schema({
  name: String,
  email: String,
  password: String,
});

module.exports = mongoose.model('User', userSchema);
```

src/routes/userRoutes.js

```
const express = require('express');
const userController = require('../controllers/userController');
const router = express.Router();

router.get('/user', userController.getUser);

module.exports = router;
```

Hy vọng cấu trúc này giúp bạn tổ chức dự án Node.js của mình một cách hiệu quả. Nếu bạn có thêm câu hỏi nào, đừng ngần ngại hỏi nhé!