

# Projeto Nest.js e Vue.js

**Alunos:** Augusto Castejon, Álvaro Dias, Luíz Felipe e Vinicius Dogonski

## 1. Configuração do Projeto Back-End com Nest.js

Abra o terminal. Em seguida:

- 1.1. Para criar um diretório com o NestJS CLI, execute o comando a seguir, o comando vai criar um novo diretório com a configuração inicial do núcleo.

→ `nest new backend`

Selecione NPM e dê enter. (Pode demorar)

- 1.2. Dê as permissões para acessar a pasta

→ `sudo chmod 777 backend`

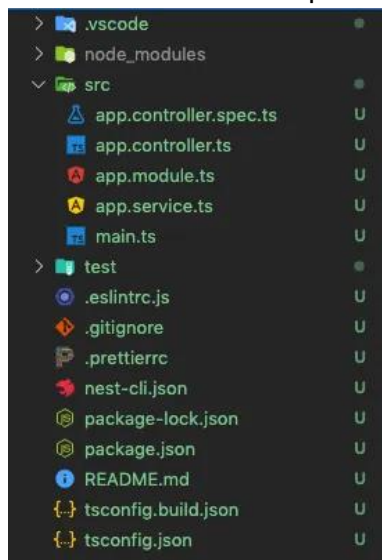
- 1.3. Entre no diretório

→ `cd backend`

- 1.4. Para visualizar o projeto criado, dê:

→ `code .`

- 1.5. Podemos encontrar arquivos principais no src/Diretório.



- `app.controller.ts`: Um controlador básico com um caminho simples.

- app.service.ts: Um serviço simples.
- app.module.ts: O módulo raiz da aplicação.
- main.ts: O arquivo de entrada do aplicativo que usa a função principal

1.6. Entre no diretório backend e instale o mongoose:

→ npm install mongoose

1.7. Configure as versões node e npm com o volta

1.7.1. Verifique a versão do Node e do NPM

→ node -v

→ npm -v

1.7.2. Definindo a versão do Node:

→ volta pin node@18.17.1

\*A versão pode alterar

1.7.3. Definindo a versão do npm:

→ volta pin npm@9.6.7

\*A versão pode alterar

## 2. Configuração do Front-End com Vue.js

2.1. Agora volte para o diretório geral do projeto:

→ cd ..

2.2. Crie um novo projeto Vue.js:

→ npm create -y vue@latest

2.3. Selecione as seguintes opções:

```
Vue.js - The Progressive JavaScript Framework
✓ Project name: ... frontend
✓ Add TypeScript? ... No / Yes
✓ Add JSX Support? ... No / Yes
✓ Add Vue Router for Single Page Application development? ... No / Yes
✓ Add Pinia for state management? ... No / Yes
✓ Add Vitest for Unit Testing? ... No / Yes
✓ Add an End-to-End Testing Solution? ... No
✓ Add ESLint for code quality? ... No / Yes
✓ Add Prettier for code formatting? ... No / Yes
```

2.4. Agora vamos executar alguns comandos no terminal para, de fato, configurar nosso frontend:

- cd frontend/
- npm install
- npm run format

2.5. Definindo a versão do Node, no diretório frontend:

- volta pin node@18.18.2
- \*A versão pode alterar

2.6. Definindo a versão do npm, no frontend:

- volta pin npm@10.2.0
- \*A versão pode alterar

### 3. Integração entre Back-End e Front-End

3.1. Navegue para a pasta do projeto Vue.js:

- cd frontend

3.2. Instale o pacote axios para fazer solicitações HTTP ao back-end:

- npm install axios

### 4. Execução dos Projetos

4.1. Em um terminal, na pasta backend, inicie o servidor NestJS:

- npm run start

4.2. Em um segundo terminal, na pasta frontend, inicie o servidor Vue.js:

- npm run dev

Agora, você deve ter o back-end do NestJS rodando em um servidor local e o front-end Vue.js sendo servido em outro. Lembre-se de que esta é uma configuração básica. À medida que você avança no desenvolvimento, pode precisar ajustar configurações e instalar mais dependências de acordo com os requisitos do seu projeto.