


# Introdução

Nesse roteiro, apresentamos as orientações em um passo a passo para você estruturar o seu ambiente de desenvolvimento com o Node.js e o JSONServer.

## Roteiro detalhado

Siga os passos que se seguem.

### 1. Instalar Node.js

Acesse <https://nodejs.org> , faça o download e instale a versão **LTS** (Long Term Support) recomendada para seu sistema operacional.

Após a instalação, confirme se a instalação está ok abrindo o terminal e digitando:

```
node -v  
npm -v
```

Você deve ver as versões do Node.js e do npm.

### 2. Criar a Pasta do Projeto

Abra o terminal e crie uma nova pasta:

```
mkdir meu-projeto-jsonserver  
cd meu-projeto-jsonserver
```

### 3. Inicializar o Projeto com npm

Abra a pasta com o Visual Studio Code ou outra IDE de sua preferência. Dentro da pasta do projeto, inicialize o projeto Node.js, executando o seguinte comando em um terminal:

```
npm init -y
```

Isso criará o arquivo `package.json` que é o arquivo descritor do projeto e traz entre outras coisas, as bibliotecas/pacotes necessários para o projeto, os scripts para execução, testes e deploy da aplicação, demais configurações para o projeto.

### 4. Instalar o JSONServer

A nossa aplicação requer o pacote JSONServer que não é nativo do Node.js e, dessa forma, vamos precisar instalar esse módulo no nosso projeto. No terminal, execute:

```
npm install json-server
```

## 5. Criar a Estrutura de Pastas

Crie as duas pastas principais do nosso projeto (db e public). Isso pode ser feito diretamente pela interface do Visual Studio Code ou via terminal.

Comandos no Linux, Mac Os e outros Unix:

```
mkdir db  
mkdir public
```

Comandos no Windows

```
md db  
md public
```

*OBS: Essas pastas devem ser criadas na raiz do projeto.*

## 6. Criar o arquivo JSON (simulação do Banco de Dados)

Dentro da pasta `db`, crie o arquivo `db.json` e inclua um exemplo básico de estrutura de dados para a entidade **usuários**:

```
{  
  "usuarios": [  
    {  
      "id": 1,  
      "nome": "Ana Souza",  
      "email": "ana@example.com"  
    },  
    {  
      "id": 2,  
      "nome": "Bruno Lima",  
      "email": "bruno@example.com"  
    }  
  ]  
}
```

## 7. Inserir os arquivos do projeto Front-end

Na pasta `public`, adicione os arquivos (HTML, CSS, JavaScript e imagens):

- `index.html`
- `detalhes.html`
- `styles.css`
- `app.js`
- (E a pasta `img/`, se houver imagens)

*OBS: Esses arquivos podem ser os do projeto anterior que você desenvolveu.*

## 8. Executar o JSON Server

Adicione um script no `package.json` para facilitar a execução:

```
"scripts": {  
  "start": "json-server --watch db/db.json --port 3000 --static ./public"  
}
```

Em seguida, abra uma janela de terminal e execute:

```
npm start
```

Isso fará:

- JSON Server rodar em `http://localhost:3000`
- Arquivos da pasta `public/` serem servidos como **páginas estáticas**

## 9. Testes Básicos

Agora que temos tudo no lugar, passaremos por uma dinâmica de testes parecida com o que é mostrado na imagem seguinte:





Vamos testar primeiro a API e depois o ambiente front end. Siga os seguintes passos.

#### ◆ Testar a API

- No navegador ou via ferramenta como Postman:
  - Acesse `http://localhost:3000/usuarios` para ver a lista de usuários.
  - Acesse `http://localhost:3000/usuarios/1` para ver o usuário com ID 1.

#### ◆ Testar o Front-end

- Acesse:
  - `http://localhost:3000/index.html` para a página inicial.
  - `http://localhost:3000/detalhes.html` para a página de detalhes (passando parâmetros se necessário).