



**UNIVERSIDADE FEDERAL DO MARANHÃO**  
**Bacharelado Interdisciplinar em Ciências e Tecnologia**  
**Projeto e Desenvolvimento de Software**

**Giordano Bruno de Araujo Mochel**  
**Vitor Ferreira Nunes**

**Sistema de Script Inteligente**

**Manual do Instalação**

# 1. Reconhecimento e Direitos Autorais:

@autor: Giordano Bruno Mochel e Vitor Ferreira Nunes

@contato:

@data última versão: 21 / 06 / 2024

@versão: 1.1

@outros repositórios:

@Agradecimentos: Universidade Federal do Maranhão (UFMA), Professor Doutor Thales Levi Azevedo Valente, e colegas de curso.

@Copyright/License

Este material é resultado de um trabalho acadêmico para a disciplina PROJETO E DESENVOLVIMENTO DE SOFTWARE, sobre a orientação do professor Dr. THALES LEVI AZEVEDO VALENTE, semestre letivo 2024.1, curso Engenharia da Computação, na Universidade Federal do Maranhão (UFMA). Todo o material sob esta licença é software livre: pode ser usado para fins acadêmicos e comerciais sem nenhum custo. Não há papelada, nem royalties, nem restrições de "copyleft" do tipo GNU. Ele é licenciado sob os termos da licença MIT reproduzida abaixo e, portanto, é compatível com GPL e também se qualifica como software de código aberto. É de domínio público. Os detalhes legais estão abaixo. O espírito desta licença é que você é livre para usar este material para qualquer finalidade, sem nenhum custo. O único requisito é que, se você usá-los, nos dê crédito.

Copyright © 2024 Educational Material

Este material está licenciado sob a Licença MIT. É permitido o uso, cópia, modificação, e distribuição deste material para qualquer fim, desde que acompanhado deste aviso de direitos autorais.

O MATERIAL É FORNECIDO "COMO ESTÁ", SEM GARANTIA DE QUALQUER TIPO, EXPRESSA OU IMPLÍCITA, INCLUINDO, MAS NÃO SE LIMITANDO ÀS GARANTIAS DE COMERCIALIZAÇÃO, ADEQUAÇÃO A UM DETERMINADO FIM E NÃO VIOLAÇÃO. EM HIPÓTESE ALGUMA OS AUTORES OU DETENTORES DE DIREITOS AUTORAIS SERÃO RESPONSÁVEIS POR QUALQUER RECLAMAÇÃO, DANOS OU OUTRA RESPONSABILIDADE, SEJA EM UMA AÇÃO DE CONTRATO, ATO ILÍCITO OU DE OUTRA FORMA, DECORRENTE DE, OU EM CONEXÃO COM O MATERIAL OU O USO OU OUTRAS NEGOCIAÇÕES NO MATERIAL.

Para mais informações sobre a Licença MIT: <https://opensource.org/licenses/MIT>.

## 2. Manual de Instalação - Frontend:

### 2.1. Pré-requisitos:

Certifique-se de que você tem os seguintes itens instalados em seu sistema:

1. Node.js (versão 14.x ou superior)
2. npm (versão 6.x ou superior)

### 2.2. Clonar o Repositório:

Clone o repositório do projeto a partir do sistema de controle de versão utilizando o comando:

```
git clone https://github.com/ProjetoDesenvolvimentoSoftwareUFMA/backAndFrontScriptInteligente.git
```

Navegue até o diretório do projeto FrontEndScriptInteligente:

```
cd backAndFrontScriptInteligente/1.3Codigo/FrontEndScriptInteligente
```

### 2.3. Instalar Dependências:

Execute o comando abaixo para instalar todas as dependências necessárias:

```
npm install
```

### 2.4 Scripts Disponíveis:

npm run start - Inicia o servidor de desenvolvimento utilizando o Webpack.

npm run build - Gera uma versão de produção do projeto.

npm run lint - Executa o ESLint em todos os arquivos .ts e .tsx no diretório src.

### 2.5. Executar o Projeto:

Para iniciar o servidor de desenvolvimento e visualizar o projeto no navegador, execute o seguinte comando:

```
npm start
```

## 3. Manual de Instalação - Backend:

### 3.1. Pré-requisitos:

Certifique-se de que você tem os seguintes itens instalados em seu sistema:

1. Node.js (versão 14.x ou superior)
2. npm (versão 6.x ou superior)
3. PostgreSQL
4. Ollama 7b

### 3.2. Clonar o Repositório:

Clone o repositório do projeto a partir do sistema de controle de versão utilizando o comando:

```
git clone https://github.com/ProjetoDesenvolvimentoSoftwareUFMA/backAndFrontScriptInteligente.git
```

Navegue até o diretório do projeto BackEndScriptInteligente:

```
cd backAndFrontScriptInteligente/1.3Codigo/BackEndScriptInteligente
```

### 3.3. Instalar Dependências:

Execute o comando abaixo para instalar todas as dependências necessárias:

```
npm install
```

### 3.4. Instalar Ollama:

Acesse o site oficial do Ollama: <https://ollama.com/download/windows>

Clique em "Download for Windows (Preview)"

Execute o arquivo baixado e siga as instruções de instalação na tela.

### 3.5. Configurar o Ollama:

Após a instalação, abra o Ollama. Verifique se a configuração do Ollama está definida para utilizar a porta correta.

No terminal do Ollama, execute o comando:

```
Ollama push codellama:latest
```

Após a instalação, execute o comando:

```
Ollama serve
```

### 3.6. Configurar o Banco de Dados:

Crie um banco de dados PostgreSQL e configure as variáveis de ambiente no arquivo .env na raiz do projeto:

```
DATABASE_URL="postgresql://<USUARIO>:<SENHA>@<HOST>:<PORTA>/<NOME_DO_BANCO>?schema=public"
```

Execute os comandos abaixo para aplicar as migrações do Prisma e configurar o banco de dados:

```
npx prisma migrate dev --name init
```

```
npx prisma generate
```

### 3.7. Scripts Disponíveis:

O arquivo package.json define vários scripts úteis que podem ser executados usando npm. Aqui estão os principais scripts que você precisará:

npm run build - Compila o projeto usando o NestJS.

npm run format - Formata o código usando Prettier.

npm run start - Inicia o servidor NestJS.

npm run start:dev - Inicia o servidor NestJS em modo de desenvolvimento com hot-reload.

### 3.8. Executar o Projeto:

Para iniciar o servidor de desenvolvimento e visualizar o projeto no navegador, execute o seguinte comando:

```
npm run start:dev
```