



MANUAL DE INSTALAÇÃO

EDULING

Guia Completo para Instalação e Configuração

Versão 1.0
Fevereiro de 2025

EDULING LTDA.
SUPORTE@EDULING.COM.BR

Sumário

| | | |
|----------|---|----------|
| 1 | Introdução | 2 |
| 1.1 | Sobre a Aplicação | 2 |
| 1.2 | Requisitos de Sistema | 2 |
| 2 | Instalação de Pré-requisitos | 3 |
| 2.1 | Instalação do Node.js | 3 |
| 2.1.1 | Para Windows | 3 |
| 2.1.2 | Para macOS | 3 |
| 2.1.3 | Para Linux (Debian/Ubuntu) | 4 |
| 2.2 | Instalação do Git (Opcional, mas recomendado) | 4 |
| 2.2.1 | Para Windows | 4 |
| 2.2.2 | Para macOS | 4 |
| 2.2.3 | Para Linux | 4 |
| 3 | Instalação da Aplicação | 5 |
| 3.1 | Clonando o Repositório | 5 |
| 3.2 | Instalando Dependências | 5 |
| 3.3 | Configurando Variáveis de Ambiente | 5 |
| 4 | Executando a Aplicação | 7 |
| 4.1 | Ambiente de Desenvolvimento | 7 |
| 4.2 | Compilação para Produção | 7 |
| 4.3 | Iniciando em Modo de Produção | 7 |
| 5 | Configuração Avançada | 7 |
| 5.1 | Configurando Porta Personalizada | 7 |
| 5.2 | Configurando um Proxy Reverso | 8 |
| 5.3 | Configuração de PM2 (Para produção) | 8 |
| 6 | Solução de Problemas | 8 |
| 6.1 | Problemas Comuns e Soluções | 8 |
| 6.2 | Logs e Depuração | 9 |
| 7 | Atualizações | 9 |
| 7.1 | Atualizando a Aplicação | 9 |
| 7.2 | Recursos Adicionais | 10 |

1 Introdução

Este manual fornece instruções passo a passo para a instalação e configuração de uma aplicação desenvolvida com Node.js e Next.js. O documento foi elaborado para usuários com conhecimentos básicos em sistemas operacionais e linha de comando.

1.1 Sobre a Aplicação

A aplicação é uma solução web moderna desenvolvida com as seguintes tecnologias:

- **Node.js** - Ambiente de execução JavaScript do lado do servidor
- **Next.js** - Framework React para aplicações web renderizadas no servidor
- **React** - Biblioteca JavaScript para criação de interfaces de usuário
- **npm/yarn** - Gerenciadores de pacotes para o ecossistema JavaScript

1.2 Requisitos de Sistema

Para instalar e executar a aplicação, seu sistema deve atender aos seguintes requisitos mínimos:

- Sistema Operacional: Windows 10/11, macOS 12+, ou Linux (Ubuntu 20.04+, Debian 11+, etc.)
- Memória RAM: mínimo de 4GB (recomendado 8GB ou mais)
- Espaço em disco: mínimo de 1GB livre
- Processador: Intel Core i3 ou equivalente (2 núcleos ou mais)
- Conexão à internet para download de dependências

Atenção

A aplicação foi testada apenas nos sistemas operacionais listados acima. O suporte para outros sistemas não é garantido.

2 Instalação de Pré-requisitos

2.1 Instalação do Node.js

O Node.js é necessário para executar a aplicação. Recomendamos a instalação da versão LTS (Long Term Support) mais recente.

2.1.1 Para Windows

1. Acesse o site oficial do Node.js: <https://nodejs.org/>
2. Faça o download da versão LTS recomendada para Windows
3. Execute o instalador baixado (.msi)
4. Siga as instruções do assistente de instalação
5. Verifique a instalação abrindo o Prompt de Comando e digitando:

```
1 node --version
2 npm --version
```

2.1.2 Para macOS

Usando o instalador:

1. Acesse o site oficial do Node.js: <https://nodejs.org/>
2. Faça o download da versão LTS recomendada para macOS
3. Execute o instalador baixado (.pkg)
4. Siga as instruções do assistente de instalação

Usando Homebrew:

1. Abra o Terminal
2. Se ainda não tiver o Homebrew instalado, instale-o com:

```
1 /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

3. Instale o Node.js com:

```
1 brew install node
```

2.1.3 Para Linux (Debian/Ubuntu)

1. Abra o Terminal
2. Atualize os repositórios:

```
1 sudo apt update
```

3. Instale o Node.js e npm:

```
1 sudo apt install nodejs npm
```

Dica

Para uma versão mais recente do Node.js, considere usar o NodeSource:

```
1 curl -fsSL https://deb.nodesource.com/setup_18.x | sudo -E bash -  
2 sudo apt install -y nodejs
```

Substitua `setup_18.x` pela versão LTS desejada.

2.2 Instalação do Git (Opcional, mas recomendado)

O Git é necessário para clonar o repositório da aplicação e para futuras atualizações.

2.2.1 Para Windows

1. Acesse: <https://git-scm.com/download/win>
2. Faça o download do instalador
3. Execute o instalador e siga as instruções
4. Verifique a instalação com:

```
1 git --version
```

2.2.2 Para macOS

1. Se você já instalou o Xcode, o Git já está disponível
2. Caso contrário, instale via Homebrew:

```
1 brew install git
```

2.2.3 Para Linux

1. Abra o Terminal e execute:

```
1 sudo apt update  
2 sudo apt install git
```

3 Instalação da Aplicação

3.1 Clonando o Repositório

1. Abra o Terminal (Linux/macOS) ou o Prompt de Comando/PowerShell (Windows)
2. Navegue até o diretório onde deseja instalar a aplicação
3. Clone o repositório com o comando:

```
1 git clone git@github.com:tiagodlb/Edulign.git
```

4. Navegue até o diretório da aplicação:

```
1 cd Edulign
```

3.2 Instalando Dependências

1. Dentro do diretório da aplicação você terá duas partes a back e a front, navega até cada uma com 'cd' e instale as dependências com npm:

```
1 npm install
```

Dica

Se preferir usar o Yarn como gerenciador de pacotes, instale-o primeiro com:

```
1 npm install -g yarn
```

E depois instale as dependências com:

```
1 yarn install
```

3.3 Configurando Variáveis de Ambiente

1. No diretório raiz da aplicação, crie um arquivo `.env.local` baseado no exemplo fornecido:

```
1 # Windows
2 copy .env.example .env.local
3
4 # Linux/macOS
5 cp .env.example .env.local
```

2. Abra o arquivo `.env.local` em um editor de texto e configure as variáveis necessárias:

```
1 NEXT_PUBLIC_API_URL=https://api.suaempresa.com.br
2 NEXT_PUBLIC_SITE_URL=https://suaempresa.com.br
3 DATABASE_URL=postgres://usuario:senha@localhost:5432/seu_banco
4 # Adicione outras variaveis conforme necessario
```

Atenção

Nunca cometa o arquivo `.env.local` no Git. Ele contém informações sensíveis que não devem ser compartilhadas.

4 Executando a Aplicação

4.1 Ambiente de Desenvolvimento

Para iniciar a aplicação em modo de desenvolvimento:

```
1 npm run dev
2 # OU com Yarn
3 yarn dev
```

A aplicação estará disponível em `http://localhost:3000`.

Dica

O modo de desenvolvimento oferece:

- Recarregamento automático quando você altera arquivos
- Mensagens de erro detalhadas
- Hot Module Replacement (HMR)

4.2 Compilação para Produção

Para compilar a aplicação para ambiente de produção:

```
1 npm run build
2 # OU com Yarn
3 yarn build
```

4.3 Iniciando em Modo de Produção

Após a compilação, inicie a aplicação em modo de produção:

```
1 npm run start
2 # OU com Yarn
3 yarn start
```

A aplicação estará disponível em `http://localhost:3000` (ou na porta configurada).

5 Configuração Avançada

5.1 Configurando Porta Personalizada

Para executar a aplicação em uma porta diferente:

```
1 # Temporariamente
2 PORT=8080 npm run dev
3
4 # Permanentemente (adicione ao .env.local)
5 PORT=8080
```


5.2 Configurando um Proxy Reverso

Para ambientes de produção, recomendamos o uso de um proxy reverso como Nginx:

```
1 server {
2     listen 80;
3     server_name seu-dominio.com www.seu-dominio.com;
4
5     location / {
6         proxy_pass http://localhost:3000;
7         proxy_http_version 1.1;
8         proxy_set_header Upgrade $http_upgrade;
9         proxy_set_header Connection 'upgrade';
10        proxy_set_header Host $host;
11        proxy_cache_bypass $http_upgrade;
12    }
13 }
```

5.3 Configuração de PM2 (Para produção)

Para manter sua aplicação rodando em produção, recomendamos o PM2:

1. Instale o PM2 globalmente:

```
1 npm install -g pm2
```

2. Crie um arquivo de configuração ecosystem.config.js:

```
1 module.exports = {
2   apps: [
3     {
4       name: "nextjs-app",
5       script: "npm",
6       args: "start",
7       env: {
8         NODE_ENV: "production",
9       },
10      instances: "max",
11      exec_mode: "cluster",
12    },
13  ],
14 };
```

3. Inicie a aplicação com PM2:

```
1 pm2 start ecosystem.config.js
```

6 Solução de Problemas

6.1 Problemas Comuns e Soluções

Problema 1: Erro "Port 3000 is already in use"

Solução: Encerre o processo que está usando a porta 3000 ou configure uma porta diferente.

```
1  # Verificar processo usando a porta 3000
2  # Windows
3  netstat -ano | findstr :3000
4
5  # Linux/macOS
6  lsof -i :3000
7
8  # Encerrar o processo (substitua PID pelo ID do processo)
9  # Windows
10 taskkill /F /PID PID
11
12 # Linux/macOS
13 kill -9 PID
14
```

Problema 2: Erro "Module not found"

Solução: Verifique se todas as dependências foram instaladas corretamente.

```
1  rm -rf node_modules
2  rm package-lock.json # ou yarn.lock se estiver usando Yarn
3  npm install # ou yarn install
4
```

Problema 3: Erro de variáveis de ambiente não definidas

Solução: Verifique se o arquivo `.env.local` existe e contém todas as variáveis necessárias.

Problema 4: Aplicação lenta em desenvolvimento

Solução: Verifique se seu sistema atende aos requisitos mínimos. Considere fechar outros aplicativos que consomem muitos recursos.

6.2 Logs e Depuração

1. Em modo de desenvolvimento, os logs são exibidos no terminal
2. Em produção com PM2, visualize os logs com:

```
1 pm2 logs
```

3. Para logs específicos da aplicação:

```
1 pm2 logs nextjs-app
```

7 Atualizações

7.1 Atualizando a Aplicação

Para atualizar a aplicação para a versão mais recente:

```
1 # Navegue ate o diretorio da aplicacao
2 cd Edulign
3
4 # Obtenha as atualizacoes mais recentes
5 git pull
6
7 # Atualize as dependencias
8 npm install # ou yarn install
9
10 # Reconstrua a aplicacao
11 npm run build # ou yarn build
12
13 # Reinicie a aplicacao (se estiver usando PM2)
14 pm2 restart nextjs-app
```

Atenção

Sempre faça backup de suas configurações e dados antes de atualizar para uma nova versão.

7.2 Recursos Adicionais

- Documentação Oficial do Next.js: <https://nextjs.org/docs>
- Documentação do Node.js: <https://nodejs.org/en/docs/>

Fim do Manual de Instalação