



# Configuração

Esta página definirá o passo a passo para compilar este projeto.

## Clonar o repositório

### Bash

```
git clone https://github.com/ZRafaF/educa-utf  
  
cd educa-utf
```

## Instalar dependências

### Bash

```
npm run install
```

## Criando variáveis de ambiente

Os seguintes arquivos são necessários para o desenvolvimento da aplicação:

### Variáveis do NEXTJS

Crie um arquivo de variável de ambiente recomendando `.env.local` e adicione as seguintes variáveis:

### Bash

```
# URL of your PocketBase db  
PB_URL=https://myproject.pockethost.io  
  
# Should Next analyze your packages during build  
ANALYZE=false
```



### Atenção

Esses são **SEGREDOS** e não deverão ser enviados a repositórios públicos. Seja GitHub, DockerHub e outros.

## Variáveis do TypeGen

Em um arquivo de variável de ambiente chamado `.env` adicione as seguintes variáveis:

### Bash

```
# URL of your PocketBase db
PB_TYPEGEN_URL=https://myproject.pockethost.io

# Email of an admin on PocketBase dashboard
PB_TYPEGEN_EMAIL=admin@myproject.com

# Password of an admin on PocketBase dashboard
PB_TYPEGEN_PASSWORD=secr3tp@ssword!
```



### Atenção

Esses são **SEGREDOS** e não deverão ser enviados a repositórios públicos. Seja GitHub, DockerHub e outros.

## Documentação

Este projeto foi desenvolvido usando a versão Python `Python 3.10.x`. A etapa a seguir é necessária apenas para aqueles que desejam **editar a documentação**.

Para compilar a documentação você precisará de **Python**, **PIP** e **GIT**.

## Criando um ambiente virtual

A etapa a seguir não é obrigatória, mas é **recomendada**. Se você quiser saber um pouco mais sobre ambientes virtuais de python visite <https://docs.python.org/3/library/venv.html>

### Bash

```
python -m pip install --user virtualenv

python -m venv venv
```

Um diretório `venv` deve ser criado na pasta raiz do projeto.

Como ativar:

### ■ Ativação no Windows

```
venv/Scripts/activate
```

or

### ■ Ativação no Linux

```
source venv/bin/activate
```

Com isso o ambiente virtual estará ativado e qualquer biblioteca instalada será aplicada apenas à esse ambiente.

## Instalando dependências

### Bash

```
pip install -r docs/requirements.txt
```

Esse comando irá instalar todas as dependências contidas no arquivo `requirements.txt`

## Build

Para compilar a documentação temos duas opções:

- **Serve:**

Esta opção é usada para depuração, ela abrirá a página estática em uma das portas localhost.

### Bash

```
mkdocs serve
```

- **Build:** Esta opção cria uma compilação da documentação e a salva no diretório `/site/`.

### Bash

```
mkdocs build
```



### Note

Esteja ciente da **Environment Variable** `ENABLE_PDF_EXPORT`, ela só irá gerar o PDF se essa variável estiver definida como `1`.

Você pode alterar o arquivo `mkdocs.yml` e remover esta linha se preferir.

## Integração Pocketbase

Para gerar os Bindings do pocket base está sendo usado a biblioteca `pocketbase-typegen`. Para gerar os tipos execute:

```
yarn typegen
```

O *output* está localizado em `/src/types/pocketbase-types.ts`.

Última atualização: 4 de março de 2024

Criado em: 4 de março de 2024