|  |  |
| --- | --- |
| **Teste Técnico para vaga de Desenvolvedor Back-end** | |
| **Requisitos do teste:** Desenvolver um sistema que cadastre empresas e o seu quadro societário | **Data:** 08/03/2024 |

**Desenvolver um sistema que cadastre empresas e o seu quadro societário**

**Pré-requisitos:**

\* Symfony Framework

\* Persistir os dados em banco de dados, utilizar o PostgreSQL.

\* Doctrine ORM.

**Diferencias**:

\* Autenticação;

\* Permissionamento;

\* Não utilizar gerador de código

\* Desenvolver os recursos em API Rest;

\* Documentação da API

\* Frontend em ANGULAR;

\* SOLID

**O que esperamos como escopo mínimo:**

\* CRUD Empresas;

\* CRUD de Sócios (vinculando a empresa);

**Dependências**

\* PHP 7.4+

\* Composer

\* Symfony Framework na última versão LTS (long-term support)

\* Git

**Instruções**

1. Crie um repositório privado na gitlab.com;

2. Desenvolva. Você terá 3 (três) dias a partir da data do envio do desafio;

3. Após concluir seu trabalho, publique;

4. É importante adicionar como membro (develop) o usuário `philipe` (philipe@voxtecnologia.com.br)

5. Crie um arquivo README.MD com a explicação de como devemos executar o projeto e com uma descrição do que foi feito

**Sumário**

[**Estrutura do projeto** 3](#_Toc160764957)

[**Diretórios Principais** 3](#_Toc160764958)

[**Fluxo de Desenvolvimento** 3](#_Toc160764959)

[**Telas Principais** 3](#_Toc160764960)

[**Requisitos mínimos do Sistema** 4](#_Toc160764961)

[**Acesso ao Sistema** 4](#_Toc160764962)

[**Clone o repositório:** 4](#_Toc160764963)

[**Instalação de Dependências** 5](#_Toc160764964)

[**Configuração do Ambiente** 5](#_Toc160764965)

[**Banco de Dados** 5](#_Toc160764966)

[**Inicialização do Servidor Local** 5](#_Toc160764967)

[**Funcionalidades do sistema** 5](#_Toc160764968)

[**API REST de Geração de PDF** 6](#_Toc160764969)

[**Visão Geral** 6](#_Toc160764970)

[**Recurso e Endpoint** 6](#_Toc160764971)

[**Requisitos** 6](#_Toc160764972)

[**Autenticação** 6](#_Toc160764973)

[**Resposta da API** 6](#_Toc160764974)

[**Exemplo de Uso** 7](#_Toc160764975)

[**Considerações Finais** 7](#_Toc160764976)

[**Login para o sistema** 7](#_Toc160764977)

**Estrutura do projeto**

O projeto foi desenvolvido utilizando o framework Symfony, juntamente com várias bibliotecas e o Composer para gerenciamento de dependências. Abaixo, descrevemos a estrutura geral do projeto.

## **Diretórios Principais**

1. config: Contém as configurações do Symfony, incluindo rotas, serviços e outras configurações gerais. Estão algumas rotas criadas e algumas alterações na segurança para o funcionamento do sistema
2. src: Este diretório contém todo o código-fonte do projeto. As principais pastas dentro de src são:
3. Controller: Controladores Symfony que respondem às solicitações HTTP.
4. Entity: Definições das entidades Doctrine, representando os modelos de dados.
5. Form: Formulários Symfony para manipular dados em formulários HTML.
6. Repository: Repositórios Doctrine para interação com o banco de dados.
7. templates: Contém modelos Twig usados para renderizar as páginas HTML.
8. var: Armazena arquivos temporários, caches e logs.
9. vendor: Diretório onde as dependências do Composer são instaladas.
10. .env: Arquivo de configuração do ambiente que armazena variáveis de ambiente. Aonde foi feito a configuração da conexão com a base de dados

## **Fluxo de Desenvolvimento**

O desenvolvimento do projeto segue o padrão MVC (Model-View-Controller), onde:

* Model: Representado pelas entidades Doctrine em src/Entity.
* View: As visualizações Twig em templates.
* Controller: Controladores Symfony em src/Controller.

O Composer é utilizado para gerenciar as dependências do projeto, e as configurações do Symfony estão presentes no diretório config.

## **Telas Principais**

* **Cadastro de Empresas**

A tela de cadastro de empresas permite a adição e gerenciamento de informações relacionadas a empresas. Os usuários com permissões adequadas podem adicionar, editar ou excluir empresas. Além disso, é possível associar sócios a empresas existentes.

* **Cadastro de Sócios**

A tela de cadastro de sócios possibilita a inclusão e administração de dados referentes aos sócios. Os usuários autorizados podem adicionar, editar ou remover sócios, além de associá-los a empresas.

* **Tela Principal**

A tela principal oferece uma visão geral do sistema e fornece links rápidos para acessar as funcionalidades principais, como o cadastro de empresas, sócios e gerar relatórios PDF.

# **Requisitos mínimos do Sistema**

1. Php 7.4 ou superior;
2. Composer;
3. PostegreSQL;
4. Symfony framework
5. nodeJS

# **Acesso ao Sistema**

Para acessar o sistema, siga as etapas abaixo:

## **Clone o repositório:**

* **Abra o Terminal**:

No caso do Windows, você pode utilizar o Git Bash ou qualquer terminal que preferir.

No Linux ou macOS, o terminal padrão do sistema é adequado.

* **Navegue até o Diretório de Destino:**

Utilize o comando cd para navegar até o diretório onde deseja armazenar o projeto. Por exemplo:

cd /caminho/do/seu/diretório

* **Clone o Repositório:**

Execute o seguinte comando para realizar o clone do repositório:

git clone <https://github.com/seu-usuario/nome-do-repositorio.git>

Certifique-se de substituir seu-usuário pelo seu nome de usuário no GitHub e nome-do-repositório pelo nome específico do repositório.

* **Acesse o Diretório do Projeto:**

Após o clone ser concluído, acesse o diretório do projeto utilizando:

cd nome-do-repositorio

ou utilize as pastas do sistema operacional que estiver utilizando.

## **Instalação de Dependências**

Após ter realizado o clone, é necessário instalar as dependências do projeto. O Composer é a ferramenta que será utilizada para essa tarefa.

* **Instale o Composer:**

Se ainda não tiver o Composer instalado, siga as instruções em getcomposer.org para realizar a instalação.

* **Instale as Dependências:**

Dentro do diretório do projeto, execute o seguinte comando:

composer install

lembre de fazer isso dentro do diretório do projeto para que seja reconhecido as dependências que estão no arquivo composer.json

## **Configuração do Ambiente**

Antes de iniciar o sistema, certifique-se de configurar o ambiente adequadamente.

Estarei colocando o arquivo .env junto aos outro no git, com isso se atente nas configurações da conexão da base de dados, deixei alguns comentários explicando oque deve ser colocado em cada campo no arquivo.

Abra o arquivo com algum editor da sua preferencia e faça as alterações necessárias como estão descritas nos comentários.

## **Banco de Dados**

Em relação a o banco de dados, estarei colocando o script junto a os arquivos no git, com isso será apenas necessário você importar o script para o PostgreSQL.

## **Inicialização do Servidor Local**

Inicie o servidor local Symfony com o seguinte comando:

php bin/console server:run

Acesse o sistema no navegador através de <http://localhost:8000>.

Lembrete: todos os comando devem ser feitos no pront de comando (da sua preferência), dentro do diretório do arquivo de preferência.

# **Funcionalidades do sistema**

* Fazer cadastros das empresas, editar ou excluir.
* Realizar cadastros de sócios e vincula-los as empresas que eles atuam, poderá editar ou excluir os cadastros
* Realizar consultas nas tabelas, buscando por empresa ou por sócios (dependendo da tela que estiver no sistema)

Exemplo:

* tela Empresas -> pesquise apenas por empresas;
* tela sócios->pesquise apenas por sócios;
* tela principal está utilizando a busca por empresas
* Dentro da tela principal temos um gerador de relatório, aonde ele gera um arquivo PDF com todos os sócios e suas empresas relacionadas que estão cadastradas na base de dados.

# **API REST de Geração de PDF**

## **Visão Geral**

A API REST de Geração de PDF permite a criação dinâmica de relatórios em formato PDF contendo informações sobre empresas e sócios registrados no sistema. A geração do PDF é baseada nos dados armazenados no banco de dados, oferecendo uma maneira rápida e eficiente de obter relatórios personalizados.

## **Recurso e Endpoint**

* Endpoint: /api/pdf
* Método: GET

## **Requisitos**

Antes de utilizar a API, é necessário que o usuário tenha previamente instalado e configurado o ambiente Symfony, bem como tenha acesso ao banco de dados contendo as informações das empresas e sócios.

## **Autenticação**

A API não requer autenticação, permitindo acesso público.

Parâmetros da Solicitação

A solicitação à API é realizada através de uma requisição HTTP GET para o endpoint /api/pdf. Não são necessários parâmetros adicionais na solicitação.

## **Resposta da API**

A API retorna um arquivo PDF como resposta. O conteúdo do PDF é gerado dinamicamente com base nos dados mais recentes do banco de dados.

## **Exemplo de Uso**

Solicitação

http://seu-domino.com/api/pdf -o relatorio.pdf

Resposta

A resposta consistirá em um arquivo PDF chamado relatorio.pdf contendo informações atualizadas sobre empresas e sócios registrados no sistema.

## **Considerações Finais**

A API REST de Geração de PDF oferece uma maneira simples e eficaz de obter relatórios personalizados do sistema. Certifique-se de que o ambiente Symfony esteja devidamente configurado e que o banco de dados contenha os dados necessários para uma geração precisa do relatório.

# **Login para o sistema**

**Usuário Master:**  
**login:** admin

**Senha:** admin123

**Usuário comum:**

**Login:** usuário

**Senha:** 123456