Estrutura

Configurando o portal SINPOM em um ambiente de produção.

Estrutura do Projeto

Estrutura do Projeto com Docker

Estrutura do Banco de Dados

Visão Geral

Requisitos

Requisitos do Sistema

Estrutura do Projeto

Estrutura do Projeto com Docker

O projeto do sinpom utiliza o Docker para gerenciamento de containeres e imagens Docker. Ele utiliza o Docker Compose para gerenciamento de containeres e imagens Docker.

```
docker-compose-dev-ubuntu.yml
docker-compose-local.yml
docker-compose.yml
Dockerfile
.env
```

Estrutura do Laravel

O projeto do sinpom utiliza o framework Laravel para desenvolvimento da aplicação. Ele utiliza o Composer para gerenciamento de dependências do projeto.

```
/app
├── Console
├── Exceptions
├── Http
│   ├── Controllers
│   ├── Middleware
│   ├── Library
│   ├── Mail
│   ├── Providers
│   └── Scopes
/bootstrap
```

Diagrama Arquitetural

Estrutura do Banco de Dados

Visão Geral

O banco de dados do SINPOM é a espinha dorsal do sistema, permitindo armazenamento e a recuperação de dados criticos da plataforma, incluindo contas de usuários, funções, logs de acesso e entidades específicas. Ele usa o MySQL como sgbd, e foi projetado para funcionar perfeitamente com o ORM do Laravel (Eloquent) e é estruturado para se alinhar à arquitetura MVC.

Tabelas Principais

1. `users`: Gerencia detalhes da conta do usuário

Field Name	Data Type	Description
id	INT	Primary key, auto-increment.
name	VARCHAR(255)	User's full name.
email	VARCHAR(255)	Unique email address.
email_verified_at	TIMESTAMP	Email verification timestamp.
password	VARCHAR(255)	Encrypted password.
remember_token	VARCHAR(100)	
google2fa_secret	VARCHAR(255)	
created_at	TIMESTAMP	Record creation timestamp.
updated_at	TIMESTAMP	Record last update timestamp.

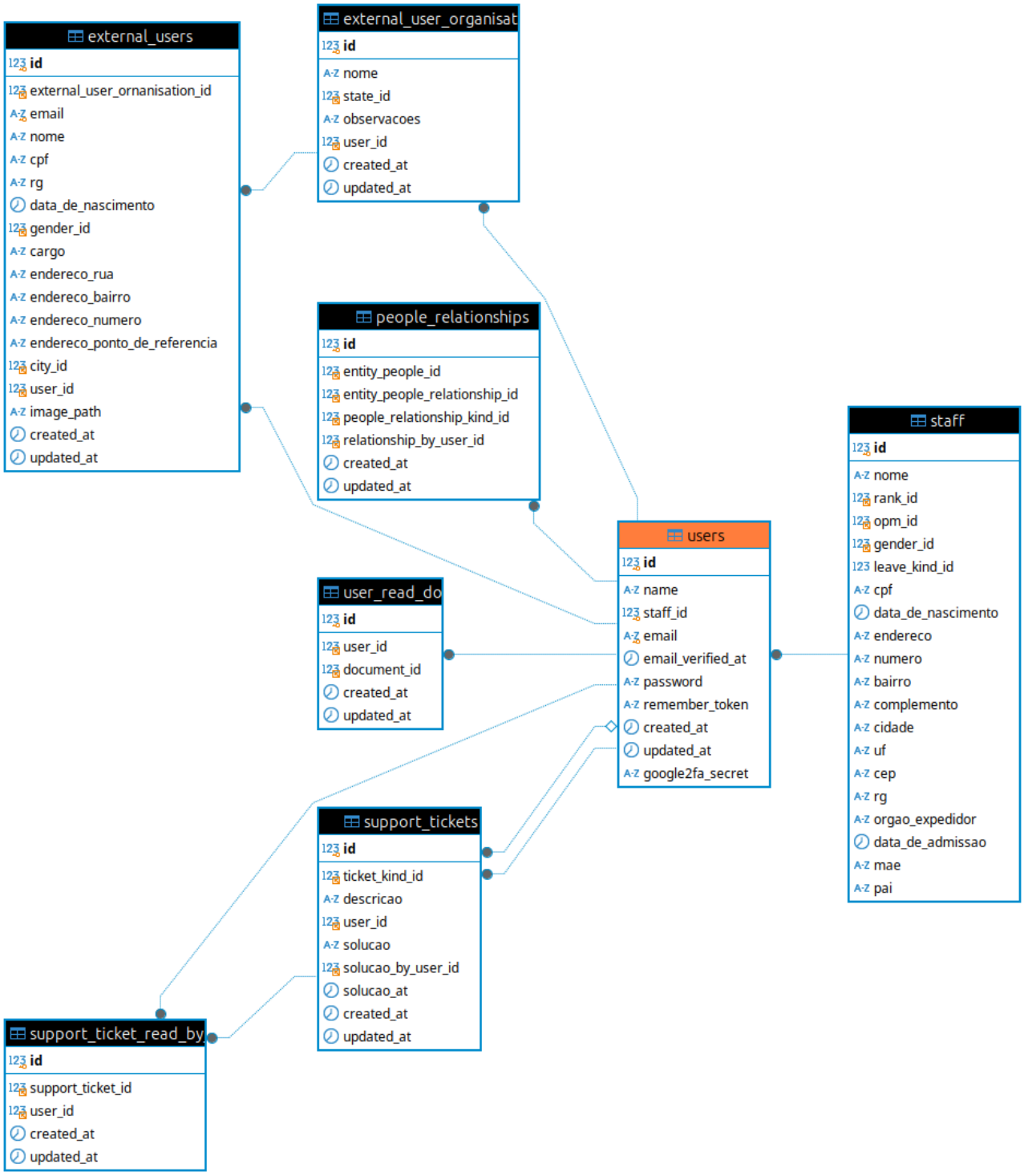
2. `documents`: Tabela de documentos.

Field Name	Data Type	Description
id	BIGINT unsigned	Primary key, auto-increment

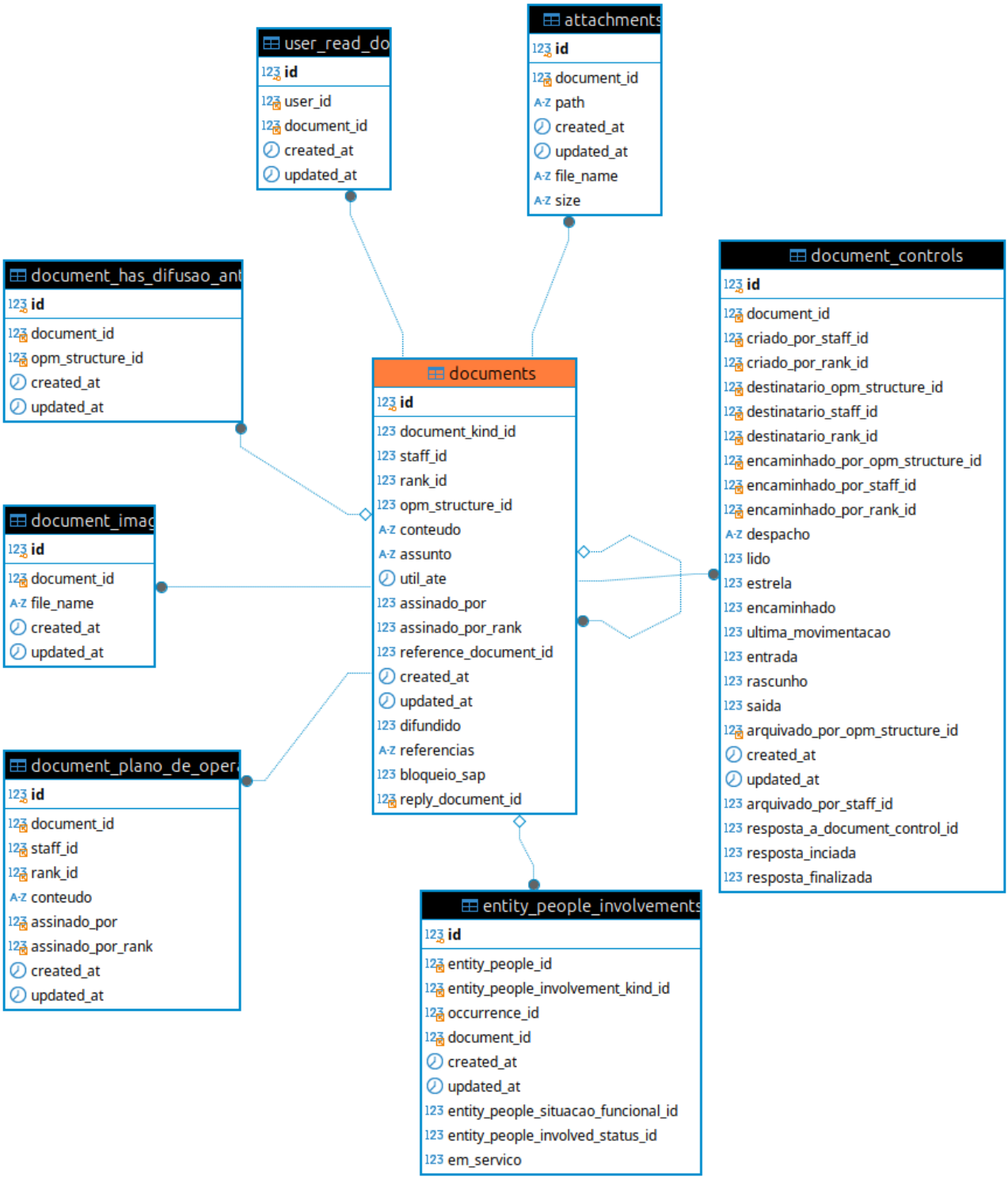
Field Name	Data Type	Description
document_kind_id	BIGINT unsigned	tipo do documento
staff_id	BIGINT unsigned	matricula do criador
rank_id	BIGINT unsigned	posto/graduacao do criador
opm_structure_id	BIGINT unsigned	seção do criador do documento
assinado_por	BIGINT unsigned	assinatura do documento finalizado
assinado_por_rank	BIGINT unsigned	assinatura do documento finalizado
reference_document_id	BIGINT unsigned	documento que deu origem a este
reply_document_id	BIGINT unsigned	
util_ate	DATE	prazo aceitável do documento
conteudo	TEXT	
difundido	TINYINT(1)	
bloqueio_sap	TINYINT(1)	
assunto	VARCHAR(191)	
referencias	VARCHAR(191)	
created_at	TIMESTAMP	Record creation timestamp.
updated_at	TIMESTAMP	Record last update timestamp.

Principais Relacionamentos

Relationamentos de `users`



Relationamentos de `documents`

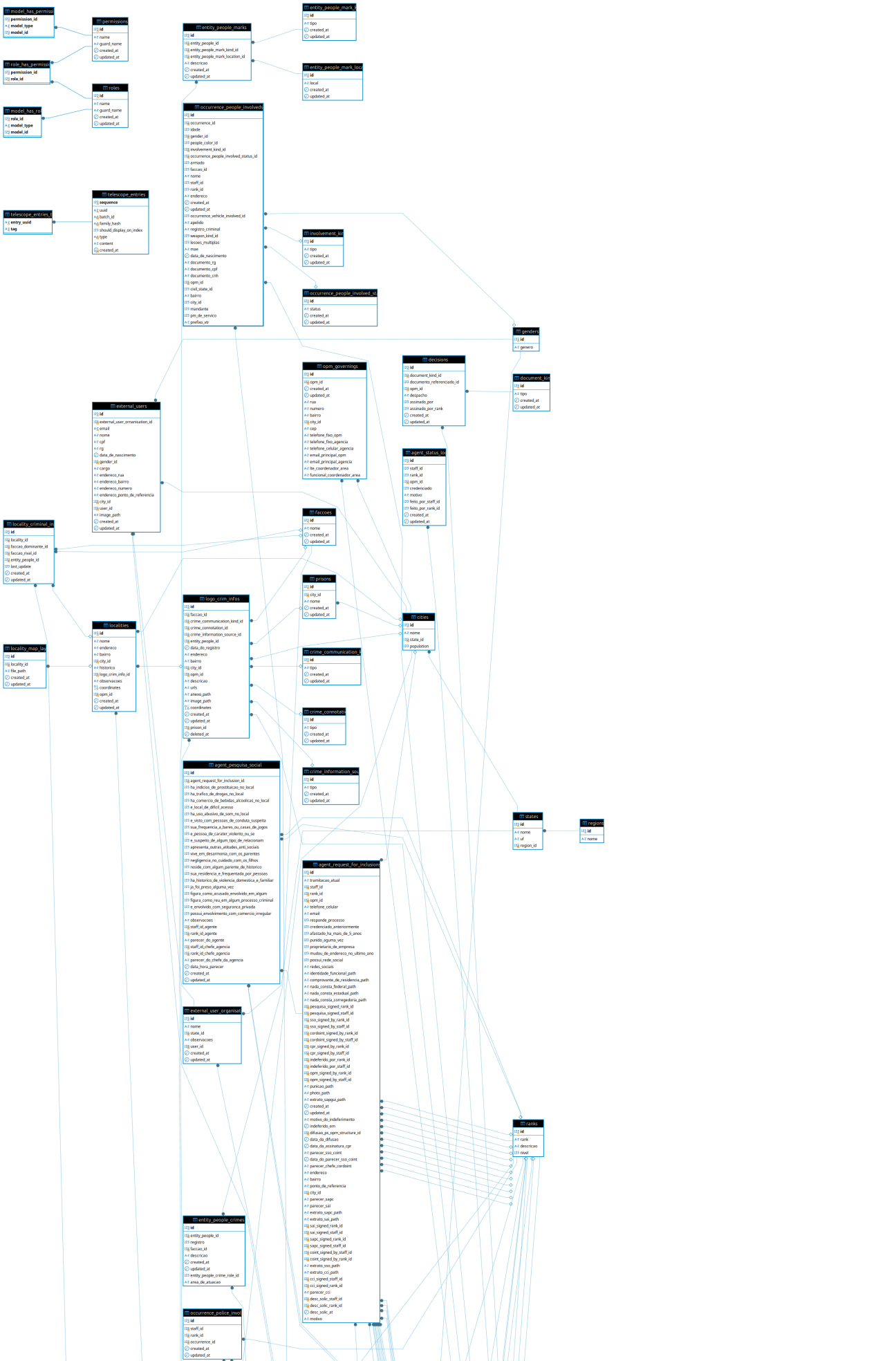


Exemplo de Consulta SQL

Lista todos os agentes de uma OPM ou de um CPR

```
   $user = Auth::user();
      $opm = Opm::find($request->opm_id);
      $isCpr = $opm ? $opm->grande_comando : null;
      $agentsQuery = User::join('staff', 'users.staff_id','=','staff.id')
         ->leftJoin('agent_request_for_inclusions',
'agent_request_for_inclusions.staff_id', '=', 'staff.id')
         ->join('opms','opms.id', '=', 'staff.opm_id')
         ->join('model_has_roles','users.id','=','model_has_roles.model_id')
         ->join('ranks','staff.rank_id','=','ranks.id')
         ->join('roles', 'roles.id', '=', 'model_has_roles.role_id')
         ->whereNotIn('roles.name', ['Comandante - OPM', 'Subcomandante - OPM', 'Comandante
- CPR', 'Subcomandante - CPR'])
         ->when(($request->opm_id == $user->staff->opm_id) and ($isCpr), function($q) use
($request){
            $q->where('opms.cpr_id', $request->opm_id);
         })
         ->when($request->opm_id, function($q) use ($request){
            $q->where('staff.opm_id',$request->opm_id);
         })
         ->when(!$request->opm_id and $user->hasAnyPermission(['Consultar agentes OPM']),
function($q) use ($user){
            $q->where('staff.opm_id',$user->staff->opm_id);
         })
         ->when(($request->staff_id) and $user->hasAnyPermission(['Consultar agentes
OPM']), function($q) use ($request, $user){
            $q->where('staff.opm_id',$user->staff->opm_id)
            ->where('users.id', Crypt::decrypt($request->staff_id));
         })
         ->when(!$request->opm_id and $user->hasAnyPermission(['Consultar agentes CPR']),
function($q) use ($user){
            $q->where('opms.cpr_id', $user->staff->opm->cpr_id);
         })
         ->when(($request->staff_id) and $user->hasAnyPermission(['Consultar agentes
CPR']), function($q) use ($request, $user){
            $q->where('opms.cpr_id', $user->staff->opm->cpr_id)
            ->where('users.id', Crypt::decrypt($request->staff_id));
         })
         ->when(($request->staff_id) and $user->hasAnyPermission(['Ver todos agentes']),
function($q) use ($request, $user){
            $q->where('users.id', Crypt::decrypt($request->staff_id));
         })->groupBy('staff.id');
      $agentsCountByRank =
$agentsQuery->select('users.*')->orderBy('ranks.nivel')->get()->countBy('staff.rank.rank');
      $agents =
```

Diagrama ER


model_has_permiss
permission_id
model_type
model_id
permissions
id
name
guard_name
created_at
updated_at
role_has_permissi
permission_id
role_id
roles
id
name
guard_name
created_at
updated_at
model_has_ro
role_id
model_type
model_id
entity_people_marks
id
entity_people_id
entity_people_mark_kind_id
entity_people_mark_location_id
descricao
created_at
updated_at
entity_people_mark_k
id
tipo
created_at
updated_at
entity_people_mark_loc
id
local
created_at
updated_at
occurrence_people_involved
id
occurrence_id
idade
gender_id
people_color_id
involvement_kind_id
occurrence_people_involved_status_id
armado
faccao_id
nome
staff_id
rank_id
endereco
created_at
updated_at
occurrence_vehicle_involved_id
apelido
registro_criminal
weapon_kind_id
lesoes_multiplas
mae
data_de_nascimento
documento_rg
documento_cpf
documento_cnh
opm_id
civil_state_id
bairro
city_id
mandante
pm_de_servico
prefixo_viv
involvement_kir
id
tipo
created_at
updated_at
occurrence_people_involved_st
id
status
created_at
updated_at
telescope_entries
sequence
uuid
batch_id
family_hash
should_display_on_index
type
content
created_at
telescope_entries_t
entry_uuid
tag
genders
id
genero
opm_governings
id
opm_id
created_at
updated_at
rua
numero
bairro
city_id
cep
telefone_fixo_opm
telefone_fixo_agencia
telefone_celular_agencia
email_principal_opm
email_principal_agencia
fte_coordenador_area
funcional_coordenador_area
decisions
id
document_kind_id
documento_referenciado_id
opm_id
despacho
assinado_por
assinado_por_rank
created_at
updated_at
document_kir
id
tipo
created_at
updated_at
external_users
id
external_user_organisation_id
email
nome
cpf
rg
data_de_nascimento
gender_id
cargo
endereco_rua
endereco_bairro
endereco_numero
endereco_ponto_de_referencia
city_id
user_id
image_path
created_at
updated_at
agent_status_loc
id
staff_id
rank_id
opm_id
credenciado
motivo
feito_por_staff_id
feito_por_rank_id
created_at
updated_at
locality_criminal_in
id
locality_id
faccao_dominante_id
faccao_rival_id
entity_people_id
last_update
created_at
updated_at
faccoes
id
nome
created_at
updated_at
prisons
id
city_id
nome
created_at
updated_at
logo_crim_infos
id
faccao_id
crime_communication_kind_id
crime_connotation_id
crime_information_source_id
entity_people_id
data_do_registro
endereco
bairro
city_id
opm_id
descricao
urls
anexo_path
image_path
coordinates
created_at
updated_at
prison_id
deleted_at
cities
id
nome
state_id
population
localities
id
nome
endereco
bairro
city_id
historico
logs_crim_info_id
observacoes
coordinates
opm_id
created_at
updated_at
locality_map_lay
id
locality_id
file_path
created_at
updated_at
crime_communication_t
id
tipo
created_at
updated_at
crime_connotati
id
tipo
created_at
updated_at
crime_information_sou
id
tipo
created_at
updated_at
agent_pesquisa_social
id
agent_request_for_inclusion_id
ha_indicios_de_prostituicao_no_local
ha_trafico_de_drogas_no_local
ha_comercio_de_bebidas_alcoolicas_no_local
e_local_de_difícil_acesso
ha_uso_abusivo_de_som_no_local
e_visto_com_pessoas_de_conduta_suspeita
sua_frequencia_a_bares_ou_casas_de_jogos
e_pessoa_de_carater_violento_ou_se
e_suspeito_de_algum_tipo_de_relacionam
apresenta_outras_atitudes_anti_sociais
vive_em_desarmonia_com_os_parentes
negligencia_no_cuidado_com_os_filhos
reside_com_algum_parente_de_historico
sua_residencia_e_frequentada_por_pessoas
ha_historico_de_violencia_domestica_e_familiar
ja_foi_preso_alguma_vez
figura_como_acusado_envolvido_em_algum
figura_como_reu_em_algum_processo_criminal
e_envolvido_com_seguranca_privada
possui_envolvimento_com_comercio_irregular
observacoes
staff_id_agente
rank_id_agente
parecer_do_agente
staff_id_chefe_agencia
rank_id_chefe_agencia
parecer_do_chefe_da_agencia
data_hora_parecer
created_at
updated_at
states
id
nome
uf
region_id
regions
id
nome
agent_request_for_inclusion
id
tramitacao_atual
staff_id
rank_id
opm_id
telefone_celular
email
responde_processo
credenciado_anteriormente
afastado_ha_mais_de_5_anos
punido_alguma_vez
proprietario_de_empresa
mudou_de_endereco_no_ultimo_ano
possui_rede_social
redes_sociais
identidade_funcional_path
comprovante_de_residencia_path
nada_consta_federal_path
nada_consta_estadual_path
nada_consta_corregedoria_path
pesquisa_signed_rank_id
pesquisa_signed_staff_id
sso_signed_by_rank_id
sso_signed_by_staff_id
cordoni_signed_by_rank_id
cordoni_signed_by_staff_id
cpr_signed_by_rank_id
cpr_signed_by_staff_id
indeferido_por_rank_id
indeferido_por_staff_id
opm_signed_by_rank_id
opm_signed_by_staff_id
punicao_path
photo_path
extrato_sapqui_path
created_at
updated_at
motivo_do_indeferimento
indeferido_em
difusao_ps_opm_structure_id
data_da_difusao
data_da_assinatura_cpr
parecer_sso_coint
data_do_parecer_sso_coint
parecer_chefe_cordaint
endereco
bairro
ponto_de_referencia
city_id
parecer_sapc
parecer_sai
extrato_sapc_path
extrato_sai_path
sai_signed_rank_id
sai_signed_staff_id
sapc_signed_rank_id
sapc_signed_staff_id
coint_signed_by_staff_id
coint_signed_by_rank_id
extrato_sso_path
extrato_cci_path
cci_signed_staff_id
cci_signed_rank_id
parecer_cci
desc_solic_staff_id
desc_solic_rank_id
desc_solic_at
motivo
external_user_organisat
id
nome
state_id
observacoes
user_id
created_at
updated_at
ranks
id
rank
descricao
nivel
entity_people_crimes
id
entity_people_id
registro
faccao_id
descricao
created_at
updated_at
entity_people_crime_role_id
area_de_atuacao
occurrence_police_involv
id
staff_id
rank_id
ocurrence_id
created_at
updated_at

Requisitos

Requisitos do Sistema

- **PHP >= 8.0:** Para o servidor
- **MySQL >= 8.0:** Para o banco de dados
- **Composer >= 2.4:** Para gerenciamento de dependências do Composer
- **Docker >= 20.10:** Para gerenciar os ambientes da aplicação.
- **Docker Compose >= 1.29:** Para gerenciamento de containeres e imagens Docker.
- **Traefik >= 2.9:** Para gerenciamento de rotas e certificados SSL.
- **Node.js >= 16:**

Configuração

Configurando o portal SINPOM para desenvolvimento local.

📄 Configuração do Ambiente

Clonando o Repositório

📄 Instalar dependências

Executar composer install para instalar as dependências do sinpom.

📄 Migrations e seeds

Executar as migrações e popular o banco de dados.

📄 Proxy/Proxy Reverso

O Traefik está configurado para gerenciar o roteamento de tráfego com base em domínios. Certifique-se de que o DNS ou o arquivo hosts aponta para os domínios configurados.

📄 Acessando o projeto

Acessando o portal sinpom em ambiente de desenvolvimento

Configuração do Ambiente

Clonando o Repositório

```
git clone https://github.com/sinpom/sinpom.git
cd sinpom
```

Configuração do Docker

1. Copie o arquivo `.env.example` para `.env`

```
cp .env.example .env
```

2. Configure as variáveis de ambiente no arquivo `.env`

Você pode usar o arquivo `.env.local` para configuração local do sinpom.

3. Inicie os containeres com o Docker Compose

Para ambiente de desenvolvimento em Ubuntu:

Criar containeres

```
docker compose -f docker-compose-dev-ubuntu.yml up -d
```

Remover containeres

```
docker compose -f docker-compose-dev-ubuntu.yml down
```

Para ambiente local:

Criar containeres

```
docker compose -f docker-compose-local.yml up -d
```

Remover containeres

```
docker compose -f docker-compose-local.yml down
```

> ⚠ **WARNING**
>
> Use o arquivo `docker-compose-local.yml` para configuração local do sinpom ou use `docker-compose-dev-ubuntu.yml` para ambiente de desenvolvimento em Ubuntu(#).

Para ambiente em produção:

Criar containeres

```
docker compose -f docker-compose-producao.yml up -d
```

Instalar dependências

Executar **composer install** para instalar as dependências do sinpom.

> **INFO**
>
> O comando **composer install** deve ser executado dentro do container do portal sinpom

Execute o comando abaixo para instalar as dependências do projeto.

```
docker exec -it sinpom-web bash -c "composer install"
```

Migrations e seeds

Executar as migrações e popular o banco de dados.

Migrações

Criando as tabelas no banco de dados.

> **INFO**
>
> O comando **php artisan migrate** deve ser executado dentro do container do portal sinpom

Execute o comando abaixo para executar a migrações do banco de dados.

```
docker exec -it sinpom-web bash -c "php artisan migrate"
```

Você pode verificar se o sistema está funcionando corretamente, acesse o portal sinpom em:

- https://sinpom.docker.localhost

Populando o banco de dados

> **INFO**
>
> O comando **php artisan db:seed** deve ser executado dentro do container do portal sinpom

Execute o comando abaixo para popular o banco de dados.

```
docker exec -it sinpom-web bash -c "php artisan db:seed"
```

Proxy/Proxy Reverso

O Traefik está configurado para gerenciar o roteamento de tráfego com base em domínios. Certifique-se de que o DNS ou o arquivo hosts aponta para os domínios configurados.

Usando o Traefik com Docker

> ⚠ **WARNING**
>
> O Traefik precisa ser executado em um container separado, fora do diretório do sinpom.

Você pode criar um arquivo `docker-compose.yml` para configuração do Traefik, fora do diretório do sinpom e então gerencia-lo com o Docker Compose.

```
# Saindo do diretório do sinpom
cd ../
# Criando diretório para o traefik
mkdir traefik
cd traefik
touch docker-compose.yml
touch dynamic.yml
```

Exemplo de arquivo `docker-compose.yml` para configuração do Traefik:

```yaml
networks:
  proxy:
    external: true
```

Use um arquivo `dynamic.yml` para configuração do Traefik. Exemplo:

```
tls:
  certificates:
    - certFile: /certs/cert.cert
      keyFile: /certs/cert.key
```

Configurando o hosts

No arquivo de hosts do seu sistema operacional, adicione a seguinte linha:

```
127.0.0.1 sinpom.docker.localhost
```

Exemplo de edição do Hosts no Ubuntu

O arquivo de **hosts** no Ubuntu é usado para mapear endereços IP a nomes de domínio de forma local, sem a necessidade de configurar um servidor DNS.

Passo 1: Abrir o Arquivo de Hosts com Permissões de Superusuário

O arquivo de hosts está localizado em `/etc/hosts`. Para editá-lo, você precisa de permissões de administrador. Use o editor de texto de sua preferência (neste exemplo, usamos `nano`).

Digite o seguinte comando no terminal:

```
sudo nano /etc/hosts
```

Passo 2: Editar o arquivo

O conteúdo do arquivo /etc/hosts será exibido no editor. Ele geralmente terá esta aparência inicial:

```
127.0.0.1 localhost
127.0.0.1 seu-host-local
```

Adicione a entrada necessária

```
127.0.0.1 sinpom.docker.localhost
```

Passo 3: Salvar e Fechar o Arquivo

```
Ctrl + O
Enter
Ctrl + X
```

Passo Opcional: Limpar o Cache DNS (se necessário)

Se você tiver problemas para resolver o domínio, limpe o cache de DNS. Para sistemas que usam o systemd-resolved, execute:

```
sudo systemctl restart systemd-resolved
```

Acessando o projeto

Acessando o portal sinpom em **ambiente de desenvolvimento**

Desenvolvimento local

Após as etapas passadas você já pode acessar o portal sinpom em:

- https://sinpom.docker.localhost

Use um dos usuários do sistema para acessar o portal sinpom.

Usuário nível **Agente**:

- Usuário: **agente@sinpom.com**
- Senha: **123**

Desenvolvimento

Configurando o portal SINPOM para desenvolvimento local.

📄 Controle de Versão

Práticas de Versionamento

📄 Casos de Uso

Login no sistema

📄 Detalhamento de Funcionalidades

Em breve

📄 Plano de Testes

Em breve

Controle de Versão

Práticas de Versionamento

- A branch `master` é a versão estável do sistema
- Desenvolva novas funcionalidades em uma branch diferente da `master` ex: 'feature/funcionalidade-nova'

Fluxo de Trabalho em Desenvolvimento

1. Crie uma branch para a nova funcionalidade

```
git checkout -b 'feature/funcionalidade-nova'
```

2. Depois de implementar as alterações, faça o commit e envie a branch para o GitHub:

```
git add .
git commit -m 'Adicionando nova funcionalidade'
git push origin 'feature/funcionalidade-nova'
```

- Use boas práticas de commit messages, conforme convention commits

Casos de Uso

Login no sistema

Descrição: O usuário deve realizar login no sistema para acessar os recursos do portal.
Atores: O usuário. **Pre condições:**

- O usuário já possui uma conta ativa no sistema

Fluxo Principal:

1. O usuário acessa a página de login: https://sinpom.docker.localhost/login
2. O usuário preenche os campos de e-mail e senha.
3. O usuário clicar em "Entrar".
4. Após validar as credenciais o sistema solicita o OTP para autenticação
5. O sistema valida o OTP e redireciona o usuário para a página de inicial

Fluxos Alternativos:

- **F1**: Caso as credenciais sejam inválidas, o sistema exibe uma mensagem de erro.
- **F2**: Uso do sistema sem OTP

1. O usuário acessa a página de login: https://sinpom.docker.localhost/login
2. O usuário preenche os campos de e-mail e senha.
3. O usuário clicar em "Entrar".
4. O sistema verifica as credenciais e redireciona o usuário para a página de inicial

---

Detalhamento de Funcionalidades

> **INFO**
>
> Em breve

Plano de Testes

> **INFO**
>
> Em breve

Implantação

Configurando o portal SINPOM em um ambiente de produção.

📄 Estratégia de Deploy

Verificar com o desenvolvedor a estratégia de deploy

📄 Backup e Recuperação

Definir se vai ser adicionada a documentação online ou mantida em arquivo separado.

Estratégia de Deploy

> **WARNING**
>
> Verificar com o desenvolvedor a estratégia de deploy

Backup e Recuperação

> ⚠ **WARNING**
>
> Definir se vai ser adicionada a documentação online ou mantida em arquivo separado. Verificar com o desenvolvedor a estratégia de backup e recuperação

Referências

📄 Changelog

Adicionar historico de alterações no sistema

Changelog

> **INFO**
>
> Adicionar historico de alterações no sistema

Contribuição

- Siga o padrão de commits convencional
- Siga os padrões de código do Laravel e as melhores práticas de desenvolvimento do PHP

Autor

- Cap PM **Vailson Marcelo** - github.com/vailsonmarcelo

Documentos Externos

> ⚠ **WARNING**
>
> Documentos que devem ser mantidos fora da documentação técnica online.

- Manual do usuário
- Plano de Gerenciamento de Riscos
- Matriz de Responsabilidades (RACI)
- Política de Backup e Recuperação (Avaliar se será incluida ou não na seção monitoramento e manutenção)