

# SAÚDE MÓVEL

Ana Cristina Moura

# Contextualização do projeto

- Este projeto foi criado para atender aos requisitos das disciplinas de Criação de empresas para Internet, Tópicos especiais em sistemas para Internet III e Desenvolvimento para dispositivos móveis II.
- Um dos requisitos é estar alinhado com o Terceiro Objetivo de Desenvolvimento Sustentável “Saúde e Bem-estar” e desenvolver uma solução para um problema real social.

# Problemática e Solução

- **Problema:** Falta de comunicação na saúde pública brasileira
- **Solução:** Aplicativo mobile que conecta os usuários a médicos e/ou entidades de saúde, oferecendo informações detalhadas e divulgação de campanhas de saúde.

# Ferramenta Utilizada

- ***Business Model Canvas (BMC)***: Estruturação do modelo de negócio (segmento de clientes, proposta de valor, canais, relacionamento com clientes, fontes de receita, recursos principais, atividades-chave, parcerias principais e estrutura de custos).

# Business Model Canvas - Detalhes

- **Segmento de Clientes:** Pacientes, familiares, profissionais de saúde e organizações de saúde.
- **Proposta de Valor:** Acesso fácil e conveniente a serviços de saúde, informações detalhadas e campanhas de saúde.
- **Canais:** Distribuição via lojas de aplicativos e *marketing* digital.
- **Relacionamento com os Clientes:** Suporte eficiente, *feedback* dos usuários e estratégias de engajamento.
- **Fontes de Receita:** Publicidade, parcerias e serviços de agendamento.
- **Recursos Principais:** Plataforma de *software*, banco de dados de saúde, integração com mapas e equipe de desenvolvimento.
- **Atividades:** Desenvolvimento e manutenção do aplicativo, atualização de informações, marketing e parcerias estratégicas.
- **Parcerias Principais:** Instituições médicas, organizações de saúde, governos e agências de *marketing*.
- **Estrutura de Custos:** Desenvolvimento e manutenção do aplicativo, servidores, equipe e campanhas de *marketing*.

# Canvas

## Principais Alianças

- Instituições médicas e organizações de saúde.
- Governos locais e nacionais para campanhas de saúde.
- Agências de marketing e publicidade.

## Principais Alianças

- Desenvolvimento e manutenção do aplicativo.
- Atualização contínua de informações e dados médicos.
- Marketing e promoção para aumentar a base de usuários.
- Parcerias estratégicas com instituições de saúde e organizações.

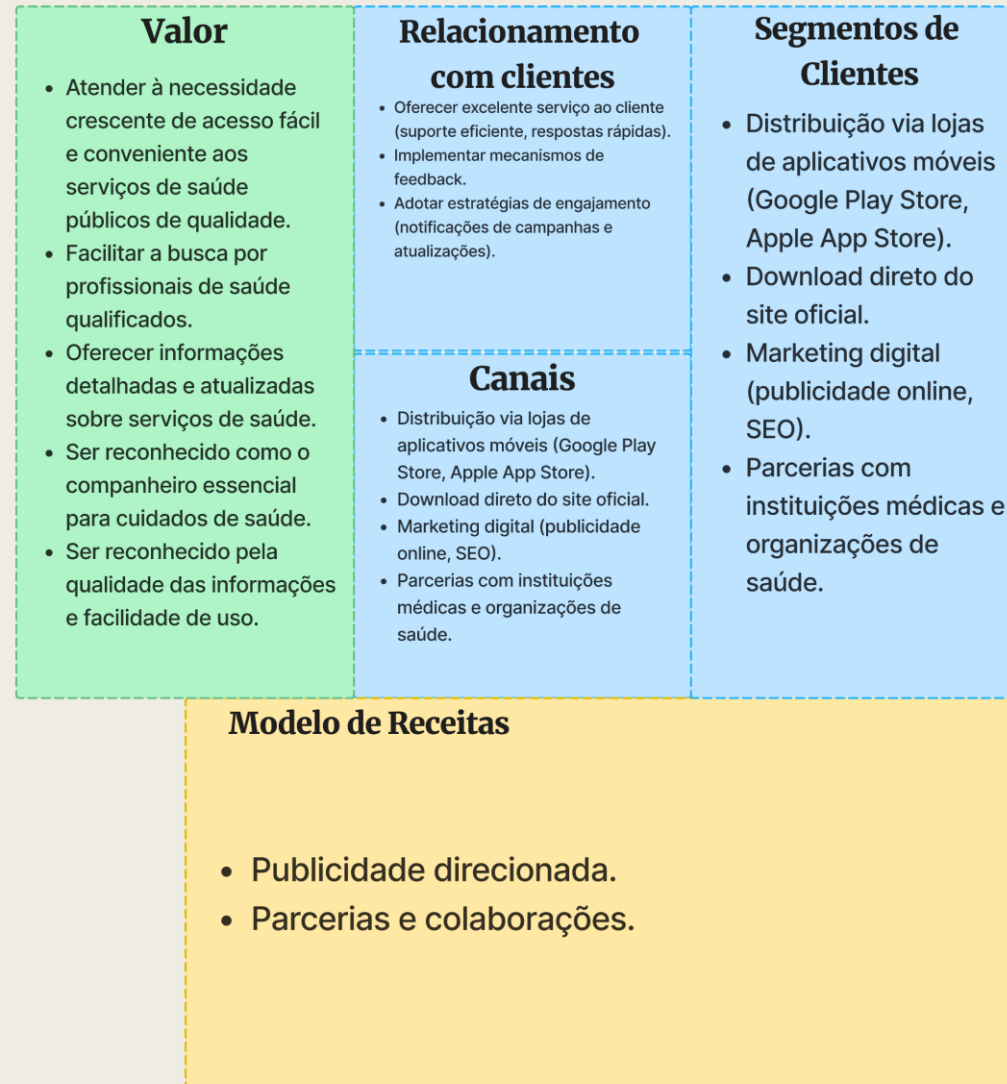
## Principais Recursos

- Plataforma de software robusta e escalável.
- Banco de dados abrangente de profissionais de saúde e instalações médicas.
- Equipe de desenvolvimento e suporte ao cliente.

## Estrutura de Custos

- Desenvolvimento e manutenção do aplicativo.
- Servidores e infraestrutura de TI.
- Marketing e publicidade.

# Canvas



# Requisitos

## ■ Funcionais:

- **Pesquisa avançada:** o sistema deve possibilitar aos usuários realizarem buscas detalhadas por especialidade médica, nome do médico, nome da clínica, sintomas, vacinas, exames, entre outros.
- **Listagem de profissionais de saúde ou instalações médicas:** o sistema deve exibir uma lista organizada por proximidade em relação ao local onde o usuário se encontra.
- **Detalhes do profissional ou da instituição:** cada item no guia deve incluir informações detalhadas, como o nome do médico ou instalação, especialidade, endereço, telefone, horário de funcionamento e uma breve biografia ou descrição dos serviços oferecidos.
- **Navegação integrada:** o sistema deve oferecer integração com serviços de mapas do próprio dispositivo, permitindo que os usuários visualizem a localização dos médicos e instalações médicas em um mapa e obtenham direções para chegar até lá.



# Requisitos

## ■ Não funcionais:

- **Compatibilidade com dispositivos móveis:** o aplicativo deve ser compatível com uma variedade de dispositivos móveis, como smartphones e tablets, para garantir uma experiência consistente para os usuários.
- **Usabilidade:** o sistema deve ser intuitivo e fácil de usar, com uma interface amigável que permita uma navegação fluida e sem complicações.

# Casos de Uso com Regras de Negócio

- **Buscar médicos:** o usuário pode pesquisar por médicos com base em critérios como especialidade médica, nome do médico etc. O fluxo de interação começa com o usuário acessando a função de busca de médicos no aplicativo, inserindo os critérios de pesquisa desejados e, então, o sistema exibe uma lista de médicos correspondentes aos critérios fornecidos, priorizando a proximidade com a localização em tempo real do usuário. A regra de negócio associada é que os médicos listados no aplicativo devem estar devidamente registrados e autorizados a exercer a profissão.
- **Informações e contato:** o usuário pode acessar informações detalhadas sobre as instituições ou os profissionais, incluindo localização, serviços oferecidos e formas de contato. O fluxo de interação envolve o usuário acessando a função de busca, selecionando um resultado específico da lista apresentada e o sistema exibindo detalhes sobre a instituição ou profissional. A regra de negócio é que as informações fornecidas sobre hospitais devem ser atualizadas regularmente para garantir precisão e confiabilidade.

# Casos de Uso com Regras de Negócio

- **Campanhas de saúde:** o sistema informa sobre campanhas de saúde, incluindo datas, locais e público-alvo, promovendo a conscientização e a participação. Essas campanhas são personalizadas de acordo com a localização atual do usuário. O fluxo de interação é o sistema exibindo informações sobre campanhas de saúde na página inicial ou em uma seção dedicada e o usuário navegando pelas diferentes campanhas listadas para obter mais detalhes. A regra de negócio estabelece que as informações sobre campanhas de saúde devem ser atualizadas regularmente e provenientes de fontes confiáveis.
- **Processo para realizar teste laboratoriais:** o usuário pode encontrar informações sobre locais de realização de testes laboratoriais e entender o processo necessário para realizá-los. O fluxo de interação começa com o usuário acessando a função de busca, seguido pelo sistema exibindo uma lista de laboratórios disponíveis para realizar testes, após isso, o usuário seleciona um laboratório específico para obter mais informações, como procedimentos de agendamento e pré-requisitos. A regra de negócio é que as informações sobre os laboratórios devem incluir os tipos de testes oferecidos, horários de funcionamento e informações sobre agendamento, quando aplicável.

# Casos de Uso com Regras de Negócio

- **Integração com o serviço de mapa do celular:** o usuário pode visualizar a localização dos médicos, hospitais e outras instalações de saúde no mapa do celular para facilitar a navegação. O fluxo de interação é o usuário selecionando um médico, hospital ou outra instalação de saúde no aplicativo, optando por visualizar a localização no mapa do celular e o sistema integrando-se com o serviço de mapa do celular. O usuário pode visualizar a localização no mapa, obter direções e navegar até o destino. A regra de negócio é que a integração com o serviço de mapa do celular deve ser compatível com os principais aplicativos de mapas disponíveis nos dispositivos móveis dos usuários.

# Protótipo - Vídeo

- Para ver o vídeo do protótipo deste projeto:
  - *É necessário ter conta no google*
  - [https://drive.google.com/file/d/1HuXKzTTNk\\_qXyFawPUTlu1DQ\\_Uo3SLbp/view?usp=sharing](https://drive.google.com/file/d/1HuXKzTTNk_qXyFawPUTlu1DQ_Uo3SLbp/view?usp=sharing)

9:41



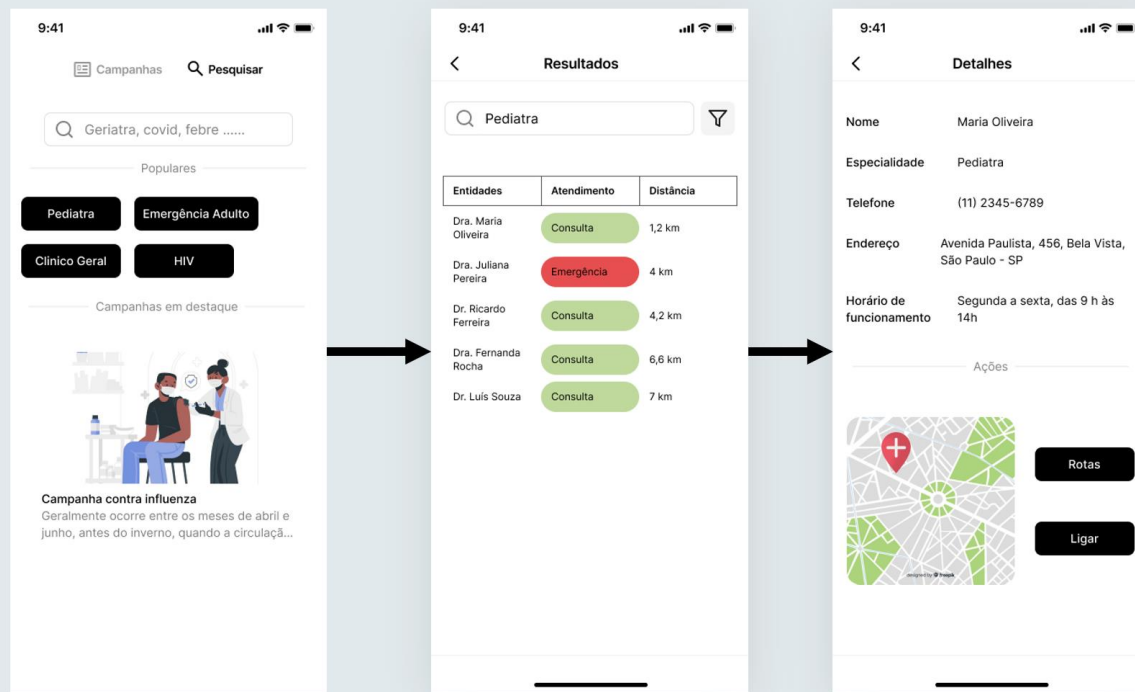
Saúde Móvel



Desenvolvido por **Ana Moura**

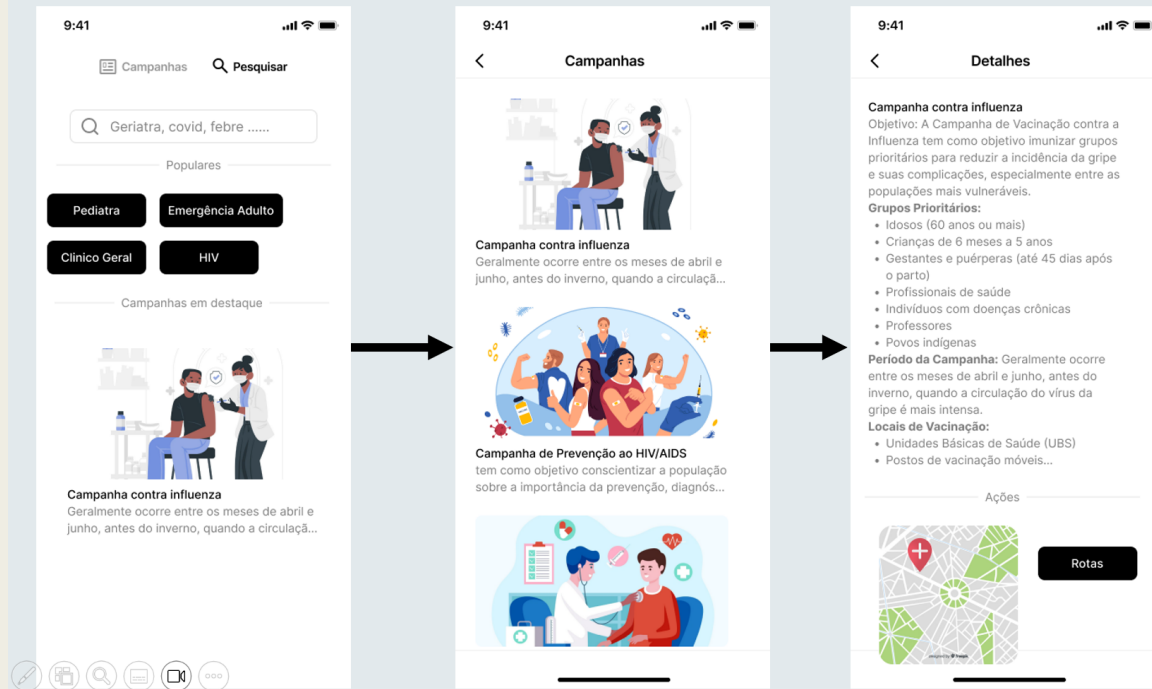
# PROTÓTIPO - SPLASHSCREEN

## Telas para buscar um profissional pediatra



# PROTÓTIPO – FUNÇÃO BUSCA

## Telas para campanhas



# PROTÓTIPO - CAMPANHAS



# Banco de Dados – Pré modelagem

- A proposta do aplicativo mobile é garantir o acesso à saúde pública brasileira. A solução foi criar um aplicativo que servirá como um guia médico. Este guia digital é uma aplicação móvel projetada para ajudar os usuários a encontrarem facilmente profissionais de saúde, instalações médicas (enfermarias, UBS, UPA, farmácias, ambulatórios, clínicas etc.) e serviços de saúde públicos próximos a eles através de uma simples busca no banco de dados, utilizando palavras-chaves como "Pediatra", "Fratura", "Hospital", "Emergência", "Dentista", "covid-19", "dengue" etc. O aplicativo acessará a localização do usuário e deverá responder à busca dele com uma lista limitada a um raio de 10 km de distância.
- Neste sistema, o usuário não possui login, pois o objetivo do aplicativo é ser um guia médico onde o usuário pesquisa por palavras-chave. Imitando assim os antigos modelos impressos de guia médico.
- Quando o usuário pesquisar um profissional da área de saúde como "Clínico geral", "Pediatra", "Dentista", "Geriatra", "Endocrinologista" etc., o resultado da busca deve ser uma lista com os nomes dos profissionais que estarão mais perto do usuário. Assim, ao cadastrar um profissional no banco de dados, é necessário incluir seu endereço. Ao clicar em um profissional, o usuário será levado para uma nova tela com as informações do profissional escolhido e terá as seguintes informações: nome, profissão, telefone, endereço e horário de funcionamento.
- É importante diferenciar especialidade médica de profissão.

# Banco de Dados – Pré modelagem

- Cada um desses profissionais foca em diferentes aspectos da saúde e no tratamento de várias condições.
- Independente do termo digitado, especialidade médica ou profissional o resultado da busca deve listar os profissionais de saúde e as instituições que abrange essa categoria. No entanto, quando o usuário pesquisar de uma profissão, o resultado deve listar somente os profissionais de saúde.

# Banco de Dados – Pré modelagem

- Cada instituição tem um nome, endereço, telefone, horário de funcionamento. O usuário pode pesquisar as instituições pelas siglas/abreviações, nome ou tipo de atendimento que elas oferecem, por exemplo, “Psiquiatria”, “Psicologia”, “Assistentência social”, “Terapia Ocupacional”, etc (especialidades médicas).
- O usuário poderá pesquisar nomes de doenças.
- Cada doença está relacionada a um único profissional, funcionando como um par de chaves, exemplo:
  - (*dengue, infectologia*),
  - (*chikungunya, infectologia*),
- Então quando o usuário pesquisar doenças, o sistema deve retornar uma lista com os profissionais de saúde indicados para tratar as doenças.
- O usuário pode pesquisar palavras chaves de exames. Como resposta a esta busca o sistema deve retornar uma lista de instituições que oferecem tal exame.

# Banco de Dados – Modelagem

- O banco de dados **saudemovel** foi projetado para suportar um aplicativo mobile que atua como um guia médico, ajudando os usuários a encontrarem profissionais de saúde, instituições médicas e serviços de saúde públicos. A modelagem do banco de dados inclui várias entidades inter-relacionadas para capturar informações detalhadas sobre especialidades médicas, profissões, profissionais de saúde, instituições e exames oferecidos.

# Banco de Dados - Tabelas

- Tabela **Cidades**:
  - *id*: Identificador único da cidade.
  - *nome*: Nome da cidade.
- Tabela **Bairros**:
  - *id*: Identificador único do bairro.
  - *nome*: Nome do bairro.
  - *cidade\_id*: Referência à cidade a que o bairro pertence.
- Tabela **Especialidade**:
  - *id*: Identificador único da especialidade médica.
  - *nome*: nome da especialidade.
- Tabela **Profissoes**:
  - *id*: Identificador único da profissão.
  - *nome*: Nome da profissão.
  - *especialidade\_id*: Referência à especialidade médica relacionada.

# Banco de Dados - Tabelas

## ■ Tabela Profissionais:

- *id*: Identificador único do profissional de saúde.
- *nome*: Nome do profissional.
- *profissao\_id*: Referência à profissão do profissional.
- *telefone*: Telefone de contato.
- *endereco*: Endereço do profissional.
- *bairro\_id*: Referência ao bairro onde o profissional atende.
- *cidade\_id*: Referência à cidade onde o profissional atende.
- *latitude* e *longitude*: Coordenadas geográficas do endereço.

## ■ Tabela HorariosFuncionamento:

- *id*: Identificador único do horário de funcionamento.
- *profissional\_id*: Referência ao profissional de saúde.
- *dia\_da\_semana*: Dia da semana.
- *hora\_inicio* e *hora\_fim*: Horário de início e fim de atendimento.

# Banco de Dados - Tabelas

## ■ Tabela Instituicoes:

- *id*: Identificador único da instituição de saúde.
- *nome*: Nome da instituição.
- *categoria*: Categoria da instituição.
- *sigla*: Sigla ou abreviação da instituição.
- *telefone*: Telefone de contato.
- *endereco*: Endereço da instituição.
- *bairro\_id*: Referência ao bairro onde a instituição está localizada.
- *cidade\_id*: Referência à cidade onde a instituição está localizada.
- *latitude e longitude*: Coordenadas geográficas do endereço.
- *horario\_funcionamento*: Horário de funcionamento da instituição.

## ■ Tabela InstituicaoEspecialidade:

- *instituicao\_id*: Referência à instituição.
- *especialidade\_id*: Referência à especialidade médica oferecida.

# Banco de Dados - Tabelas

## ■ Tabela Doencas:

- *id*: Identificador único da doença.
- *nome*: Nome da doença.
- *profissao\_id*: Referência à profissão do profissional de saúde relacionado ao tratamento da doença.

## ■ Tabela Exames:

- *id*: Identificador único do exame.
- *nome*: Nome do exame.
- *instituicao\_id*: Referência à instituição que oferece o exame.



# Banco de Dados - Relações

- **Cidades e Bairros:** Uma cidade pode ter muitos bairros, mas cada bairro pertence a uma única cidade.
- **Especialidade e Profissões:** Uma especialidade pode ter muitas profissões relacionadas, mas cada profissão pertence a uma única especialidade.
- **Profissões e Profissionais:** Uma profissão pode ter muitos profissionais, mas cada profissional exerce uma única profissão.
- **Bairros e Profissionais:** Um bairro pode ter muitos profissionais, mas cada profissional está localizado em um único bairro.
- **Cidades e Profissionais:** Uma cidade pode ter muitos profissionais, mas cada profissional está localizado em uma única cidade.
- **Profissionais e Horários de Funcionamento:** Um profissional pode ter muitos horários de funcionamento, mas cada horário de funcionamento está associado a um único profissional.

# Banco de Dados - Relações

- **Bairros e Instituicoes:** Um bairro pode ter muitas instituições, mas cada instituição está localizada em um único bairro.
- **Cidades e Instituicoes:** Uma cidade pode ter muitas instituições, mas cada instituição está localizada em uma única cidade.
- **Especialidade e InstituicaoEspecialidade:** Uma especialidade pode estar relacionada a muitas instituições e vice-versa.
- **Profissoes e Doencas:** Uma profissão pode estar relacionada a muitas doenças, mas cada doença está relacionada a uma única profissão.
- **Instituicoes e Exames:** Uma instituição pode oferecer muitos exames, mas cada exame é oferecido por uma única instituição.

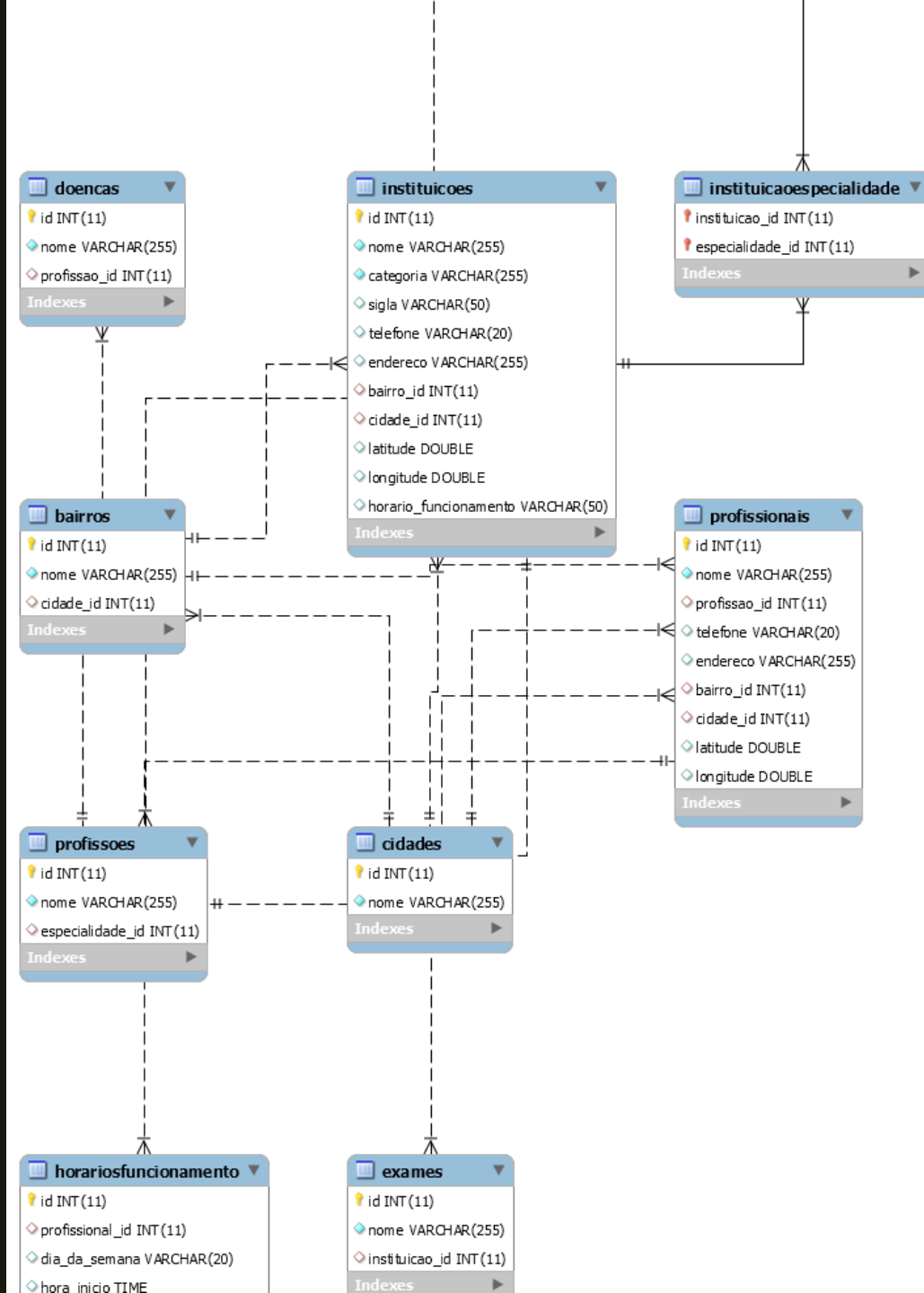
# Banco de Dados - Consultas

- Pesquisa por Especialidade:
  - *Retorna resultados das tabelas profissionais e instituições baseados na especialidade pesquisada.*
- Pesquisa por Profissão:
  - *Retorna apenas resultados da tabela profissionais baseados na profissão pesquisada.*
- Filtro de Instituições por Especialidade:
  - *Retorna apenas resultados da tabela instituição baseados na especialidade pesquisada.*
- Filtro de Profissionais por Especialidade:
  - *Retorna apenas resultados da tabela profissional baseados na especialidade pesquisada.*
- Pesquisa por Exames:
  - *Retorna uma lista de instituições que realizam um exame específico baseado na palavra-chave digitada pelo usuário.*
- Pesquisa por Nome de Instituição, Sigla ou Categoria:
  - *Retorna registros de instituições que contenham a palavra-chave nas colunas de nome, sigla ou categoria.*
- Pesquisa por Doença:
  - *Retorna profissionais de saúde indicados para tratar uma doença específica.*

# Banco de Dados - Consultas

- Pesquisa por Início de Nome de Doença:
  - *Retorna doenças e os profissionais de saúde relacionados baseados nas primeiras letras do nome da doença digitadas pelo usuário.*
- Pesquisa por Nome de Profissional ou Especialidade:
  - *Retorna profissionais ou instituições baseados nas primeiras letras do nome da especialidade ou profissão digitadas pelo usuário.*
- Essas consultas permitem aos usuários do aplicativo realizarem buscas eficazes e obter informações detalhadas sobre profissionais de saúde, instituições e serviços disponíveis, facilitando o acesso aos cuidados médicos necessários.

# BANCO DADOS



# Desenvolvimento Back-end

- O *front-end* é o que seus usuários veem e inclui elementos visuais, como botões, caixas de seleção, gráficos e mensagens de texto. Ele permite que seus usuários interajam com sua aplicação. O *back-end* consiste nos dados e na infraestrutura que fazem sua aplicação funcionar (AWS, 2024).

# Desenvolvimento Back-end – Tecnologias




- Docker para Windows:
  - *Necessário ter os recursos wsl 2 e ubuntu devidamente instalados.*
  - <https://docs.docker.com/desktop/install/windows-install/> (Docker)
  - <https://learn.microsoft.com/pt-br/windows/wsl/install-manual> (WSL e Ubuntu)
- Visual Studio Code:
  - *Editor de código*
  - <https://code.visualstudio.com/download>
- Imagens docker:
  - <https://laravel.build/example-app?with=mysql> (laravel e MySQL)

# Desenvolvimento Back-end - Container










- No terminal WSL, digite o comando:
  - **`curl -s https://laravel.build/example-app?with=mysql | bash`**
    - Troque **example-app** pelo nome do seu projeto
- Depois, no diretório do projeto, faça:
  - Digite o comando: **`code .`**, para abrir o projeto no VS Code
- Para construir e rodar os dois containers (aplicação e banco de dados):
  - No terminal integrado do VS code, abra o terminal do WSL e digite:  
**`./vendor/bin/sail up -d`**. Para verificar use: **`./vendor/bin/sail ps`**.
- Para configurar o Laravel:
  - Depois que os containers estiverem rodando, você pode acessar o container do Laravel para executar comandos adicionais, como a migração do banco de dados: **`./vendor/bin/sail artisan migrate`**.



# IMAGENS DOCKER

<input type="checkbox"/>	Name	Tag	Status
<input type="checkbox"/>	<a href="#">sail-8.3/app</a> 83ed7ba9f29a 	latest	<a href="#">In use</a>
<input type="checkbox"/>	<a href="#">laravelsail/php83-composer</a> 6569b0956b65 	latest	<a href="#">In use</a>
<input type="checkbox"/>	<a href="#">mysql/mysql-server</a> 1d9c2219ff69 	8.0	<a href="#">In use</a>

# CONTAINER DOCKER

	Name	Image	Status	Port(s)
<input checked="" type="checkbox"/>	  <a href="#">saude-movel-1</a>		Running (2/2)	
<input checked="" type="checkbox"/>	 <a href="#">mysql-1</a> add358d130d4 	<a href="#">mysql/mysql-serv</a>	Running	<a href="#">3306:3306</a> 
<input type="checkbox"/>	 <a href="#">laravel.test-1</a> 20dc96fba6c6 	<a href="#">sail-8.3/app</a>	Running	<a href="#">5173:5173</a>  <a href="#">Show all ports (2)</a>

# Desenvolvimento Back-end – Laravel + sail + mysql

- Laravel Sail é um ambiente de desenvolvimento minimalista para aplicações Laravel, alimentado por Docker. Ele oferece uma interface simples para gerenciar os diferentes serviços necessários para o desenvolvimento de aplicações Laravel, como PHP, MySQL, Redis, e outros, sem a necessidade de instalar esses serviços diretamente na sua máquina local.

# Desenvolvimento Back-end – Container MySQL

- Para configurar o Banco de Dados:
  - *Dentro do seu contêiner MySQL, você pode executar o script SQL fornecido para criar e popular o banco de dados.*
    - **docker-compose exec mysql mysql -u root -p** (para entrar no container do MYSQL)
  - *No prompt do MySQL, execute o script sql. Para verificar se deu certo digite o comando: **show databases ;***
  - *Dentro do container do mysql, os comandos devem ser finalizados com ponto e vírgula.*

# Desenvolvimento Back-end – Container Laravel

- Para criar as migrations:
  - `php artisan make:migration create_cidades_table`
- Para criar os modelos:
  - `docker-compose exec laravel.test php artisan make:model Cidade -m`
- Para criar os controllers:
  - `docker-compose exec laravel.test php artisan make:controller CidadeController`
  - *Depois implemente pois no controlador lida com as requisições HTTP*
- Configure as rotas:
  - *No arquivo **routes/api.php** e defina as rotas para a sua API*

# EXEMPLO DE MIGRATION

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateCidadesTable extends Migration
{
    public function up()
    {
        Schema::create('cidades', function (Blueprint $table) {
            $table->id();
            $table->string('nome', 255);
            $table->timestamps();
        });
    }

    public function down()
    {
        Schema::dropIfExists('cidades');
    }
}
```

# EXEMPLO DE CONTROLLER

```
<?php

namespace App\Http\Controllers;

use App\Models\Cidade;
use Illuminate\Http\Request;

class CidadeController extends Controller
{
    public function index()
    {
        return Cidade::all();
    }

    public function show($id)
    {
        return Cidade::find($id);
    }

    public function store(Request $request)
    {
        return Cidade::create($request->all());
    }

    public function update(Request $request, $id)
    {
        $cidade = Cidade::findOrFail($id);
        $cidade->update($request->all());
        return $cidade;
    }

    public function destroy($id)
    {
        Cidade::destroy($id);
    }
}
```

# EXEMPLO ROTAS

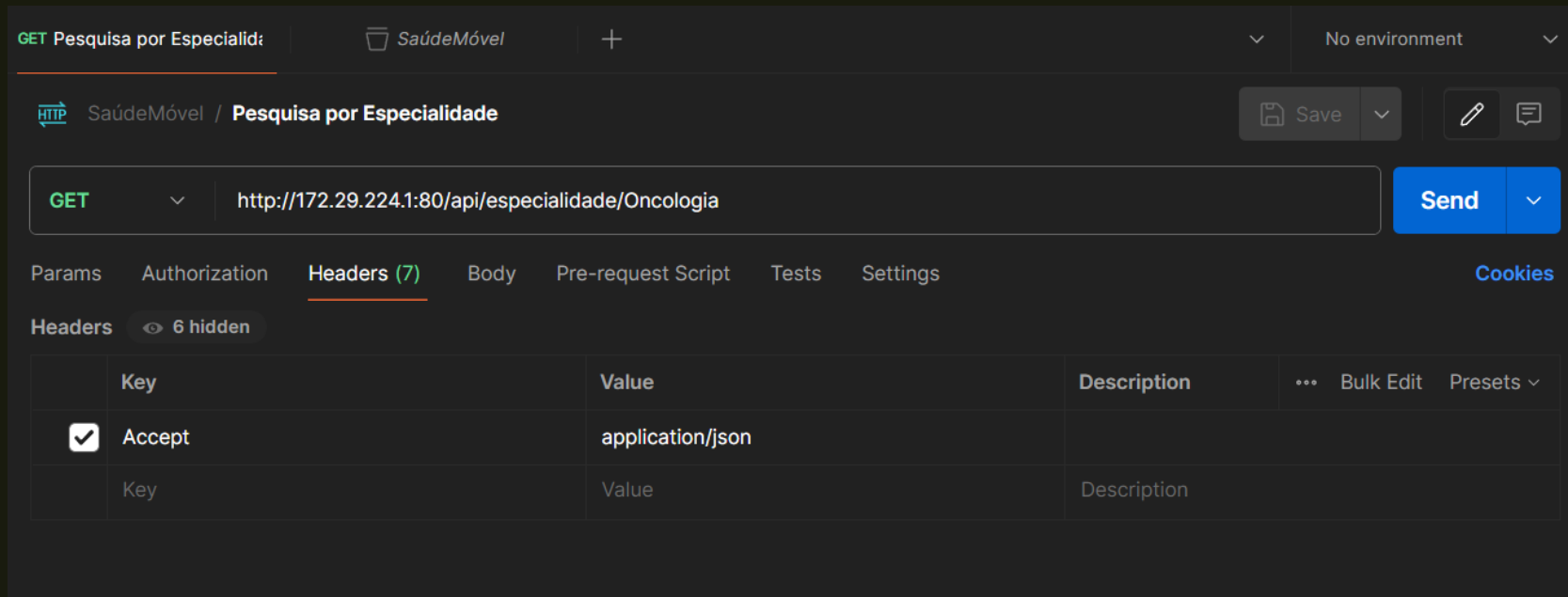
```
<?php

use App\Http\Controllers\EspecialidadeController;
use App\Http\Controllers\ProfissaoController;
use App\Http\Controllers\ProfissionalController;
use App\Http\Controllers\InstituicaoController;
use App\Http\Controllers\ExameController;

Route::get('/especialidade/{specialty}', [EspecialidadeController::class, 'searchBySpecialty']);
Route::get('/profissao/{profession}', [ProfissaoController::class, 'searchByProfession']);
Route::get('/instituicao/especialidade/{specialty}', [InstituicaoController::class, 'searchBySpecialtyForInstitutions']);
Route::get('/profissional/especialidade/{specialty}', [ProfissionalController::class, 'searchBySpecialtyForProfessionals']);
Route::get('/exame/{exam}', [ExameController::class, 'searchByExam']);
Route::get('/instituicao/unidade/{unit}', [InstituicaoController::class, 'searchByUnit']);
Route::get('/doenca/{disease}', [ProfissionalController::class, 'searchByDisease']);
Route::get('/doenca/nome/{name}', [ProfissionalController::class, 'searchDiseaseByName']);
```



# DESENVOLVIMENTO BACK-END – TESTE NO POSTMAN



# Para saber mais

- Link para o github do projeto:
  - <https://github.com/fateclins/ana.saudemovel>
- Link para o protótipo do projeto:
  - *Necessário ter conta no figma*
  - <https://www.figma.com/design/9n5GPliCmX9tyBpzvFKUqi/Sa%C3%BAdemove-M%C3%B3vel?node-id=0-1&t=OMwWDGiAZz1NMuUr-1> (Design)
  - <https://www.figma.com/proto/9n5GPliCmX9tyBpzvFKUqi/Sa%C3%BAdemove-M%C3%B3vel?node-id=22-652&t=OMwWDGiAZz1NMuUr-0&scaling=scale-down&content-scaling=fixed&page-id=0%3A1&starting-point-node-id=22%3A652> (protótipo)
- Documentação e imagens deste projeto:
  - [https://drive.google.com/drive/folders/1-Xwos8K\\_f5ZX9n4IIOkWcUpw6ZvlQ9vw?usp=sharing](https://drive.google.com/drive/folders/1-Xwos8K_f5ZX9n4IIOkWcUpw6ZvlQ9vw?usp=sharing)

# Contato com a autora

- Meu portfólio:
  - <https://portfolio.fateclins.edu.br/~1920832121016/>
- LinkedIn:
  - [www.linkedin.com/in/ana-moura-14a3951b3](https://www.linkedin.com/in/ana-moura-14a3951b3)
- Github:
  - <https://github.com/ahmourao>



**OBRIKADA!**