

## **SISTEMA DE LOCALIZAÇÃO E AGENDAMENTOS MÉDICOS.**

Filipe Velho da Silva<sup>1</sup>  
MSc. Silvio Cesar Viegas

### **RESUMO**

Este artigo apresenta o desenvolvimento de um sistema web de localização de médicos, visando facilitar o acesso da população a serviços médicos em sua cidade. O sistema foi projetado para permitir que os médicos façam seu cadastro, incluindo informações como especialidade, cidade de atendimento e contatos. Os pacientes, por sua vez, podem utilizar o sistema para buscar médicos sem a necessidade de login, inserindo o nome da cidade ou a especialidade desejada. A implementação da solução foi realizada utilizando PHP no lado do servidor, MySQL para armazenamento de dados, e JavaScript (com AJAX) no lado do cliente, para garantir uma experiência de busca dinâmica e sem recarga de página. O artigo descreve as ferramentas utilizadas no desenvolvimento, como PHP, MySQL, HTML e JavaScript, além de apresentar os diagramas de caso de uso e de classes que modelam a estrutura do sistema. Como resultados, o sistema mostrou-se funcional na busca de médicos, atingindo os objetivos propostos para a primeira entrega, que incluiu o cadastro dos médicos e a interface de busca para pacientes. Este sistema contribui para a democratização do acesso à saúde, permitindo que pessoas localizem profissionais de forma rápida e eficiente.

Palavras chave: Gestão, Médico, Agendamento, Sistema, Localização.

## **1. Introdução**

### **1.1 Tema:**

Sistema de localização e agendamento médico.

### **1.2 Delimitação do Tema:**

Desenvolvimento de um sistema web para que médicos possam cadastrar seus serviços em suas especialidades e os pacientes possam localizar os médicos na sua cidade e agendar uma consulta, para a primeira entrega do projeto aplicado estará contemplado o sistema de cadastro dos médicos e o front de busca dos pacientes, com isto, o fluxo de agendamento não será contemplado nesta entrega.

### 1.3 Objetivo Geral:

Desenvolver um sistema de busca de médicos de fácil acesso, para que as pessoas possam encontrar o melhor profissional na sua região.

### 1.4 Objetivos Específicos:

1. Implementar um módulo de login para os médicos participantes, que para o cadastro, seja incluído dados de localidade e contatos, além da especialidade do serviço disponibilizado.
2. Criar um front-end para usuários de forma pública, sendo possível acessar sem a necessidade de um cadastro prévio.
3. Criar um mecanismo de busca por cidade e/ou especialidade médica.
4. Disponibilizar para o usuário o endereço de atendimento e os dados de contato como e-mail e telefone para que o usuário entre em contato.

### 1.5 Problema:

A dificuldade de encontrar de forma centralizada as informações sobre serviços médicos disponíveis e que seja gratuito, acaba não dando acesso à saúde às pessoas que não possuem um plano de saúde ou um bom convênio, que por muitas vezes são as que mais precisam de um serviço de qualidade.

### 1.6 Justificativa:

Os serviços médicos estão cada vez mais acessíveis para as pessoas, tanto com atendimentos via convênios como via consultas particulares, porém, o acesso à informação e a profissionais capacitados é restrito, ou para quem tenha acesso aos médicos diretamente. Com isto, a criação de um sistema web, disponível de forma gratuita, tem como objetivo facilitar o acesso a serviços médicos de qualidade para qualquer pessoa.

## 2. Fundamentação Teórica

Ao longo deste projeto, decidi adotar algumas ferramentas essenciais para o desenvolvimento da aplicação de consultas médicas. Optei por utilizar as linguagens **PHP** e **JavaScript**, além do banco de dados **MySQL** e a linguagem de marcação **HTML**. A escolha dessas tecnologias se baseia tanto na robustez quanto na sua capacidade de atender as necessidades de uma aplicação web moderna e funcional.

Primeiramente, utilizei **PHP** (Hypertext Preprocessor) por ser uma linguagem de programação que eu já tenho alguma familiaridade e por ela se destacar no desenvolvimento de aplicações web dinâmicas. Como aponta Vasconcelos (2021), "o

PHP é amplamente utilizado devido à sua flexibilidade e capacidade de gerar conteúdo dinâmico em páginas web". O que mais me atraiu no PHP foi justamente essa facilidade de integrar a lógica do servidor com o conteúdo dinâmico das páginas HTML, algo que considero fundamental para o projeto.

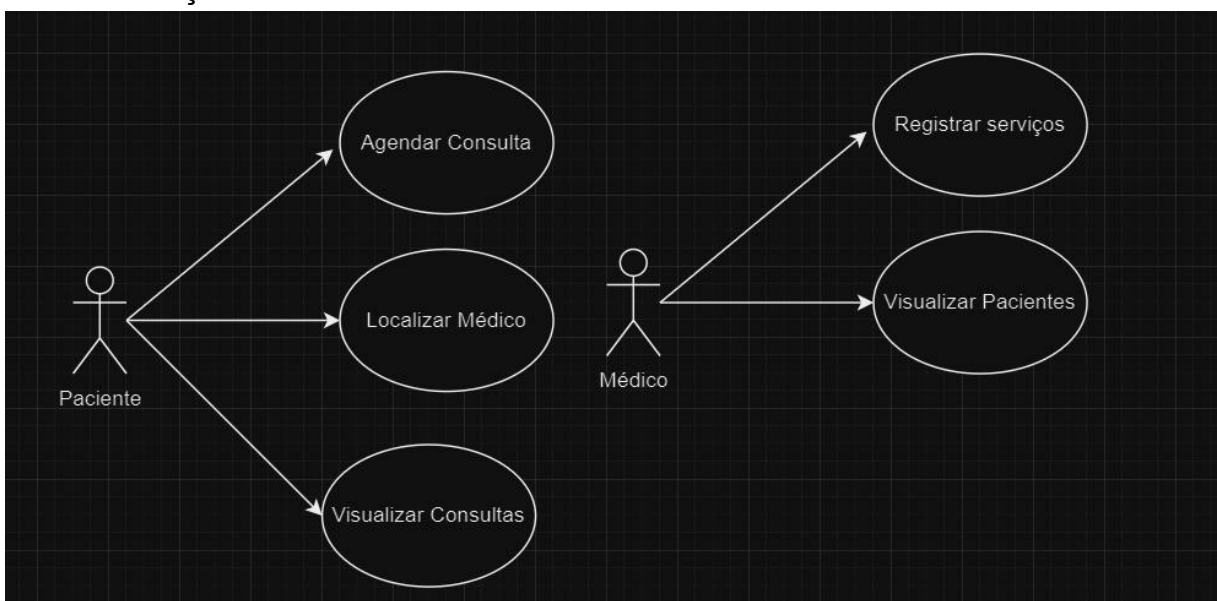
Além disso, optei por **JavaScript**, que é executado no lado do cliente, permitindo que a interação com o sistema seja imediata, sem a necessidade de recarregar a página a cada ação. Para Rocha e Almeida (2019), "o JavaScript torna a experiência do usuário mais fluida, especialmente em interfaces interativas e dinâmicas", o que corrobora com a minha visão de que uma boa interface deve ser o mais responsiva possível, melhorando a usabilidade do sistema para os médicos e pacientes.

No que diz respeito ao armazenamento de dados, optei por utilizar o **MySQL**, que é um sistema de gerenciamento de banco de dados bastante popular e confiável. Borges (2020) afirma que "o MySQL é eficiente na manipulação de grandes volumes de dados e na segurança da informação", algo essencial para o projeto, considerando a natureza sensível dos dados médicos e pessoais envolvidos.

Já o **HTML (Hypertext Markup Language)** é a base de toda a estrutura da aplicação web. Segundo Pereira (2018), "o HTML permite estruturar o conteúdo das páginas web, organizando textos, imagens, vídeos e outros elementos visuais". No desenvolvimento do sistema de consultas médicas, o HTML é a espinha dorsal da interface visual que será acessada pelos usuários, e, por isso, é de extrema importância para mim garantir que as páginas sejam bem estruturadas e funcionais.

## 2.1 Diagrama de Caso de Uso

O diagrama de caso de uso que elaborei (Figura 1) ilustra as principais interações dos usuários no sistema. Os dois atores principais são o médico, que pode gerenciar seu perfil e consultas, e o paciente, que pode agendar consultas e verificar suas informações.



## 2.2 Diagrama de Classes

Por fim, desenvolvi um diagrama de classes (Figura 2) que descreve a estrutura interna do sistema. As classes Médico, Paciente e Consulta são as principais, e o diagrama reflete como os dados circulam e como as funcionalidades se conectam entre elas. Isso me ajudou a visualizar de forma clara a arquitetura do sistema, facilitando o processo de implementação.

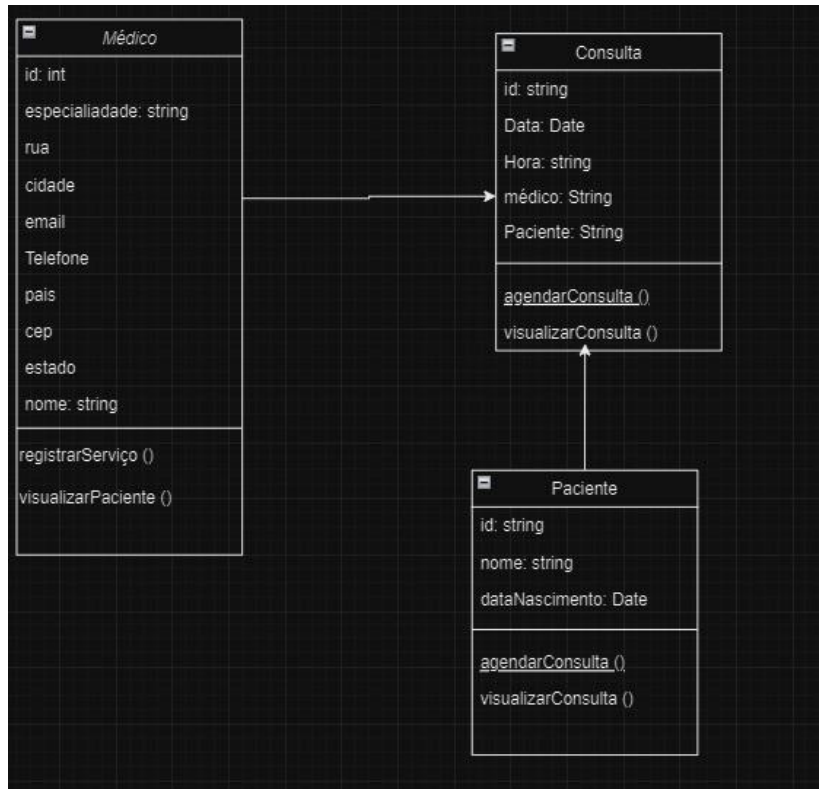


Figura 2 - Diagrama de Classes

## 3. Metodologia

A metodologia aplicada no desenvolvimento deste projeto se baseia em um modelo de ciclo de vida iterativo e incremental, conhecido como **modelo ágil**. Esta abordagem foi escolhida devido à necessidade de adaptação rápida às mudanças nos requisitos do sistema e à entrega contínua de funcionalidades ao longo do projeto. Para isso, utilizamos práticas comuns como **sprints**, com entregas planejadas de acordo com as funcionalidades previstas, começando pelo módulo de cadastro de médicos e a interface de busca de médicos por parte dos pacientes.

Durante o processo de desenvolvimento, as ferramentas principais utilizadas foram PHP, JavaScript, HTML e MySQL. A escolha dessas tecnologias foi motivada pela sua flexibilidade, robustez e pelo amplo suporte da comunidade de desenvolvedores, que permite uma curva de aprendizado relativamente mais suave para quem já tem alguma experiência.

A coleta e estruturação dos requisitos do sistema foi feita com base em análises de necessidades reais de usuários e em entrevistas com médicos que necessitam de uma plataforma para divulgar seus serviços e pacientes que buscam por atendimentos

de forma mais prática e acessível. Através dessas análises, construímos os principais módulos do sistema, que incluíram:

- Módulo de cadastro de médicos;
- Interface pública para busca de médicos, onde o usuário pode buscar por cidade e especialidade;
- Página de listagem dos médicos com os dados de contato visíveis para o usuário final.

Essas funcionalidades foram priorizadas para a primeira fase do projeto, deixando o fluxo de agendamento de consultas para uma entrega posterior.

## 4. Desenvolvimento

O desenvolvimento do sistema de consultas médicas seguiu uma abordagem modular, onde cada parte do sistema foi construída separadamente e integrada ao final do processo. Inicialmente, foram levantados os requisitos com base nas necessidades tanto dos médicos quanto dos pacientes. Esses requisitos foram então traduzidos em módulos que englobam o back-end (lógica do sistema, banco de dados e API) e o front-end (interface com o usuário).

O primeiro passo foi o planejamento da arquitetura do sistema. Optamos por uma arquitetura **cliente-servidor**, onde o servidor gerencia a lógica de negócio e o banco de dados, enquanto o cliente (navegador) apresenta a interface visual para o usuário final. Essa separação é fundamental para garantir a escalabilidade e manutenção do sistema.

### 4.1 Tecnologias Utilizadas

Para garantir a eficiência e funcionalidade do sistema, as seguintes tecnologias foram escolhidas:

- **PHP**: Usamos PHP no back-end para tratar requisições e processar a lógica de negócio. A linguagem permitiu criar rotinas de cadastro de médicos, validação de login, e consultas ao banco de dados. O PHP foi escolhido por sua flexibilidade e capacidade de integração com bancos de dados SQL.
- **JavaScript (com AJAX)**: Utilizamos JavaScript no front-end para dinamizar a interação com o usuário. Com a ajuda de AJAX, foi possível implementar uma busca de médicos sem recarregar a página. Por exemplo, o paciente pode digitar sua cidade e/ou especialidade, e a lista de médicos será atualizada em tempo real, sem interrupção na experiência do usuário.
- **MySQL**: No armazenamento de dados, utilizamos o MySQL, um banco de dados relacional que possui ampla compatibilidade com PHP. A modelagem de dados incluiu tabelas para **médicos**, **especialidades**, **localidades** e um sistema simples de **autenticação** de usuários (médicos).
- **HTML e CSS**: A interface com o usuário foi construída utilizando **HTML5** para estruturar as páginas e **CSS3** para estilizar os elementos visuais, garantindo uma interface clara, limpa e acessível. O foco foi criar uma interface

responsiva que se adapta a dispositivos móveis, uma vez que muitos pacientes poderão acessar o sistema via smartphones.

- **Bootstrap:** O framework **Bootstrap** foi utilizado para agilizar o desenvolvimento da interface. Ele permitiu a criação de layouts responsivos e pré-definidos, como formulários de cadastro, listas e tabelas para exibição de médicos, sem que fosse necessário codificar tudo do zero.
- **Modelagem de Dados:** Criamos um diagrama de classes e um diagrama de caso de uso para organizar e visualizar a estrutura do sistema. Abaixo, segue a descrição da modelagem de dados:
  - **Tabela Médicos:** Responsável por armazenar dados como nome, e-mail, especialidade e cidade.
  - **Tabela Especialidades:** Contém as especialidades médicas disponíveis (como pediatria, dermatologia, etc.).
  - **Tabela Localidades:** Armazena as cidades em que os médicos atuam.
  - **Tabela Login:** Sistema básico para autenticar médicos, utilizando senha e e-mail.

A implementação dos módulos de login e cadastro de médicos seguiu a seguinte sequência:

1. **Página de Cadastro de Médicos:** Esta página contém um formulário onde médicos podem se cadastrar. Além dos campos tradicionais (nome, e-mail, senha), foram incluídos campos específicos para a especialidade médica e local de atuação.
  - O sistema valida todos os campos antes de enviar o formulário para o servidor. Caso algum dado seja inválido, como um e-mail incorreto ou campo obrigatório vazio, o usuário é notificado imediatamente via JavaScript.
2. **Autenticação:** Após o médico se cadastrar, ele pode fazer login no sistema. A autenticação foi construída utilizando PHP e MySQL. Ao efetuar login, os dados do médico são armazenados em uma sessão para garantir que apenas usuários autenticados possam acessar o painel administrativo.
3. **Busca de Médicos (Visão do Paciente):** Na página inicial, os pacientes podem utilizar o sistema de busca. Aqui, a tecnologia AJAX desempenha um papel importante, permitindo que os resultados apareçam em tempo real à medida que o paciente digita sua cidade ou escolhe uma especialidade.
  - O back-end, por sua vez, faz uma consulta ao banco de dados MySQL e retorna uma lista com os médicos que correspondem aos critérios de busca. As informações exibidas incluem o nome do médico, sua especialidade e informações de contato, como e-mail e telefone.

Com essas funcionalidades desenvolvidas, o sistema atinge seu objetivo principal nesta primeira fase: permitir que os médicos se cadastrem e que os pacientes consigam localizar médicos de forma fácil e rápida.

## 5. Resultados

Para demonstrar os resultados práticos obtidos até o momento, abaixo segue uma série de imagens que ilustram as principais funcionalidades do sistema.

1. **Página Inicial - Busca de Médicos:** Esta imagem mostra a interface de busca de médicos, onde os pacientes podem inserir sua cidade ou a especialidade desejada para localizar médicos na região. A busca é realizada em tempo real, sem necessidade de recarregar a página.

2. **Resultado da Busca:** Após realizar a busca, a aplicação retorna uma lista de médicos que correspondem aos filtros aplicados. Cada médico tem suas informações de contato exibidas, permitindo que o paciente entre em contato diretamente para agendar uma consulta.

Nome	Especialidade	Cidade	Telefone
Ana Ferreira	Oftalmologia	Porto Alegre	21977777777

3. **Página de Cadastro de Médicos:** Esta página contém o formulário que os médicos devem preencher para se cadastrar no sistema. Inclui campos para especialidade, cidade, e dados de contato. Além disso, o sistema realiza validação de dados em tempo real, informando o médico caso algum campo tenha sido preenchido incorretamente.

## Cadastro de Médico

### Cadastro de Médico

Nome:

Especialidade:

Cidade:

E-mail:

Telefone:

Senha:

4. **Painel do Médico:** Após o login, os médicos têm acesso a um painel onde podem visualizar e editar suas informações. Aqui, o médico pode atualizar seus dados de contato ou alterar sua especialidade, caso necessário.



The image shows a web interface titled "Painel do Médico" (Doctor's Dashboard) in a blue header. Below the header, a white box contains the text "Bem-vindo, Dr. Filipe Test!" (Welcome, Dr. Filipe Test!). Underneath, there is a section titled "Gerenciar Perfil" (Manage Profile). This section contains four input fields: "Nome:" (Name) with the value "Filipe Test", "E-mail:" (Email) with the value "testeFilipe@gmail.com", "Telefone:" (Phone) with the value "5199885554", and "Especialidade:" (Specialty) with the value "Ortopedista". A blue button labeled "Atualizar Dados" (Update Data) is positioned below these fields. At the bottom left of the white box, there is a small blue button labeled "Sair" (Logout).

Essas imagens ilustram o sistema em pleno funcionamento, demonstrando que os objetivos propostos foram atingidos, especialmente no que diz respeito à busca por médicos e ao cadastro de profissionais. O sistema é totalmente funcional para a busca de médicos por pacientes e para o gerenciamento de perfis médicos, facilitando o acesso a profissionais de saúde em diferentes localidades.

---

## Referências

BORGES, Marcelo. *Banco de dados MySQL para iniciantes*. São Paulo: Editora de Tecnologia, 2020.

PEREIRA, João. *Desenvolvimento Web com HTML e CSS*. 2. ed. Rio de Janeiro: TechBooks, 2018.

ROCHA, Fabiano; ALMEIDA, Mariana. *Interatividade na Web com JavaScript*. Porto Alegre: Ed. Nova Web, 2019.

VASCONCELOS, Bruno. *Programação em PHP para a Web*. 3. ed. São Paulo: CodeBooks, 2021.