

**MINISTÉRIO DA EDUCAÇÃO  
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA  
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA FARROUPILHA  
CAMPUS SANTO ÂNGELO**

**SISTEMA DE AGENDAMENTO PARA CLÍNICAS EM GERAL**

**TRABALHO DE CONCLUSÃO DE CURSO**

**Julio de Souza Carvalho**

**Santo Ângelo, RS, Brasil.  
2024**

# **SISTEMA DE AGENDAMENTO PARA CLÍNICAS EM GERAL**

**por**

**Julio de Souza Carvalho**

Trabalho de Conclusão de Curso apresentado ao Instituto Federal de Educação, Ciência e Tecnologia Farroupilha, como requisito parcial para obtenção do título de Tecnólogo em Sistemas para Internet.

Orientador: John Soldera

**Santo Ângelo, RS, Brasil**

**2024**

**Ministério da Educação**  
**Secretaria de Educação Profissional e Tecnológica**  
**Instituto Federal de Educação Ciência e Tecnologia Farroupilha**

A Comissão Examinadora, abaixo assinada,  
aprova o Trabalho de Conclusão de Curso

**SISTEMA DE AGENDAMENTO PARA CLÍNICAS EM GERAL**

elaborada por  
**Julio de Souza Carvalho**

como requisito parcial para obtenção do título de  
**Tecnólogo em Sistemas para Internet**

**COMISSÃO EXAMINADORA**

---

**John Soldera, Dr.**  
(Orientador)

---

**Marta Breunig Loose, Dra.**

---

**Eno Renato Geiss, Esp.**

Conceito Final: \_\_\_\_\_

Santo Ângelo, 12 de dezembro de 2024.

*Dedico este trabalho aos meus pais.*

## **AGRADECIMENTOS**

Gostaria de expressar minha sincera gratidão aos meus pais pelo amor e apoio incondicionais. Agradeço também ao meu orientador, John Soldera, e à professora Marta Breunig pela orientação valiosa ao longo dessa jornada. Não poderia deixar de agradecer aos demais professores pelo conhecimento compartilhado, que foi essencial para o meu crescimento acadêmico.

*“A persistência é o caminho do êxito.”*

Charles Chaplin

## RESUMO

Este trabalho desenvolve um Sistema de Agendamento de Consultas para clínicas, otimizando a marcação de consultas e facilitando a gestão para administradores e pacientes. Muitas clínicas ainda dependem de processos manuais, que são lentos e propensos a erros. O sistema permite que usuários cadastrem suas informações pessoais, agendem consultas de forma intuitiva e acessível. Além disso, inclui funcionalidades para encaminhar solicitações de consultas e distribuir receitas médicas. Desenvolvido em PHP, utilizando a arquitetura MVC, o sistema visa oferecer diversas vantagens ao usuário, melhorando a eficiência, reduzindo o tempo de espera dos pacientes e aumentando a produtividade das clínicas.

**Palavras-chave:** sistema web; saúde; PHP; MVC.

## **ABSTRACT**

This work develops a Scheduling System for clinics, optimizing appointment booking and facilitating management for administrators and patients. Many clinics still rely on manual processes, which are slow and error-prone. The system allows users to register their personal information and schedule appointments in an intuitive and accessible manner. Additionally, it includes functionalities to forward appointment requests and distribute medical prescriptions. Developed in PHP, using the MVC architecture, the system improves efficiency, reduces patient wait times, and increases clinic productivity.

**Keywords:** web system; health; PHP; MVC.

## LISTA DE FIGURAS

Figura 1 - Tela de Dashboard.....	13
Figura 2 - Tela de Agendamento de Consultas.....	14
Figura 3 - Tela de Agendamento do Administrador.....	14
Figura 4 - Exemplo de Diagrama de Casos de Uso.....	22
Figura 5 - Diagrama Entidade Relacionamento.....	41
Figura 6 - Projeto Lógico.....	42
Figura 7 - Diagrama de Classes.....	43
Figura 8 - Tela de Login.....	43
Figura 9 - Tela de Cadastro.....	44
Figura 10 -Tela de Visualização do Perfil.....	44
Figura 11 -Tela histórico de consultas e download de prescrição (paciente).....	45
Figura 12 -Tela de Médicos (Administrador).....	45
Figura 13 - Tela para solicitar consultas.....	46
Figura 14 - Tela para aceite/recusa de solicitações de consultas e upload/download de prescrição e histórico de consultas (Administrador).....	46
Figura 16 -Tela após login do paciente.....	48
Figura 17 -Tela após login do Administrador.....	48
Figura 18 -Tela de perfil.....	49
Figura 19 -Tela de listagem de pacientes (administrador).....	49
Figura 20 -Tela listagem de médicos (administrador).....	50
Figura 21 -Tela edição de médicos (administrador).....	51
Figura 22 -Tela de upload de anexo (administrador).....	51
Figura 23 -Tela de confirmação de upload de anexo (administrador).....	52
Figura 24 -Tela de minhas consultas (paciente).....	52
Figura 25 -Tela de download de receita (paciente/admin).....	53

## LISTA DE TABELAS

Tabela 1 - Efetuar Login.....	22
Tabela 2 - Realizar Cadastro.....	23
Tabela 3 - Visualizar horários.....	23
Tabela 4 - Visualizar disponibilidade de médicos.....	24
Tabela 5 - Solicitar agendamento.....	24
Tabela 6 - Cancelar agendamento.....	25
Tabela 7 - Realizar download de receita médica.....	25
Tabela 8 - Gerir disponibilidade de médicos.....	26
Tabela 9 - Confirmar agendamento.....	26
Tabela 10 - Recusa agendamento.....	27
Tabela 11 - Enviar lembrete de consulta.....	27
Tabela 12 - Realiza upload de receitas médicas.....	28
Tabela 13 - Efetuar Login.....	29
Tabela 14 - Realizar Cadastro.....	30
Tabela 15 - Visualizar horários.....	31
Tabela 16 - Visualizar disponibilidade de médicos.....	32
Tabela 17 - Solicitar agendamento.....	33
Tabela 18 - Cancelar agendamento.....	34
Tabela 19 - Realizar download de receita médica.....	35
Tabela 20 - Gerir disponibilidade de médicos.....	36
Tabela 21 - Confirmar agendamento.....	37
Tabela 22 - Recusa agendamento.....	38
Tabela 23 - Enviar lembrete de consulta.....	39
Tabela 24 - Realiza upload de receitas médicas.....	40
Tabela 25 - Questionário de Avaliação.....	54

## **LISTA DE ABREVIATURAS**

- API Application Programming Interface
- CSS Cascading Style Sheets
- HTML HyperText Markup Language
- MVC Model-View-Controller
- SGBD Sistema de gerenciamento de banco de dados
- UML Unified Modeling Language
- XML eXtensible Markup Language
- PHP Hypertext Preprocessor

# SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>11</b>
1.1 Objetivos.....	11
1.1.1 Objetivo Geral.....	11
1.1.2 Objetivos Específicos.....	12
1.2 Trabalhos Relacionados.....	12
1.2.1 iClinic.....	13
1.2.2 AgendarConsulta.com.....	13
1.2.3 Central de Consultas.....	14
1.2.4 Diferencial do Trabalho.....	15
1.3 Fundamentação Teórica.....	15
<b>2 DESENVOLVIMENTO.....</b>	<b>20</b>
2.1 Requisitos do Sistema.....	20
2.2 Casos de Uso.....	21
2.1.1 Diagrama de Casos de Uso.....	21
2.1.2 Especificações de Casos de Uso.....	22
2.3 Diagramas de Sequência.....	28
2.4 Banco de Dados.....	40
2.3.1 Diagrama Entidade Relacionamento.....	40
2.3.2 Modelo Lógico.....	41
2.5 Diagrama de Classes.....	42
2.6 Telas do Sistema.....	43
<b>3 RESULTADOS E DISCUSSÕES.....</b>	<b>48</b>
3.1 Relatórios e Telas de Visualização do Sistema.....	48
3.2 Questionário de avaliação e testes do Sistema.....	48
<b>4 CONCLUSÃO.....</b>	<b>49</b>
<b>REFERÊNCIAS.....</b>	<b>51</b>
<b>APÊNDICE A : SCRIPT SQL DE CRIAÇÃO DAS TABELAS.....</b>	<b>53</b>

# 1 INTRODUÇÃO

Nos últimos anos, o avanço da tecnologia tem desempenhado um papel crucial na transformação de diversos setores, e a área da saúde não é exceção. A digitalização tem permitido melhorias significativas no atendimento aos pacientes, na gestão hospitalar e na organização de processos administrativos, como o agendamento de consultas. Um exemplo disso são os sistemas de agendamento online, que oferecem aos pacientes mais comodidade e aos administradores das clínicas uma gestão mais eficiente dos horários e recursos.

A implementação de sistemas digitais tem mostrado, globalmente, benefícios consideráveis para a qualidade dos serviços de saúde. A adoção de soluções tecnológicas no agendamento de consultas, por exemplo, reduz consideravelmente o risco de erros humanos, otimiza o tempo de espera dos pacientes e facilita a comunicação entre os envolvidos no processo. Essas melhorias contribuem para uma experiência mais satisfatória para os pacientes e aumentam a produtividade e a eficiência dos profissionais de saúde.

No entanto, muitas clínicas ainda enfrentam desafios ao utilizar processos manuais ou sistemas antiquados para o agendamento de consultas, o que resulta em sobrecarga administrativa, falhas na comunicação e uma gestão pouco eficiente dos horários dos médicos. Essa realidade dificulta a organização das agendas, aumenta o tempo de espera e pode impactar diretamente a qualidade do atendimento.

A realização deste trabalho se justifica pela necessidade de desenvolver uma ferramenta que não apenas resolva esses problemas, mas que também proporcione uma experiência mais ágil, segura e conveniente para os pacientes, além de aumentar a eficiência operacional das clínicas. Além disso, um sistema de agendamento bem estruturado pode trazer benefícios econômicos, ao reduzir custos administrativos e otimizar a utilização dos recursos disponíveis, aumentando a receita das clínicas.

## 1.1 Objetivos

### 1.1.1 Objetivo Geral

Desenvolver um Sistema de Agendamento de Consultas para clínicas em geral que otimize o processo de marcação de consultas, permitindo aos usuários agendar consultas de forma eficiente e proporcionando aos administradores um controle completo sobre o

agendamento, incluindo a gestão de solicitações, disponibilidade de médicos e distribuição de receitas médicas.

### 1.1.2 Objetivos Específicos

Os objetivos específicos do Sistema de Agendamento de Consultas visam facilitar o agendamento para os pacientes, proporcionar controle e segurança para os administradores e garantir uma experiência eficiente e satisfatória para todos os envolvidos.

- Permitir que os pacientes cadastrem suas informações pessoais para facilitar o agendamento de consultas.
- Implementar um sistema de agendamento simples, intuitivo e acessível para os usuários, permitindo que selecionem datas, horários e médicos desejados.
- Integrar um mecanismo de solicitação de consulta que encaminhe as requisições para o atendente da clínica, proporcionando aos administradores a capacidade de aprovar ou recusar as solicitações.
- Incluir funcionalidades para visualizar a disponibilidade de médicos, informando sobre férias, afastamentos ou outros impedimentos.
- Integrar um sistema de distribuição de receitas médicas após as consultas, garantindo que os pacientes tenham acesso rápido e fácil às suas prescrições.
- Permitir cadastro e manutenção de médicos da clínica.
- Permitir cadastro e manutenção dos dados dos pacientes.

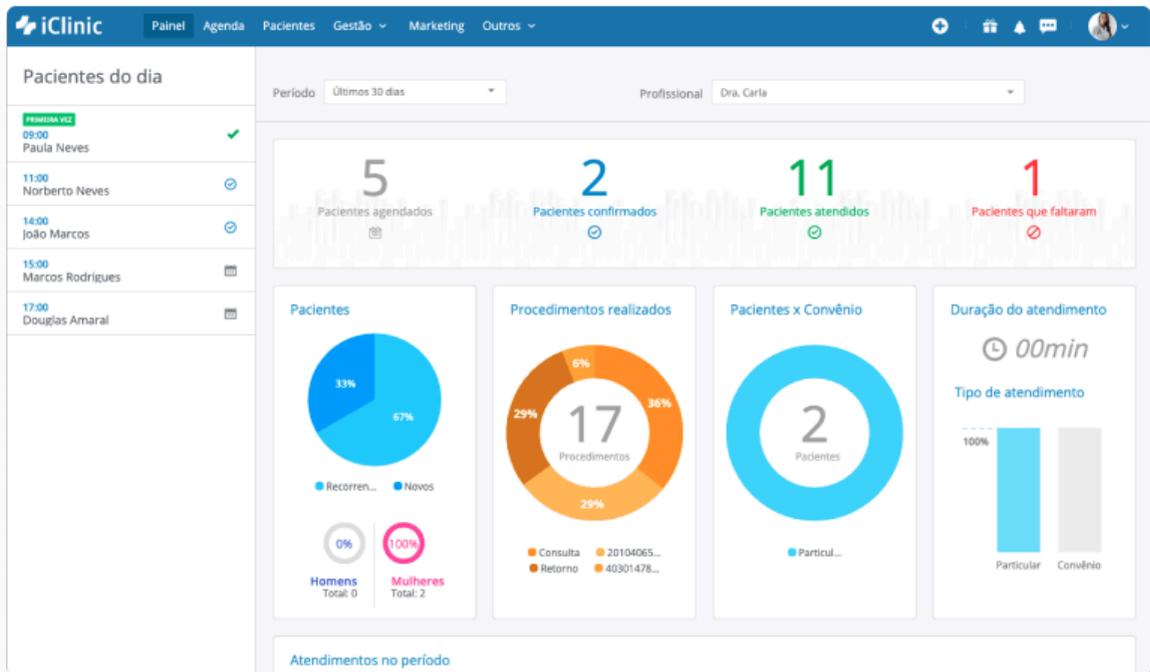
## 1.2 Trabalhos Relacionados

Esta seção dedica-se à revisão de trabalhos que detêm relevância para o escopo do presente estudo. A escolha dessas obras decorre de uma pesquisa voltada para identificar soluções previamente propostas para a problemática em questão. Estes trabalhos em particular foram escolhidos para análise, abrangendo características fundamentais como registros, funcionalidades, relatórios e as tecnologias adotadas.

### 1.2.1 iClinic

O iClinic é um software de pagamento criado pela Afya Digital Health, destinado exclusivamente a médicos especialistas. Este sistema oferece uma ampla gama de funcionalidades, incluindo Agenda Médica, Teleconsultas e Gestão Financeira, proporcionando uma solução abrangente para profissionais da área médica.

**Figura 1 - Tela de Dashboard**

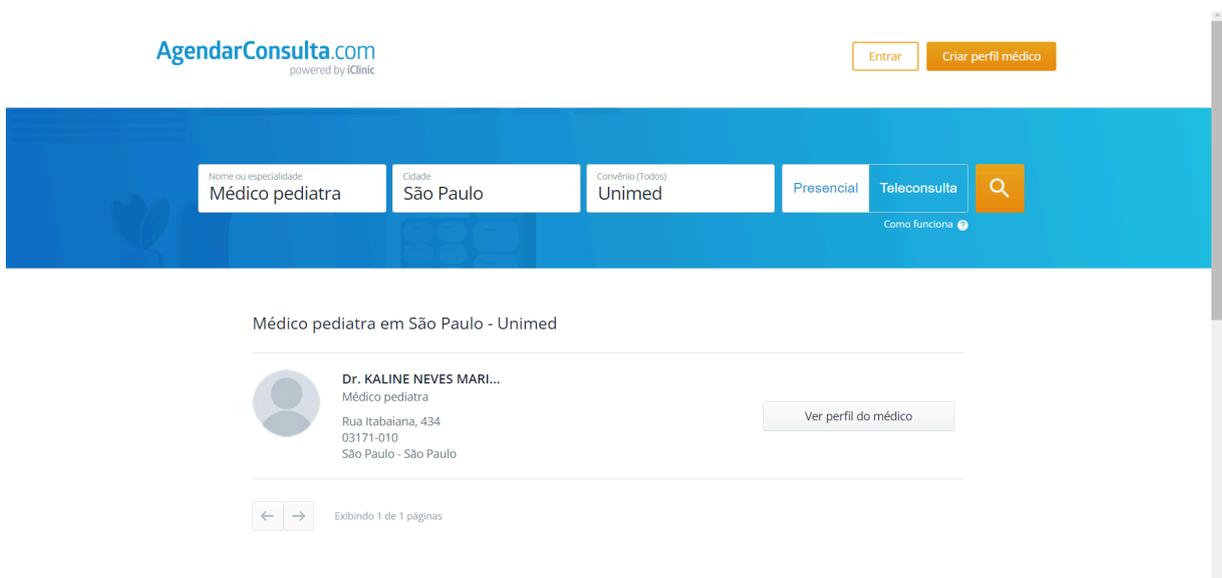


Fonte: iclinic.com.br (2024).

### 1.2.2 AgendarConsulta.com

Outro produto da Afya Digital Health, o AgendarConsulta.com é um site voltado para a conveniência do cliente, permitindo a marcação de consultas com especialistas no horário desejado, eliminando a necessidade de enfrentar filas ou aguardar em chamadas em espera.

**Figura 2 - Tela de Agendamento de Consultas**



Fonte: agendarconsulta.com (2024).

### 1.2.3 Central de Consultas

Desenvolvida pela Clínica Acessível, a Central de Consultas é um software projetado para agendar consultas médicas e odontológicas com especialistas de diversas áreas. Além disso, também é possível agendar exames em específico.

**Figura 3 - Tela de Agendamento do Administrador**



Fonte: centraldeconsultas.med.br (2024).

### 1.2.4 Diferencial do Trabalho

Este sistema visa aumentar a produtividade das clínicas ao digitalizar as solicitações de agendamento de consultas, reduzindo erros administrativos e otimizando o tempo dos pacientes. A interface intuitiva facilita o agendamento para os pacientes, tornando o processo rápido e eficiente.

## 1.3 Fundamentação Teórica

Nesta seção serão descritos os fundamentos teóricos, ou seja, os principais conceitos relacionados à área em que o trabalho está inserido, bem como às ferramentas, técnicas e tecnologias utilizadas para o seu desenvolvimento.

### 1.3.1 Sistemas de Agendamento

Sistemas de agendamento online permitem que clientes marquem compromissos diretamente, acessando o calendário de empresas ou profissionais, e oferecem recursos como lembretes automáticos, integração com sites e compartilhamento de agendas."

(Capterra, 2024)

Os sistemas de agendamento são ferramentas essenciais em diversos contextos, oferecendo uma maneira eficiente e conveniente para os usuários agendarem serviços. Com interfaces intuitivas e processos simplificados, esses sistemas facilitam o agendamento tanto para empresas quanto para clientes, permitindo personalização de horários e notificações para evitar cancelamentos. Além disso, essas plataformas muitas vezes incluem recursos de análise de dados, possibilitando às empresas entenderem melhor os padrões de agendamento e otimizarem o uso de recursos, enquanto garantem a segurança e privacidade das informações dos usuários.

### 1.3.2 Segurança em Aplicações Web

Em aplicações web PHP, a segurança é uma preocupação central. Para prevenir ataques como injeção de SQL, é essencial adotar práticas como o uso de consultas preparadas e a validação cuidadosa dos dados de entrada. A autenticação de usuários também é crucial, e técnicas como criptografia de senhas e verificação em duas etapas podem garantir que apenas indivíduos autorizados tenham acesso aos recursos sensíveis da aplicação.

Ao implementar essas melhores práticas de segurança, os desenvolvedores podem reduzir significativamente o risco de comprometimento da aplicação e proteger os usuários contra ameaças online. Investir em segurança desde o início do desenvolvimento é essencial para garantir a confiabilidade e a integridade das aplicações web PHP em um ambiente cada vez mais propenso a ataques cibernéticos.

A segurança de aplicações web abrange uma variedade de processos, tecnologias e métodos projetados para proteger servidores, aplicações e serviços web, incluindo APIs, contra ameaças baseadas na internet. É fundamental para salvaguardar dados, clientes e organizações contra roubo de informações, interrupções nos negócios e outros impactos prejudiciais do crime cibernético (F5 Networks).

### 1.3.3 Desenvolvimento Web com PHP

No desenvolvimento web usando PHP, a estrutura básica de uma aplicação envolve a interação entre o servidor e o cliente. PHP é frequentemente integrado em páginas HTML para realizar regras de negócio e carregar dados dinamicamente no servidor, enquanto HTML, CSS e JavaScript são utilizados para criar a interface do usuário e oferecer uma experiência interativa.

O uso de bancos de dados relacionais é essencial no desenvolvimento web com PHP. A linguagem oferece suporte para integração com uma variedade de sistemas de gerenciamento de banco de dados, como MySQL e PostgreSQL, permitindo aos desenvolvedores armazenar e recuperar informações de maneira eficiente.

A interação com o cliente é facilitada pela combinação de HTML, CSS e JavaScript. Enquanto HTML fornece a estrutura da página, CSS estiliza elementos para uma interface atraente, e JavaScript proporciona interatividade, possibilitando validação de formulários e atualização dinâmica de conteúdo, melhorando a experiência do usuário. Essas tecnologias formam a base do desenvolvimento web moderno com PHP.

De acordo com Sadik (2023), “sendo uma das linguagens de programação mais amplamente utilizadas para desenvolvimento web, o PHP oferece vários frameworks MVC que simplificam o processo de desenvolvimento e promovem a organização e a manutenção do código”.

#### 1.3.4 Modelo-Visão-Controlador (MVC)

O padrão de arquitetura MVC (Model-View-Controller) é uma abordagem organizacional amplamente utilizada no desenvolvimento de sistemas web. No MVC, a lógica de negócios é encapsulada no "Model", onde os dados são manipulados e processados. A "View" é responsável pela apresentação dos dados ao usuário, exibindo a interface gráfica. O "Controller" atua como intermediário entre o Model e a View, gerenciando as requisições do usuário, processando dados vindos da View e atualizando o Model conforme necessário.

No desenvolvimento de sistemas web com PHP, o padrão MVC é aplicado dividindo-se o código em três camadas distintas: o Model, que contém classes e funções para manipular dados e regras de negócio; a View, que inclui arquivos HTML, CSS e JavaScript para criar a interface do usuário; e o Controller, que trata das requisições do usuário, executa a lógica de negócios adequada no Model e atualiza a View correspondente com os resultados.

De acordo com Adam Freeman (2011), o MVC é uma arquitetura bem-sucedida porque separa as preocupações da aplicação em três partes: modelo, visualização e controle.

Essa separação de responsabilidades proporciona diversos benefícios, como a manutenção simplificada do código, melhor organização e escalabilidade do sistema. Ao adotar o padrão MVC no desenvolvimento web com PHP, os desenvolvedores podem escrever código mais limpo, modular e fácil de dar manutenção, resultando em sistemas mais robustos e eficientes.

### 1.3.5 Bancos de Dados e MySQL

Segundo Milani (2007), o MySQL é reconhecido pela sua alta velocidade no processamento de consultas SQL, resultado da otimização de seus códigos e funções. No entanto, essa otimização pode levar à falta de implementação de alguns padrões mais recentes do SQL, priorizando a velocidade do banco de dados. Apesar disso, essa limitação geralmente não impacta negativamente nas aplicações.

Os bancos de dados relacionais são uma parte essencial do desenvolvimento de aplicações web PHP, fornecendo um meio eficiente de armazenar e organizar dados. O MySQL é uma das opções mais populares para sistemas desenvolvidos em PHP, oferecendo uma combinação de desempenho, confiabilidade e recursos avançados. Como um sistema de gerenciamento de banco de dados relacional (SGBDR), o MySQL permite que os desenvolvedores criem e manipulem tabelas, execute consultas complexas e garanta a integridade dos dados.

A integração entre PHP e MySQL é bastante direta, graças às bibliotecas e extensões específicas disponíveis para PHP. Os desenvolvedores podem usar funções e comandos específicos do MySQL dentro de scripts PHP para interagir com o banco de dados, como inserir dados, recuperar informações ou atualizar registros. Essa combinação oferece aos desenvolvedores flexibilidade e controle sobre o armazenamento e a recuperação de dados em suas aplicações web, garantindo um funcionamento eficiente e confiável do sistema.

Além disso, o MySQL oferece recursos avançados, como suporte transacional, indexação eficiente e replicação de dados, tornando-o uma escolha robusta para uma ampla gama de aplicações web. Sua escalabilidade também é notável, permitindo que sistemas cresçam conforme a demanda sem comprometer o desempenho. Em resumo, a combinação de PHP e MySQL proporciona uma base sólida para o desenvolvimento de aplicações web dinâmicas e escaláveis, atendendo às demandas dos usuários e garantindo a integridade dos dados.

### 1.3.6 Casos de Uso

Os casos de uso desempenham um papel crucial na engenharia de requisitos de software, fornecendo uma representação clara e concisa das interações entre os usuários e o sistema. Eles descrevem as funcionalidades do sistema a partir da perspectiva do usuário, identificando suas metas e ações específicas para alcançá-las. Os casos de uso são frequentemente representados por meio de diagramas, facilitando a comunicação entre desenvolvedores, stakeholders e demais partes interessadas.

Na prática, os casos de uso são úteis para capturar os requisitos funcionais do sistema, ajudando a garantir que o software desenvolvido atenda às necessidades dos usuários finais. Eles também auxiliam no processo de validação e verificação do sistema, permitindo que os desenvolvedores testem e confirmem se todas as funcionalidades foram implementadas conforme o esperado. Ao identificar os principais casos de uso, os desenvolvedores podem priorizar e planejar as atividades de desenvolvimento de forma mais eficaz.

De acordo com Ivar Jacobson (1992), pode-se dizer que um caso de uso é um "documento narrativo que descreve a sequência de eventos de um ator que usa um sistema para completar um processo".

Além disso, os casos de uso fornecem uma base sólida para a elaboração de testes de aceitação e cenários de teste, ajudando a garantir a qualidade e a confiabilidade do software. Eles também servem como uma ferramenta eficaz para documentar e comunicar os requisitos do sistema ao longo do ciclo de vida do projeto, facilitando a colaboração entre equipes de desenvolvimento, gerenciamento de projetos e stakeholders. Em resumo, os casos de uso desempenham um papel fundamental no desenvolvimento de software, contribuindo para o sucesso e a eficácia dos projetos de engenharia de software.

### 1.3.7 VsCode

De acordo com a documentação oficial, o Visual Studio Code (VSCode) é uma ferramenta de desenvolvimento amplamente elogiada por sua simplicidade e eficiência. Desenvolvido pela Microsoft, o VSCode suporta diversas linguagens de programação e oferece uma vasta gama de recursos, tornando-se uma escolha popular entre os desenvolvedores (Code.visualstudio.com, 2024).

O VSCode se destaca por sua interface intuitiva e altamente personalizável, permitindo que os usuários adaptem o ambiente de trabalho conforme suas preferências. Com

temas, configurações de editor e extensões, os desenvolvedores podem ajustar a aparência e o comportamento da IDE para aumentar sua produtividade.

Com recursos poderosos como realce de sintaxe, sugestões de código e depuração integrada, o VSCode oferece uma experiência de desenvolvimento robusta. Além disso, sua extensa comunidade de desenvolvedores contribui regularmente com novas extensões e melhorias, enriquecendo ainda mais a experiência do usuário.

### 1.3.8 Laravel

Conforme a documentação disponibilizada no site oficial, o Laravel é um framework PHP open-source, lançado por Taylor Otwell em 2011, que facilita o desenvolvimento de aplicações web através de uma arquitetura MVC. Ele oferece uma sintaxe expressiva e fácil de entender, integrando funcionalidades essenciais como roteamento, autenticação, gerenciamento de sessões e caching, que agilizam o processo de desenvolvimento (laravel.com, 2024).

Além de suas funcionalidades principais, Laravel inclui o sistema de templating Blade, que facilita a criação de layouts dinâmicos e reutilizáveis. O framework também se integra facilmente com tecnologias de frontend para criar interfaces de usuário interativas e responsivas.

## 2 DESENVOLVIMENTO

Neste capítulo serão descritas as principais etapas do desenvolvimento do trabalho. O foco principal está na documentação do sistema, contendo a descrição dos requisitos e modelagem do sistema, modelagem do banco de dados, classes e telas do sistema.

### 2.1 Requisitos do Sistema

#### 2.1.1 Requisitos Funcionais

- Cadastro de Pacientes

O sistema deve permitir que os pacientes cadastrem suas informações pessoais e médicas necessárias para o agendamento de consultas.

- Agendamento de Consultas

Deve haver uma interface intuitiva que permita aos pacientes selecionar datas, horários e especialidades médicas desejadas para agendar consultas.

- Solicitação de Consultas

Deve existir um mecanismo que encaminhe as solicitações de consulta para a equipe administrativa da clínica para revisão e aprovação.

- Gestão de Disponibilidade de Médicos

O sistema deve permitir que os administradores visualizem a disponibilidade dos médicos, incluindo informações sobre férias, afastamentos e outros impedimentos.

- Distribuição de Receitas Médicas

Deve ser possível distribuir rapidamente receitas médicas aos pacientes após as consultas, garantindo acesso rápido e fácil às prescrições.

- Cadastro de Médicos

O sistema deve permitir que os administradores cadastrem as informações referente aos médicos, como especialidade e situação atual.

- Usabilidade

O sistema deve possuir uma interface intuitiva para facilitar o uso por parte dos usuários.

- Desempenho

O sistema deve ser capaz de lidar com diversos acessos simultâneos sem comprometer a velocidade de resposta.

- Segurança

Medidas de segurança, como controle de acesso, devem ser implementadas

para proteger os dados dos pacientes e informações confidenciais do sistema.

- **Confiabilidade**

O sistema deve estar disponível durante o horário comercial da clínica, com um tempo mínimo de inatividade planejada para manutenção.

- **Escalabilidade**

O sistema deve ser escalável para lidar com o crescimento da clínica, suportando um aumento no número de pacientes e consultas.

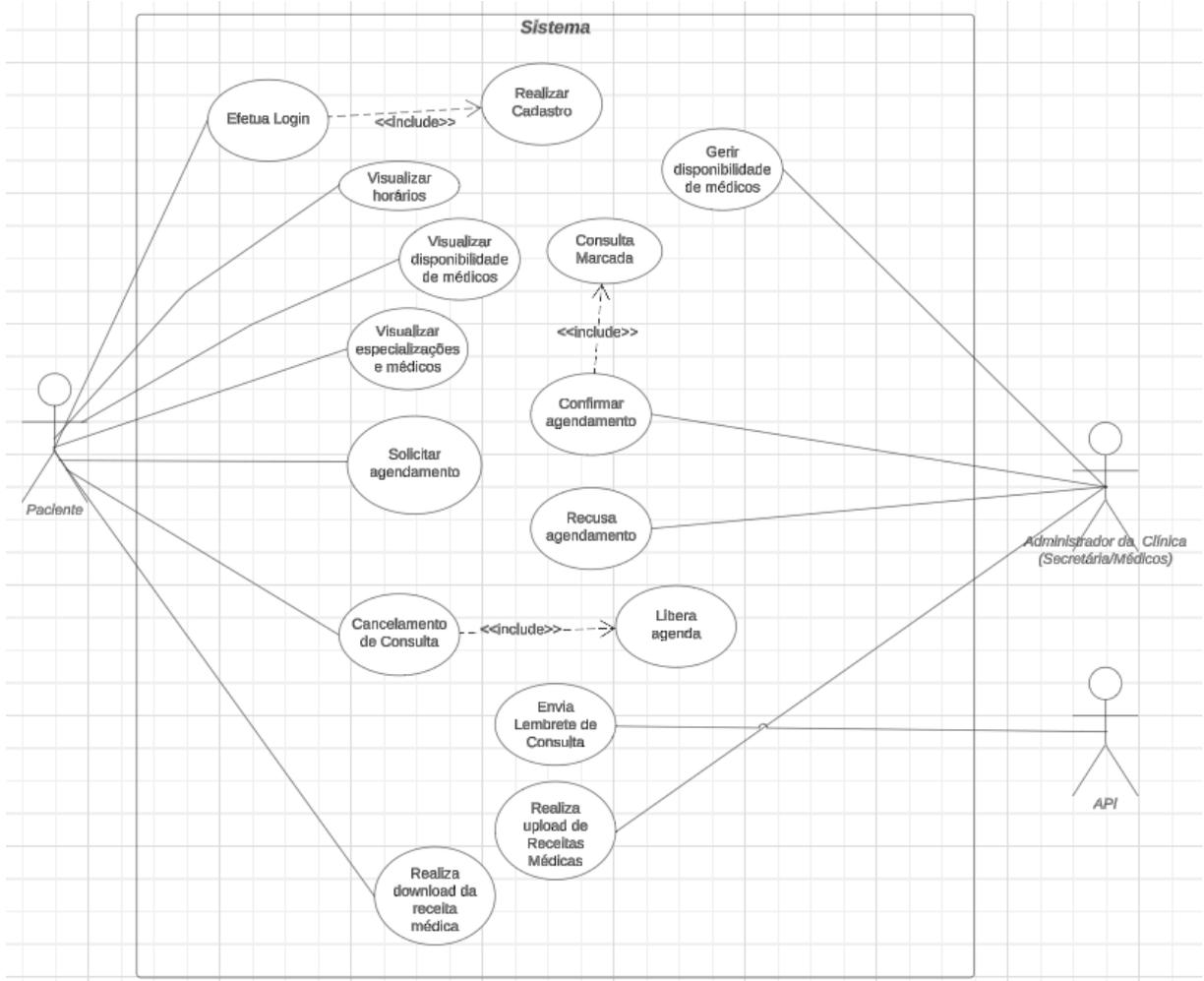
- **Compatibilidade**

O sistema deve ser compatível com diferentes dispositivos e navegadores web para garantir acessibilidade para todos os usuários.

## 2.2 Casos de Uso

### 2.1.1 Diagrama de Casos de Uso

**Figura 4 - Exemplo de Diagrama de Casos de Uso**



Fonte: Criado pelo Autor (2024).

### 2.1.2 Especificações de Casos de Uso

Nesta seção, serão apresentadas as tabelas contendo as descrições dos Casos de Uso.

**Tabela 1 - Efetuar Login**

<b>Identificação UC:</b> Efetuar Login	<b>Nome UC:</b> Efetuar Login
<b>Ator:</b> Paciente	
<b>Resumo:</b> Paciente realiza login no sistema.	
<b>Pré-condições:</b> O paciente não está autenticado no sistema.	

Fluxo principal	
Ações do Ator	Ações do Sistema
1. O paciente acessa a página de login. 2. O paciente insere seu nome de usuário e senha.	3. O sistema verifica as credenciais do paciente. 4. O paciente é autenticado no sistema.
Fluxo alternativo	
2.1 Caso o paciente não tenha uma conta, ele pode escolher a opção de realizar o cadastro.	

Fonte: elaborado pelo autor (2024).

**Tabela 2 - Realizar Cadastro**

Identificação UC: Realizar Cadastro	Nome UC: Realizar Cadastro
<b>Ator:</b> Paciente	
<b>Resumo:</b> Paciente realiza o cadastro do usuário no sistema.	
<b>Pré-condições:</b> O paciente não possui uma conta no sistema.	
Fluxo principal	
Ações do Ator	Ações do Sistema
1. O paciente acessa a página de cadastro 2. O paciente preenche um formulário com suas informações pessoais e médicas.	3. O sistema valida os dados inseridos pelo paciente. 4. O sistema registra o novo usuário no banco de dados.
Fluxo alternativo	

Fonte: elaborado pelo autor (2024).

**Tabela 3 - Visualizar horários**

Identificação UC: Visualizar horários	Nome UC: Visualizar horários
<b>Ator:</b> Paciente	
<b>Resumo:</b> Paciente visualiza os horários e datas disponíveis no sistema.	
<b>Pré-condições:</b> O paciente estar autenticado no sistema	
Fluxo principal	
Ações do Ator	Ações do Sistema

1. O paciente acessa a página de visualização de horários.	2. O sistema exibe os horários disponíveis para agendamento de consultas.
--	---

Fonte: elaborado pelo autor (2024).

**Tabela 4 - Visualizar disponibilidade de médicos**

<b>Identificação UC:</b> Visualizar disponibilidade de médicos	<b>Nome UC:</b> Visualizar disponibilidade de médicos
<b>Ator:</b> Paciente	
<b>Resumo:</b> Paciente visualiza os médicos disponíveis no sistema.	
<b>Pré-condições:</b> O paciente está autenticado no sistema.	
<b>Fluxo principal</b>	
<b>Ações do Ator</b>	<b>Ações do Sistema</b>
1. O paciente acessa a página de visualização de disponibilidade de médicos.	2. O sistema exibe a lista de médicos disponíveis para consulta, juntamente com suas especialidades e horários disponíveis.

Fonte: elaborado pelo autor (2024).

**Tabela 5 - Solicitar agendamento**

<b>Identificação UC:</b> Solicitar agendamento	<b>Nome UC:</b> Solicitar agendamento
<b>Ator:</b> Paciente	
<b>Resumo:</b> Paciente solicitar um agendamento de consulta.	
<b>Pré-condições:</b> O paciente está autenticado no sistema.	
<b>Fluxo principal</b>	
<b>Ações do Ator</b>	<b>Ações do Sistema</b>
1. O paciente seleciona a opção de agendar consulta. 2. O paciente escolhe a data, horário e médico desejados para a consulta.	3. O sistema verifica a disponibilidade do médico e do horário selecionado. 4. O sistema encaminha a solicitação de consulta para a atendente da clínica.

Fonte: elaborado pelo autor (2024).

**Tabela 6 - Cancelar agendamento**

<b>Identificação UC:</b> Cancelar agendamento	<b>Nome UC:</b> Cancelar agendamento
<b>Ator:</b> Paciente	
<b>Resumo:</b> Paciente solicita o cancelamento de uma consulta.	

<b>Pré-condições:</b> O paciente possui uma consulta agendada.	
<b>Fluxo principal</b>	
<b>Ações do Ator</b>	<b>Ações do Sistema</b>
1. O paciente acessa a página de cancelamento de consulta. 2. O paciente seleciona a consulta que deseja cancelar.	3. O sistema confirma o cancelamento da consulta.

Fonte: elaborado pelo autor (2024).

**Tabela 7 - Realizar download de receita médica**

<b>Identificação UC:</b> Realizar download de receita médica	<b>Nome UC:</b> Realizar download de receita médica
<b>Ator:</b> Paciente	
<b>Resumo:</b> Paciente realiza o download da receita médica de uma consulta concluída.	
<b>Pré-condições:</b> O paciente possui uma consulta realizada	
<b>Fluxo principal</b>	
<b>Ações do Ator</b>	<b>Ações do Sistema</b>
1. O paciente acessa a página de receitas médicas. 2. O sistema exibe as receitas médicas disponíveis para download.	3. O paciente seleciona a receita que deseja baixar. 4. O sistema permite o download da receita médica.

Fonte: elaborado pelo autor (2024).

**Tabela 8 - Gerir disponibilidade de médicos**

<b>Identificação UC:</b> Gerir disponibilidade de médicos	<b>Nome UC:</b> Gerir disponibilidade de médicos
<b>Ator:</b> Administrador da clínica	
<b>Resumo:</b> Administrador cadastra e edita informações dos médicos.	
<b>Pré-condições:</b> O administrador da clínica está autenticado no sistema.	
<b>Fluxo principal</b>	
<b>Ações do Ator</b>	<b>Ações do Sistema</b>
1. O administrador acessa a página de gestão de disponibilidade de médicos.	2. O administrador visualiza e atualiza a disponibilidade de cada médico, informando férias, afastamentos ou outros impedimentos.

--	--

Fonte: elaborado pelo autor (2024).

**Tabela 9 - Confirmar agendamento**

<b>Identificação UC:</b> Confirmar agendamento	<b>Nome UC:</b> Confirmar agendamento
<b>Ator:</b> Administrador da clínica	
<b>Resumo:</b> Administrador visualiza e confirma uma solicitação de agendamento.	
<b>Pré-condições:</b> A clínica recebe uma solicitação de agendamento.	
<b>Fluxo principal</b>	
<b>Ações do Ator</b>	<b>Ações do Sistema</b>
1. O administrador acessa a lista de solicitações de agendamento. 3. O administrador confirma o agendamento da consulta.	2. O sistema apresenta as consultas solicitadas.

Fonte: elaborado pelo autor (2024).

**Tabela 10 - Recusa agendamento**

<b>Identificação UC:</b> Recusa agendamento	<b>Nome UC:</b> Recusa agendamento
<b>Ator:</b> Administrador da clínica	
<b>Resumo:</b> O administrador visualiza e recusa uma solicitação de agendamento.	
<b>Pré-condições:</b> A clínica recebe uma solicitação de agendamento.	
<b>Fluxo principal</b>	
<b>Ações do Ator</b>	<b>Ações do Sistema</b>
1. O administrador acessa a lista de solicitações de agendamento. 3. O administrador recusa o agendamento da consulta.	2. O sistema apresenta as consultas solicitadas.

Fonte: elaborado pelo autor (2024).

**Tabela 11 - Enviar lembrete de consulta**

<b>Identificação UC:</b> Enviar lembrete de consulta	<b>Nome UC:</b> Enviar lembrete de consulta
<b>Ator:</b> API	

<b>Resumo:</b>	
<b>Pré-condições:</b> O paciente possui uma consulta agendada.	
<b>Fluxo principal</b>	
<b>Ações do Ator</b>	<b>Ações do Sistema</b>
1. A API acessa a lista de consultas agendadas. 2. A API verifica consultas com 24h de antecedência. 4. A API envia um lembrete por WhatsApp para o paciente, contendo os detalhes da consulta.	3. O sistema obtém o número de telefone do paciente associado à consulta.

Fonte: elaborado pelo autor (2024).

**Tabela 12 - Realiza upload de receitas médicas**

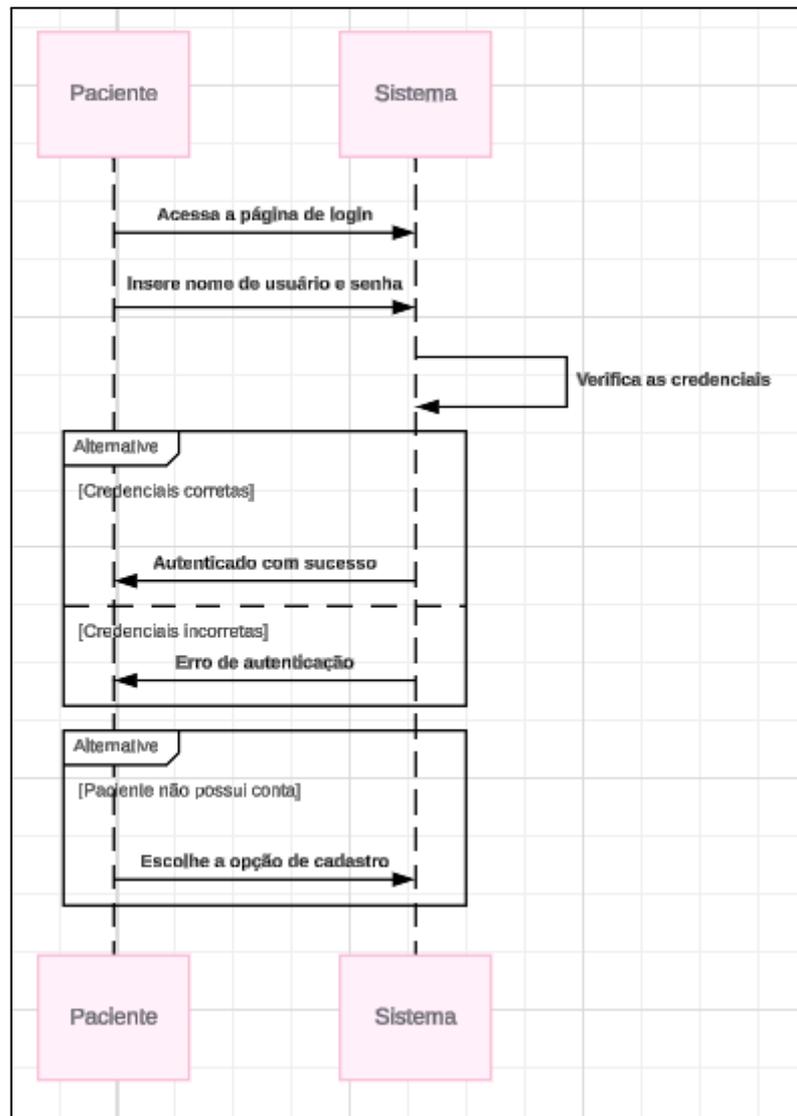
<b>Identificação UC:</b> Realiza upload de receitas médicas	<b>Nome UC:</b> Realiza upload de receitas médicas
<b>Ator:</b> Administrador da Clínica	
<b>Resumo:</b> O administrador realiza upload de uma receita médica na consulta concluída.	
<b>Pré-condições:</b> O médico da clínica descreve uma receita médica e repassa ao administrador.	
<b>Fluxo principal</b>	
<b>Ações do Ator</b>	<b>Ações do Sistema</b>
<ol style="list-style-type: none"> <li>1. O administrador acessa a página de upload de receitas médicas.</li> <li>2. O administrador seleciona a receita gerada.</li> </ol>	<ol style="list-style-type: none"> <li>3. O sistema permite o upload da receita médica.</li> </ol>

Fonte: elaborado pelo autor (2024).

## 2.3 Diagramas de Sequência

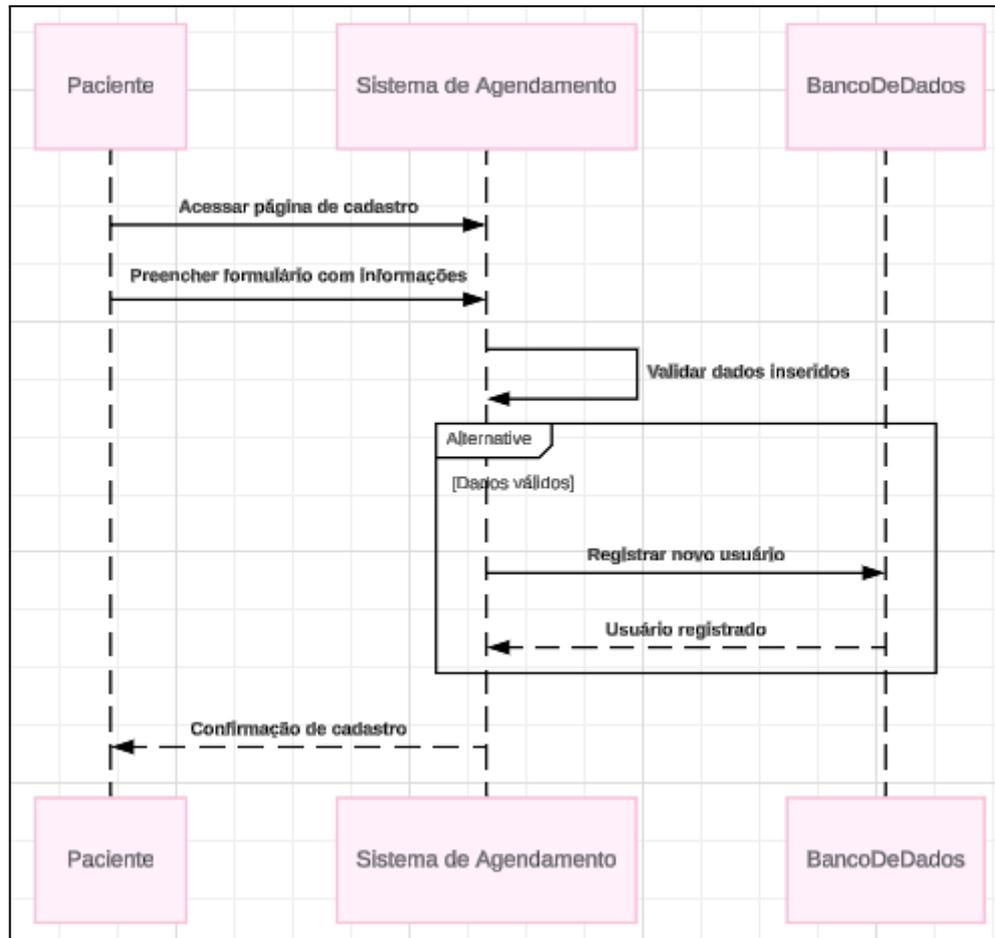
Nesta seção, serão apresentados os Diagramas de Sequência de cada funcionalidade do sistema. Basicamente, esses diagramas seguem a definição do fluxo da Especificação do Caso de Uso.

**Tabela 13 - Efetuar Login**



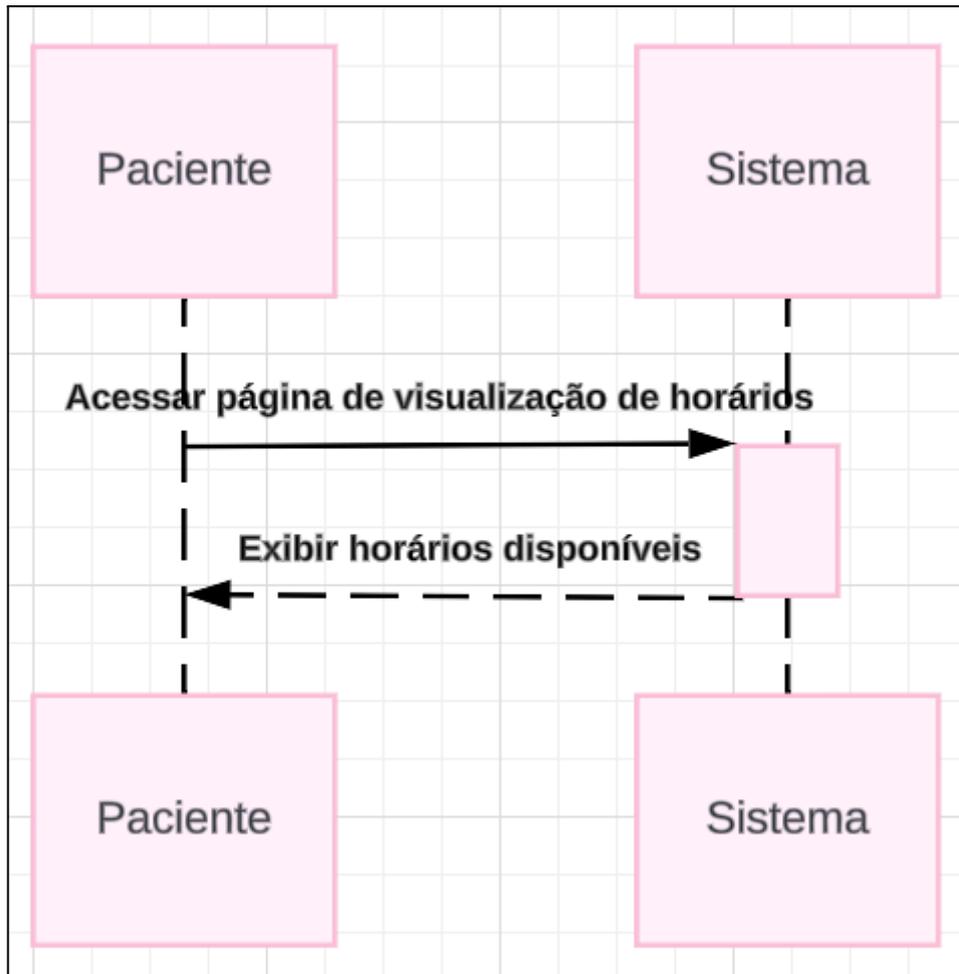
Fonte: elaborado pelo autor (2024).

Tabela 14 - Realizar Cadastro



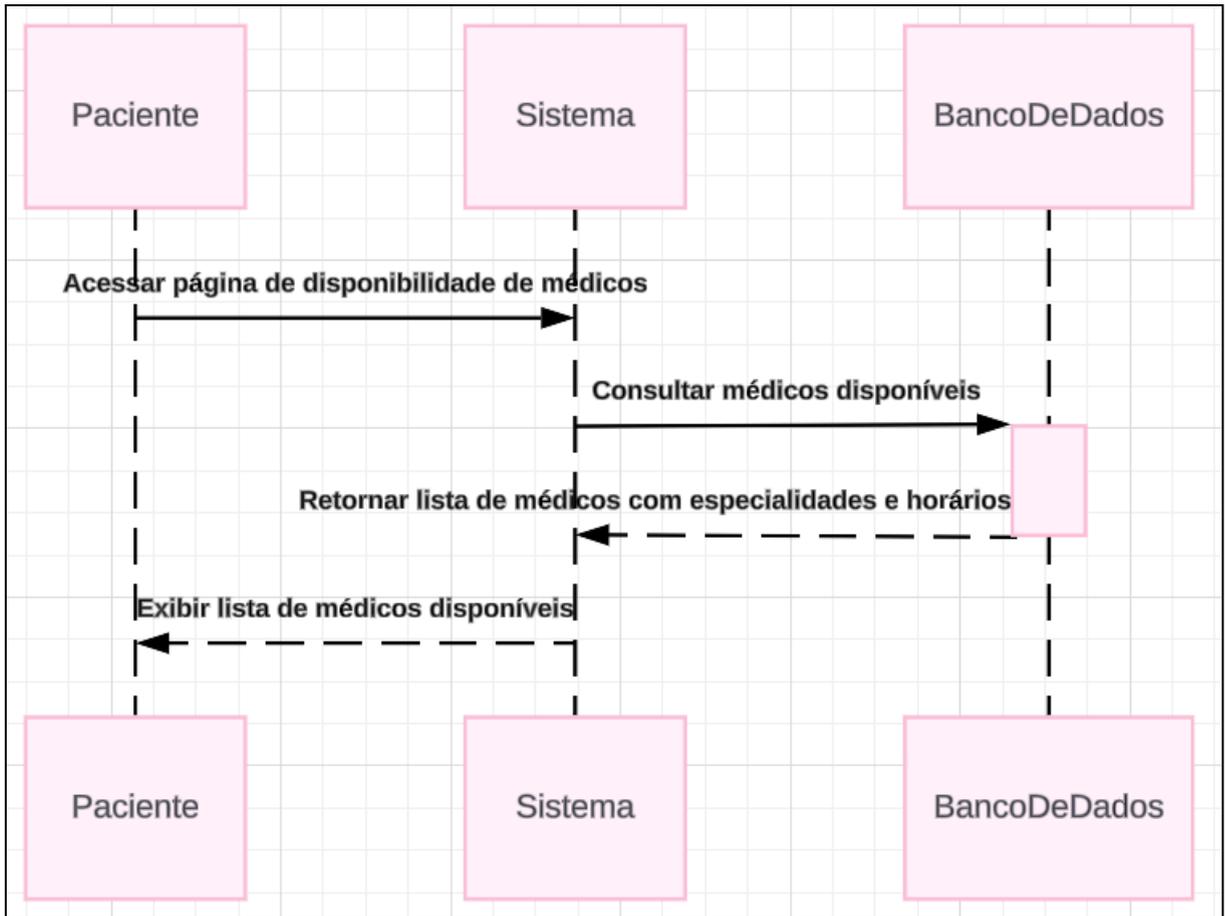
Fonte: elaborado pelo autor (2024).

Tabela 15 - Visualizar horários



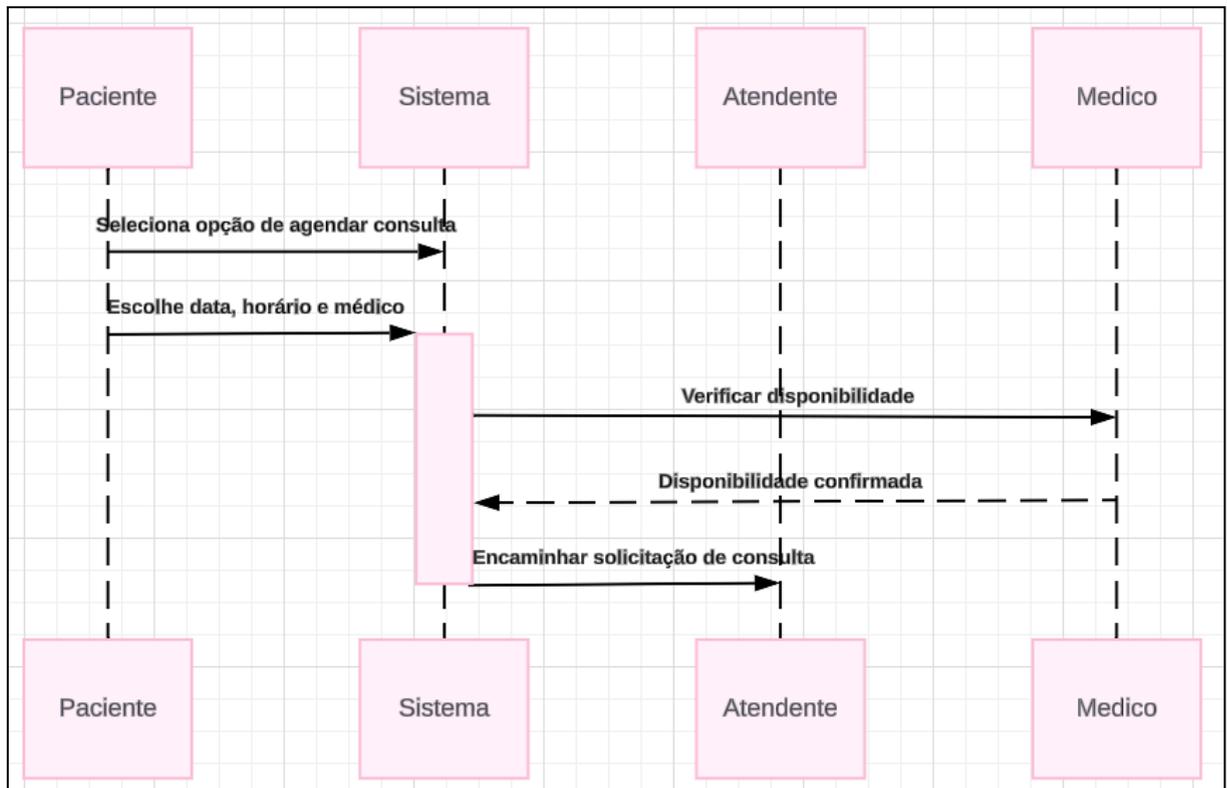
Fonte: elaborado pelo autor (2024).

Tabela 16 - Visualizar disponibilidade de médicos



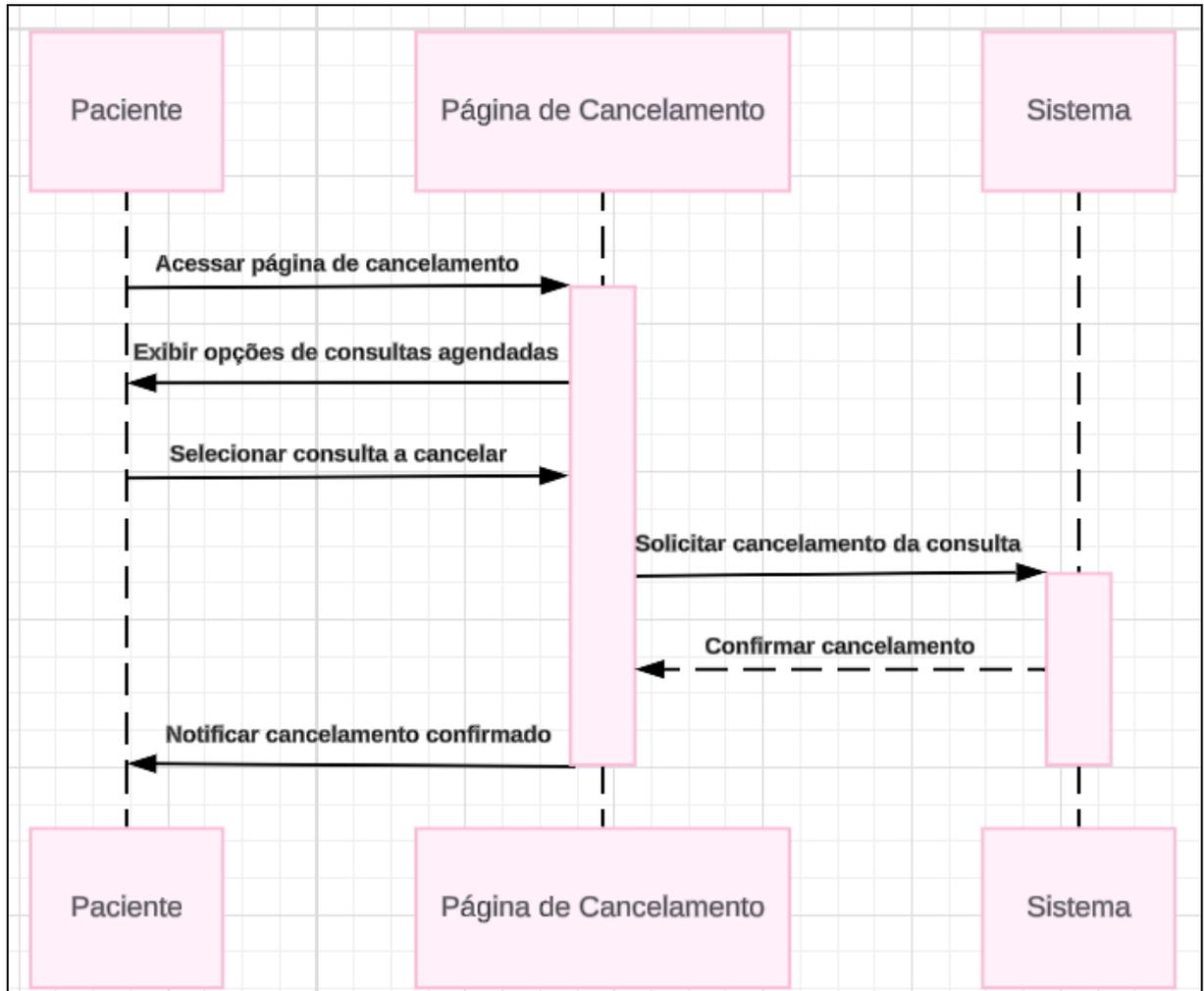
Fonte: elaborado pelo autor (2024).

Tabela 17 - Solicitar agendamento



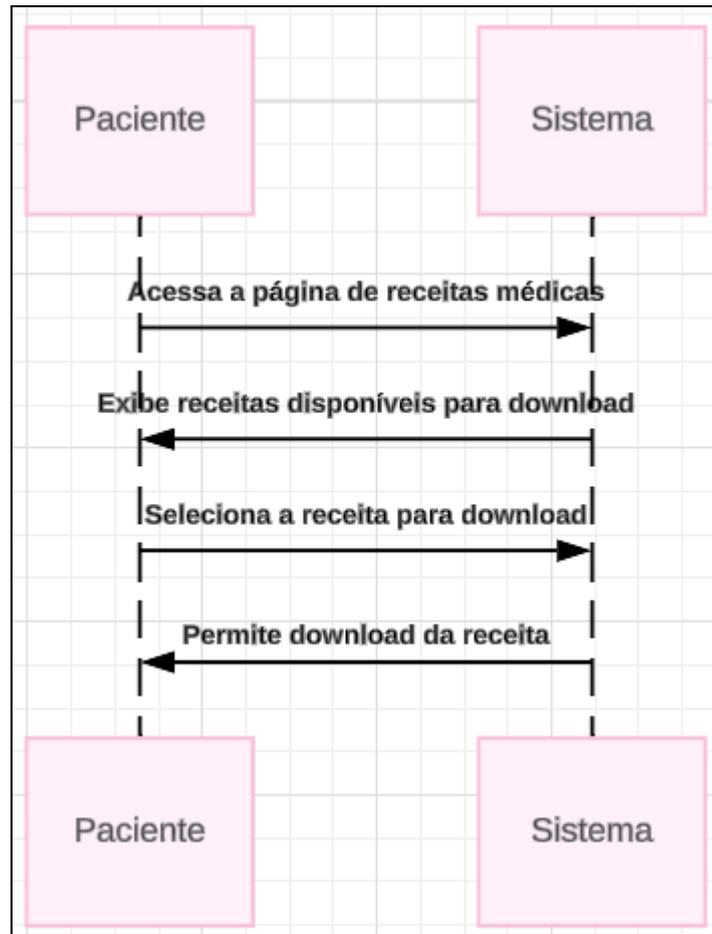
Fonte: elaborado pelo autor (2024).

Tabela 18 - Cancelar agendamento



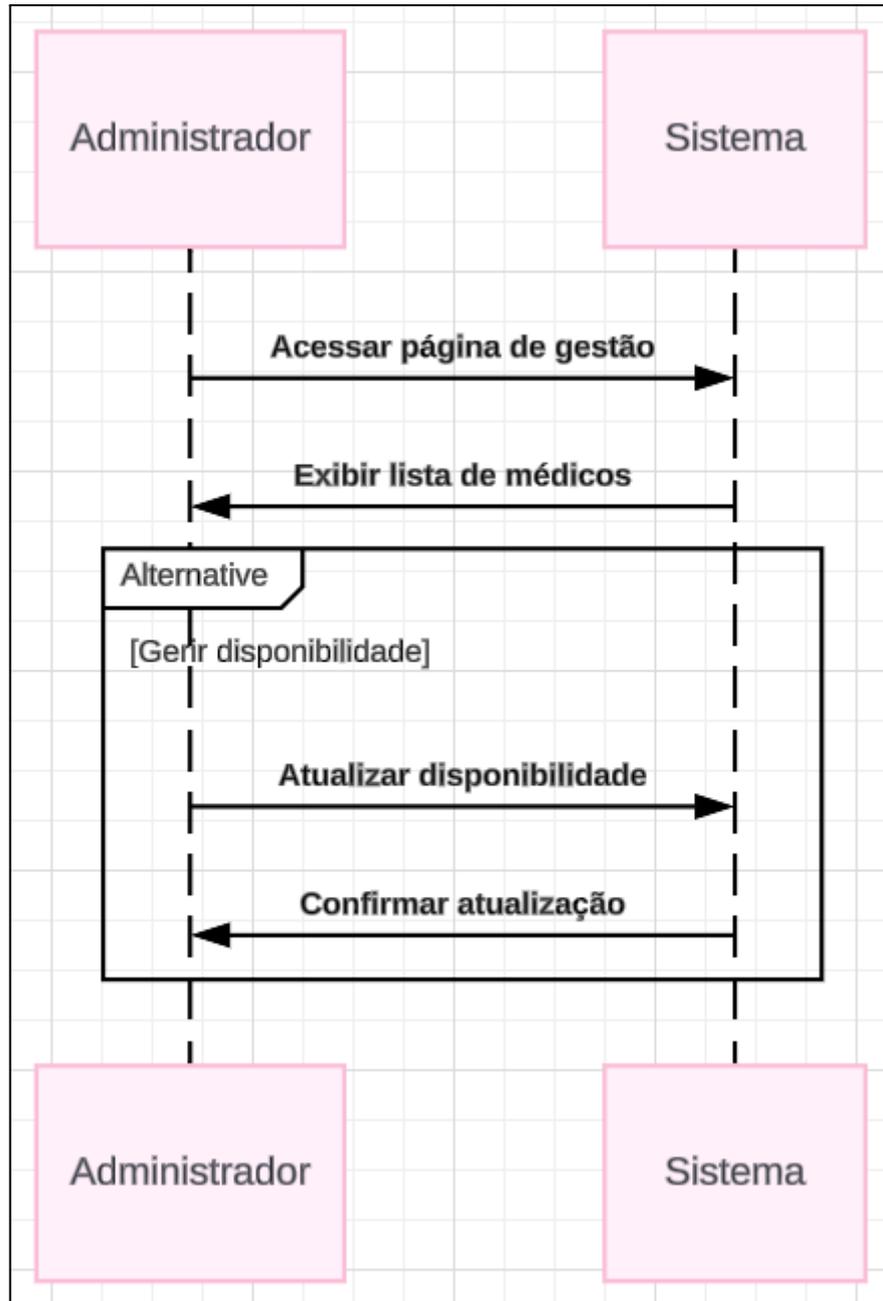
Fonte: elaborado pelo autor (2024).

Tabela 19 - Realizar download de receita médica



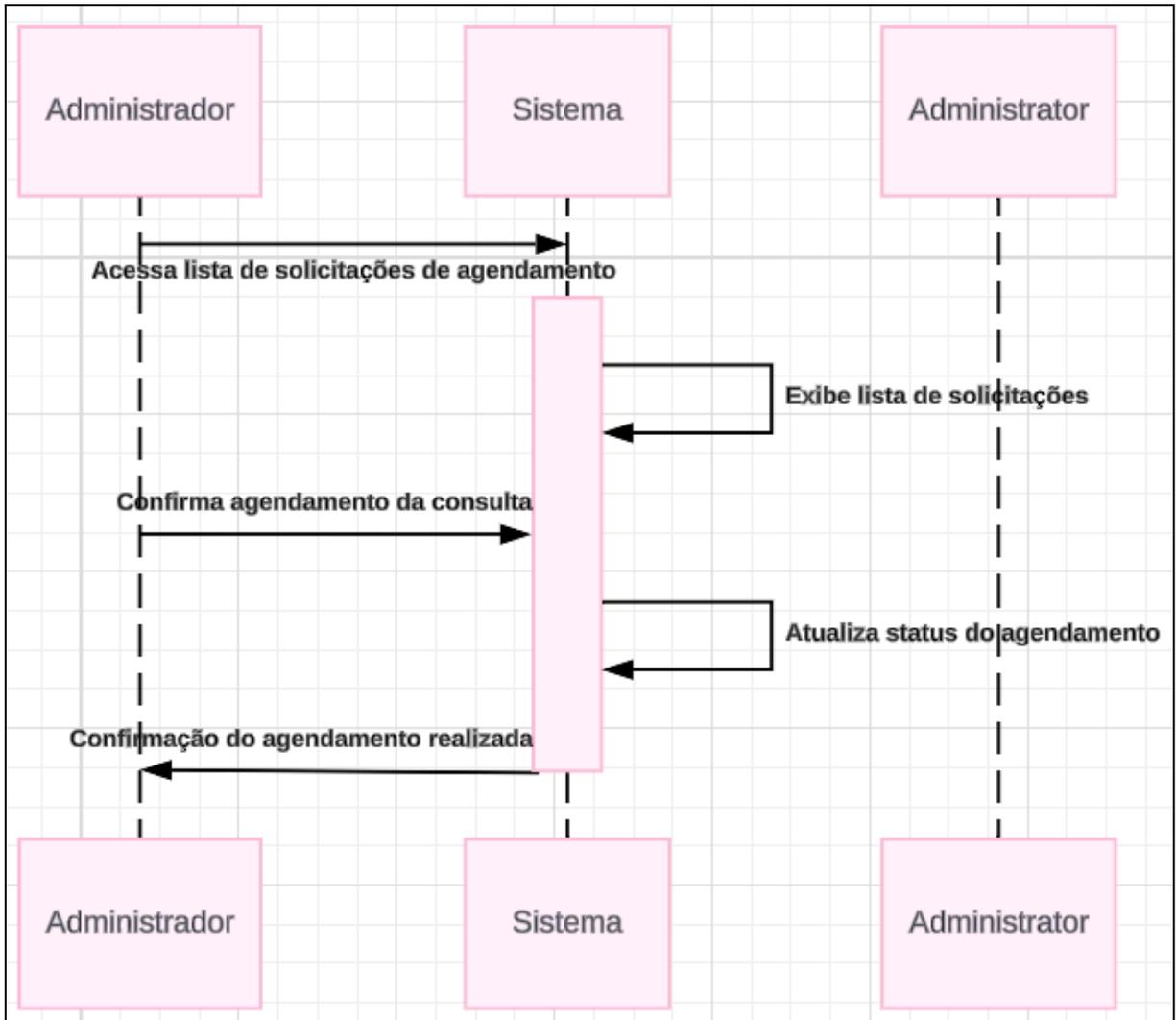
Fonte: elaborado pelo autor (2024).

Tabela 20 - Gerir disponibilidade de médicos



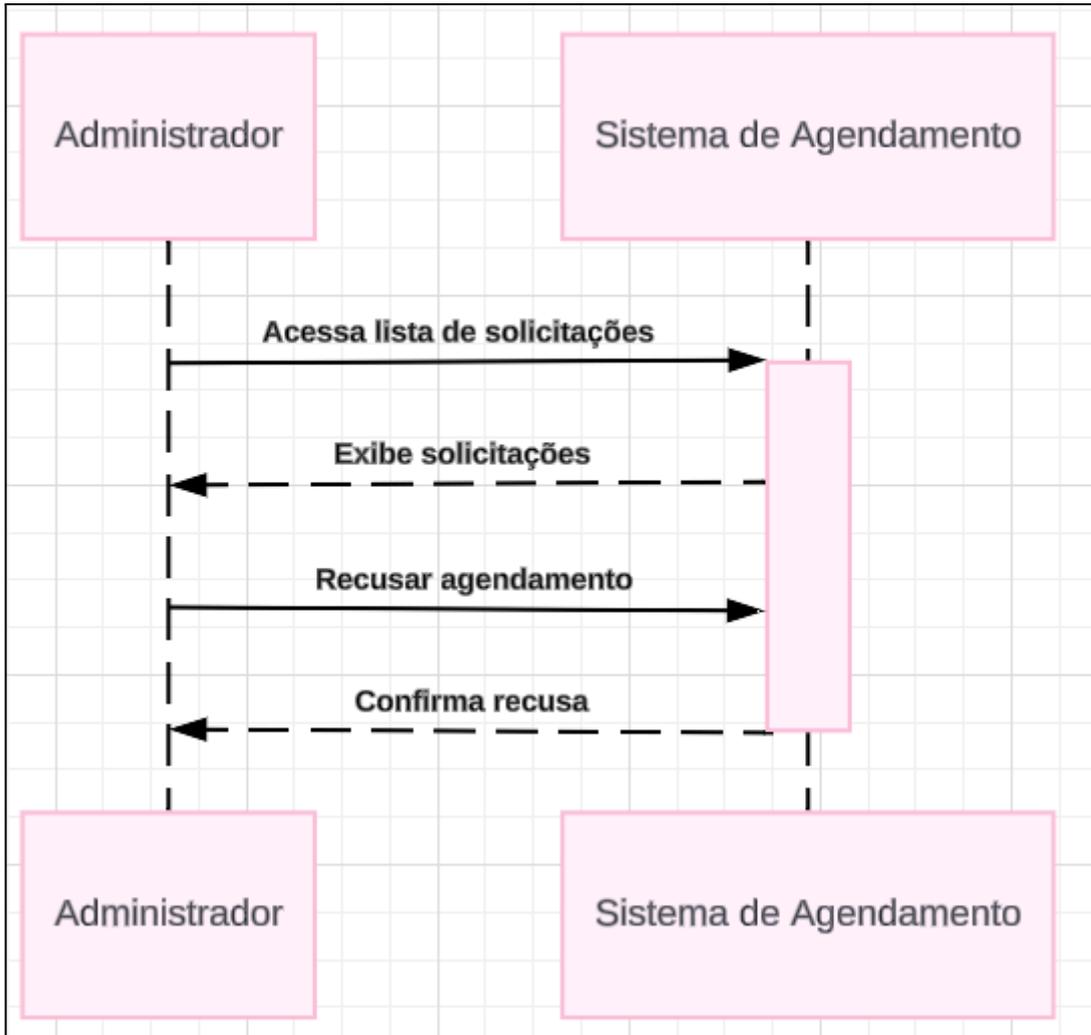
Fonte: elaborado pelo autor (2024).

Tabela 21 - Confirmar agendamento



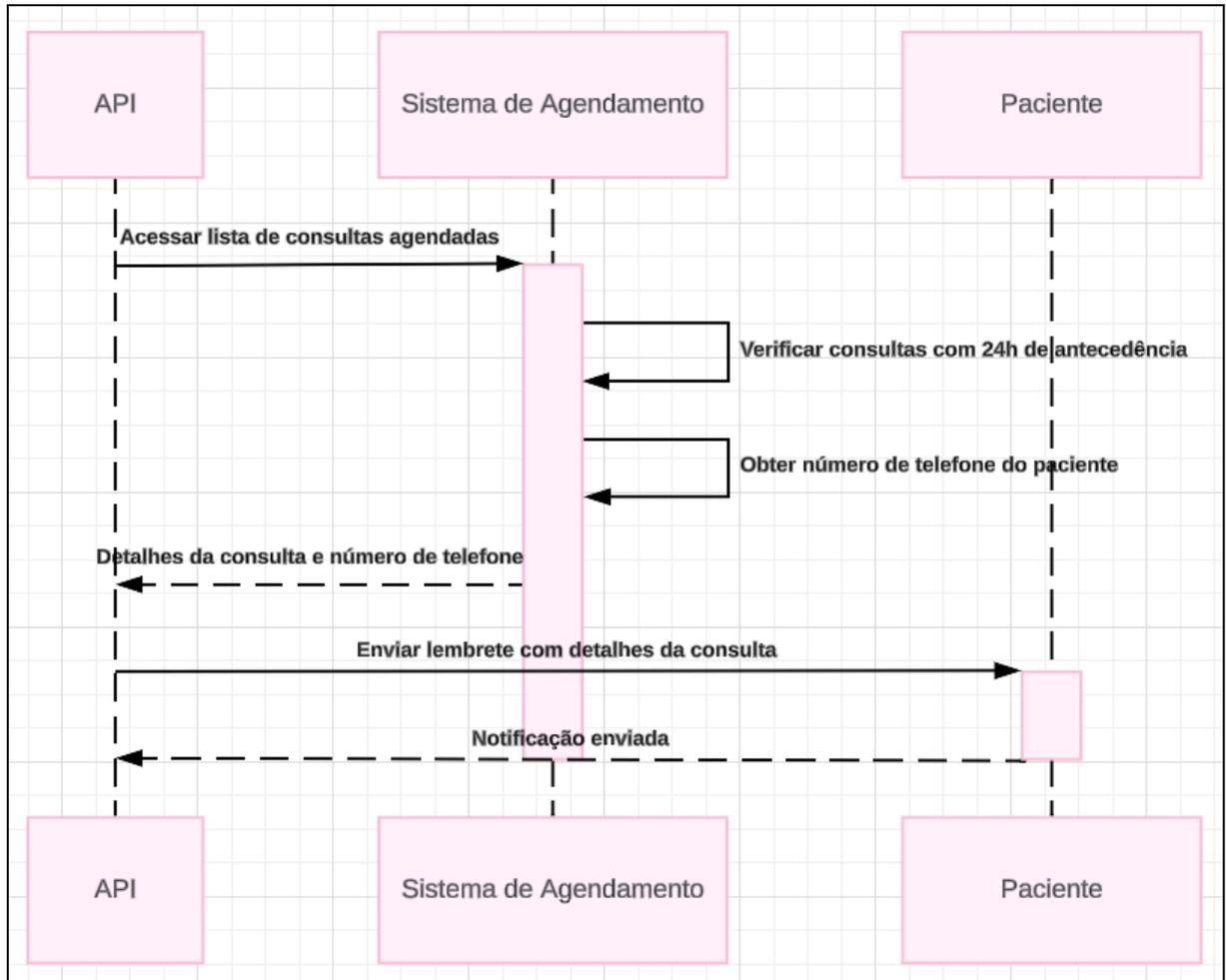
Fonte: elaborado pelo autor (2024).

Tabela 22 - Recusa agendamento



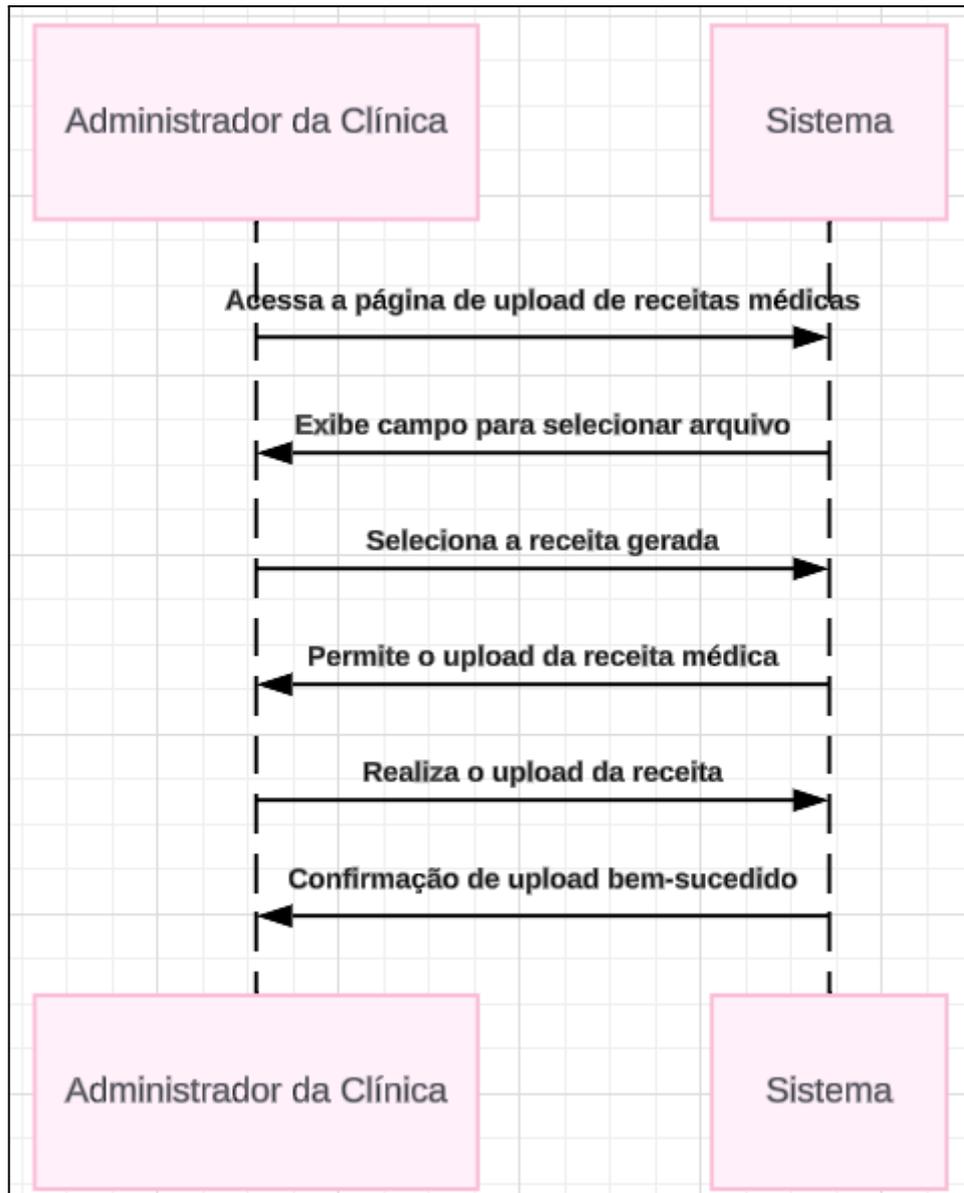
Fonte: elaborado pelo autor (2024).

Tabela 23 - Enviar lembrete de consulta



Fonte: elaborado pelo autor (2024).

Tabela 24 - Realiza upload de receitas médicas



Fonte: elaborado pelo autor (2024).

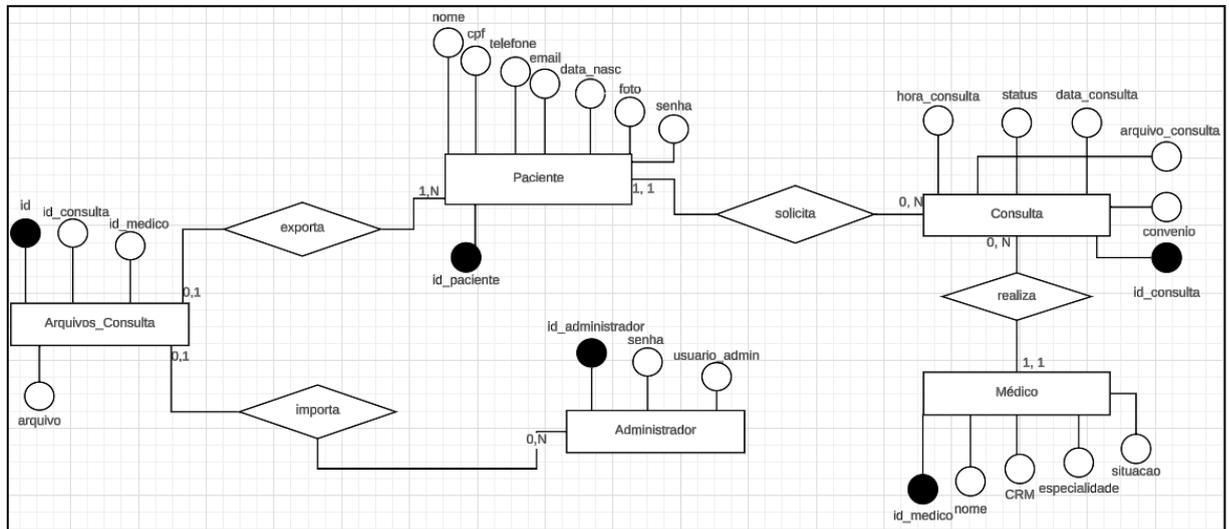
## 2.4 Banco de Dados

Esta seção descreve a modelagem do Banco de Dados do sistema, contendo o Modelo Conceitual, apresentado através do Diagrama Entidade Relacionamento e Projeto Lógico de tabelas.

### 2.3.1 Diagrama Entidade Relacionamento

A seguir será apresentado um exemplo de Diagrama Entidade Relacionamento.

**Figura 5 - Diagrama Entidade Relacionamento**



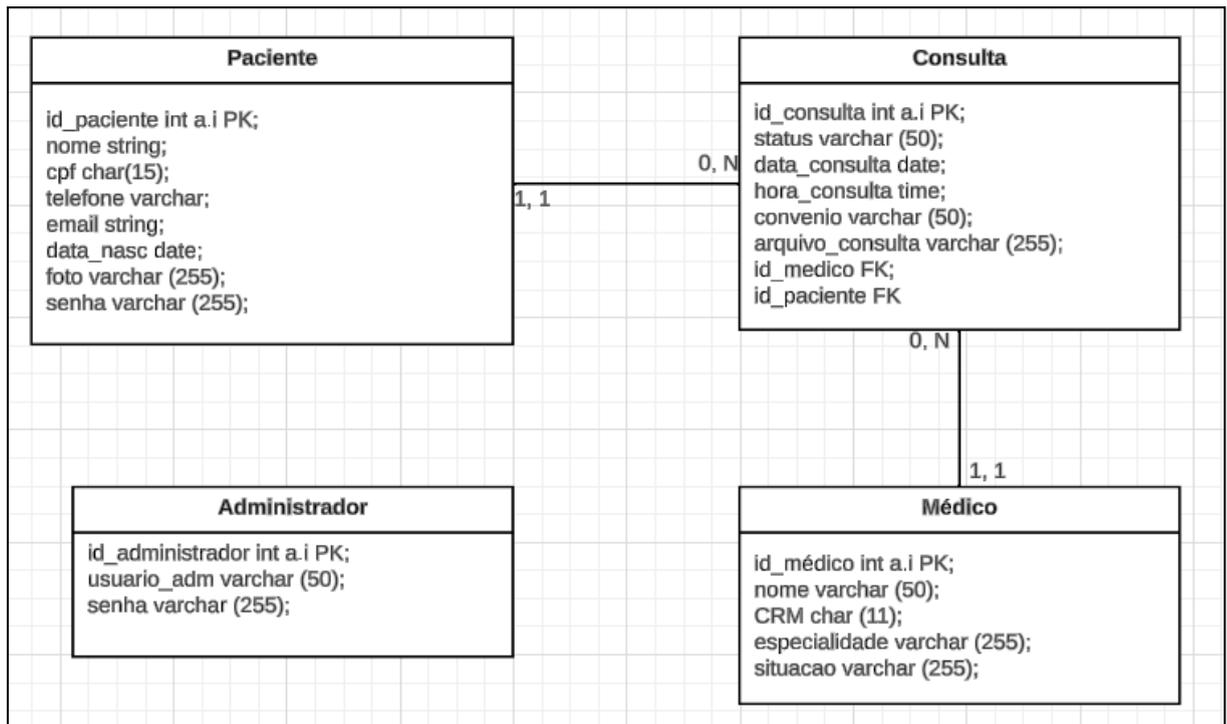
Fonte: elaborado pelo autor (2024).

Um paciente pode ter várias consultas, cada uma realizada por um único médico em uma única sala. Administradores e Pacientes devem ter uma conta de usuário. Médicos podem realizar várias consultas. O status da consulta pode ser utilizado para gerenciar seu estado (agendada, cancelada) e a especialidade do médico é registrada (como clínico geral, cardiologista).

### 2.3.2 Modelo Lógico

Nesta seção é ilustrado um exemplo de projeto lógico das tabelas.

**Figura 6 - Projeto Lógico**



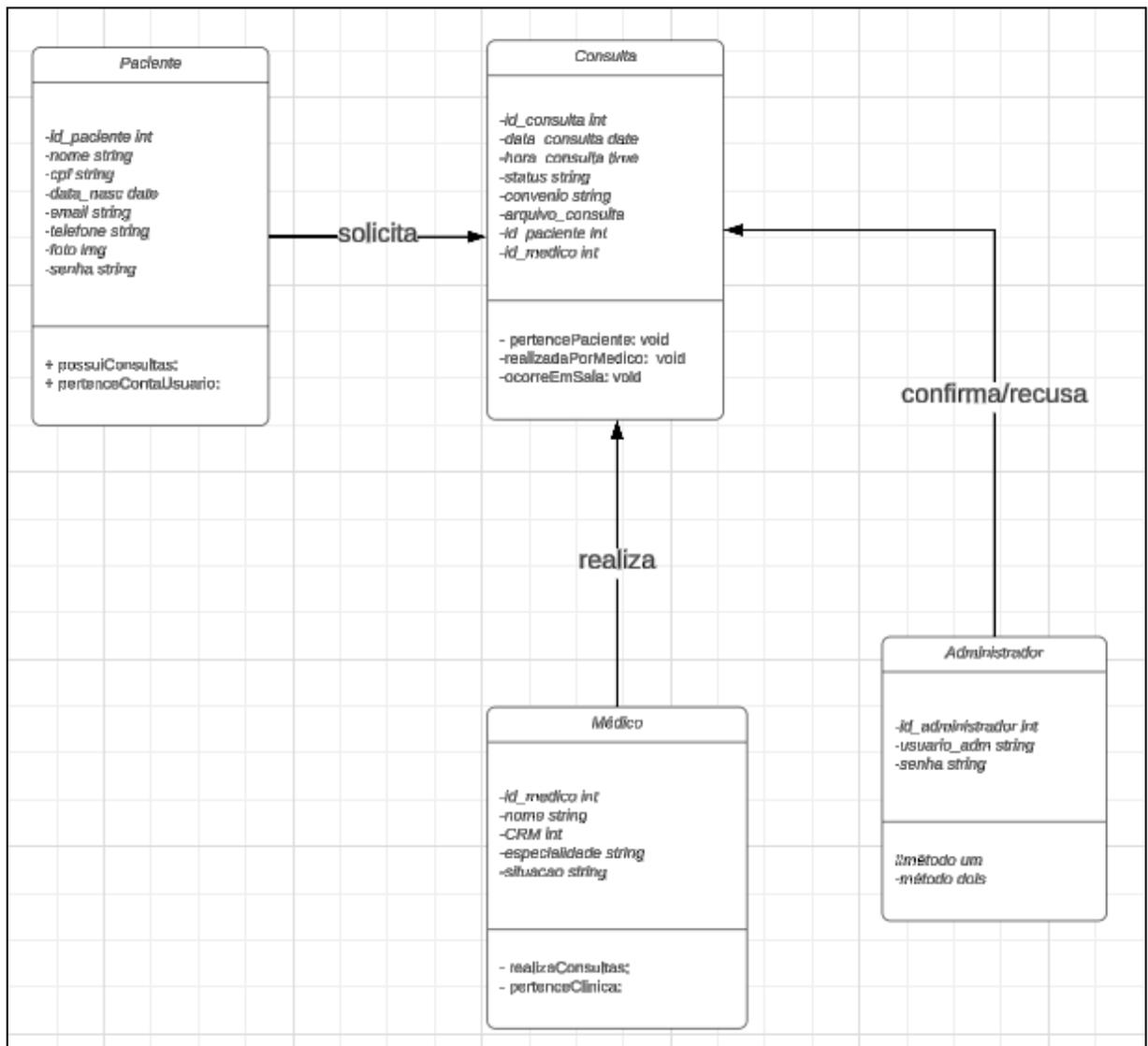
Fonte: elaborado pelo autor (2024).

No modelo lógico, pacientes têm consultas realizadas por médicos. Médicos podem realizar várias consultas. Administradores e Pacientes devem ter uma conta de usuário. Consultas têm status, e médicos têm especialidades. As relações são definidas por chaves primárias e estrangeiras.

## 2.5 Diagrama de Classes

A seguir será apresentado um exemplo de Diagrama de Classes do Sistema.

**Figura 7 - Diagrama de Classes**



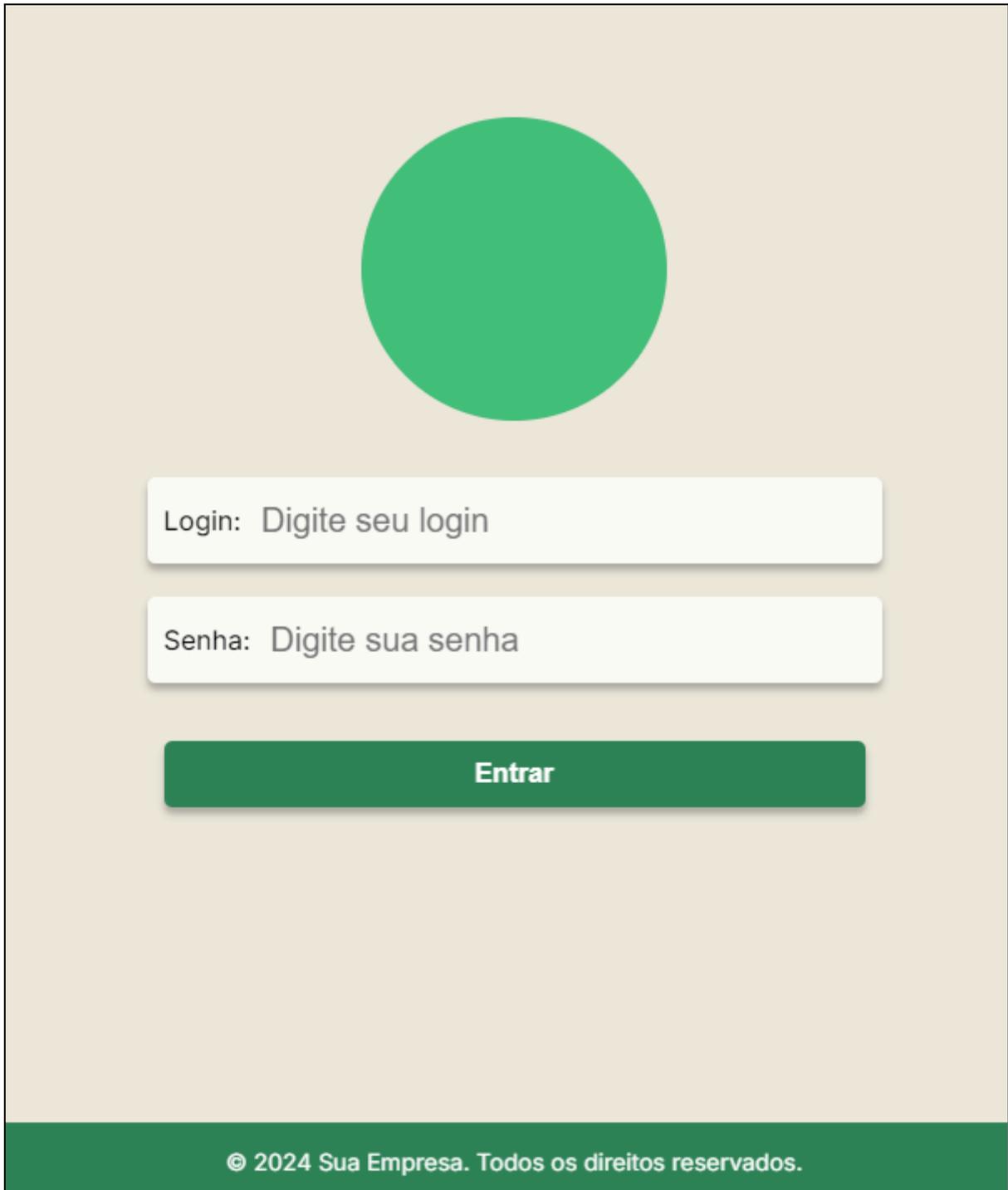
Fonte: Elaborado pelo autor (2024)

## 2.6 Telas do Sistema

Nesta seção serão descritas e apresentadas as principais telas do sistema com o layout definitivo e com exemplos reais de funcionamento.

### Figura 8 - Tela de Login

Tela para realização de login de usuários, tanto pacientes quanto administradores da clínica.

A interface de usuário para o cadastro de usuários. No topo, há um círculo verde sólido. Abaixo dele, há dois campos de entrada de texto brancos com bordas arredondadas e sombras. O primeiro campo contém o texto "Login: Digite seu login" e o segundo contém "Senha: Digite sua senha". Abaixo dos campos, há um botão verde sólido com o texto "Entrar" em branco. Na base da interface, há uma barra verde sólida com o texto "© 2024 Sua Empresa. Todos os direitos reservados." em branco.

Login: Digite seu login

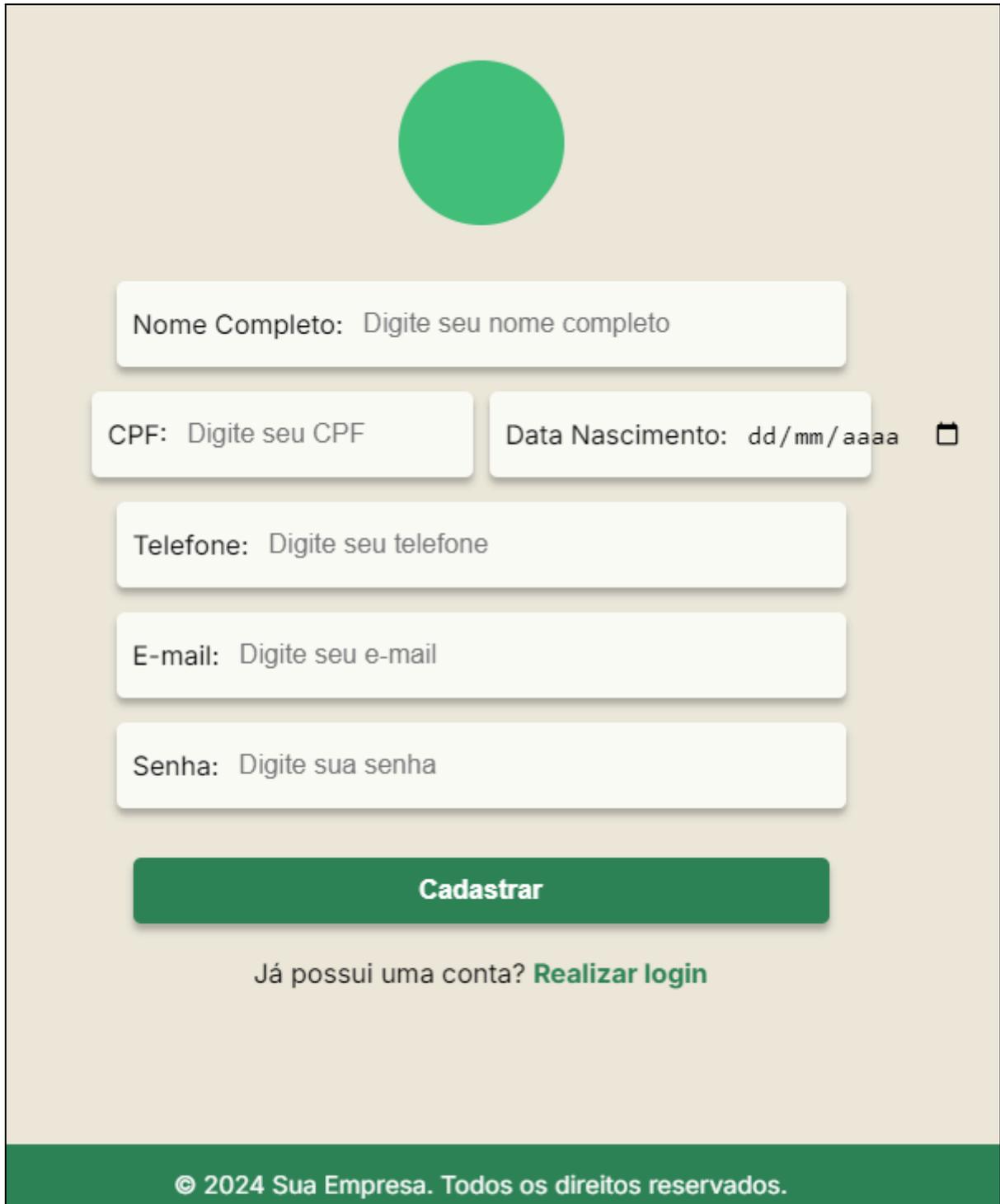
Senha: Digite sua senha

Entrar

© 2024 Sua Empresa. Todos os direitos reservados.

### Figura 9 - Tela de Cadastro

A seguir, na Figura 9, é apresentada a tela para realização de cadastro dos usuários, necessária para o primeiro acesso.



Nome Completo: Digite seu nome completo

CPF: Digite seu CPF

Data Nascimento: dd/mm/aaaa 

Telefone: Digite seu telefone

E-mail: Digite seu e-mail

Senha: Digite sua senha

**Cadastrar**

Já possui uma conta? [Realizar login](#)

© 2024 Sua Empresa. Todos os direitos reservados.

### Figura 10 -Tela de Visualização do Perfil

A seguir, na Figura 10, é apresentada a tela para visualização do perfil do paciente, com suas informações de cadastro.



**Figura 11 -Tela histórico de consultas e download de prescrição (paciente)**

A seguir, na Figura 11, é apresentada a tela utilizada pelo paciente para visualização do histórico de consultas e download de prescrições médicas



**Figura 12 -Tela de Médicos (Administrador)**

A seguir, na Figura 12, é apresentada a tela para realização de cadastro e edição dos médicos, que são realizados pelo administrador da clínica.

Nome	CRM	Especialidade	Situação	Ações
Dra. Marta Santos	45236	Ortotrauma	Férias	<input type="button" value="Editar"/>
Dr. João Silva	12345	Cardiologista	Ativo	<input type="button" value="Editar"/>
Dra. Mariana Oliveira	67890	Pediatra	Ativo	<input type="button" value="Editar"/>
Dr. Lucas Ferreira	54321	Ortopedista	Inativo	<input type="button" value="Editar"/>
Dra. Isabela Souza	09876	Ginecologista	Férias	<input type="button" value="Editar"/>
Dr. Gustavo Almeida	43215	Neurologista	Ativo	<input type="button" value="Editar"/>

© 2024 Facilitando Web. Todos os direitos reservados.

### Figura 13 - Tela para solicitar consultas

A seguir, na Figura 13, é apresentada a tela para solicitação de uma nova consulta pelo usuário, onde o mesmo seleciona o médico, data e horário da consulta desejada.

# Solicitar Consulta

**Login com sucesso!**

Médico:  ▼

Convênio:  ▼

Data:  📅

Hora:  ▼

**Solicitar Consulta**

© 2024 Facilitando Web

**Figura 14 - Tela para aceite/recusa de solicitações de consultas e upload/download de prescrição e histórico de consultas (Administrador)**

A seguir, na Figura 14, é apresentada a tela para confirmação ou recusa de solicitações de consulta, de acordo com a disponibilidade e diferentes situações que podem ocorrer na clínica.

The screenshot shows a web application interface with a dark green header. The header contains a navigation menu with the following items: Perfil, Gerenciar Consultas (highlighted), Pacientes Cadastrados, Médicos, and Sair. Below the header, the main content area has a light beige background. At the top of this area is the title 'Consultas Solicitadas' in a large, bold, dark green font. Below the title is a green notification message: 'Login com sucesso!'. The main content area displays a list of three consultation requests, each in a white card with a light green border. Each card contains the following information: 'Paciente: Luan Médico: Marta Convênio: Unimed', 'Data: 16/10/2024 Hora: 09:00:00 Status: Finalizado', and three buttons: 'Confirmar' (green), 'Recusar' (red), and 'Anexar receita' (green). The second card shows 'Paciente: Braitwaite Médico: Vitor Convênio: Ipe', 'Data: 13/10/2024 Hora: 17:30:00 Status: Finalizado', and the same three buttons. The third card shows 'Paciente: Braitwaite Médico: Leonardo Jaim Convênio: Fusex' and the same three buttons. At the bottom of the page, there is a dark green footer with the text '© 2024 Facilitando Web'.

Perfil Gerenciar Consultas Pacientes Cadastrados Médicos Sair

## Consultas Solicitadas

Login com sucesso!

Paciente: Luan Médico: Marta Convênio: Unimed  
Data: 16/10/2024 Hora: 09:00:00 Status: Finalizado

Confirmar Recusar Anexar receita

Paciente: Braitwaite Médico: Vitor Convênio: Ipe  
Data: 13/10/2024 Hora: 17:30:00 Status: Finalizado

Confirmar Recusar Anexar receita

Paciente: Braitwaite Médico: Leonardo Jaim Convênio: Fusex

© 2024 Facilitando Web

### 3 RESULTADOS E DISCUSSÕES

O presente capítulo destina-se a apresentar os principais resultados obtidos a partir da utilização do sistema, bem como discutir as implicações e melhorias que ele oferece tanto para os pacientes quanto para os administradores das clínicas.

#### 3.1 Relatórios e Telas de Visualização do Sistema

##### Figura 16 -Tela após login do paciente

Como resultado do login realizado pelo paciente, o mesmo é redirecionado para a tela de solicitação de consultas com uma mensagem de sucesso, como pode-se observar na figura 16.

Perfil Solicitar Consulta Minhas Consultas Sair

## Solicitar Consulta

Login com sucesso!

Médico: Vitor

Convênio: Particular

Data: dd/mm/aaaa

Hora: 08:00

Solicitar Consulta

© 2024 Facilitando Web

##### Figura 17 -Tela após login do Administrador

A seguir na figura 17, pode-se visualizar a mensagem de sucesso apresentada ao realizar login com o usuário Administrador, sendo direcionado para o gerenciamento de consultas.



**Figura 18 -Tela de perfil**

A seguir, na figura 18, pode-se visualizar a tela de perfil do usuário, que segue o mesmo padrão tanto para paciente quanto administrador.



**Figura 19 -Tela de listagem de pacientes (administrador)**

A seguir na figura 19, é apresentada a tela de listagem de pacientes, em que apenas o administrador tem acesso, e pode realizar as ações desejadas, como novos cadastros ou edição de perfil.



Perfil Gerenciar Consultas Pacientes Cadastrados Médicos Sair

## Lista de Pacientes Cadastrados

ID	Nome	CPF	Data de Nascimento	Telefone	E-mail	Ações		
1	Julio Carvalho	04333333333	2024-09-04	5999999999	julio@gmail.com	Visualizar	Editar	Excluir
2	Braitwaite	15226334188	2024-09-04	55999998888	teste@gmail.com	Visualizar	Editar	Excluir
15	Luan	5555555	2024-10-01	55555555	teste@gmail.com	Visualizar	Editar	Excluir

Adicionar Paciente

© 2024 Facilitando Web. Todos os direitos reservados.

**Figura 20 -Tela listagem de médicos (administrador)**

Na figura 20, é apresentada a tela para cadastro de novos médicos, listagem e edição dos mesmos.



Perfil Gerenciar Consultas Pacientes Cadastrados Médicos Sair

## Médicos

Nome Médico: Digite o nome do médico CRM: Digite o CRM

Especialidade: Digite a especialidade Situação: Selecione a situação

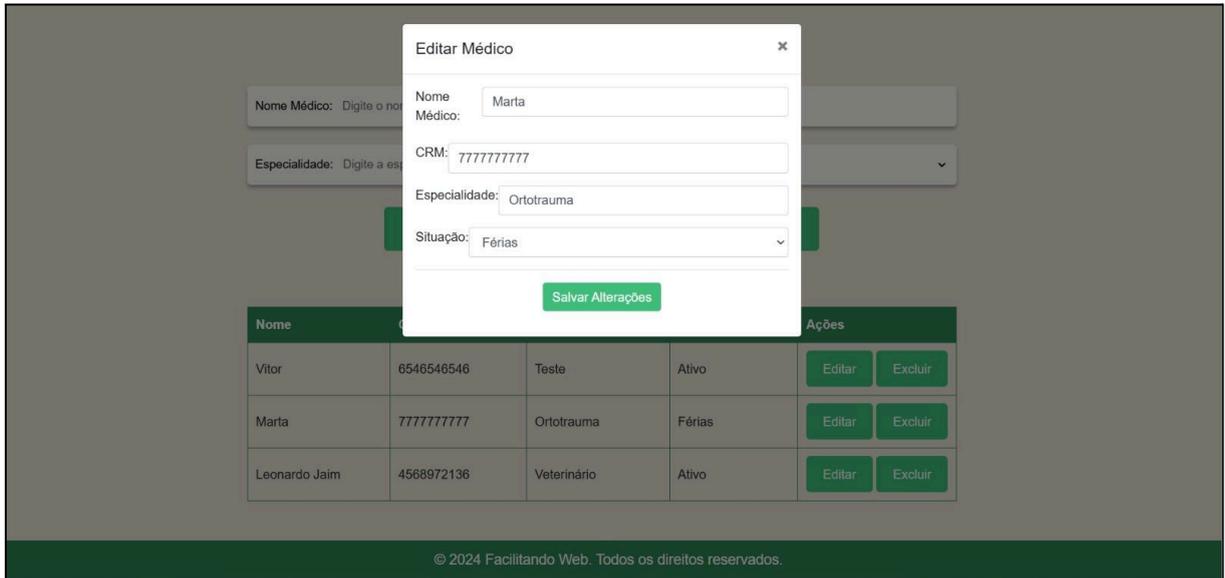
Adicionar Médico

Nome	CRM	Especialidade	Situação	Ações	
Vitor	6546546546	Teste	Ativo	Editar	Excluir
Marta	7777777777	Ortotrauma	Férias	Editar	Excluir

© 2024 Facilitando Web. Todos os direitos reservados.

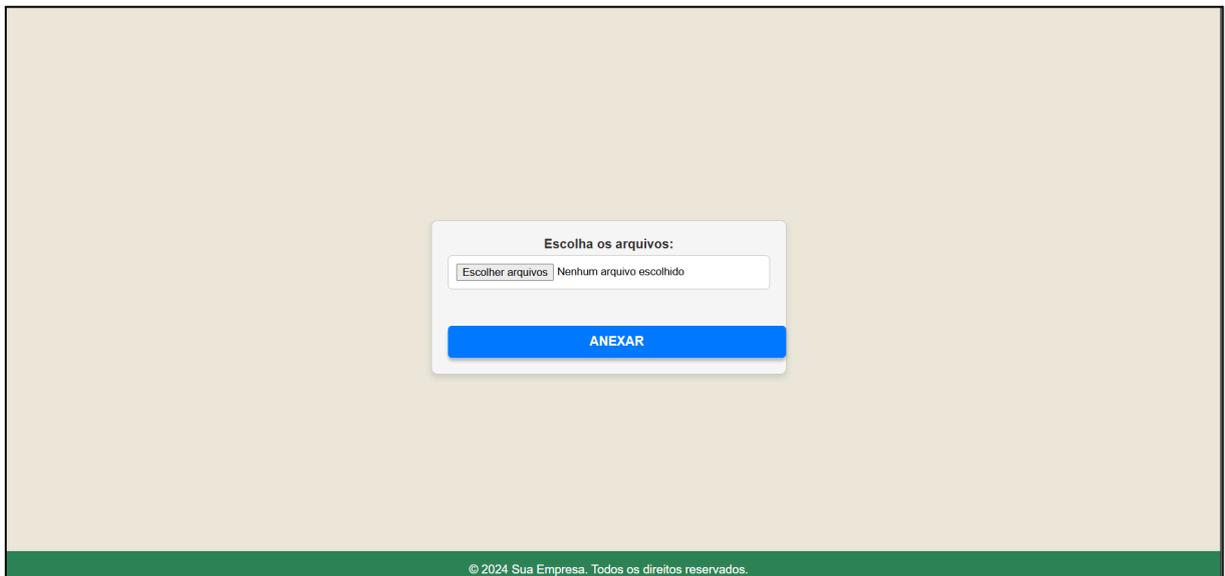
### Figura 21 -Tela edição de médicos (administrador)

A seguir, é apresentado o modal para edição de médicos.



### Figura 22 -Tela de upload de anexo (administrador)

Na figura seguinte, é mostrada a tela disponibilizada para inclusão de anexo referente a consulta, que pode ser realizado pelo administrador através do botão “Anexar Receita” do gerenciamento de consultas.



### Figura 23 -Tela de confirmação de upload de anexo (administrador)

Após a realização do upload, é apresentada uma mensagem de confirmação de anexo, de acordo com a figura 23.



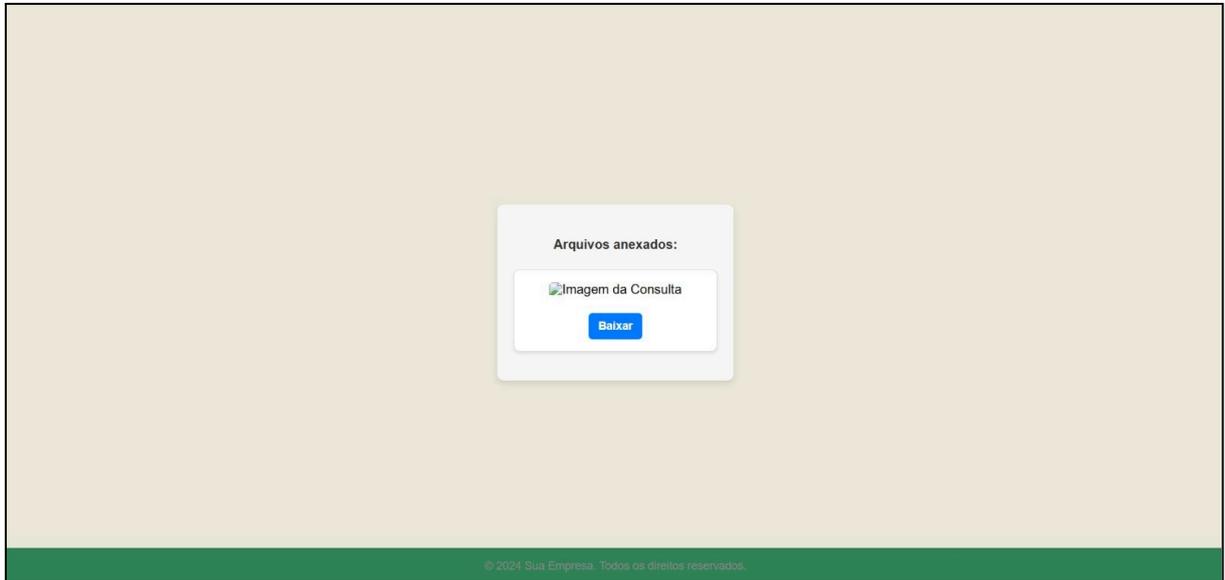
### Figura 24 -Tela de minhas consultas (paciente)

Na figura 24, é apresentada a tela referente às "Minhas Consultas" do usuário, quando o mesmo ainda não possui consultas solicitadas ou concluídas.



### Figura 25 -Tela de download de receita (paciente/admin)

Na figura seguinte, é apresentada a tela disponibilizada ao acessar o anexo da consulta para realizar download do mesmo.



**Figura 26 -Tela de download de receita (paciente/admin)**

Na figura seguinte, é apresentada a validação de consulta ao solicitar diferentes consultas no mesmo horário com o mesmo médico, utilizando pacientes diferentes. A mesma validação ocorre quando um paciente tenta marcar duas consultas no mesmo horário.

# Solicitar Consulta

**Ocorreram erros no preenchimento do formulário:**

- Já existe uma consulta marcada com esse médico, nessa data e horário.

Médico:

Convênio:

Data:

Hora:

**Solicitar Consulta**

© 2024 Facilitando Web

### Figura 27 -Tela de download de receita (paciente/admin)

Na figura a seguir, pode se verificar que em consultas que estão com o status “Finalizado”, o botão de “Excluir” é desabilitado.



A maioria dos objetivos específicos foi plenamente alcançada, incluindo o cadastro e manutenção de dados de pacientes e médicos, a implementação de um sistema intuitivo de agendamento, a gestão da disponibilidade de médicos, e a distribuição de receitas médicas de forma eficiente.

Contudo, a funcionalidade de oferecer relatórios detalhados para os administradores, com foco no controle de fluxo de pacientes, monitoramento do sistema e análise de dados de agendamento, não foi concluída nesta fase do projeto. Esse objetivo será tratado como um trabalho futuro, garantindo que o sistema possa evoluir e atender a todas as necessidades previstas com ainda mais eficiência.

### 3.2 Questionário de avaliação e testes do Sistema

Na próxima seção será abordado o item questionário de Avaliação e Testes do Sistema. Essa etapa inclui a aplicação de um questionário estruturado para coleta de feedback dos usuários, além da realização de testes práticos.

O objetivo foi verificar o desempenho, a usabilidade e a conformidade do sistema com os requisitos estabelecidos. Os testes foram realizados com sete (7) usuários do tipo “Paciente” selecionados de forma aleatória, permitindo uma avaliação imparcial e diversificada. Os valores apresentados no questionário representam a média das respostas

fornecidas pelos participantes.

**Tabela 25 - Questionário de Avaliação**

<i>Afirmção</i>	<i>Nota</i>
O design do sistema é atraente.	9
No meu dia a dia, o sistema é relevante para as minhas necessidades.	7
O sistema é intuitivo.	6
Eu recomendaria o sistema para amigos e familiares.	8
Me senti bem utilizando o sistema.	8
O sistema é fácil de aprender a usar.	6
A experiência geral com o sistema é positiva.	7
As informações apresentadas no sistema são precisas e confiáveis.	8
A interface do sistema é moderna e agradável.	8
O sistema tem impacto na sociedade.	9

## **4 CONCLUSÃO**

Este trabalho teve como objetivo o desenvolvimento de um Sistema de Agendamento de Consultas para clínicas, com foco na melhoria da eficiência administrativa e na otimização da experiência dos pacientes. Através da implementação de funcionalidades como login de usuários, cadastro de pacientes, visualização de horários e disponibilidades de médicos, o sistema propõe uma solução prática para problemas recorrentes em clínicas que ainda dependem de processos manuais e desorganizados.

A análise dos casos de uso, como a solicitação, confirmação e recusa de agendamentos, bem como o gerenciamento da disponibilidade dos médicos e o download de receitas médicas, demonstra como a tecnologia pode agilizar e integrar processos críticos no ambiente clínico. A possibilidade de automatizar tarefas administrativas e garantir um fluxo contínuo de informações entre pacientes e administradores reduz a margem para erros, melhora a comunicação e, conseqüentemente, contribui para um atendimento mais ágil e eficaz.

Além dos benefícios operacionais, a adoção de um sistema de agendamento digital traz vantagens econômicas consideráveis para as clínicas, como a redução de custos administrativos, a otimização do uso dos recursos disponíveis e o aumento da produtividade dos profissionais de saúde. A capacidade de gerar prontuários e monitorar a utilização do sistema também proporciona aos administradores um controle mais preciso sobre a dinâmica das consultas e o fluxo de pacientes, facilitando a tomada de decisões.

Em um cenário onde a digitalização tem se tornado essencial para a evolução do setor de saúde, o sistema proposto representa uma alternativa viável para clínicas que buscam não apenas modernizar sua gestão, mas também proporcionar uma melhor experiência para seus pacientes. Além disso, o sistema pode servir como base para futuras melhorias e adaptações, como a integração com outros sistemas de saúde ou a inclusão de novos recursos, sempre com o objetivo de atender às demandas de um setor em constante evolução.

Portanto, é possível concluir que a implementação de tecnologias como o sistema de agendamento proposto pode impactar positivamente tanto a eficiência administrativa das clínicas quanto a qualidade do atendimento aos pacientes, contribuindo para um ambiente de saúde mais organizado, acessível e satisfatório para todos os envolvidos. Como trabalhos futuros, sugere-se a implementação de relatórios detalhados de tratamentos dos pacientes; o envio de lembretes por meio de APIs, para reforçar o acompanhamento do cuidado e aumentar a taxa de comparecimento às consultas; a inclusão de parametrizações para o cadastro individual de cada clínica, de forma que possa ser personalizável de acordo com cada situação; a implementação de visualização de agenda geral e filtrada para os administradores da clínica; e a inclusão de funcionalidades que contribuam para uma melhor robustez e usabilidade do sistema, como a incorporação de filtros de pesquisa avançados e maior detalhamento no cadastro de pacientes.

## REFERÊNCIAS

- AGENDAR CONSULTA.** Disponível em: <https://agendarconsulta.com>. Acesso em: 15 mar. 2024.
- BENTO, Evaldo Junior.** **Desenvolvimento web com PHP e MySQL.** 2021.
- CENTRAL DE CONSULTAS.** Disponível em: <https://centraldeconsultas.med.br/agenda/>. Acesso em: 10 abril. 2024.
- COCKBURN, Alistair.** **Escrevendo Casos de Usos Eficazes: Um guia prático para desenvolvedores de software.** 2005.
- DEVMEDIA.** **PHP: adicionando segurança às suas aplicações.** Disponível em: <https://www.devmedia.com.br/php-adicionando-seguranca-a-suas-aplicacoes/38054>. Acesso em: 05 mai. 2024.
- F5 Networks.** **Web Application Security.** Disponível em: [https://www.f5.com/pt\\_br/glossary/web-application-security#:~:text=A%20seguran%C3%A7a%20de%20aplica%C3%A7%C3%B5es%20Web,por%20amea%C3%A7as%20baseadas%20na%20Internet](https://www.f5.com/pt_br/glossary/web-application-security#:~:text=A%20seguran%C3%A7a%20de%20aplica%C3%A7%C3%B5es%20Web,por%20amea%C3%A7as%20baseadas%20na%20Internet). Acesso em 03 julho 2024.
- ICLINIC.** **Sistema inteligente para clínicas e consultórios.** Disponível em: <https://lps.iclinic.com.br/homepage/>. Acesso em: 15 mar. 2024.
- KHAN, Sadik; KHANAM, Aaisha T.** Study on MVC Framework for Web Development in PHP. **Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol**, p. 414-419, 2023.
- LARAVEL.** **Laravel - The PHP Framework For Web Artisans.** Disponível em: <https://laravel.com>. Acesso em: 17 jun. 2024.
- MEDEIROS, Otávio Lopes.** **Segurança em aplicações Web.** 2016.
- MICROSOFT.** **Documentation for Visual Studio Code.** Disponível em: <https://code.visualstudio.com/docs>. Acesso em: 22 abril. 2024.
- MILANI, André.** **MySQL-guia do programador.** 2007.
- PHP GROUP.** **PHP: Hypertext Preprocessor (PHP) - Copyright.** Disponível em: <https://www.php.net/copyright.php>. Acesso em: 17 jun. 2024.
- WIKIPÉDIA.** **Diagrama de caso de uso.** Disponível em: [https://pt.wikipedia.org/wiki/Diagrama\\_de\\_caso\\_de\\_uso#:~:text=Segundo%20Ivar%20Jacobs on%20podemos%20dizer,outro%20software](https://pt.wikipedia.org/wiki/Diagrama_de_caso_de_uso#:~:text=Segundo%20Ivar%20Jacobs on%20podemos%20dizer,outro%20software). Acesso em: 08 jun. 2024.



## APÊNDICE A : SCRIPT SQL DE CRIAÇÃO DAS TABELAS

```

-- Banco de dados: `testenovo-tcc`
--
-----

--
-- Estrutura para tabela `administrador`
--

CREATE TABLE `administrador` (
  `id_administrador` int(11) NOT NULL,
  `usuario_adm` varchar(50) DEFAULT NULL,
  `senha` varchar(255) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Despejando dados para a tabela `administrador`
--

INSERT INTO `administrador` (`id_administrador`, `usuario_adm`, `senha`) VALUES
(1, 'admin',
'$2y$10$j7b6TOLILwf6.zKJuSvVEemlcsEWQCLvcQTNAJEgybchOhK6lvFMi');

-----

--
-- Estrutura para tabela `cache`
--

```

```
CREATE TABLE `cache` (  
  `key` varchar(255) NOT NULL,  
  `value` mediumtext NOT NULL,  
  `expiration` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

-----

```
--  
-- Estrutura para tabela `cache_locks`  
--
```

```
CREATE TABLE `cache_locks` (  
  `key` varchar(255) NOT NULL,  
  `owner` varchar(255) NOT NULL,  
  `expiration` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

-----

```
--  
-- Estrutura para tabela `consulta`  
--
```

```
CREATE TABLE `consulta` (  
  `id_consulta` int(11) NOT NULL,  
  `data_consulta` date DEFAULT NULL,  
  `hora_consulta` time DEFAULT NULL,
```

```

`status` varchar(50) DEFAULT NULL,
`convenio` varchar(50) DEFAULT NULL,
`arquivo_consulta` varchar(255) DEFAULT NULL,
`id_paciente` int(11) DEFAULT NULL,
`id_medico` int(11) DEFAULT NULL,
`created_at` timestamp NOT NULL DEFAULT current_timestamp(),
`updated_at` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE
current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Despejando dados para a tabela `consulta`
--

INSERT INTO `consulta` (`id_consulta`, `data_consulta`, `hora_consulta`, `status`,
`convenio`, `arquivo_consulta`, `id_paciente`, `id_medico`, `created_at`, `updated_at`)
VALUES
(4, '2024-10-16', '09:00:00', 'Confirmada', 'Unimed', NULL, 15, 6, '2024-10-09 05:25:55',
'2024-10-18 05:00:21'),
(25, '2024-10-13', '17:30:00', 'Confirmada', 'Ipe', NULL, 2, 2, '2024-10-23 15:52:01',
'2024-10-24 02:10:38'),
(26, '2024-10-28', '16:00:00', 'Pendente', 'Fusex', NULL, 2, 9, '2024-10-23 15:53:58',
'2024-10-23 15:53:58'),
(27, '2024-11-01', '08:00:00', 'Pendente', 'Unimed', NULL, 2, 6, '2024-10-23 17:55:18',
'2024-10-23 17:55:18'),
(28, '2024-10-25', '08:00:00', 'Pendente', 'Ipe', NULL, 1, 9, '2024-10-23 17:56:18', '2024-10-23
17:56:18');

-----

--
-- Estrutura para tabela `failed_jobs`

```

--

```
CREATE TABLE `failed_jobs` (  
  `id` bigint(20) UNSIGNED NOT NULL,  
  `uuid` varchar(255) NOT NULL,  
  `connection` text NOT NULL,  
  `queue` text NOT NULL,  
  `payload` longtext NOT NULL,  
  `exception` longtext NOT NULL,  
  `failed_at` timestamp NOT NULL DEFAULT current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

-----

--

-- Estrutura para tabela `jobs`

--

```
CREATE TABLE `jobs` (  
  `id` bigint(20) UNSIGNED NOT NULL,  
  `queue` varchar(255) NOT NULL,  
  `payload` longtext NOT NULL,  
  `attempts` tinyint(3) UNSIGNED NOT NULL,  
  `reserved_at` int(10) UNSIGNED DEFAULT NULL,  
  `available_at` int(10) UNSIGNED NOT NULL,  
  `created_at` int(10) UNSIGNED NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

-----

```
--  
-- Estrutura para tabela `job_batches`  
--  
  
CREATE TABLE `job_batches` (  
  `id` varchar(255) NOT NULL,  
  `name` varchar(255) NOT NULL,  
  `total_jobs` int(11) NOT NULL,  
  `pending_jobs` int(11) NOT NULL,  
  `failed_jobs` int(11) NOT NULL,  
  `failed_job_ids` longtext NOT NULL,  
  `options` mediumtext DEFAULT NULL,  
  `cancelled_at` int(11) DEFAULT NULL,  
  `created_at` int(11) NOT NULL,  
  `finished_at` int(11) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

-----

```
--  
-- Estrutura para tabela `medico`  
--  
  
CREATE TABLE `medico` (  
  `id_medico` int(11) NOT NULL,  
  `nome` varchar(50) DEFAULT NULL,  
  `CRM` char(11) DEFAULT NULL,  
  `especialidade` varchar(255) DEFAULT NULL,
```

```

`situacao` varchar(20) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Despejando dados para a tabela `medico`
--

INSERT INTO `medico` (`id_medico`, `nome`, `CRM`, `especialidade`, `situacao`) VALUES
(2, 'Vitor', '6546546546', 'Teste', 'Ativo'),
(6, 'Marta', '7777777777', 'Ortotrauma', 'Férias'),
(9, 'Leonardo Jaim', '4568972136', 'Veterinário', 'Ativo');

-----

--
-- Estrutura para tabela `migrations`
--

CREATE TABLE `migrations` (
  `id` int(10) UNSIGNED NOT NULL,
  `migration` varchar(255) NOT NULL,
  `batch` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

--
-- Despejando dados para a tabela `migrations`
--

INSERT INTO `migrations` (`id`, `migration`, `batch`) VALUES

```

```
(1, '0001_01_01_000000_create_users_table', 1),
(2, '0001_01_01_000001_create_cache_table', 1),
(3, '0001_01_01_000002_create_jobs_table', 1);
```

```
-----
```

```
--
-- Estrutura para tabela `paciente`
--
```

```
CREATE TABLE `paciente` (
  `id_paciente` int(11) NOT NULL,
  `nome` varchar(50) DEFAULT NULL,
  `cpf` varchar(14) DEFAULT NULL,
  `data_nasc` date DEFAULT NULL,
  `email` varchar(255) DEFAULT NULL,
  `telefone` char(11) NOT NULL,
  `foto` varchar(255) DEFAULT NULL,
  `senha` varchar(255) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

```
--
-- Despejando dados para a tabela `paciente`
--
```

```
INSERT INTO `paciente` (`id_paciente`, `nome`, `cpf`, `data_nasc`, `email`, `telefone`,
`foto`, `senha`) VALUES
(1, 'Julio Carvalho', '04333333333', '2024-09-04', 'julio@gmail.com', '5999999999', NULL,
'$2y$10$qCi.9RYOO0bAdSpjfBAzGeVbAqe2tJ3zvrtpB4BWmI6RCyfWtDHnO'),
(2, 'Braitwaite', '15226334188', '2024-09-04', 'teste@gmail.com', '55999998888', NULL,
```

```
'$2y$10$yvmvPaQUIfVe.70GdZma5uXUexBA43YE98TtgqpE11W.orOtmU8..'),  
(15, 'Luan', '5555555', '2024-10-01', 'teste@gmail.com', '5555555', NULL, '55555');
```

```
-----
```

```
--
```

```
-- Estrutura para tabela `password_reset_tokens`
```

```
--
```

```
CREATE TABLE `password_reset_tokens` (  
  `email` varchar(255) NOT NULL,  
  `token` varchar(255) NOT NULL,  
  `created_at` timestamp NULL DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
-----
```

```
--
```

```
-- Estrutura para tabela `sessions`
```

```
--
```

```
CREATE TABLE `sessions` (  
  `id` varchar(255) NOT NULL,  
  `user_id` bigint(20) UNSIGNED DEFAULT NULL,  
  `ip_address` varchar(45) DEFAULT NULL,  
  `user_agent` text DEFAULT NULL,  
  `payload` longtext NOT NULL,  
  `last_activity` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

--

-- Despejando dados para a tabela `sessions`

--

```
INSERT INTO `sessions` (`id`, `user_id`, `ip_address`, `user_agent`, `payload`,
`last_activity`) VALUES
```

```
('nzDxMmJY9F2gQk2HuGOS8H2HLvXGN9dsp7KaToXZ', NULL, '127.0.0.1', 'Mozilla/5.0
(Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/130.0.0.0 Safari/537.36',
'YTozOntzOjk6Il9wcmV2aW91cyI7YToxOntzOjM6InVybCI7czozMzoiaHR0cDovLzEyNy4
wLjAuMTo4MDAwL2FkbWluL2xvZ2luljt9czo2OiJfZmxhc2giO2E6Mjp7czozOiJvbGQiO2
E6MDp7fXM6MzoibmV3IjthOjA6e319czo2OiJfdG9rZW4iO3M6NDA6ImtzN2xpSExSTTF
UUzdRVFJvNVJYaFZQOUtDUWE4b0cwTG14UXpaV2ciO30=', 1729725151),
```

```
('rtFXwz9RZMyHzRQv52Axs3Hs6FxAscJpBrlG5C8T', NULL, '127.0.0.1', 'Mozilla/5.0
(Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/130.0.0.0 Safari/537.36',
'YTozOntzOjk6Il9wcmV2aW91cyI7YToxOntzOjM6InVybCI7czozMzoiaHR0cDovLzEyNy4
wLjAuMTo4MDAwL2FkbWluL2xvZ2luljt9czo2OiJfZmxhc2giO2E6Mjp7czozOiJvbGQiO2
E6MDp7fXM6MzoibmV3IjthOjA6e319czo2OiJfdG9rZW4iO3M6NDA6IjFjVThjTENsa2thc
DFZdXE1UHl4aGJvZEIMWVdGb0NrSDVWRUc2ZHAiO30=', 1729702932);
```

-----

--

-- Estrutura para tabela `users`

--

```
CREATE TABLE `users` (
  `id` bigint(20) UNSIGNED NOT NULL,
  `name` varchar(255) NOT NULL,
  `email` varchar(255) NOT NULL,
  `email_verified_at` timestamp NULL DEFAULT NULL,
  `password` varchar(255) NOT NULL,
```

```
`remember_token` varchar(100) DEFAULT NULL,  
`created_at` timestamp NULL DEFAULT NULL,  
`updated_at` timestamp NULL DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```