

**CENTRO UNIVERSITÁRIO PARA O DESENVOLVIMENTO DO ALTO VALE DO  
ITAJAÍ - UNIDAVI**

**KEVIN MARCHI**

**CATÁLOGO MÉDICO: DESENVOLVIMENTO DE PROTÓTIPO DE APLICAÇÃO  
WEB RESPONSIVA**

**RIO DO SUL  
2021**

**CENTRO UNIVERSITÁRIO PARA O DESENVOLVIMENTO DO ALTO VALE DO  
ITAJAÍ - UNIDAVI**

**KEVIN MARCHI**

**CATÁLOGO MÉDICO: DESENVOLVIMENTO DE PROTÓTIPO DE APLICAÇÃO  
WEB RESPONSIVA**

Trabalho de Conclusão de Curso a ser apresentado ao curso de Sistemas da Informação, da Área das Ciências Naturais, da Computação e das Engenharias, do Centro Universitário para o Desenvolvimento do Alto Vale do Itajaí, como condição parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

Prof. Orientador: M.e Jullian Hermann Creutzberg

**RIO DO SUL  
2021**

**CENTRO UNIVERSITÁRIO PARA O DESENVOLVIMENTO DO ALTO VALE DO  
ITAJAÍ - UNIDAVI**

**KEVIN MARCHI**

**CATÁLOGO MÉDICO: DESENVOLVIMENTO DE PROTÓTIPO DE APLICAÇÃO  
WEB RESPONSIVA**

Trabalho de Conclusão de Curso a ser apresentado ao curso de Sistemas da Informação, da Área das Ciências Naturais, da Computação e das Engenharias, do Centro Universitário para o Desenvolvimento do Alto Vale do Itajaí- UNIDAVI, a ser apreciado pela Banca Examinadora, formada por:

---

Professor Orientador: M.e Jullian Hermann Creutzberg

Banca Examinadora:

---

Prof. M.e Fernando Andrade Bastos

---

Prof. M.e Jeancarlo Visentainer

Rio do Sul, 01 de dezembro de 2021.

## RESUMO

O setor da saúde tem se mostrado uma das áreas mais importantes para o bem-estar da sociedade, surgindo, portanto, a necessidade da criação de novas tecnologias para o auxílio de toda a população. Este trabalho tem como objetivo desenvolver um protótipo de aplicação web responsiva visando disponibilizar um catálogo médico, que permita o agendamento de consultas para a população em geral. Quanto a metodologia, se caracteriza como uma pesquisa aplicada e descritiva. Para que os objetivos sejam alcançados foi realizada uma revisão da literatura sobre as tecnologias e linguagens de programação utilizadas no desenvolvimento. Também foi realizada uma pesquisa por aplicações semelhantes ao protótipo proposto visando a construção da seção do estado da arte. O capítulo de desenvolvimento compreendeu a análise que contém o levantamento de requisitos, juntamente com os diagramas de atividade, caso de uso e entidade relacionamento. Por fim a seção de implementação estão apresentados todos os detalhes sobre o desenvolvimento e funcionamento do protótipo. O catálogo de médicos, oferece aos usuários a praticidade e confiança na hora de marcar uma consulta médica, poupando tempo dos utilizadores, bem como aos médicos auxilia na gestão de suas agendas e consultórios. A aplicação proposta permite ainda que os pacientes avaliem os atendimentos visando a criação de um ranqueamento dos melhores profissionais do município.

**Palavras-Chave:** catálogo de médicos, desenvolvimento web, sistemas de informação.

## **ABSTRACT**

The health sector has been highlighted as one of the most important areas for the well-being of society, thus emerging the need to create new technologies to help the entire population. This work aims to develop a prototype of a responsive web application to provide a medical catalog that allows the scheduling of appointments for the general population. As for the methodology, it is characterized as an applied and descriptive research. To achieve the objectives, a literature review was carried out on the technologies and programming languages used in the development. A search for applications like the proposed prototype was also carried out, aiming to build the state-of-the-art section. The development chapter comprised the analysis that contains the requirements gathering, along with the activity, use case and entity relationship diagrams. Finally, in the implementation section, all the details about the development and functioning of the prototype are presented. The medical catalog offers users the convenience and confidence when making a medical appointment, saving users time, as well as helping physicians in managing their schedules and offices. The proposed application also allows patients to evaluate the assistance aimed at creating a ranking of the best professionals in the city.

**Keywords:** doctors catalog, web development, information systems.

## LISTA DE FIGURAS

Figura 1 - Regra para definição de seletores em CSS .....	20
Figura 2 - BoaConsulta para Android.....	30
Figura 3 - MeuMedico versão Web.....	31
Figura 4 - Diagrama de Caso de Uso.....	39
Figura 5 - Diagrama de Atividade do Agendamento de Consulta.....	41
Figura 6 - Diagrama de Modelo Entidade Relacionamento .....	42
Figura 7 – Tela de Apresentação (RF18).....	45
Figura 8 – Login no Sistema (RF01).....	46
Figura 9 – Logout no Sistema (RF02).....	47
Figura 10 - Cadastro de Usuário (RF03) .....	47
Figura 11 – Menu Principal (RF04) .....	48
Figura 12 – Cadastro de Contato do Usuário (RF14).....	49
Figura 13 - Catálogo de Médicos (RF05).....	50
Figura 14 – Cadastro de Consultas (RF11) .....	51
Figura 15 - Tela de Consultas (RF11) .....	51
Figura 16 - Cadastro de Avaliações (RF12).....	52
Figura 17 - Consulta de Avaliações (RF12).....	53
Figura 18 – Cadastro de Médicos x Consultórios (RF09).....	54
Figura 19 - Consulta de Equipe Administrativa (RF19) .....	54
Figura 20 – Cadastro de Equipe Administrativa (RF19).....	55
Figura 21 - Consulta do Cadastro de Agenda (RF10) .....	56
Figura 22 – Cadastro de Agenda (RF10).....	56
Figura 23 - Consulta de Itens da Agenda (RF20).....	57
Figura 24 – Efetivação de Consultas (RF13) .....	58
Figura 25 - Consulta de Estados (RF06) .....	59

Figura 26 - Consulta de Cidades (RF07).....	59
Figura 27 – Cadastro de Endereço (RF21) .....	60
Figura 28 - Consulta de Especialidade (RF15).....	61
Figura 29 – Cadastro de Médico x Especialidade (RF22).....	61
Figura 30 - Consulta de Calendário (RF16) .....	62
Figura 31 – Cadastro de Item do Calendário (RF23) .....	63
Figura 32 - Consulta de Consultórios (RF08) .....	63
Figura 33 – Cadastro de Horários do Consultório (RF17) .....	64
Figura 34 - Consulta de Contatos do Consultório.....	65

## LISTA DE QUADROS

Quadro 1 - Características essenciais de qualidade de software .....	15
Quadro 2 - Características do Banco de Dados .....	17
Quadro 3 - Utilização de APIs no HTML .....	19
Quadro 4 - Nomes e Versões do Javascript.....	22
Quadro 5 - Vantagens no uso do atributo src .....	24
Quadro 6 - Vantagens da utilização do PHP .....	25
Quadro 7 - Princípios da LGPD .....	28
Quadro 8 - Recursos nos sistemas avaliados .....	33
Quadro 9 - Requisitos Funcionais .....	33
Quadro 10 - Requisitos Funcionais Opcionais .....	36
Quadro 11 - Requisitos não funcionais .....	37
Quadro 12 - Regras de Negócio .....	37

## SUMÁRIO

<b>1. INTRODUÇÃO</b> .....	<b>10</b>
1.1 PROBLEMAS DE PESQUISA.....	11
1.2 OBJETIVOS.....	11
<b>1.2.1 Geral</b> .....	<b>11</b>
<b>1.2.2 Específicos</b> .....	<b>11</b>
1.3 JUSTIFICATIVA .....	12
<b>2. REVISÃO DA LITERATURA</b> .....	<b>13</b>
2.1 ENGENHARIA DE SOFTWARE .....	13
<b>2.1.1 Requisitos Funcionais e Não Funcionais</b> .....	<b>13</b>
<b>2.1.2 Qualidade de Software</b> .....	<b>15</b>
2.2 BANCO DE DADOS .....	17
<b>2.2.1 PostgreSQL</b> .....	<b>18</b>
2.3 HTML.....	19
2.4 CSS .....	20
<b>2.4.1 Bootstrap</b> .....	<b>21</b>
2.5 JAVASCRIPT .....	22
<b>2.5.1 JQuery</b> .....	<b>24</b>
2.6 PHP.....	25
<b>2.6.1 Laravel</b> .....	<b>26</b>
2.7 SEGURANÇA DE DADOS.....	27
<b>2.7.1 LGPD</b> .....	<b>27</b>
<b>3. METODOLOGIA DA PESQUISA</b> .....	<b>29</b>
3.1 ESTADO DA ARTE .....	29
<b>3.1.1 BoaConsulta</b> .....	<b>29</b>
<b>3.1.2 MeuMedico</b> .....	<b>30</b>
<b>4. CATÁLOGO MÉDICO: DESENVOLVIMENTO DE PROTÓTIPO DE APLICAÇÃO WEB RESPONSIVA</b> .....	<b>32</b>
4.1 ANÁLISE .....	32
<b>4.1.1 Visão Geral do Sistema</b> .....	<b>32</b>
<b>4.1.2 Comparação do Protótipo com o Estado da Arte</b> .....	<b>33</b>
<b>4.1.3 Requisitos</b> .....	<b>33</b>
<b>4.1.4 Diagramas</b> .....	<b>38</b>
4.2 IMPLEMENTAÇÃO .....	43
<b>4.2.1 Técnicas e Ferramentas Utilizadas</b> .....	<b>43</b>
<b>4.2.2 Utilização e Funcionamento</b> .....	<b>44</b>

<b>4.2.2.1 Aplicação Web Cliente .....</b>	<b>49</b>
<b>4.2.2.2 Aplicação Web Administrativa.....</b>	<b>53</b>
<b>4.2.2.3 Aplicação Web Médico.....</b>	<b>58</b>
<b>5. CONSIDERAÇÕES FINAIS.....</b>	<b>66</b>
<b>5.1 RECOMENDAÇÕES DE TRABALHOS FUTUROS .....</b>	<b>67</b>

## 1. INTRODUÇÃO

O século XXI tem demonstrado que área da saúde se tornou uma das áreas mais importantes para o bem-estar da sociedade brasileira, ainda mais a partir de 2020 quando a pandemia de COVID-19 se instaurou no país e no mundo. Com isso a carência de processos eficientes e pouco burocráticos se tornou evidente perante a sociedade, surgindo a necessidade da criação e adoção de novas tecnologias para auxílio de toda a população.

Saúde e educação são as duas principais políticas de um país. São elas que podem produzir um mínimo de igualdade num país tão desigual. Prover saúde de qualidade significa garantir e respeitar a dignidade de cada cidadão que integra e constrói a sociedade brasileira. (OLIVEIRA, 2018).

Com o avanço da tecnologia as tarefas do dia a dia se tornaram mais práticas para as pessoas, o que não pode ser diferente para o setor da saúde. Vários artifícios para facilitar a vida do profissional da saúde já foram criados, para assim otimizar ao máximo os procedimentos burocráticos relacionados ao setor.

A proposta da criação da aplicação web tem o objetivo de centralizar os médicos que atendem em um determinado município. O público-alvo desta ferramenta é a população em geral, sendo que estes poderão ter acesso a uma lista completa dos profissionais de sua cidade separados pelas determinadas especialidades, assim após escolher o profissional desejado, o sistema disponibilizará recurso para realizar o agendamento de uma consulta em uma data que melhor se encaixa na disponibilidade do paciente e do médico. Por fim, será possível interagir com a ferramenta por meio de uma avaliação, que será fundamental para um ranqueamento mais eficiente dos profissionais listados.

As tecnologias da informação representam uma alternativa promissora para a redução dos custos, para a ampliação do acesso e para a melhoria dos serviços de saúde, por sua vez trazendo mais conforto e comodidade para a população. (NEGRI, 2019).

Neste contexto, com o uso da aplicação tem-se o intuito de poupar tempo das pessoas, pois com uma busca eficiente e as disponibilidades de cada profissional sendo exibidas em tempo real, o processo se torna menos oneroso e com uma probabilidade de ocorrência de erros menor do que as tradicionais ligações praticadas atualmente. Também, com o uso da aplicação será possível verificar a qualidade do atendimento de cada profissional com base nas avaliações descritas por pessoas que fizeram alguma consulta anteriormente.

Além deste capítulo introdutório, este trabalho possui uma revisão da literatura onde são apresentadas todas as tecnologias usadas no desenvolvimento do protótipo com a apresentação de um conceito sobre cada uma delas. No segundo capítulo temos a metodologia da pesquisa que explica em detalhes o processo metodológico deste trabalho, juntamente com a seção do estado da arte que descreve ferramentas com propósitos semelhantes ao protótipo apresentado. Já no quarto capítulo é apresentado o desenvolvimento da aplicação junto com a análise do projeto, com o levantamento dos requisitos e os diagramas necessários para facilitar a construção do protótipo, também a seção de implementação onde são demonstradas todas as rotinas da aplicação, e são detalhadas as tecnologias utilizadas durante o desenvolvimento. No último capítulo, são listadas as considerações finais e recomendações para trabalhos futuros.

## 1.1 PROBLEMAS DE PESQUISA

Como melhorar a busca por um médico especialista?

Como mensurar o nível de satisfação e avaliar a consulta médica?

Como otimizar o tempo gasto até achar o profissional mais adequado?

## 1.2 OBJETIVOS

### 1.2.1 Geral

- Desenvolver um protótipo de aplicação web responsiva visando disponibilizar um catálogo médico, que permita o agendamento de consultas para a população em geral.

### 1.2.2 Específicos

- Detalhar as tecnologias que serão utilizadas para a construção do protótipo;
- Identificar ferramentas já existentes no mercado que possuem uma finalidade semelhante e explicar seu funcionamento;
- Descrever os requisitos e elaborar os diagramas visando a construção da análise do protótipo;
- Aplicar as tecnologias selecionadas para o desenvolvimento do protótipo.

### 1.3 JUSTIFICATIVA

Historicamente, procurar médicos tem sido uma tarefa difícil para a sociedade, pois deve ser olhado sempre em consultórios únicos, ou seja, encontrar um médico de agrado para o paciente é necessário fazer uma longa busca até achar um profissional. A parte de avaliações também é feita de maneira informal, ou seja, a própria sociedade passa a informação de uma pessoa para a outra.

Por essa problemática, foi proposta a criação de uma aplicação para centralização de médicos de uma cidade, cada profissional vai poder realizar seu cadastro na aplicação, sendo separados por especialização. Assim os clientes poderão ter acesso a lista de profissionais de sua cidade, sendo que será ranqueada conforme as avaliações de consultas realizadas por outros usuários. Também será possível o usuário escolher uma data que melhor se enquadra a sua necessidade, uma vez que será disponibilizado para escolha os horários disponíveis de cada profissional.

Dessa forma com o uso da aplicação, pretende-se que o usuário consiga poupar tempo de pesquisa, pois os dados já estarão dispostos de forma fácil e poderá conseguir uma consulta mais rápida de acordo com sua necessidade de urgência, podendo sempre escolher um profissional pela sua respectiva avaliação, o que justifica a importância da realização do presente projeto.

## 2. REVISÃO DA LITERATURA

Neste capítulo serão abordados assuntos referentes as ferramentas de desenvolvimento e modelagem, bem como as tecnologias utilizadas durante todo o processo de construção do protótipo.

### 2.1 ENGENHARIA DE SOFTWARE

Pressman (1995, p. 231) afirma que “Não importa quão bem projetado ou quão bem codificado seja, um programa mal analisado e especificado desapontará o usuário e trará aborrecimentos ao desenvolvedor.”.

A importância dos métodos de desenvolvimento de software advém do fato de que eles definem, por meio de suas notações, um canal de comunicação uniforme entre os membros da equipe de desenvolvimento. Também, os métodos estabelecem produtos de trabalho padronizados que facilitam as atividades de manutenção de software. Além disso, permitem que novos colaboradores sejam treinados melhorando a qualidade do software. Os métodos podem ser usados como referência para a escolha e aquisição de ferramentas CASE que podem dar maior produtividade ao processo de desenvolvimento de software. (HIRAMA, 2011, p. 11).

Segundo Hirama (2011), a atividade de Engenharias de Sistemas tem como um de seus objetivos entender as necessidades de negócio de um cliente, a partir disso são estabelecidos requisitos de um ponto de vista sistêmico, até ser possível definir em alto nível o que será desenvolvido. Uma das tarefas é a identificação de funções, restrições e desempenho do sistema, uma vez que o objetivo é executar um método, procedimento ou realizar o processamento de dados.

#### 2.1.1 Requisitos Funcionais e Não Funcionais

Segundo Pressman (1995), a análise de requisitos é uma tarefa que efetua a ligação de software em nível de sistema e projeto, ela possibilita que seja especificado a função e o desempenho do *software*, juntamente com as restrições do sistema. O analista constrói modelos de processo dos dados e domínios comportamentais do *software*. Essa análise de requisitos proporcionará uma representação da informação que pode ser traduzida em projeto procedimental, arquitetônico e de dados.

“A análise de requisitos de *software* pode ser dividida em cinco áreas de esforço: (1) reconhecimento do problema, (2) avaliação e síntese, (3) modelagem, (4) especificação e (5) revisão.”. (PRESSMAN, 1995, p. 232).

O amplo espectro de tarefas e técnicas que levam a um entendimento dos requisitos é chamado de engenharia de requisitos. Do ponto de vista do processo de software, a engenharia de requisitos é uma ação de engenharia de software importante que se inicia durante a atividade de comunicação e continua na de modelagem. Ela deve ser adaptada às necessidades do processo, do projeto, do produto e das pessoas que estão realizando o trabalho. (PRESSMAN, 2016, p.132).

Pressman (2016, p.133) afirma que “A engenharia de requisitos abrange sete tarefas distintas: concepção, levantamento, elaboração, negociação, especificação, validação e gestão. É importante notar que algumas delas ocorrem em paralelo e que todas são adaptadas às necessidades do projeto.”.

Processos de software costumam estipular um ou mais tipos de cadastro dos requisitos. Esse é o artefato que registra os diversos tipos de requisitos do produto, ajudando a determinar o impacto de alterações de requisitos, e servindo de insumo para cálculo do tamanho funcional do produto. Os requisitos primários, levantados diretamente a partir das necessidades das partes interessadas, abrangem requisitos de estrutura (requisitos lógicos de dados, assim como usuários e sistemas externos) e de comportamento (regras de negócio, requisitos funcionais e requisitos não funcionais). (PÁDUA, 2019, p. 125).

“Um requisito não funcional (NFR, *nonfunctional requirement*) pode ser descrito como um atributo de qualidade, de desempenho, de segurança ou como uma restrição geral em um sistema. Frequentemente, os envolvidos têm dificuldade de articulá-los.”. (PRESSMAN, 2016, p.141).

O gerenciamento de requisitos é essencialmente um processo para gerenciar as mudanças dos requisitos do sistema. As mudanças devem ser gerenciadas para assegurar que elas atendam a questões econômicas e contribuam para as necessidades de negócio da organização que adquire o sistema. A viabilidade das mudanças deve ser avaliada dentro do prazo e do orçamento do projeto. (HIRAMA, 2011, p.55).

“Uma vez definidos os requisitos do software, eles são desenvolvidos ao longo do ciclo de desenvolvimento. Embora, estes requisitos tenham sido satisfatoriamente entendidos antes de iniciar a sua implementação, o fato é que eles irão mudar em algum momento do desenvolvimento.” (HIRAMA, 2011, p.55).

De acordo com Pressman (2016), é possível definir duas fases para ajudar na identificação de requisitos não funcionais. A primeira fase é levantada um conjunto de diretrizes para o sistema que será construído, algumas dessas diretrizes podem ser de estilo arquitetural e do uso de padrões de projeto. Já na segunda fase é criado um conjunto de requisitos não

funcionais utilizando um conjunto de regras que estabelecerão quais diretrizes serão implementadas ou rejeitadas.

### 2.1.2 Qualidade de Software

Pressman (2016, p. 414) afirma que “No desenvolvimento de software, a qualidade de um projeto engloba o grau de atendimento às funções e características especificadas no modelo de requisitos.”.

A necessidade de maior qualidade de software surgiu realmente a partir do momento que o software ficou cada vez mais integrado a todas as atividades de nossas vidas. Na década de 1990, as principais empresas reconheciam que bilhões de dólares por ano eram desperdiçados em software que não apresentava as características e as funcionalidades prometidas. Pior ainda, tanto o governo quanto as empresas estavam cada vez mais preocupados com o fato de que uma falha grave de software poderia inutilizar importantes infraestruturas, aumentando o custo em dezenas de bilhões. (PRESSMAN, 2016, p. 412).

“No sentido mais geral, a qualidade de *software* pode ser definida como: uma gestão de qualidade efetiva aplicada de modo a criar um produto útil que forneça valor mensurável para aqueles que o produzem e para aqueles que o utilizam.”. (PRESSMAN, 2016, p. 414).

De acordo com Pressman (2016) existem algumas características principais do processo de qualidade de software, sendo essa estrutura formada pelo Quadro 1.

**Quadro 1 - Características essenciais de qualidade de software**

<b>Gestão de Qualidade</b>	Estabelece a infraestrutura que dá suporte a qualquer tentativa de construir um produto de software de alta qualidade. Os aspectos administrativos do processo criam mecanismos de controle e equilíbrio de poderes que ajudam a evitar o caos no projeto – um fator-chave para uma qualidade inadequada. As práticas de engenharia de software permitem ao desenvolvedor analisar o problema e elaborar uma solução - consistentes aspectos críticos na construção de software de alta qualidade. Finalmente, as atividades de apoio, como o gerenciamento de mudanças e as revisões técnicas, têm muito a ver com a qualidade, assim como qualquer outra parte da prática de engenharia de software.
<b>Produto Útil</b>	Fornecer o conteúdo, as funções e os recursos que o usuário deseja; além disso, e não menos importante, deve fornecer confiabilidade e isenção de erros. Um produto útil sempre satisfaz às exigências definidas explicitamente pelos envolvidos. Além disso, ele satisfaz a um conjunto de requisitos implícitos (por exemplo, facilidade de uso) que se espera de todo software de alta qualidade.
<b>Agregar Valor</b>	Um software de alta qualidade gera benefícios para a empresa de software e para a comunidade de usuários. A empresa fabricante do software ganha valor agregado pelo fato de um software de alta qualidade exigir menos manutenção, menos correções de erros e menos suporte ao cliente. Isso permite que os engenheiros de software despendam mais tempo criando aplicações novas e menos tempo em manutenções. A comunidade de usuários ganha um valor agregado, pois a aplicação fornece a capacidade

	de agilizar algum processo de negócio. O resultado final é: (1) maior receita gerada pelo produto de software, (2) maior rentabilidade quando uma aplicação suporta um processo de negócio e/ou (3) maior disponibilidade de informações cruciais para o negócio.
--	---

Fonte: Adaptado de Pressman (2016).

“A qualidade de *software* não aparece simplesmente do nada. Ela é o resultado de um bom gerenciamento de projeto e uma prática consistente de engenharia de *software*.” (PRESSMAN, 2016, p. 427).

O custo da qualidade inclui todos os custos necessários para a busca de qualidade ou para a execução de atividades relacionadas à qualidade, assim como os custos causados pela falta de qualidade. Para compreender esses custos, uma organização deve reunir métricas para prover uma base para o custo corrente da qualidade, identificar oportunidades para reduzir esses custos e fornecer uma base normalizada de comparação. O custo da qualidade pode ser dividido em custos associados à prevenção, avaliação e falhas. (PRESSMAN, 2016, p. 422).

De acordo com Pressman (2016), a qualidade e segurança de um *software* se define à medida que o caráter crítico de uma aplicação que esteja na internet aumente, ou seja quando a aplicação vai se tornando mais importante. Por consequência, um *software* que não possui uma alta qualidade é mais fácil de ser copiado, também com um software de baixa qualidade pode aumentar de maneira indireta os riscos de segurança, assim como os problemas e custos associados.

À medida que o caráter crítico das aplicações e dos sistemas móveis baseados na Internet aumenta, a segurança da aplicação tem se tornado cada vez mais importante. Ou seja, software que não apresente alta qualidade é mais fácil de ser copiado (“pirateado”), e, como consequência, o software de baixa qualidade pode aumentar indiretamente o risco de segurança, assim como todos os problemas e custos associados. (PRESSMAN, 2016, p. 425).

Para construirmos um *software* de alta qualidade, precisamos entender integralmente o problema a ser resolvido. Também deve ser criado um projeto que seja adequado ao problema, apresentando as características que levem o *software* a possuir as dimensões e fatores de qualidade. (PRESSMAN, 2016).

Para Hirama (2011), a qualidade de *software* é algo intangível que se encontra presente em um produto, por outro ponto de vista, a qualidade é algo que precisa ser conquistado e não pode ser simplesmente obtido, portanto havendo um esforço a ser dispendido. O *software* é um produto resultante da necessidade de um cliente, não possuindo um espaço físico, apenas lógico em um computador e embute uma série de tecnologias, o *software* como um produto também tem a necessidade de ter qualidade.

## 2.2 BANCO DE DADOS

Segundo Alves (2014a, p.17) “Um banco de dados é um conjunto lógico e ordenado de dados que possuem algum significado, e não uma coleção aleatória sem um fim ou objetivo específico.”.

Se um banco de dados é um conjunto de dados relacionados, um Sistema de Gerenciamento de Banco de Dados (SGBD) é uma coleção de ferramentas e programas que permitem aos usuários a criação e manutenção do próprio banco de dados. Desta forma o SGBD pode ser considerado como um sofisticado software destinado à definição, construção e manipulação. (ALVES, 2014a, p.17).

“Para se ter um banco de dados, são necessários três ingredientes: uma fonte de informação, da qual os dados são derivados, uma interação com o mundo real e um público que demonstra interesse nos dados contidos no banco.” (ALVES, 2014a, p.17).

Da mesma forma que na matemática, podem ser efetuadas operações entre dois ou mais conjuntos, como, por exemplo, obter os elementos que são comuns a ambos os conjuntos (tabelas/relações) num banco de dados relacional. Também é possível executar certas operações com essas tabelas, como ligar duas ou mais por meio de campos comuns em ambas. Quando uma operação de consulta é executada, o resultado é um conjunto de registros que pode ser tratado como uma tabela virtual (que só existe enquanto a consulta está ativa). Isso significa que não há comandos para efetuar uma navegação pelos registros, do tipo MOVE ou SKIP, que permitem posicionar o ponteiro de registro no próximo registro da tabela. (ALVES, 2014a, p.20).

De acordo com Neves (2002) existem algumas características principais do banco de dados, sendo essa estrutura formada pelo Quadro 2.

**Quadro 2 - Características do Banco de Dados**

<b>Sistema Gerenciador de Banco de Dados (SGDB)</b>	O SGDB permite que bancos de dados “persistentes” sejam concorrentemente partilhados por muitos usuários e aplicações utilizando manuseio de armazenamento e estratégias de otimização. Precisamos manter os dados para executar as tarefas de leitura a partir de banco de dados, preservando a integridade dos dados.
<b>Banco de Dados Relacional</b>	O banco de dados relacional é sustentado pelo modelo relacional, que é matematicamente perfeito, completo e consistente internamente. O banco de dados relacional é muito mais utilizado comercialmente e pode ser usado para resolver muitos problemas.
<b>Banco de Dados Orientado a Objetos (OODBs)</b>	Integra a orientação a objeto com aptidões de banco de dados, oferecendo modelos diretivos e intuitivos para o desenvolvimento de aplicações.

Fonte: Adaptado de Neves (2002).

Segundo Alves (2014a), a maioria dos sistemas de gerenciamento de banco de dados utilizados hoje em dia, são do tipo relacional, este se caracteriza pelo fato da organização dos dados em tabelas, possuindo suas linhas e colunas, assim essas tabelas são semelhantes a conjuntos de elementos, ou objetos, da maneira que relacionam as informações referentes a um assunto em comum de modo organizado.

Atualmente há dois modelos muito utilizados: modelo de dados relacional e modelo de dados orientado a objetos. Com os avanços tecnológicos ocorridos na área de banco de dados, os modelos relacionais têm incorporado ultimamente conceitos que originalmente foram desenvolvidos para uso com modelos de dados de objetos, o que levou à criação de outra categoria denominada banco de dados objeto-relacional. (ALVES, 2014a, p.20).

De acordo com Alves (2014a), o banco de dados orientado a objetos, surgiu da necessidade do armazenamento de dados onde não era possível a utilização de um banco de dados relacional, devido a sua limitação. Basicamente este sistema é caracterizado pela definição dos dados por meio de objetos, com suas propriedades e operações, assim como é feito em algumas linguagens de programação.

“Um banco de dados, principalmente os que suportam vários usuários, precisa ser capaz de gerenciar múltiplas visões dos dados. Essas visões são subconjuntos de dados temporários, que existem somente durante uma determinada operação, como um comando de consulta SQL.”. (ALVES, 2014a, p.19).

Segundo Alves (2014a), existem duas possibilidades quanto a localização do SGBD, o centralizado e o distribuído. No sistema gerenciador de banco de dados centralizado o sistema está localizado em uma única máquina, denominada servidor de banco de dados, porém mesmo centro centralizado ele pode ter suporte a acesso de vários usuários. O SGDB distribuído, é caracterizado pelo sistema estar localizado em múltiplas máquinas, sendo todos interligados em rede. Independente do tipo utilizado, o SGDB trabalha dentro de uma arquitetura cliente/servidor, composta pelo servidor de banco de dados e estação de trabalho do cliente.

### **2.2.1 PostgreSQL**

Neves (2002, p.18) afirma que “O PostgreSQL ou apenas seu nome original Postgres é um Sistema Gerenciador de banco de dados relacional e orientado a objetos.”.

De acordo com Neves (2002), o PostgreSQL é um dos melhores bancos de dados para o sistema operacional GNULINUX, voltado para aplicações simples ou complexas, possui uma robustez excepcional e um suporte a várias linguagens. Algumas das características deste banco

de dados, é não possuir custo de utilização, possuir o processo de transações, segurança contra falhas, chaves primárias e estrangeiras e a confirmação ou cancelamento de operações realizadas.

### 2.3 HTML

Segundo Alves (2014b, p.26), “A linguagem HTML (sigla de *HyperText Markup Language* - Linguagem de Marcação de Hipertexto) é responsável pela criação de documentos para a web, comumente conhecidos como páginas.”

Por meio da linguagem HTML, o desenvolvedor pode especificar atributos para um texto (como fonte, tamanho, cor etc.) ou criar marcações que definem um vínculo com outro documento, que deve ser apresentado quando o usuário clicar na marcação. Essa é a ideia de hipertexto: vínculos entre diversos documentos criados a partir de marcações específicas. (ALVES, 2014b p. 26).

De acordo com Alves (2014b), existem algumas características de destaque no HTML, como por exemplo a independência da plataforma utilizada, ou seja, qualquer tipo de computador pode ler e interpretar o conteúdo de uma página escrita em HTML. Os arquivos possuem um tamanho pequeno, sendo vinculados entre si pelos *hiperlinks* definidos na estrutura de documentos, como o tamanho dos arquivos são reduzidos é possível fazer a transmissão deles com uma velocidade aceitável, mesmo em meios de comunicação com baixa velocidade. Por fim a linguagem não necessita de um editor especial, podendo ser escrita por qualquer editor de texto disponível.

De acordo com Teruel (2014), o HTML permite a utilização de uma série de APIs, sendo que algumas delas estão formadas pelo Quadro 3.

**Quadro 3 - Utilização de APIs no HTML**

<b>WebFoms 2.0</b>	Disponibiliza mais tipos de entrada de dados ( <i>button, checkbox, color, date, datetime, datetime-local, email, file, hidden, image, month, number, password, radio, range, reset, search, submit, tel, text, time, url e week</i> ) em formulários, validação de entrada mais simples (por meio do atributo <i>pattern</i> no elemento <code>&lt;input&gt;</code> ) e menos scripts na página.
<b>Web SQL Database</b>	Conjunto de APIs para manipular bancos de dados do lado do cliente usando SQL via JavaScript.
<b>Web Storage</b>	Permite manipular o histórico de navegação e dados no computador do cliente utilizando uma forma mais eficiente do que os antigos cookies. Armazena até 5MB de dados sem expiração definida e sem os complexos cálculos de data dos cookies.
<b>Web Sockets</b>	API que permite que páginas web usem o protocolo <i>Web Socket</i> para a comunicação de duas vias com um servidor. Essa API permite que os aplicativos web mantenham comunicação bidirecional assíncrona com o servidor.

<b>Web Workers</b>	Utilizada para a execução de scripts em segundo plano (em paralelo), independentemente de qualquer script em execução na <i>interface</i> do usuário. Isso permite que scripts de execução demorada não sejam interrompidos por scripts que respondem a cliques ou outras interações com o usuário. As tarefas longas são executadas sem atrapalhar as transações de requisição/resposta.
<b>Geolocation</b>	Define uma interface de alto nível para as informações de localização associada apenas com o dispositivo que hospeda a aplicação, como latitude e longitude. A API <i>Geolocation</i> permite por meio de JavaScript a integração e consumo de serviços de geolocalização no globo, como, por exemplo, serviços Google Maps.
<b>Vídeo &amp; Áudio</b>	Executa conteúdo multimídia por meio dos elementos <audio>, <video> e chamadas a métodos de controle via JavaScript.
<b>Canvas e SVG</b>	Possibilita desenhar na tela do navegador utilizando o elemento <canvas> e JavaScript. A API SVG é vetorial, e baseada em XML, logo, acessível via DOM. A API Canvas desenha pixel a pixel via JavaScript.
<b>Aplicações Offline</b>	Executa aplicações web <i>offline</i> referenciadas em cache pelo atributo <i>manifest</i> do elemento <html>. Esse atributo aponta para um arquivo de manifesto, indicando as URLs que devem ser postas em cache para execução <i>offline</i> . Uma vez colocadas em cache, o usuário pode navegar pelas páginas em cache sem estar online.
<b>Drag &amp; Drop</b>	Em meio visual, permite operações para arrastar e soltar elementos. Uma operação de arrastar pode ser a ação padrão de um evento <i>MouseDown</i> que é seguido por uma série de eventos <i>MouseMove</i> , e a parada (queda) pode ser desencadeada quando o mouse é liberado. Para acessar as funções dessa API, utilizam-se scripts JavaScript.

Fonte: Adaptado de Teruel (2014).

“A força da HTML5 não está apenas nos novos elementos e atributos, mas também na possibilidade de acessá-los, manipulá-los e formatá-los com JavaScript e CSS utilizando o *Document Object Model* (DOM) [...]” (TERUEL, 2014 p. 19).

## 2.4 CSS

Segundo Alves (2014b, p.78), “As folhas de estilo, popularmente conhecidas pela sigla CSS, do inglês *Cascading Style Sheets*, fazem parte dos recursos que foram adicionados à linguagem HTML a fim de aprimorar o design das páginas web.”.

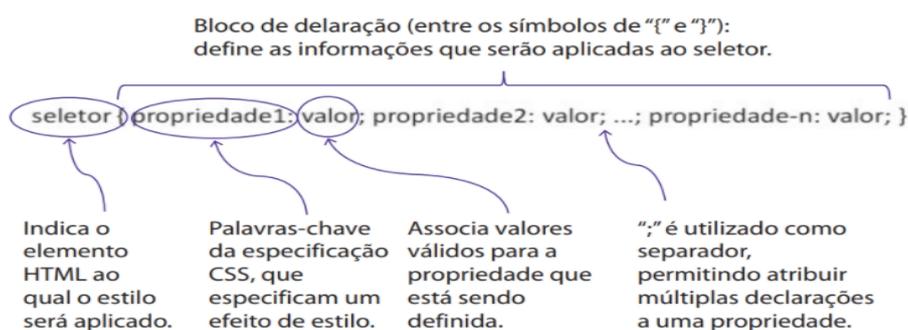
As folhas de estilo em cascata mudam a forma de organização das páginas. O HTML passa a ser utilizado somente como elemento para estruturar as páginas, e o CSS é utilizado na formatação da aparência das páginas. Com o CSS, é possível definir em um único local a formatação que será utilizada por cada TAG. Com isso, apenas um arquivo é alterado, sendo que a mudança é automaticamente propagada a todas as páginas que compõem o site. (MILLETO, 2014 p. 80).

De acordo com Alves (2014b), com o CSS é possível definir características que podem ser aplicadas a diversos arquivos HTML, com o objetivo de aprimorar o *design* das páginas. Alguns exemplos de implementação são alterar a cor padrão aplicada a um texto, sendo que

quando necessário fazer a alteração dessa cor novamente todas as páginas serão atualizadas automaticamente a partir do momento que estejam vinculados a folha de estilo.

De acordo com Milleteo (2014), o CSS possui uma regra para definição de seus seletores, sendo esta formada pela Figura 1.

**Figura 1 - Regra para definição de seletores em CSS**



Fonte: Milleteo (2014, p.81).

“As folhas de estilo possibilitam criar estilos personalizados para títulos, listas, imagens, etc., além de permitirem a definição de cores, fontes, bordas, alinhamentos, entre outras características vinculadas à aparência das páginas Web.” (MILLETO, 2014, p.80).

### 2.4.1 Bootstrap

O Bootstrap é uma biblioteca de código aberto que auxilia no desenvolvimento de componentes para estilização de uma aplicação web, sendo possível projetar e personalizar rapidamente sites responsivos para dispositivos móveis e computadores, consistindo, portanto, em um kit de ferramentas para o *front-end*. (BOOTSTRAP, 2021).

Uma das grandes vantagens da utilização da biblioteca é o sistema de *grid* responsivo que pode facilmente ser utilizado, juntamente com extensos componentes pré-construídos para estilização das telas. Uma outra vantagem é a grande quantidade de plug-ins de Javascript que podem auxiliar no processo de desenvolvimento (BOOTSTRAP, 2021).

## 2.5 JAVASCRIPT

Flanagan (2014, p. 18) afirma que “[...] JavaScript é uma linguagem de alto nível, dinâmica, interpretada e não tipada, conveniente para estilos de programação orientados a objetos e funcionais.”.

JavaScript é a linguagem de programação da Web. A ampla maioria dos sites modernos usa JavaScript e todos os navegadores modernos – em computadores de mesa, consoles de jogos, tablets e smartphones – incluem interpretadores JavaScript, tornando-a a linguagem de programação mais onipresente da história. JavaScript faz parte da tríade de tecnologias que todos os desenvolvedores Web devem conhecer: HTML, para especificar o conteúdo de páginas Web; CSS, para especificar a apresentação dessas páginas; e JavaScript, para especificar o comportamento delas. (FLANAGAN, 2014, P.18).

De acordo com Flanagan (2014) apresentam os nomes e versões que o JavaScript possuiu ao longo do tempo, sendo essa estrutura formada pelo Quadro 4.

**Quadro 4 - Nomes e Versões do Javascript**

<b>Criação</b>	JavaScript foi criada na Netscape na fase inicial da Web e, tecnicamente, “JavaScript” é marca registrada, licenciada pela Sun Microsystems (agora Oracle), usada para descrever a implementação da linguagem pelo Netscape (agora Mozilla). A Netscape enviou a linguagem para a ECMA – <i>European Computer Manufacturer’s Association</i> – para padronização e, devido a questões relacionadas à marca registrada, a versão padronizada manteve o nome estranho “ECMAScript”. Pelos mesmos motivos ligados à marca registrada, a versão da Microsoft da linguagem é formalmente conhecida como “JScript”. Na prática, quase todo mundo chama a linguagem de JavaScript. Este livro usa o nome “ECMAScript” apenas para se referir ao padrão da linguagem.
<b>Versões</b>	Na última década, todos os navegadores Web implementaram a versão 3 do padrão ECMAScript e não havia necessidade de se pensar em números de versão: o padrão da linguagem era estável e as implementações dos navegadores eram, na maioria, interoperáveis. Recentemente, uma importante nova versão da linguagem foi definida como ECMAScript versão 5. Às vezes, você vai ver essas versões da linguagem abreviadas como ES3 e ES5, assim como às vezes vai ver o nome JavaScript abreviado como JS.
<b>Interpretadores</b>	Quando falamos da linguagem em si, os únicos números de versão relevantes são ECMAScript versões 3 ou 5. (A versão 4 da ECMAScript esteve em desenvolvimento por anos, mas se mostrou ambiciosa demais e nunca foi lançada). Contudo, às vezes você também vai ver um número de versão de JavaScript, como JavaScript 1.5 ou JavaScript 1.8. Esses são números da versão do Mozilla: a versão 1.5 é basicamente a ECMAScript 3 e as versões posteriores incluem extensões não padronizadas da linguagem. Por fim, também existem números de versão vinculados a interpretadores ou “engines” de JavaScript específicos. O Google chama seu interpretador JavaScript de V8, por exemplo, e quando este livro estava sendo produzido a versão corrente do mecanismo V8 era a 3.0.

Fonte: Adaptado de Flanagan (2014).

“O uso de JavaScript em documentos Web normalmente deve ser controlado e moderado. O papel apropriado de JavaScript é melhorar a experiência de navegação do usuário, tornando mais fácil obter ou transmitir informações.” (FLANAGAN, 2014, p. 317).

É interessante pensarmos nas páginas que exibimos nos navegadores Web. Algumas delas apresentam informações estáticas e podem ser chamadas de documentos. (A apresentação dessas informações pode ser bastante dinâmica – por causa de JavaScript –, mas as informações em si são estáticas.) Outras páginas Web mais parecem aplicativos do que documentos. Essas páginas podem carregar novas informações dinamicamente, conforme a necessidade, podem ser gráficas em vez de textuais e podem funcionar *off-line* e salvar dados de forma local, para que possam restaurar seus estados quando você visitá-las novamente. Ainda outras páginas Web ficam em algum ponto no meio desse espectro e combinam recursos de documentos e de aplicativos. (FLANAGAN, 2014, P. 314).

De acordo com Flanagan (2014), a experiência do usuário não depende totalmente do Javascript, porém ela facilitará o processo, como criando animações e efeitos visuais que guiam o usuário para ajudar na navegação da página, ordenando colunas em uma tabela, tornando mais fácil para o usuário a disposição dos dados que mais necessita e ocultar algum conteúdo, revelando-o gradativamente à medida que o usuário se aprofunda no determinado conteúdo.

Para realmente entender os aplicativos Web, é importante perceber que os navegadores Web foram muito além de sua função original como ferramentas para exibir documentos e se transformaram em sistemas operacionais simples. Considere o seguinte: um sistema operacional tradicional permite organizar ícones (que representam arquivos e aplicativos) na área de trabalho e em pastas. Um navegador Web permite organizar *bookmarks* (que representam documentos e aplicativos Web) em uma barra de ferramentas e em pastas. Um sistema operacional executa vários aplicativos em janelas separadas; um navegador Web exibe vários documentos (ou aplicativos) em guias (ou abas) separadas. Um sistema operacional define APIs de baixo nível para conexão em rede, desenho de elementos gráficos e salvamento de arquivos. Os navegadores Web definem APIs de baixo nível para conexão em rede, salvamento de dados e desenho de elementos gráficos. (FLANAGAN, 2014, P. 317).

Segundo Flanagan (2014, p. 320) “Um arquivo JavaScript contém JavaScript puro, sem marcações `<script>` ou qualquer outro código HTML. Por convenção, os arquivos de código JavaScript têm nomes que terminam com `.js`.”

Uma marcação `<script>` com o atributo `src` especificado se comporta exatamente como se o conteúdo do arquivo JavaScript especificado aparecesse diretamente entre as marcações `<script>` e `</script>`. Note que a marcação de fechamento `</script>` é obrigatória em documentos HTML, mesmo quando o atributo `src` é especificado, e que não há qualquer conteúdo entre as marcações `<script>` e `</script>`. Em XHTML, pode-se usar a marcação de atalho `<script/>` nesse caso. (FLANAGAN, 2014, P. 320).

De acordo com Flanagan (2014), existem vantagens no uso do atributo *src*, sendo estas vantagens formadas pelo Quadro 5.

**Quadro 5 - Vantagens no uso do atributo *src***

<b>Arquivos HTML</b>	Ele simplifica seus arquivos HTML, permitindo remover deles grandes blocos de código JavaScript – isto é, ajuda a manter conteúdo e comportamento separados.
<b>Páginas Web</b>	Quando várias páginas Web compartilham o mesmo código JavaScript, o uso do atributo <i>src</i> permite que você mantenha apenas uma cópia desse código, em vez de ter de editar cada arquivo HTML quando o código mudar.
<b>Compartilhamento</b>	Se um arquivo de código JavaScript é compartilhado por mais de uma página, ele só precisa ser baixado uma vez, pela primeira página que o utilizar – as páginas subsequentes podem recuperá-lo da cache do navegador.
<b>URL</b>	Como o atributo <i>src</i> recebe um URL arbitrário como valor, um programa JavaScript ou uma página Web de um servidor pode empregar o código exportado por outros servidores Web. Muitos anúncios na Internet contam com isso.
<b>Carregamento de scripts</b>	A capacidade de carregar scripts de outros sites nos permite levar as vantagens do uso da cache um passo adiante: o Google está promovendo o uso de URLs padrão bem conhecidos para as bibliotecas do lado do cliente mais comumente usadas, permitindo que o navegador coloque no cache uma única cópia para uso compartilhado por qualquer site. Vincular código JavaScript aos servidores do Google pode diminuir o tempo de inicialização de suas páginas Web, pois é provável que a biblioteca já exista na cache do navegador do usuário; porém, você precisa estar disposto a confiar em terceiros para fornecer código fundamental para seu site.

Fonte: Adaptado de Flanagan (2014).

“Evidentemente, JavaScript é mais pertinente a aplicativos Web do que a documentos Web. Ela aprimora documentos Web, mas um documento bem projetado vai continuar a funcionar mesmo com JavaScript desativada.” (FLANAGAN, 2014, p. 318).

### 2.5.1 JQuery

De acordo com Flanagan (2014), o jQuery torna fácil encontrar os elementos importantes de um documento e então manipulá-los de maneira mais simples, editando atributos HTML e seletores CSS, definindo tratamentos de evento e animações. A biblioteca também tem utilitários como por exemplo fazer requisições HTTP em Ajax, juntamente com a possibilidade de trabalhar com objetos e *arrays*.

Numa tradução livre, de acordo com o próprio site da ferramenta, jQuery é uma biblioteca JavaScript que simplifica a manipulação de documentos HTML, eventos, animações e interações com AJAX para desenvolvimento rápido de aplicações web. (BALDUINO, 2012).

De acordo com Flanagan (2014), existem algumas características que são base para o poder e utilidade da biblioteca jQuery, dentre elas estão a sintaxe expressiva para se referir aos elementos do documento, possui um método de consulta eficiente para localizar elementos no documento que correspondem a seletores CSS, um conjunto amplo de métodos para manipulação dos elementos selecionados, técnicas de programação funcional que servem para operar em conjuntos de elementos em grupo e também um encadeamento de métodos sucinto para expressar a sequência das operações.

## 2.6 PHP

“O PHP é uma linguagem criada por Ramus Ledorf, em 1995, que explora a criação de scripts que são normalmente interpretados em um servidor Web no qual esses scripts estejam armazenados.” (MILETTO, 2014, p.172).

Um dos destaques do PHP está na sua apresentação em conjunto com as marcações de linguagem HTML, o que possibilita a adição de dinamicidade às páginas desenvolvidas nessa linguagem. Para a identificação, pelo servidor Web, dos trechos que devem ser interpretados como scripts em PHP, é preciso usar delimitadores (TAGs) iniciais (<?php é o mais utilizado) e finais (?>) nos trechos que envolvem esta linguagem, diferenciando-a, por exemplo, do HTML ou do JavaScript. (MILETTO, 2014, p.172).

Welling e Thomsom (2005, p. 27) afirmam que “O PHP é um produto de código-fonte aberto, o que significa que você tem acesso ao seu código-fonte. É possível utilizá-lo, alterá-lo e redistribuí-lo sem pagar nada.”.

Quando um determinado cliente, através de seu navegador Web, solicita o acesso ao script (página Web) exemplo.php, o servidor procurará pelo arquivo e passará a interpretar o código apresentado, processando os trechos dentro das TAGs de identificação do PHP. Após esse processamento, o servidor enviará ao cliente (navegador do usuário) o resultado em conjunto com os demais códigos em HTML (ou também JavaScript) anteriormente presentes no arquivo. Dessa forma, no lado do cliente não há conhecimento referente aos códigos previamente existentes na linguagem PHP, pois as páginas são recebidas devidamente processadas pelo servidor. (MILETTO, 2014, p. 172).

De acordo com Welling e Thomsom (2005), existem algumas vantagens para utilização do PHP, sendo estas vantagens formadas pelo Quadro 6.

**Quadro 6 - Vantagens da utilização do PHP**

<b>Desempenho</b>	O PHP é muito eficiente. Utilizando um único servidor barato, você pode atender a milhões de acessos por dia. Os benchmarks publicados pela Zend Technologies ( <a href="http://www.zend.com">http://www.zend.com</a> ) mostram o PHP superando o desempenho do concorrente.
-------------------	--

<b>Integração de banco de dados</b>	O PHP tem conexões nativas disponíveis para muitos sistemas de banco de dados. Além do MySQL, você pode conectar-se diretamente a bancos de dados PostgreSQL, mSQL, Oracle, dbm, FilePro, HyperWave, Informix, InterBase e Sybase, entre outros. O PHP 5 também tem uma interface integrada SQL para muitos arquivos simples, chama SQLite.
<b>Bibliotecas integradas</b>	Como o PHP foi projetado para utilização na Web, ele tem muitas funções integradas para realizar muitas tarefas úteis relacionadas à Web. Você pode gerar imagens GIF instantâneas, conectar-se a outros serviços de rede, enviar e-mail, trabalhar com cookies e gerar documentos PDF, tudo com apenas algumas linhas de código.
<b>Custo</b>	O PHP é gratuito. Você pode fazer download da última versão a qualquer momento em <a href="http://www.php.net">http://www.php.net</a> , gratuitamente.
<b>Aprendizagem do PHP</b>	A sintaxe do PHP está baseada em outras linguagens de programação, principalmente C e Perl. Se já conhece Perl ou C ou uma linguagem do tipo C, como C++ ou Java, você conseguirá produzir quase imediatamente utilizando o PHP.
<b>Suporte orientado a objetos</b>	A versão 5 do PHP possui recursos orientado a objetos muito bem elaborados. Se você aprendeu a programar em Java ou C++, encontrará os recursos (e geralmente sintaxe) esperados, tal como herança, atributos e métodos privados e protegidos, classes e métodos abstratos, interfaces, construtores e destruidores. Você até encontrará recursos menos comuns, como o comportamento de iteração integrado. Algumas dessas funcionalidades já estavam disponíveis no PHP versões 3 e 4, mas o suporte orientado a objetos na versão 5 é muito mais completo.
<b>Portabilidade</b>	O PHP está disponível para muitos sistemas operacionais diferentes. Você pode escrever código de PHP em sistemas operacionais do tipo Unix gratuitos como o Linux e o FreeBSD, versões comerciais do Unix como o Solaris e IRIX, ou em versões diferentes do Microsoft Windows. Código bem escrito normalmente funcionará sem modificação em um sistema diferente executando o PHP.
<b>Código-fonte</b>	Você tem acesso ao código-fonte do PHP. Ao contrário dos produtos comerciais e de código-fonte fechado, se houver algo que deseje modificar ou adicionar à linguagem, você é livre para fazer isso. Não é preciso esperar o fabricante lançar os patches e nem se preocupar com a possibilidade de o fabricante largar o negócio ou decidir parar de oferecer suporte ao produto.
<b>Disponibilidade de suporte</b>	A Zend Technologies ( <a href="http://www.zend.com">www.zend.com</a> ), a empresa responsável pelo mecanismo que mantém o PHP, financia o desenvolvimento do PHP oferecendo suporte e softwares relacionados numa base comercial.

Fonte: Adaptado de Welling e Thomsom (2005).

“A interpretação dos scripts PHP possibilita a geração de códigos HTML, JavaScript, além de documentos PDF, XML, imagens ou textos, os quais podem ser enviados ao cliente ou simplesmente armazenados no servidor.” (MILETTO, 2014, p.172).

### 2.6.1 Laravel

O Laravel é um *framework* para utilização em aplicações web, que fornece um ponto de partida para a criação da aplicação desejada, o *framework* disponibiliza alguns recursos poderosos como por exemplo uma injeção de dependência completa, abstração do banco de

dados expressiva, filas de trabalhos agendados, testes de unidade e uma ampla integração, além de várias outras funcionalidades interessantes que ajudam o desenvolvedor a ter mais produtividade. (LARAVEL, 2021).

Uma das grandes vantagens para utilização do *framework* são, que ele possui uma estrutura progressiva, ou seja, o Laravel cresce junto com o desenvolvedor, sendo que a vasta biblioteca de documentação, guias e tutoriais ajudam a aprender o básico do *framework* sem maiores dificuldades, para desenvolvedores já com mais experiência o *framework* está preparado para lidar com cargas de trabalho corporativas, com base em todas as suas funcionalidades. (LARAVEL, 2021).

Outra vantagem é uma estrutura escalável pois juntamente com a estrutura de escalonamento do PHP existe um suporte embutido no Laravel para sistemas de cache rápido e distribuído, assim preparando as aplicações para lidar com centenas de milhões de requisições por mês. Por fim o *framework* conta também com uma estrutura de comunidade, sendo que combina os melhores pacotes do ecossistema do PHP, sendo que milhares de desenvolvedores já contribuíram para o melhoramento da sua estrutura. (LARAVEL, 2021).

## 2.7 SEGURANÇA DE DADOS

Sistemas de comunicação como o da internet são controlados por computadores ligados em uma determinada rede, pela qual as informações são trafegadas de uma origem até seu respectivo destino, com isso alguns problemas de sigilo e autenticidade podem surgir durante o percurso, como por exemplo como esconder informações sigilosas e como evitar que informações caiam nas mãos de pessoas mal-intencionadas. (TERADA, 2008).

Uma das formas de manter a segurança dos dados é a criptografia. “Algoritmos criptográficos basicamente objetivam “esconder” informações sigilosas de qualquer pessoa desautorizada a lê-las, isto é, de qualquer pessoa que não conheça a chamada chave secreta de criptografia.” (TERADA, 2008, p.19).

### 2.7.1 LGPD

“A Lei Geral de Proteção de Dados Pessoais (LGPD), Lei nº 13.709/2018, foi promulgada para proteger os direitos fundamentais de liberdade e de privacidade e a livre formação da personalidade de cada indivíduo.” (BRASIL, 2021a).

“A Lei fala sobre o tratamento de dados pessoais, dispostos em meio físico ou digital, feito por pessoa física ou jurídica de direito público ou privado, englobando um amplo conjunto de operações que podem ocorrer em meios manuais ou digitais.” (BRASIL, 2021a).

Existem alguns princípios para a lei geral de proteção de dados, sendo estes princípios apresentados no Quadro 7.

**Quadro 7 - Princípios da LGPD**

<b>Finalidade</b>	A realização do tratamento deve ocorrer para propósitos legítimos, específicos, explícitos e informados ao(à) titular, sem possibilidade de tratamento posterior de forma incompatível com essas finalidades.
<b>Adequação</b>	A compatibilidade do tratamento deve ocorrer conforme as finalidades informadas ao(à) titular, de acordo com o contexto do tratamento.
<b>Necessidade</b>	O tratamento deve se limitar à realização de suas finalidades, com abrangência dos dados pertinentes, proporcionais e não excessivos em relação às finalidades do tratamento de dados.
<b>Livre acesso</b>	É a garantia dada aos(às) titulares de consulta livre, de forma facilitada e gratuita, à forma e à duração do tratamento, bem como à integralidade de seus dados pessoais.
<b>Qualidade dos dados</b>	É a garantia dada aos(às) titulares de exatidão, clareza, relevância e atualização dos dados, de acordo com a necessidade e para o cumprimento da finalidade de seu tratamento.
<b>Transparência</b>	É a garantia dada aos(às) titulares de que terão informações claras, precisas e facilmente acessíveis sobre a realização do tratamento e os respectivos agentes de tratamento, observados os segredos comercial e industrial.
<b>Segurança</b>	Trata-se da utilização de medidas técnicas e administrativas qualificadas para proteger os dados pessoais de acessos não autorizados e de situações acidentais ou ilícitas de destruição, perda, alteração, comunicação ou difusão.
<b>Prevenção</b>	Compreende a adoção de medidas para prevenir a ocorrência de danos por causa do tratamento de dados pessoais.
<b>Não discriminação</b>	Sustenta que o tratamento dos dados não pode ser realizado para fins discriminatórios, ilícitos ou abusivos.
<b>Responsabilização e prestação de contas</b>	Demonstração, pelo Controlador ou pelo Operador, de todas as medidas eficazes e capazes de comprovar o cumprimento da lei e a eficácia das medidas aplicadas.

Fonte: Adaptado de Brasil (2021b).

A “Lei estabelece uma estrutura legal de direitos dos(as) titulares de dados pessoais. Esses direitos devem ser garantidos durante toda a existência do tratamento dos dados pessoais realizado pelo órgão ou entidade.” (BRASIL, 2021a).

### 3. METODOLOGIA DA PESQUISA

Este trabalho de conclusão de curso se caracteriza como uma pesquisa aplicada e descritiva, com o objetivo de analisar e descrever o processo de desenvolvimento de uma aplicação web responsiva que ajude a população na busca por médicos de todas as especialidades disponíveis no município. O trabalho procura em responder as seguintes questões: Como melhorar a busca de um médico especialista? Como mensurar o nível de satisfação e avaliar a consulta médica? Como otimizar o tempo gasto até achar o profissional adequado?

Para a construção dessa pesquisa, inicialmente foi realizada a revisão da literatura onde são apresentados os conceitos de tecnologias necessárias para o desenvolvimento do protótipo. Em segundo lugar, foi construído a seção de estado da arte, a fim de identificar sistemas com um funcionamento semelhante ao proposto pela pesquisa. O estado da arte também auxiliou no levantamento de requisitos, que foram realizados na etapa seguinte juntamente com a construção de diagramas. No capítulo de desenvolvimento foi realizada a análise que visa facilitar o entendimento das necessidades para em seguida iniciar o processo de implementação. Nesta etapa também foi realizada a modelagem e estrutura do banco de dados para o PostgreSQL. Durante a implementação a parte de *backend* foi desenvolvida com a tecnologia Laravel, um *framework* de PHP, e o *frontend* foi desenvolvido com auxílio do *Bootstrap*, sendo que a aplicação é responsiva, possível utilizar em computadores, notebooks, tablets e celulares.

#### 3.1 ESTADO DA ARTE

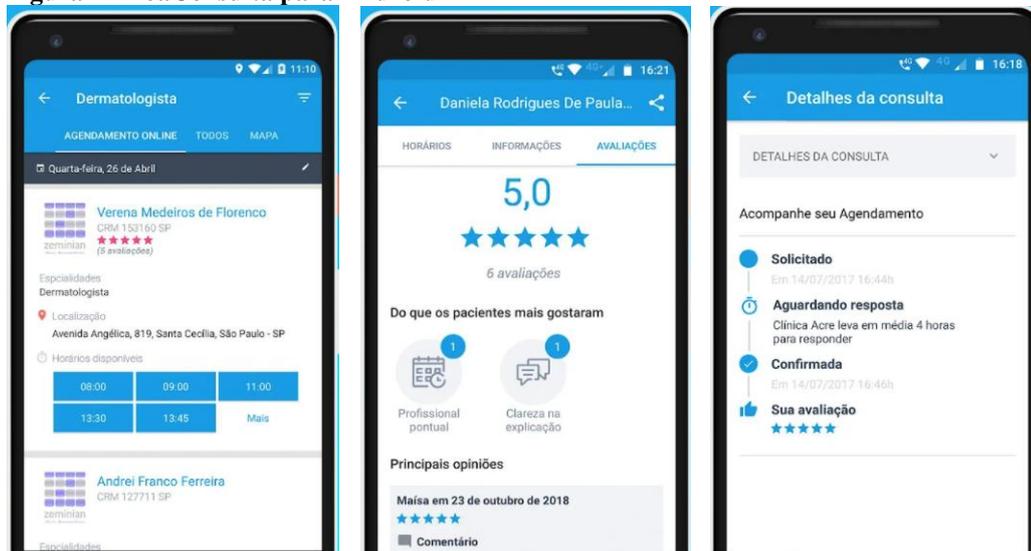
Nesta seção serão apresentadas duas ferramentas já disponíveis no mercado que são similares ao proposto para desenvolvimento. Serão listadas vantagens e desvantagens destas ferramentas, servindo como uma complementação na análise do levantamento de requisitos para o sistema.

##### 3.1.1 BoaConsulta

O aplicativo BoaConsulta (Figura 2) possui diversas especialidades e convênios, sendo que é possível fazer o agendamento desde um psicólogo até um cirurgião. O sistema além de marcar consultas presenciais, também possibilita a tele consulta, onde possibilita ao usuário uma experiência cem por cento online. A aplicação conta com mais de 58 mil profissionais

cadastrados e 1,7 milhões de pacientes registrados, seguindo todos os padrões de segurança, inclusive os estabelecidos pela Lei Geral de Proteção de Dados. (BOACONSULTA, 2021).

**Figura 2 - BoaConsulta para Android**



Fonte: BoaConsulta (2021).

A aplicação possui 3 versões, web, Android e IOS, abrangendo a maioria dos dispositivos disponíveis no mercado, contando com diferentes ferramentas para contratação, como a agenda online, gestão para clínicas e telemedicina, tornando uma solução abrangente para o consultório.

O aplicativo possui uma interface de simples utilização e bastante intuitiva, sendo prático e rápido. Uma de suas desvantagens é possuir diversos módulos, assim sendo como não há uma especialização em uma determinada área pode ser que o sistema perca um pouco de sua usabilidade, rapidez em correções de problemas e lançamentos de novas funcionalidades para um módulo em específico. Outra desvantagem é que é necessário solicitar a agenda de um profissional, não sendo possível visualizá-la em tempo real a qualquer momento.

### 3.1.2 MeuMedico

Outro aplicativo que propõe uma funcionalidade semelhante ao protótipo deste trabalho é o MeuMedico (Figura 3), ele conta com a emissão digital ou entrega física de receituário especial, oferece uma alternativa de consulta presencial na residência do paciente, a possibilidade de agendamento de consultas presenciais em consultório e agendamento de consultas totalmente online. (MEUMEDICO, 2020).

**Figura 3 - MeuMedico versão Web**

The image shows a web interface for 'MeuMedico'. On the left, there is a card for 'Consulta Online' with a price of R\$ 89,90 and a red 'Agendar' button. The main area is titled 'Agenda online' and shows a calendar for 'ago 2021'. A date selector shows 'Hoje' (today). A calendar grid displays dates from 1 to 11. A 'Consulta Online' box on the right indicates a 20-minute consultation for R\$ 89,90 with a 'Próximo' button. A 'Política de cancelamento' dropdown is also visible.

Dom.	Seg.	Ter.	Qua.	Qui.	Sex.	Sáb.
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4
5	6	7	8	9	10	11

Fonte: Acervo do Autor (2021).

A aplicação possui uma versão apenas Web com diferentes funcionalidades disponíveis além do agendamento como por exemplo a renovação de receitas e requisições e renovações com entregas de vias físicas. Além de disponibilizar um chat para conversa em que é possível solucionar algumas dúvidas.

O aplicativo possui uma interface simples e rápida, porém a desvantagem é possuir poucos profissionais disponíveis para agendamento pois está em fase de crescimento, também a falta de informações apresentadas no sistema para o usuário pode se tornar um problema, como por exemplo a dificuldade de encontrar em quais cidades os profissionais estão atuando.

## **4. CATÁLOGO MÉDICO: DESENVOLVIMENTO DE PROTÓTIPO DE APLICAÇÃO WEB RESPONSIVA**

Neste capítulo serão abordados assuntos referentes a construção do protótipo, demonstrando a análise do projeto bem como a implementação do trabalho.

### **4.1 ANÁLISE**

Conforme apontado na revisão da literatura a engenharia de software tem como um de seus objetivos entender as necessidades de negócio do cliente. Com base nisso a análise é importante para definir o que será desenvolvido e como isso será feito em alto nível. A manutenção do software também se torna mais fácil, uma vez que mapeado todos os processos de desenvolvimento na análise a identificação e correção dos problemas se torna mais simples.

#### **4.1.1 Visão Geral do Sistema**

Este protótipo de aplicação tem como objetivo facilitar a realização de agendamento de consultas médicas. A ideia central é que o próprio cliente consiga fazer o agendamento de sua consulta, sendo que após a realização desta um sistema de avaliação também será disponibilizado.

Com a utilização da aplicação o usuário poderá poupar tempo, pois não precisará mais fazer buscas por consultórios médicos na internet e perder tempo em ligações para conseguir agendar uma consulta, pois a agenda de cada profissional será disponibilizada na aplicação. Com isso também fica mais fácil verificar a demanda de cada profissional e agendar uma consulta o mais breve possível dependendo da urgência necessária.

Outra vantagem do sistema é que ele contará com um sistema de avaliações do profissional, ou seja, quando uma consulta é efetivada, o usuário poderá cadastrar uma avaliação do médico, esta por sua vez será crucial para o ranqueamento do profissional no município, atendendo uma melhor ou pior posição no catálogo de médicos.

Para o médico, ele terá um amplo controle de sua agenda, podendo gerenciar todos os horários que irá atender, possibilitando o cadastro de restrições de determinados horários em que o profissional não pode atuar. Também permitindo a efetivação de cada solicitação de consulta feita pelos clientes. Além disso o sistema disponibilizará para que usuários do sistema consigam fazer a manutenção completa na agenda de um médico, a partir do momento que ele delegue os privilégios necessários para isso.

Através do uso da aplicação o cliente poderá ter acesso a lista de todos os médicos de um determinado município, facilitando assim a escolha de um profissional, com base em seu ranqueamento e demanda. O cliente ainda terá acesso a informações como visualizar o consultório em que será agendada a consulta e informações gerais sobre o médico selecionado.

#### 4.1.2 Comparação do Protótipo com o Estado da Arte

De forma breve, foram realizadas algumas comparações entre o protótipo deste trabalho e as aplicações listadas no estado da arte, apresentando os recursos existentes nos sistemas avaliados. O Quadro 8, representa a diferença dos recursos de cada aplicação.

**Quadro 8 - Recursos nos sistemas avaliados**

Funcionalidade	Catálogo de Médicos	BoaConsulta	MeuMedico
Acesso simplificado a agenda dos médicos.	X		
Acesso a informação de médicos e consultórios.	X	X	
Controle sobre a agenda do profissional	X	X	X
Sistema de avaliação para ranqueamento dos profissionais.	X	X	
Sistema especialista em agendamento de consultas.	X		
Aplicativo específico para dispositivos móveis.		X	

Fonte: Acervo do Autor (2021).

#### 4.1.3 Requisitos

Nesta seção foram elencados os requisitos necessários para o correto funcionamento do sistema. Os requisitos listados são funcionalidades do sistema para que ele cumpra com o seu propósito. O Quadro 9, representa os requisitos funcionais do protótipo.

**Quadro 9 - Requisitos Funcionais**

Número	Nome	Descrição	Regras de Negócio
RF01	Login do Sistema	O sistema deverá permitir o usuário realizar login para acessar o sistema, será necessário informar o CPF e senha.	
RF02	Logout do Sistema	O sistema deverá permitir o usuário fazer logout na aplicação.	
RF03	Cadastro de Usuário	O sistema deverá permitir o cadastro de usuário. O cadastro de médicos será uma especialização. Os campos necessários para o cadastro de usuários são os seguintes, nome	RN01, RN02

		(obrigatório), cpf (obrigatório), data de nascimento (obrigatório), tipo (obrigatório), senha (obrigatório), e-mail (obrigatório) e o endereço, além disso para os médicos é necessário informar o número do registro de atuação, de forma obrigatória.	
RF04	Menu Principal	O sistema deverá conter um menu, sendo que as rotinas serão listadas com base no tipo de usuário que fez login no sistema.	RN03, RN04, RN05
RF05	Tela Principal	O sistema deverá conter uma tela principal, que será sempre a apresentação do catálogo de médicos. A tela poderá ser filtrada por cidade, nome do médico e consultório.	
RF06	Cadastro de Estados	O sistema deverá permitir o cadastro de estados. Sendo que os campos necessários serão os seguintes: nome (obrigatório) e UF (obrigatório).	
RF07	Cadastro de Cidades	O sistema deverá permitir o cadastro de cidades. Sendo que cada cidade está relacionada a um estado obrigatoriamente. O campo necessário para o cadastro é o seguinte: nome (obrigatório).	
RF08	Cadastro de Consultórios	O sistema deverá permitir o cadastro de consultórios. Sendo que o consultório possuirá um relacionamento obrigatório com a rotina de endereço.	
RF09	Cadastro de Médicos x Consultórios	O sistema deverá permitir cadastrar vários médicos para um determinado consultório, os relacionamentos obrigatórios serão o de médico e consultório para a construção da rotina.	RN10
RF10	Cadastro de Agenda	O sistema deverá permitir o cadastro de agenda de um médico. Para esse cadastro o médico poderá definir em quais horários ele possui sua agenda disponível para consultas. Também deverá ser disponibilizado uma tela de consulta para listar as informações de cadastros da rotina.	RN06
RF11	Cadastro de Consultas	O sistema deverá permitir o cadastro de consultas sendo que elas poderão ter quatro situações, aguardando, confirmada, finalizada e cancelada. Quando o Cliente selecionar um horário para marcar sua consulta, as informações pessoais do médico serão listadas juntamente com informações sobre o consultório em questão.	RN07
RF12	Cadastro de Avaliações	O sistema deverá permitir o cadastro de avaliações. Também é necessário listar as informações da avaliação em uma tela de consulta.	RN08
RF13	Efetivação de Consultas	O sistema deverá dispor de uma tela para listar as consultas que estão com a situação aguardando, e possibilitar o cancelamento ou efetivação destas determinadas consultas.	
RF14	Cadastro de Contato do Usuário	O sistema deverá permitir o cadastro de contatos do usuário, onde vários registros poderão ser vinculados a um usuário. A rotina deverá ter os campos de tipo	

		(obrigatório), descrição (obrigatório), preferencial (obrigatório), ativo (obrigatório) e usuário que virá preenchido com o usuário logado ou o solicitante de uma consulta. Além disso também deve ser disponibilizada uma consulta para listagem dos registros de contato.	
RF15	Cadastro de Especialidade	O sistema deverá permitir o cadastro de especialidades, onde cada médico poderá estar relacionado a várias especialidades. O campo necessário para o cadastro é o de nome (obrigatório).	
RF16	Cadastro de Calendário	O sistema deverá permitir o cadastro de calendário, onde será listado todos os dias de um determinado ano para criação da agenda.	RN09
RF17	Cadastro de Horários do Consultório	O sistema deverá permitir o cadastro de um horário padrão de atuação de um consultório. Na rotina serão listados os horários de início e fim dos períodos de funcionamento do consultório.	
RF18	Tela de Apresentação	O sistema deverá conter uma tela de apresentação do sistema, onde informações de abertura serão apresentadas, contendo as principais funcionalidades do sistema e avaliações.	
RF19	Cadastro de Equipe Administrativa	O sistema deverá permitir o cadastro da Equipe Administrativa de um médico por consultórios. Será necessário o preenchimento apenas do campo de usuário. Também é necessário listar as informações da equipe administrativa em uma tela de consulta.	
RF20	Cadastro de Itens da Agenda	O sistema deverá permitir o cadastro de itens da agenda, que serão os horários e datas que o médico terá disponibilidade de atendimento. Também deverá ser disponibilizada uma tela de consulta que listará todas as informações da rotina.	
RF21	Cadastro de Endereço	O sistema deverá permitir o cadastro de endereços, que posteriormente serão vinculados a consultórios. A rotina deverá ter os seguintes campos para cadastro: logradouro (obrigatório), número (obrigatório), complemento (opcional) e cidade (obrigatório). Além disso também deve ser disponibilizada uma tela de consulta para listar todas as informações da rotina.	
RF22	Cadastro de Médico x Especialidade	O sistema deverá permitir que o médico seja vinculado a várias especialidades. A rotina deverá ter apenas os campos de médico (obrigatório) e especialidade (obrigatório). Além disso também deve ser disponibilizado uma tela de consulta listando todos os dados.	
RF23	Cadastro de Item do Calendário	O sistema deverá permitir o cadastro de itens do calendário, que consistem basicamente nos dias que o determinado calendário possuirá. A rotina deverá ter o	

		campo de calendário (obrigatório) e data (obrigatório). Também deve ser disponibilizada uma consulta que lista todos os dias do determinado calendário.	
RF24	Cadastro de Contatos do Consultório	O sistema deverá permitir o cadastro de contatos do consultório, onde vários registros poderão ser vinculados a um consultório. A rotina deverá ter os campos de tipo (obrigatório), descrição (obrigatório), preferencial (obrigatório), ativo (obrigatório) e consultório (obrigatório). Além disso também deve ser disponibilizada uma consulta para listagem dos registros de contato.	

Fonte: Acervo do autor (2021).

O Quadro 10, representa os requisitos funcionais opcionais para o desenvolvimento do protótipo. Estes poderão ser implementados em uma segunda versão do protótipo. Apesar de estarem listados, não fazem parte do escopo inicial.

**Quadro 10 - Requisitos Funcionais Opcionais**

Número	Nome	Descrição
RF25	Envio de e-mail	O sistema deverá permitir o envio de e-mail para os clientes informando o horário de agendamento e a confirmação da consulta. Também devem ser enviados e-mails como forma de lembrete do horário e data da consulta.
RF26	Cadastro de planos de saúde	O sistema deverá permitir o cadastro de planos de saúde. Os médicos serão os únicos responsáveis pelo cadastro destes registros.
RF27	Cadastro de Usuário x Plano de Saúde	O sistema deverá permitir o vínculo do usuário com seus respectivos planos de saúde. Sendo que o usuário pode ter n planos de saúde cadastrados.
RF28	Reserva de horários para planos de saúde	O sistema deverá permitir a reserva de horários na agenda do médico, destinados especificamente a planos de saúde. O responsável pela reserva deverá informar quais planos de saúde o médico atende.
RF29	Tela de preços e planos	O sistema deverá conter uma tela que listará todos os preços e planos disponíveis para que o médico possa contratar o serviço diretamente no sistema.
RF30	Cadastro do médico	O sistema deverá permitir que o médico se cadastre no sistema, após a escolha de um plano que mais lhe agrade. Na rotina será obrigatório o preenchimento do campo de registro do profissional.
RF31	Sistema de pagamentos	O sistema deverá disponibilizar um sistema de pagamentos, por boleto, cartão de crédito e PIX para que os médicos consigam pagar por seus planos, deixando assim o sistema sempre ativo.
RF32	Inclusão múltipla de itens do calendário	O sistema deverá disponibilizar uma ação para inclusão múltipla de itens do calendário, onde será informado uma data de início e uma data de término, com base nas duas datas o sistema deve incluir os itens do calendário com todos os dias úteis entre o intervalo selecionado.

RF33	Cadastro de Usuário por Administrativos e Médicos	O sistema deverá permitir que usuários administrativos e médicos façam o cadastro de usuários clientes no sistema. Sendo que o preenchimento dos campos fica obrigatório, conforme informado no requisito RF03.
RF34	Cadastro de Consultas por Administrativos e Médicos	O sistema deverá permitir que usuários administrativos e médicos agendem consultas para outros usuários clientes do sistema. As regras para o cadastro de consulta devem ser as mesmas do requisito RF11.
RF35	Integração no sistema de login	O sistema deverá permitir a integração de login com as API's do Google e Facebook, para que o cadastro de usuário se torne mais simplificado com os dados já preenchidos a partir destas integrações.
RF36	Integração da agenda	O sistema deverá permitir a integração com a agenda do Google, onde ao ser confirmado uma consulta o sistema cria um compromisso na agenda pessoal do usuário.

Fonte: Acervo do autor (2021).

O Quadro 11, representa os requisitos não funcionais da aplicação, ou seja, os requisitos relacionados ao uso da aplicação e suas características.

**Quadro 11 - Requisitos não funcionais**

Número	Descrição
RNF01	O sistema deverá ser desenvolvido com a versão 8 do PHP ou superior.
RNF02	O sistema deverá ser a prova de <i>SQL Injection</i> .
RNF03	O sistema não poderá ter rotinas em que a requisição demore mais de 10 segundos para ser concluída.
RNF04	O sistema deverá ser capaz de voltar ao funcionamento em caso de falhas em no máximo 4 horas.
RNF05	O sistema deverá ser desenvolvido com a versão 13 do PostgreSQL ou superior.
RNF06	O sistema deverá ser responsivo, sendo acessível pelos principais navegadores do mercado em computadores ou dispositivos móveis.
RNF07	O sistema deverá possuir princípios de segurança da informação e estar aderente a LGPD.

Fonte: Acervo do autor (2021).

O Quadro 12, representa as regras de negócio da aplicação, ou seja, as regras que precisam ser implementadas para o correto desenvolvimento dos requisitos.

**Quadro 12 - Regras de Negócio**

Número	Descrição
RN01	Os usuários poderão ser de 3 tipos: médicos, administrativos e clientes.
RN02	O cadastro de médico precisa obrigar o preenchimento do número do registro.

RN03	Para os Clientes: serão listadas as telas de agendamento, consultas e avaliações.
RN04	Para o pessoal Administrativo: serão listadas as telas de agendamento, consultas, avaliações e agendas.
RN05	Para os profissionais Médicos: serão listadas as telas de agendamento, consultas, avaliações, consultórios, agendas e especialidades.
RN06	Os horários da agenda dos médicos precisam obedecer aos horários do consultório.
RN07	Cada cliente poderá ter várias consultas cadastradas, porém apenas uma para uma mesma faixa de horários.
RN08	O cadastro será liberado apenas para consultas que estiverem finalizadas.
RN09	O calendário deverá ser único por ano.
RN10	A inclusão de registros deve ser habilitada apenas para usuários que sejam médicos.

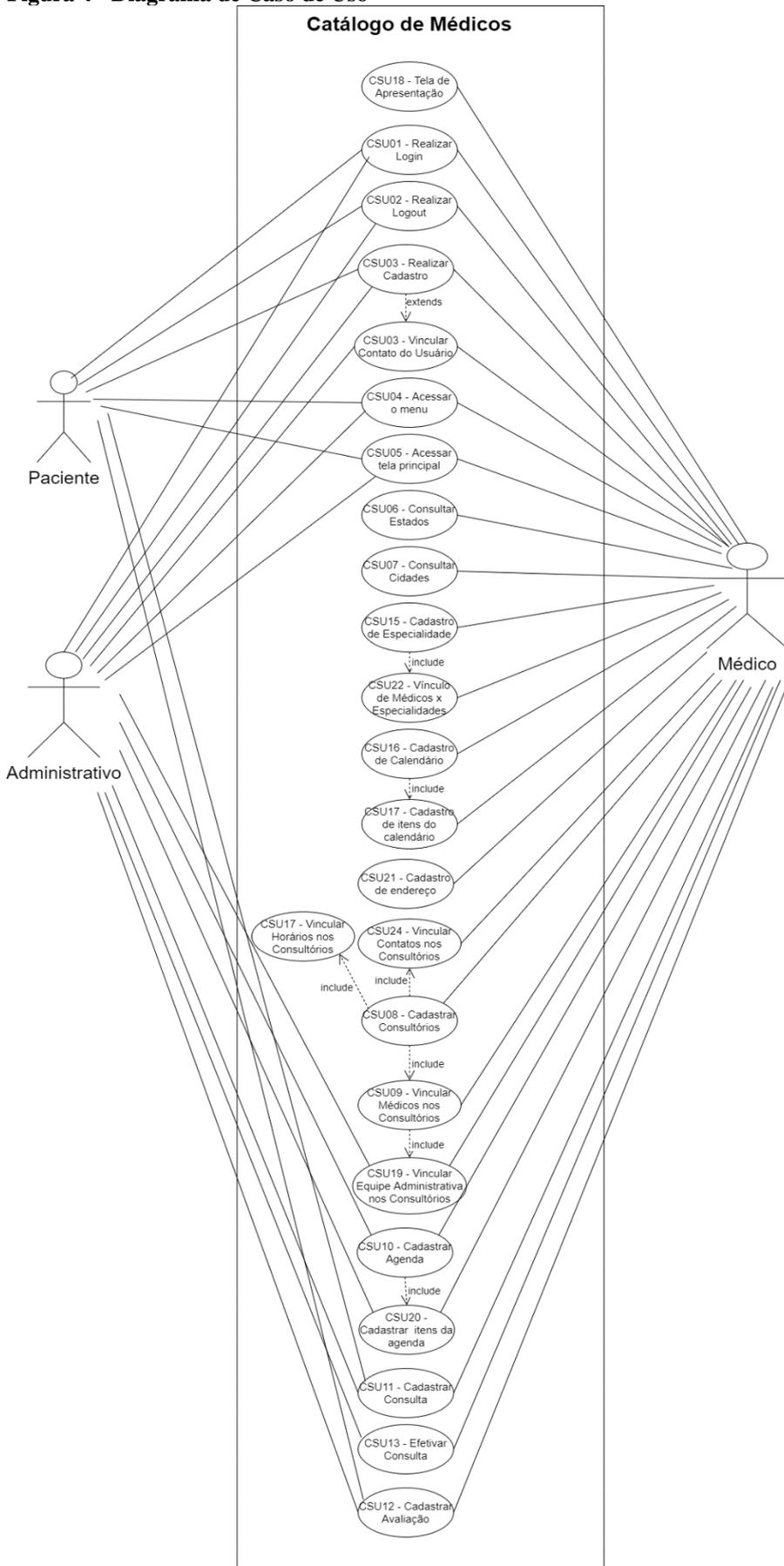
Fonte: Acervo do autor (2021).

#### 4.1.4 Diagramas

Nos diagramas a seguir são listados processos de utilização do sistema e casos de uso da aplicação.

Na Figura 4 é apresentado o diagrama de caso de uso, onde o objetivo é demonstrar as funcionalidades do sistema, inclusive o que cada ator poderá executar dentro do protótipo.

Figura 4 - Diagrama de Caso de Uso



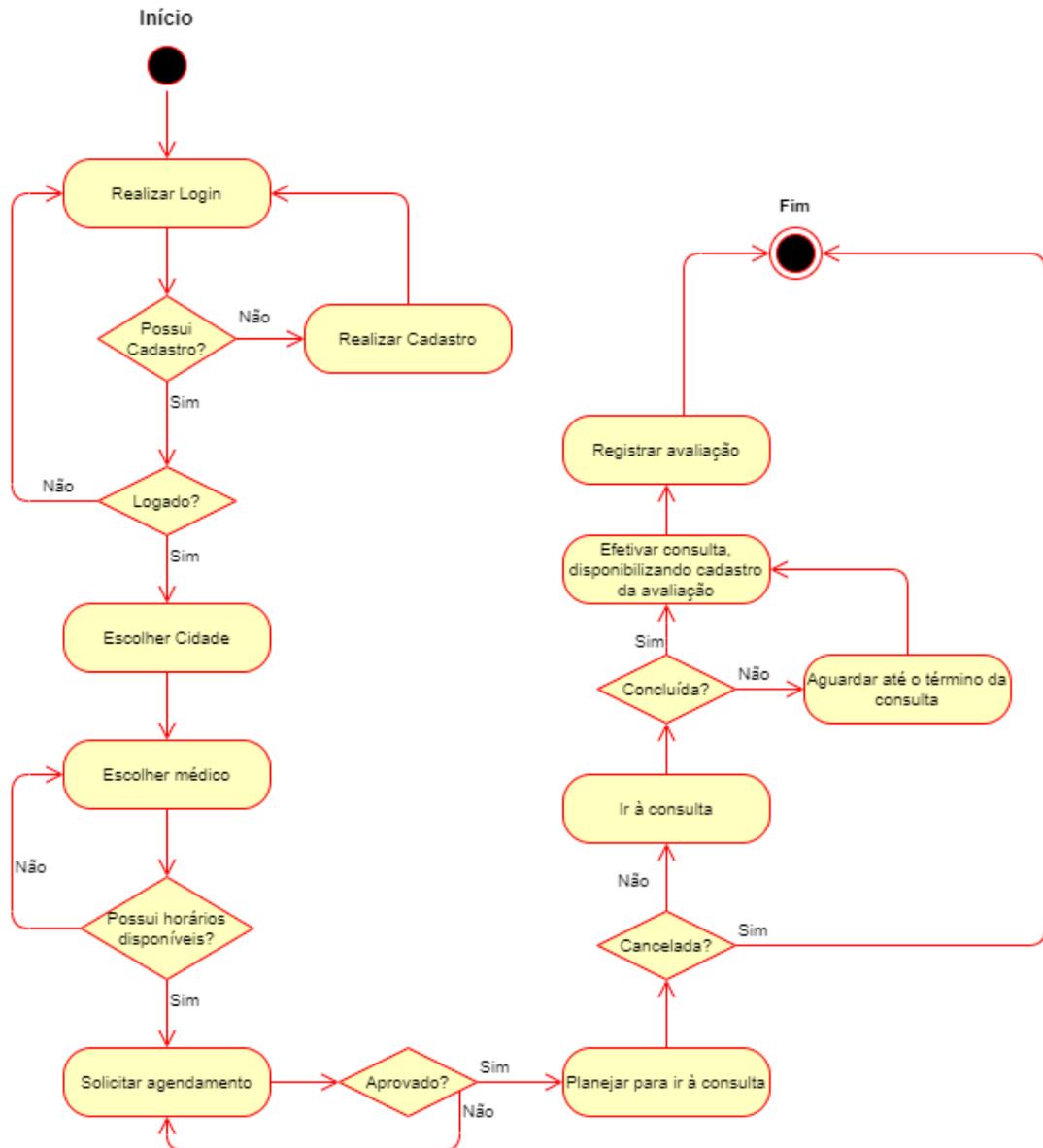
Fonte: Acervo do Autor (2021).

Pode-se perceber no diagrama de caso de uso que todos os tipos de usuário terão acesso para fazer *login* e *logout*, fazer o seu respectivo cadastro de usuário, acessar os menus e tela principal do sistema, marcar uma consulta com qualquer médico que esteja disponível para agendamento, sendo também disponibilizado a avaliação caso a consulta for efetivada. Para a parte administrativa e os médicos do sistema será possível confirmar as consultas que estão com a situação aguardando e efetivá-las para disponibilizar o sistema de avaliação, fazer a troca de horários e o cancelamento caso necessário, também será possível realizar a consulta de cidades e estados juntamente com o cadastro de calendário com seus respectivos horários. E por fim para o médico além de todas as funcionalidades citadas será possível também cadastrar consultórios e especialidades no sistema.

Na Figura 5 é apresentado um diagrama de atividade, demonstrando o processo do agendamento da consulta por um usuário do sistema. Primeiramente é necessário a realização do login no sistema, caso o usuário não possua cadastro então deverá realizá-lo previamente, após isso é necessário a seleção de uma cidade e um médico para a realização da consulta. Após a seleção do médico é necessário fazer a escolha de um horário disponível na agenda do profissional, solicitar o agendamento e aguardar a aprovação.

O médico ou a equipe administrativa deverá confirmar a consulta, por fim após a consulta ser realizada um sistema de avaliação e disponibilizado para o usuário. Caso a consulta seja cancelada então o sistema de avaliação não será disponibilizado e o processo é encerrado.

**Figura 5 - Diagrama de Atividade do Agendamento de Consulta**



Fonte: Acervo do Autor (2021).

Na Figura 6 é apresentado o diagrama de entidade relacionamento (MER), detalhando as tabelas e seus relacionamentos no banco de dados do protótipo.



## 4.2 IMPLEMENTAÇÃO

Nesta seção são detalhadas as ferramentas e tecnologias utilizadas para o desenvolvimento do protótipo, bem como são apresentadas as rotinas implementadas.

### 4.2.1 Técnicas e Ferramentas Utilizadas

Para a etapa da análise, (apresentada no item 4.1), foram utilizadas as ferramentas *Draw.io* e *SQL Power Architect* para a construção dos diagramas de caso de uso, atividades e entidade-relacionamento.

O protótipo constitui-se de uma aplicação web responsiva, onde através dela que são realizados todos os cadastros acessadas as e funcionalidades da aplicação. Para o desenvolvimento foram utilizadas tecnologias web. A principal linguagem de desenvolvimento foi o PHP, sendo a facilidade de integração com outras tecnologias como o HTML, CSS, Bootstrap e Javascript, o principal motivo de sua escolha.

O HTML versão 5 foi utilizado para a marcação de texto no *front-end*, ou seja, a base da parte visual da aplicação. Por ser utilizado em praticamente todos os navegadores e possuir uma fácil implementação se torna uma ferramenta indispensável na construção de uma aplicação web.

Para a estilização da página foi utilizado o CSS versão 3, essa tecnologia permite criar estilos personalizados para diferentes elementos, deixando as telas mais amigáveis para o usuário e possibilitando uma maior acessibilidade e usabilidade do protótipo.

Juntamente com o CSS foi utilizada a biblioteca Bootstrap que possui inúmeros estilos já prontos para implementação de uma aplicação tornando assim o processo de desenvolvimento mais ágil.

O *Javascript* foi utilizado como uma linguagem *cliente-side* para manipulação de elementos e criação de comportamentos, executados no navegador do cliente, já que com a linguagem é possível receber informações e carregá-las em tela em tempo de execução para o usuário. Além de possuir uma ampla aceitabilidade nos navegadores atuais.

Unido ao *JavaScript*, foi utilizada a biblioteca *jQuery*, que é voltada para manipulação de dados do *JavaScript*, tornando mais ágil a chamada e criação de métodos uma vez que é mais simplificada e intuitiva a criação de comportamentos no sistema.

Para a programação voltada ao lado do servidor, foi utilizada a linguagem de programação PHP. Visando auxiliar o processo de desenvolvimento, consequente tornando-o

mais ágil foi utilizado o *framework* Laravel. Com isso foi possível ter um maior desempenho na aplicação e uma maior segurança nos dados.

Com a utilização do *framework* Laravel, foi possível realizar a manipulação com o banco de dados de maneira simplificada com o *Eloquent (Object Relational Mapping)*, que é uma biblioteca interna que permite a criação de tabelas até a manipulação de registros.

Para uma maior segurança nas requisições o Laravel disponibiliza o sistema de rotas, sendo que para definir uma nova rota é necessário definir a *URL* e para qual *Controller* será responsável para fazer todos os processamentos necessários, inclusive camadas de validações podem ser aplicadas, como por exemplo exigir que o usuário esteja logado no sistema para acessar determinada rota.

Para o banco de dados, foi utilizado o *PostgreSQL*, sendo um banco de dados relacional amplamente utilizado por desenvolvedores e organizações. Além disto, é gratuito e disponibiliza diversos recursos úteis e modernos como o controle de transações, gatilhos, visões e permitir a distribuição do banco de dados.

A ferramenta de desenvolvimento utilizada foi a *IDE PhpStorm*, que é uma ferramenta que conta com todo o apoio para o desenvolvimento das tecnologias utilizadas, mapeando todas as bibliotecas e sugerindo métodos e classes para a implementação agilizada no desenvolvimento, além da ampla possibilidade de modificações como o uso de extensões.

Para mensurar a qualidade do software foram realizados diversos testes exploratórios nas rotinas da aplicação, simulando o uso diário de um usuário. Assim foi possível a correção de inconformidades nas rotinas, identificadas através destes testes executados no protótipo.

A segurança dos dados, foi garantida através da criptografia feita nas senhas dos usuários, onde apenas eles têm acesso a sua senha. Já a não exposição a dados pessoais foi verificada na visualização das avaliações onde não é apresentado o nome completo dos usuários, sendo listado apenas o primeiro nome. Outros recursos para garantir total aderência a LGPD podem ser necessários conforme novas funcionalidades forem disponibilizadas ao sistema.

#### **4.2.2 Utilização e Funcionamento**

O protótipo trata-se de uma aplicação web responsiva, sendo que ele possui três níveis de acesso, para clientes, administrativo e médicos. No nível de cliente ele poderá cadastrar consultas e avaliações. Já para o nível administrativo, além de todas as funcionalidades do nível de cliente, ou seja, ao aplicação permite que o usuário possa agendar consultas apenas para si

mesmo, o sistema também permite a gestão das consultas, da agenda e de outros membros da equipe administrativa. Por fim, para o nível dos médicos, todas as funcionalidades do sistema estarão habilitadas, inclusive o agendamento de consultas para si próprio, juntamente com as rotinas de endereço, consultório, especialidade, calendário e o vínculo de especialidades e consultórios ao profissional. A página inicial que a aplicação oferece está representada na Figura 7, e pode ser acessada para os três níveis de usuários.

**Figura 7 – Tela de Apresentação (RF18)**

Catálogo de Médicos Login Cadastrar

# Catálogo de Médicos

Catálogo de médicos líder no agendamento de consultas e avaliações de médicos.

Estamos no mercado há 1 ano, sempre oferecendo um produto com a melhor experiência possível para todos os nossos clientes, desenvolvendo com preocupação para que tudo funcione corretamente e de forma rápida, facilitando assim a vida de toda à população.

[Confira as avaliações!](#)

### Agendamento de Consultas

Nosso sistema permite um agendamento extremamente eficiente de consultas, basta fazer o cadastro no site e todos os médicos cadastrados para uma determinada cidade serão listados, após isso é só escolher o profissional, um horário disponível e pronto, sua consulta está marcada.





### Sistema de Avaliação

Após a efetivação de uma consulta, será disponibilizado para todos os usuários um sistema de avaliação, assim é possível ranquear os profissionais de uma determinada cidade, facilitando portanto a escolha do profissional mais adequado.

### Gestão de agendas

O sistema permite que os consultórios tenham equipes que façam a gestão das agendas dos médicos, possibilitando uma maior produtividade ao profissional.





### Confiabilidade

Com a utilização do sistema, será minimizado o agendamento de consultas erradas ou que o paciente não possa comparecer na determinada data, pois o cancelamento das consultas pode ser feito de maneira totalmente transparente e rápida, flexibilizando a agenda do profissional em tempo real.

### Avaliações

**Roberto Ford**  
Experiência incrível, o sistema funciona de maneira simples e muito rápida.

**Arnold Veber**  
O MELHOR SOFTWARE DE AGENDAMENTO! Marquei diversas consultas pelo sistema, e nunca tive nenhum tipo de problema, não vejo a hora de precisar ir no médico novamente!

**Antônio Hopinz**  
Consegui poupar muito tempo em busca de um médico, como todos são listados por especialidade o sistema facilita em muito nossas vidas.

**Rasmus Lerdorf**  
Bom.

© 2021 Todos os direitos reservados

Fonte: Acervo do Autor (2021).

Para acessar o sistema é necessário fazer o login, informando o e-mail e senha, conforme representado na Figura 8.

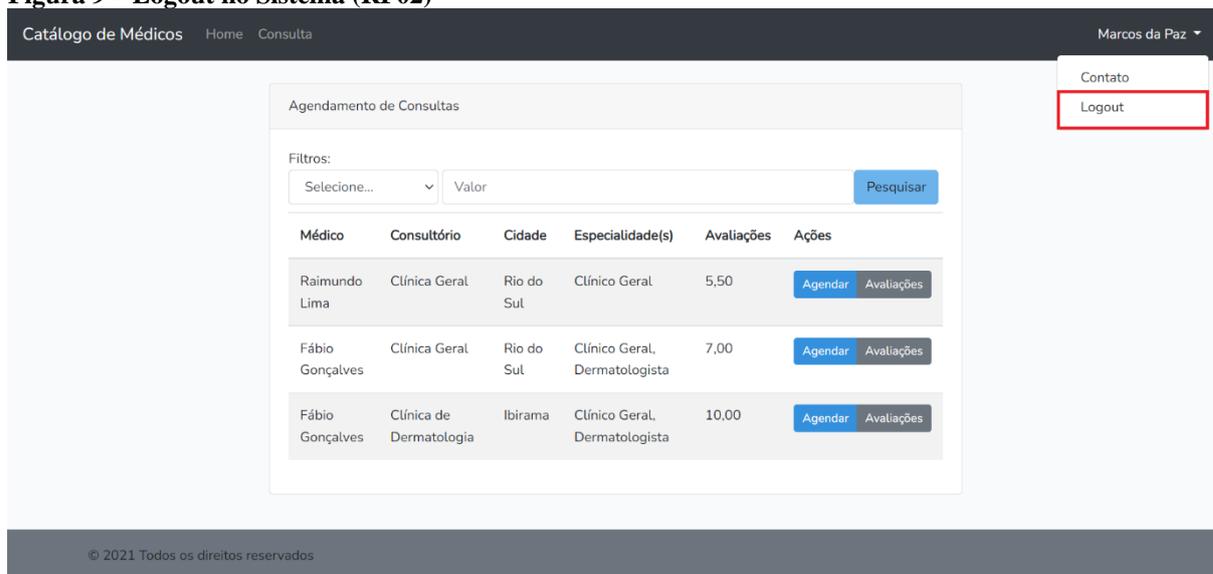
**Figura 8 – Login no Sistema (RF01)**

A imagem mostra a interface de login de um sistema web. No topo, há uma barra de navegação escura com o texto 'Catálogo de Médicos' à esquerda e os links 'Login' e 'Cadastrar' à direita. O formulário de login é centralizado e contém os seguintes elementos: um campo de entrada rotulado 'Login' com o placeholder 'Digite seu e-mail', um campo de entrada rotulado 'Senha', uma caixa de seleção desativada rotulada 'Lembrar-me' e um botão azul rotulado 'Login'. Na base da página, há uma barra de rodapé escura com o texto '© 2021 Todos os direitos reservados'.

Fonte: Acervo do Autor (2021).

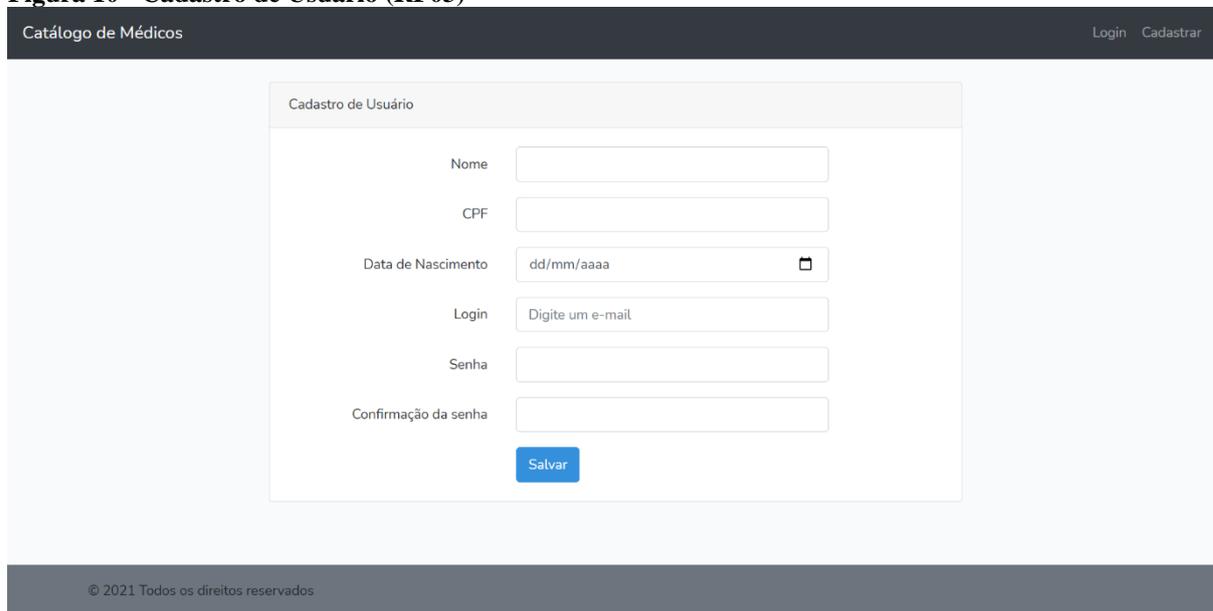
As informações de login serão validadas no banco de dados, ou seja, se o e-mail e senha são válidos. Após a realização do login o usuário será direcionado à tela de agendamento de consultas e será disponibilizado o menu com as rotinas disponíveis de acordo com os privilégios de acesso do usuário.

O sistema permite com que o usuário possa fazer logout no sistema, ou seja, a partir do momento em que o usuário clica no botão de logout sua sessão será encerrada e caso quiser utilizar o sistema, um novo login será necessário. O acesso ao logout fica no menu principal, dentro do agrupamento de ações do usuário logado, no canto superior direito. Como pode ser observado através da Figura 9.

**Figura 9 – Logout no Sistema (RF02)**

Fonte: Acervo do Autor (2021).

Todos os níveis de usuário também podem fazer o cadastro de usuário no sistema, para isso é necessário acessar o botão de cadastrar no menu superior da tela. A Figura 10 demonstra a tela de cadastro de usuário.

**Figura 10 - Cadastro de Usuário (RF03)**

Fonte: Acervo do Autor (2021).

O sistema exige preencher as seguintes informações: nome, CPF, data de nascimento, e-mail, senha e confirmação de senha. O campo e-mail precisa estar em um formato válido e a

senha precisa ter no mínimo 8 caracteres. Todos os usuários cadastrados inicialmente recebem o tipo de cliente por padrão.

A aplicação possui um menu principal, que se adequa com base nos privilégios do usuário logado no sistema. Para um usuário com o nível de acesso cliente, o menu possui as opções: home, consulta e usuário logado. Para o nível administrativo o menu conta com as opções: home, consulta, agrupamento médico com a opção de consultório e usuário logado. Para o nível de médico é disponibilizada as opções: home, consulta, estado, cidade, endereço, consultório, especialidade, o agrupamento médico com as opções de especialidade e consultório, calendário e usuário logado. Como pode ser observado na Figura 11.

**Figura 11 – Menu Principal (RF04)**

© 2021 Todos os direitos reservados

Fonte: Acervo do Autor (2021).

O cadastro de contato do usuário pode ser feito por todos os usuários do sistema, onde é possível cadastrar seus próprios contatos, sendo diferenciados por dois tipos, que são: telefone e e-mail. Como pode ser observado através da Figura 12.

**Figura 12 – Cadastro de Contato do Usuário (RF14)**

Incluir Contato

Tipo  
Telefone

Descrição

Usuário  
Danilo Monteiro

Consultório

Preferencial

Ativo

Criar Contato

© 2021 Todos os direitos reservados

Fonte: Acervo do Autor (2021).

As informações requeridas para o cadastro são as seguintes: tipo, descrição e o usuário que já vem preenchido automaticamente de acordo com o usuário logado, também são disponibilizados os campos de preferencial e ativo.

Para usuários administrativos e médicos é também possível cadastrar contatos para usuários que tenham feito a solicitação de uma consulta, como pode ser observado através da Figura 22.

#### 4.2.2.1 Aplicação Web Cliente

Assim que o usuário faz o cadastro no sistema e realiza o login com suas informações pessoais, o usuário é redirecionado para a tela principal do sistema que seria a do catálogo de médicos. Nessa tela são listados todos os médicos disponíveis no sistema para realizar o agendamento de uma consulta, como pode-se observar na Figura 13.

**Figura 13 - Catálogo de Médicos (RF05)**

Agendamento de Consultas

Filtros:

Selezione... Valor

Médico	Consultório	Cidade	Especialidade(s)	Avaliações	Ações
Danilo Monteiro	Clínica Dermatológica	Rio do Sul	Dermatologista,	8,50	<input type="button" value="Agendar"/> <input type="button" value="Avaliações"/>

© 2021 Todos os direitos reservados

Fonte: Acervo do Autor (2021).

Nesta rotina são listadas informações do médico, consultório, cidade especialidades do médico e sua média de avaliações. Também são disponibilizadas duas ações, a de agendar e visualizar as avaliações que o profissional teve. Além disso a tela conta com o mecanismo de filtros, onde podem ser filtradas as cidades, nomes de médicos e consultórios.

Após acessarem a ação de agendar o usuário é redirecionado para uma tela onde é feito o agendamento de consultas, nessa tela são listadas as informações do médico que são as seguintes: nome, CPF, registro e todas as suas especialidades. Também são listadas as informações do consultório, que são: nome, endereço, cidade, estado e contatos. Por fim é disponibilizado para seleção um horário em que o médico tem disponível para realizar a consulta, conforme pode ser observado na Figura 14.

**Figura 14 – Cadastro de Consultas (RF11)**

Agendar Consulta Médica

**Informações do Médico**  
 Nome: Danilo Monteiro  
 CPF: 956.815.979-70  
 Registro: Registro 1  
 Especialidade(s): Dermatologista.

**Informações do Consultório**  
 Nome: Clínica Dermatológica  
 Endereço: Rua Vila Inconfidência, Jardim América, CEP: 89160-200,  
 Número: 54, Complemento: ...  
 Cidade: Rio do Sul.  
 Estado: Santa Catarina  
 Contatos:  
 Tipo: Telefone, Descrição: 4799058563

**Horários disponíveis**

Dia: 07/06/2021 Horário: 11:00:00 - 12:00:00  
 Dia: 08/06/2021 Horário: 09:00:00 - 10:00:00  
 Dia: 08/06/2021 Horário: 10:00:00 - 11:00:00  
 Dia: 09/06/2021 Horário: 09:00:00 - 10:00:00  
 Dia: 09/06/2021 Horário: 10:00:00 - 11:00:00  
 Dia: 09/06/2021 Horário: 11:00:00 - 12:00:00  
 Dia: 10/06/2021 Horário: 09:00:00 - 10:00:00  
 Dia: 10/06/2021 Horário: 10:00:00 - 11:00:00  
 Dia: 10/06/2021 Horário: 11:00:00 - 12:00:00  
 Dia: 11/06/2021 Horário: 09:00:00 - 10:00:00  
 Dia: 11/06/2021 Horário: 11:00:00 - 12:00:00

Agendar Consulta

© 2021 Todos os direitos reservados

Fonte: Acervo do Autor (2021).

Quando o usuário escolhe um horário e agenda a consulta, é redirecionado para a tela de consultas, nela são listadas as informações básicas da consulta agendada pelo usuário, os seguintes campos são listados: código da consulta, médico, data, horário de início e horário de término, como pode-se observar através da Figura 15.

**Figura 15 - Tela de Consultas (RF11)**

Consulta

Usuário

Código	Médico	Consultório	Data	Horário de Início	Horário de Término	Situação	Ações
1	Danilo Monteiro	Clínica Dermatológica	07/06/2021	09:00	10:00	Finalizada	Cancelar Avaliação
3	Danilo Monteiro	Clínica Dermatológica	08/06/2021	11:00	12:00	Aguardando	Cancelar
4	Danilo Monteiro	Clínica Dermatológica	11/06/2021	10:00	11:00	Aguardando	Cancelar

© 2021 Todos os direitos reservados

Fonte: Acervo do Autor (2021).

Na tela também são listadas duas ações de cancelamento e avaliação, o usuário pode cancelar a consulta a qualquer momento e a avaliação só fica disponível para preenchimento após a consulta já ter sido finalizada.

O cadastro de avaliação consiste no preenchimento de uma descrição e uma nota que pode ir de zero a dez para demonstrar o índice de satisfação com o atendimento da consulta, a tela de cadastro de avaliação pode ser observada na Figura 16. O sistema permite o cadastro de uma avaliação por consulta, portanto quando é a primeira vez que é cadastrada a avaliação a tela de inclusão é chamada com os dados em branco. Caso contrário é chamada a tela de alteração com os dados previamente preenchidos.

**Figura 16 - Cadastro de Avaliações (RF12)**

A imagem mostra a interface de usuário para alterar uma avaliação. No topo, há uma barra de navegação com os links "Catálogo de Médicos", "Home" e "Consulta", e o nome de usuário "Mateus Levi Carlos Figueiredo" com uma seta para baixo. O formulário principal, intitulado "Alterar Avaliação", contém dois campos de entrada: "Descrição" com o texto "Atendimento razoável." e "Nota" com o valor "7". Abaixo dos campos, há um botão verde com o texto "Alterar avaliação". Na base da página, há uma barra de rodapé com o texto "© 2021 Todos os direitos reservados".

Fonte: Acervo do Autor (2021).

A consulta de avaliações pode ser acessada através da ação de avaliações na consulta do catálogo de médicos, na tela são dispostos os dados do usuário, descrição e nota, conforme pode ser observado pela Figura 17. Para não expor o nome completo do cliente optou-se por exibir apenas seu primeiro nome na lista de avaliações.

**Figura 17 - Consulta de Avaliações (RF12)**

Usuário	Descrição	Nota
Raimundo	A consulta foi muito boa, me atendeu de maneira satisfatória.	10
Mateus	Atendimento razoável.	7

Fonte: Acervo do Autor (2021).

A tela de visualizar as avaliações é apenas de consulta, portanto nenhuma ação além desta é disponibilizada para o usuário.

#### 4.2.2.2 Aplicação Web Administrativa

Além de todas as funcionalidades disponibilizadas para o nível de cliente, que basicamente consiste no agendamento de consultas para si próprio e o cadastro de avaliações para estas consultas. O nível administrativo conta com o gerenciamento do consultório do médico, sendo que a tela de consulta lista as informações do médico e consultório, conforme pode-se observar na Figura 18. Para que o usuário administrativo consiga acessar a informação da tela é necessário que o médico tenha cadastrado um consultório e que tenha incluído o vínculo entre o médico e o consultório.

**Figura 18 – Cadastro de Médicos x Consultórios (RF09)**

Código	Médico	Consultório	Ações
1	Danilo Monteiro	Clínica Dermatológica	Deletar Equipe Administrativa Agenda

Fonte: Acervo do Autor (2021).

A tela só lista os registros em que o usuário logado é o médico do determinado registro ou faz parte da equipe administrativa do respectivo consultório. Além das informações também são apresentadas as ações de deletar, equipe administrativa e agenda. A ação de inclusão da rotina é liberada apenas para usuários que sejam médicos.

O sistema permite a consulta da equipe administrativa de um médico vinculado a um consultório, são listadas as seguintes informações na tela: código, usuário, consultório e médico, juntamente com a ação de deletar e inclusão, conforme demonstra a Figura 19.

**Figura 19 - Consulta de Equipe Administrativa (RF19)**

Código	Usuário	Consultório	Médico	Ações
1	Raimundo Cavalcanti	Clínica Dermatológica	Danilo Monteiro	Deletar

Fonte: Acervo do Autor (2021).

Para a inclusão da equipe do médico por consultório o sistema obriga que o relacionamento de médico x consultório venha preenchido de acordo com a informação do registro que foi selecionado previamente e obriga o preenchimento do usuário, conforme pode-se observar na Figura 20.

**Figura 20 – Cadastro de Equipe Administrativa (RF19)**

Incluir Equipe do Médico x Consultório

Médico x Consultório

Danilo Monteiro x Clínica Dermatológica

Usuário

Kevin Marchi

Criar Equipe do Médico x Consultório

© 2021 Todos os direitos reservados

Fonte: Acervo do Autor (2021).

O sistema permite com que o usuário faça a manutenção da agenda do médico. A rotina conta com ações que são padrão utilizadas em diversas rotinas que é a de consulta, inclusão, alteração e exclusão. Essa padronização consiste em listar as informações na consulta, a tela de inclusão e alteração são as mesmas, a única diferença é a disposição dos dados, na tela de inclusão nenhum dado é preenchido nos campos, porém na tela de alteração os dados já são preenchidos com as informações do registro em que o usuário tenha selecionado previamente.

Na consulta da agenda o sistema lista as seguintes informações: código, descrição, código do calendário e ano do calendário, conforme demonstrado na Figura 21.

**Figura 21 - Consulta do Cadastro de Agenda (RF10)**

Código	Descrição	Calendário - Código	Calendário - Ano	Ações
1	Agenda 2021	1	2021	<a href="#">Editar</a> <a href="#">Deletar</a> <a href="#">Itens</a>

Fonte: Acervo do Autor (2021).

Para a tela de inclusão da agenda são necessárias o preenchimento das seguintes informações: médico x consultório que já vem preenchido de acordo com o registro selecionado para criação da agenda, calendário e descrição, pode-se observar a tela na Figura 22. Para a alteração segue a padronização informada previamente para o sistema.

**Figura 22 – Cadastro de Agenda (RF10)**

Médico x Consultório: Danilo Monteiro - Clínica Dermatológica

Calendário: 2021

Descrição:

[Criar Agenda](#)

Fonte: Acervo do Autor (2021).

O sistema permite o cadastro de itens da agenda, onde na consulta são listadas as informações de código, agenda, descrição, data, horário de início, horário de término e situação

que pode ser preenchida com duas opções: disponível e alocado, essa tela pode ser observada através da Figura 23.

**Figura 23 - Consulta de Itens da Agenda (RF20)**

Código	Agenda	Descrição	Data	Horário de Início	Horário de Término	Situação	Ações
3	Agenda 2021	Item dia: 2021-06-07 De: 11:00:00 Até: 12:00:00	07/06/2021	11:00:00	12:00:00	Disponível	<a href="#">Editar</a> <a href="#">Deletar</a>
4	Agenda 2021	Item dia: 2021-06-08 De: 09:00:00 Até: 10:00:00	08/06/2021	09:00:00	10:00:00	Disponível	<a href="#">Editar</a> <a href="#">Deletar</a>
5	Agenda 2021	Item dia: 2021-06-08 De: 10:00:00 Até: 11:00:00	08/06/2021	10:00:00	11:00:00	Disponível	<a href="#">Editar</a> <a href="#">Deletar</a>
7	Agenda 2021	Item dia: 2021-06-09 De: 09:00:00 Até: 10:00:00	09/06/2021	09:00:00	10:00:00	Disponível	<a href="#">Editar</a> <a href="#">Deletar</a>
8	Agenda 2021	Item dia: 2021-06-09 De: 10:00:00 Até: 11:00:00	09/06/2021	10:00:00	11:00:00	Disponível	<a href="#">Editar</a> <a href="#">Deletar</a>

© 2021 Todos os direitos reservados

Fonte: Acervo do Autor (2021).

A rotina também é filtrada pela agenda selecionada previamente e conta com as ações de inclusão, alteração e exclusão, que funcionam conforme a padronização do sistema previamente explicada. A inclusão dos itens da agenda é feita de maneira automática com base nos dias do calendário e horários do consultório.

O sistema permite a efetivação de consultas, ou seja, confirmar a consulta para que o usuário possa comparecer na data solicitada, esse processo é realizado por meio de uma tela destinada aos médicos e usuários administrativos, como pode ser observado através da Figura 24.

**Figura 24 – Efetivação de Consultas (RF13)**

Código	Usuário	Médico	Consultório	Data	Horário de Início	Horário de Término	Situação	Ações
1	Mateus Levi Carlos Figueiredo	Danilo Monteiro	Clínica Dermatológica	07/06/2021	09:00:00	10:00:00	Finalizada	Cancelar Confirmar Finalizar Contato
2	Raimundo Cavalcanti	Danilo Monteiro	Clínica Dermatológica	07/06/2021	10:00:00	11:00:00	Finalizada	Cancelar Confirmar Finalizar Contato
3	Mateus Levi Carlos Figueiredo	Danilo Monteiro	Clínica Dermatológica	08/06/2021	11:00:00	12:00:00	Aguardando	Cancelar Confirmar Finalizar Contato
4	Mateus Levi Carlos Figueiredo	Danilo Monteiro	Clínica Dermatológica	11/06/2021	10:00:00	11:00:00	Aguardando	Cancelar Confirmar Finalizar Contato

Fonte: Acervo do Autor (2021).

Além disso a tela também permite por meio de uma ação efetuar a visualização dos contatos do usuário que solicitou a consulta, a ação para finalizar uma consulta, que é quando o atendimento foi concluído e por fim caso for necessário o cancelamento da consulta. São listadas as informações de código, usuário solicitante da consulta, médico, consultório, data, horário de início, horário de término e situação na tela de consulta.

#### 4.2.2.3 Aplicação Web Médico

O nível de médico da aplicação é o mais abrangente em relação aos privilégios, pois nele possui os privilégios dos demais níveis, que permite o agendamento de consultas para si próprio como um paciente, também o restante das rotinas disponíveis no sistema, que são as rotinas de: estado, cidade, endereço, consultório, especialidade, calendário e o vínculo das especialidades do médico.

A inclusão de um médico no sistema é feita manualmente, por meio do banco de dados, ou seja, após fechar o contrato de fornecimento do sistema o médico é cadastrado e possuirá o acesso necessário para o gerenciamento de seu consultório. A rotina que contempla o cadastro do médico do sistema está listada no requisito opcional RF-30 do capítulo de análise e faz parte de implementações futuras.

A consulta de estados é uma rotina que já vem povoada por padrão pelo sistema, listando todos os estados brasileiros, as informações listadas na tela são: código, nome e UF, como pode-se observar na Figura 25.

**Figura 25 - Consulta de Estados (RF06)**

The screenshot shows a web application interface with a dark navigation bar at the top containing the text 'Catálogo de Médicos', 'Home', 'Consulta', 'Estado', 'Cidade', 'Endereço', 'Consultório', 'Especialidade', 'Médico', and 'Calendário'. On the right side of the navigation bar, the name 'Danilo Monteiro' is displayed with a dropdown arrow. Below the navigation bar, the main content area is titled 'Estados' and contains a table with the following data:

Código	Nome	UF
1	Acre	AC
2	Alagoas	AL
3	Amazonas	AM
4	Amapá	AP
5	Bahia	BA
6	Ceará	CE
7	Distrito Federal	DF
8	Espírito Santo	ES

At the bottom of the interface, there is a footer with the text '© 2021 Todos os direitos reservados'.

Fonte: Acervo do Autor (2021).

De uma forma semelhante o sistema também possui a consulta de cidades, que é povoada por padrão, listando todas as cidades brasileiras, os registros são separados por paginação, onde 50 registros são apresentados por página. Os campos listados na consulta são os de: código, nome e estado, conforme demonstrado na Figura 26.

**Figura 26 - Consulta de Cidades (RF07)**

The screenshot shows a web application interface with a dark navigation bar at the top containing the text 'Catálogo de Médicos', 'Home', 'Consulta', 'Estado', 'Cidade', 'Endereço', 'Consultório', 'Especialidade', 'Médico', and 'Calendário'. On the right side of the navigation bar, the name 'Danilo Monteiro' is displayed with a dropdown arrow. Below the navigation bar, the main content area is titled 'Cidades' and contains a table with the following data:

Código	Nome	Estado
1	Alta Floresta D'oeste	RO
2	Ariquemes	RO
3	Cabixi	RO
4	Cacoal	RO
5	Cerejeiras	RO
6	Colorado do Oeste	RO
7	Corumbiara	RO
8	Costa Marques	RO

At the bottom of the interface, there is a footer with the text '© 2021 Todos os direitos reservados'.

Fonte: Acervo do Autor (2021).

O sistema permite ao médico o cadastro de um endereço vinculado a uma cidade, a rotina consiste na consulta, inclusão, alteração e exclusão de registros, onde na consulta é listada as informações de código, logradouro, número, complemento e cidade. Para a manutenção da rotina é obrigatório o preenchimento dos campos de logradouro, número e cidade, assim como pode-se observar na Figura 27.

**Figura 27 – Cadastro de Endereço (RF21)**

A imagem mostra a interface de usuário para a funcionalidade 'Alterar Endereço'. No topo, há uma barra de navegação com links para 'Catálogo de Médicos', 'Home', 'Consulta', 'Estado', 'Cidade', 'Endereço', 'Consultório', 'Especialidade', 'Médico' (com uma seta para baixo) e 'Calendário'. No canto superior direito, o nome de usuário 'Daniilo Monteiro' é exibido com uma seta para baixo. O formulário principal, intitulado 'Alterar Endereço', contém os seguintes campos: 'Logradouro' com o texto 'Rua Vila Inconfidência, Jardim América, CEP: 89160-200'; 'Número' com o valor '54'; 'Complemento' com um campo vazio; e 'Cidade' com um menu suspenso selecionando 'Rio do Sul'. Abaixo dos campos, há um botão verde com o texto 'Alterar Endereço'. Na base da página, uma barra cinza contém o texto '© 2021 Todos os direitos reservados'.

Fonte: Acervo do Autor (2021).

O sistema permite o cadastro de especialidades para o médico, onde é disponibilizado as ações de inclusão, alteração e exclusão, conforme a padronização adotada no protótipo, a manutenção da rotina obriga apenas o campo de nome a ser preenchido e a consulta lista as informações de código e nome, conforme demonstrado através da Figura 28.

**Figura 28 - Consulta de Especialidade (RF15)**

© 2021 Todos os direitos reservados

Fonte: Acervo do Autor (2021).

Uma rotina associada a especialidade é o vínculo entre o médico e a determinada especialidade, para a rotina são disponibilizadas as ações de inclusão e exclusão, na consulta que segue a padronização do sistema, são listadas as informações de código, médico e especialidade, já para a manutenção o sistema preenche automaticamente o registro de médico com o usuário logado e obriga o preenchimento do campo de especialidade que lista todos os registros cadastrados no sistema, conforme pode-se observar através da Figura 29.

**Figura 29 – Cadastro de Médico x Especialidade (RF22)**

© 2021 Todos os direitos reservados

Fonte: Acervo do Autor (2021).

Outra rotina disponibilizada pelo sistema é o cadastro de calendários, onde é disponibilizada as ações padrão do sistema e o cadastro de itens, na manutenção é necessário informar o ano e se o calendário está ativo. Na consulta são listadas as informações de código, ano e ativo com as seguintes opções: sim e não, além disso também é disponibilizada a ação de itens, como pode se observar através da Figura 30.

**Figura 30 - Consulta de Calendário (RF16)**

Código	Ano	Ativo	Ações
1	2021	Sim	<a href="#">Editar</a> <a href="#">Deletar</a> <a href="#">Itens</a>

Fonte: Acervo do Autor (2021).

Após o cadastro do calendário o sistema disponibiliza o cadastro de seus respectivos itens, onde é disponibilizado apenas as ações de inclusão e exclusão, na consulta da rotina é disponibilizada as informações do código, calendário e data. Já para a manutenção o campo de calendário vem preenchido e desabilitado com a informação do calendário selecionada previamente e o campo de data de obrigatório preenchimento, sendo responsável por cadastrar cada dia do calendário, como pode-se observar através da Figura 31.

**Figura 31 – Cadastro de Item do Calendário (RF23)**

Fonte: Acervo do Autor (2021).

O sistema também permite para o médico o cadastro de Consultórios, onde é disponibilizado as ações padrão do sistema e as ações para o cadastro de contatos e horários. A tela de manutenção obriga o preenchimento dos campos de descrição, endereço que lista todos os registros da rotina cadastrados no sistema e informar se o consultório está ativo ou não. Já na consulta os campos de código, descrição, logradouro do endereço e ativo com as opções: sim e não são listados, conforme demonstrado através da Figura 32.

**Figura 32 - Consulta de Consultórios (RF08)**

Código	Descrição	Endereço - Código	Endereço - Logradouro	Ativo	Ações
1	Clínica Dermatológica	1	Rua Vila Inconfidência, Jardim América, CEP: 89160-200	Sim	Editar Deletar Contatos Horário

Fonte: Acervo do Autor (2021).

Após a inclusão do consultório o sistema disponibiliza a definição dos horários em que aquele consultório funcionará, a rotina conta com as ações de consulta, inclusão, alteração e exclusão. Na consulta estão listadas as informações de código, consultório, tipo, horário de início e término. Os campos necessários para a manutenção são: consultório que já vem preenchido e desabilitado conforme o registro selecionado previamente, tipo que pode ser uma consulta ou intervalo e a hora de início e término ambos sendo campos obrigatórios, conforme pode-se observar através da Figura 33.

**Figura 33 – Cadastro de Horários do Consultório (RF17)**

A imagem mostra a interface de usuário para o cadastro de horários do consultório. No topo, há uma barra de navegação com links para 'Catálogo de Médicos', 'Home', 'Consulta', 'Estado', 'Cidade', 'Endereço', 'Consultório', 'Especialidade', 'Médico' e 'Calendário'. O nome do usuário 'Danilo Monteiro' está no canto superior direito. O formulário principal, intitulado 'Incluir Horário do Consultório', contém os seguintes campos: 'Consultório' (menu suspenso com 'Clínica Dermatológica' selecionado), 'Tipo' (menu suspenso com 'Consulta' selecionado), 'Hora de Início' (campo de texto com máscara '--:--' e ícone de olho) e 'Hora de Término' (campo de texto com máscara '--:--' e ícone de olho). Um botão verde 'Criar Horário do Consultório' está localizado na base do formulário. Na barra inferior, há o texto '© 2021 Todos os direitos reservados'.

Fonte: Acervo do Autor (2021).

Outra rotina que possui vínculo com o consultório é o registro de contatos, onde é necessário o preenchimento do tipo de contato que pode ser: telefone ou e-mail, a descrição de obrigatório preenchimento e o consultório já é preenchido automaticamente com base no registro selecionado previamente, também é disposto para preenchimento se o contato é preferencial e se está ativo. As ações sobre a rotina são as padrão das demais rotinas do sistema. Na consulta como pode ser observado através da Figura 34, são listados os campos de código, tipo, descrição, preferencial, ativo e consultório.

**Figura 34 - Consulta de Contatos do Consultório (RF24)**

Contato

[Criar Contato](#)

Código	Tipo	Descrição	Preferencial	Ativo	Usuário	Consultório	Ações
1	Telefone	4799058563	Não	Sim		Clínica Dermatológica	<a href="#">Editar</a> <a href="#">Deletar</a>

© 2021 Todos os direitos reservados

Fonte: Acervo do Autor (2021).

Diante disso se conclui-se a apresentação dos detalhes sobre o desenvolvimento do protótipo, na próxima seção são apresentadas as considerações finais e recomendações para trabalhos futuros.

## 5. CONSIDERAÇÕES FINAIS

O trabalho constituiu-se no desenvolvimento de um protótipo de uma aplicação web voltada para o agendamento de consultas médicas, centralizando os médicos de um determinado município para que a população em geral consiga agendar um atendimento de acordo com o profissional que mais lhe agrade.

As tecnologias utilizadas, que incluem as ferramentas e linguagens de programação auxiliaram em todo o processo de desenvolvimento e permitiram que o protótipo pudesse ser concluído. Na parte do *backend* o Laravel dispõe de muitas facilidades para agilizar o processo de desenvolvimento com PHP, já no *frontend* com a biblioteca do Bootstrap é possível implementar diversos componentes de maneira rápida, integrado ao HTML, CSS e Javascript. Por fim o banco de dados PostgreSQL foi responsável por manter todos os dados íntegros e seguros na aplicação.

Com relação aos objetivos específicos do trabalho, primeiramente em relação ao que propunha detalhar sobre as tecnologias utilizadas para a construção do protótipo, este foi alcançado através da construção da revisão da literatura.

Em segundo lugar, referente ao objetivo de identificar ferramentas já existentes no mercado que possuem uma finalidade semelhante e explicar seu funcionamento, este foi cumprido através da seção do estado da arte onde foi demonstrado os aplicativos do BoaConsulta e MeuMedico.

O terceiro objetivo visava descrever os requisitos e elaborar diagramas, sendo que este foi realizado através da seção de análise, onde foram detalhados os requisitos funcionais, requisitos não funcionais e regras de negócio da aplicação, bem como onde foram apresentados os diagramas de caso de uso, atividade e entidade-relacionamento.

Por fim, o último objetivo específico propunha aplicar as tecnologias selecionadas para o desenvolvimento do protótipo. Este objetivo foi cumprido por meio do apresentado capítulo de desenvolvimento, onde foram apresentadas as técnicas e ferramentas utilizadas e demonstrado o funcionamento do protótipo, com a explicação de todas as telas e rotinas da aplicação.

O protótipo desenvolvido oferece a população em geral um mecanismo para que seja poupado tempo no dia a dia, pois na necessidade do agendamento de uma consulta médica, após acessar a aplicação o usuário pode ter acesso a todos os médicos cadastrados em seu município, optando pelo profissional que melhor se adequa a sua necessidade, podendo considerar ainda os mais bem avaliados.

Já para os médicos, a aplicação oferece um controle total dos horários de consultas, listando todos os usuários pendentes de confirmação da consulta e os já agendados, inclusive oferecendo a delegação de funções a uma equipe administrativa para facilitar ainda mais todo o processo.

Desta forma, pode-se concluir que os objetivos específicos propostos para o trabalho foram alcançados, bem como seu objetivo geral, contudo, considerando as limitações deste trabalho, a seguir são listadas algumas recomendações para trabalhos futuros, incluindo melhorias e novas funcionalidades para o protótipo.

## 5.1 RECOMENDAÇÕES DE TRABALHOS FUTUROS

Para a continuidade deste trabalho a primeira recomendação seria implementar os requisitos opcionais (Quadro 10), descritos no capítulo de análise. A primeira seria o envio de e-mails para que o usuário seja informado por outros meios quando a consulta foi efetivada e como forma de lembrete do horário e data da consulta. A segunda implementação seria permitir ao médico o cadastro de planos de saúde no sistema. Outro ponto seria o usuário vincular os planos de saúde que ele possui. O terceiro seria permitir ao médico reservar horários do dia, específicos para consulta de planos de saúde, deixando assim o gerenciamento mais fácil para o profissional que opte por trabalhar com essa metodologia.

O quinto requisito opcional trata do desenvolvimento de uma tela que listasse todos os preços e planos que o médico teria disponibilidade de contratar, informando todas as vantagens de cada plano. Neste caso, trata-se do médico aderindo ao sistema de Catálogo de Médicos. Outro requisito atrelado a este seria o sistema disponibilizar o cadastro do médico através de uma rotina do sistema. E como sétimo requisito opcional a aplicação contaria com um sistema de pagamentos, tendo diversas formas de pagamentos implementadas e faria a cobrança mensal, semestral ou anual para os médicos, assim o sistema permanecendo ativo para o profissional. Para o oitavo requisito opcional, seria necessário a implementação de inclusão de múltiplos itens no calendário, onde o usuário informaria um intervalo de datas e o sistema incluiria os dias úteis entre o intervalo selecionado como itens do respectivo calendário. O nono e décimo requisito opcional tratam da inclusão de usuários e agendamento de consultas pelos usuários administrativos e médicos do sistema, ou seja, estes dois níveis de usuário conseguiriam fazer o agendamento de consultas para si próprios e para os demais usuários clientes do sistema.

No âmbito da homologação do sistema, seria recomendado a implementação em um cliente piloto, que serviria para validar as regras de negócio implementadas e a usabilidade do sistema. Após a homologação e validação o sistema poderia entrar em operação.

Uma última recomendação seria a realização de integrações com API's do Google e Facebook para a realização de login no sistema, tornando o acesso mais fácil por parte do usuário pois menos dados seriam necessários para preenchimento no cadastro de usuário. Também seria interessante integrar a agenda do Google com a aplicação, onde ao confirmar uma consulta, o sistema criaria um compromisso na agenda pessoal do usuário.

## REFERÊNCIAS

- ALVES, William Pereira. **Banco de dados**. São Paulo Erica 2014a. *Ebook*
- ALVES, William Pereira. **Desenvolvimento e design de sites**. São Paulo: Érica, 2014b. *Ebook*.
- BALDUINO, Plínio. **Dominando Javascript com jQuery**. São Paulo Casa do Código 2012. *Ebook*.
- BOACONSULTA. **Conheça o BoaConsulta**. 2021. Disponível em: <https://www.boaconsulta.com/sobre-nos>. Acesso em: 22 ago. 2021.
- BOOTSTRAP. **Build fast, responsive sites with Bootstrap**. 2021. Disponível em: <https://getbootstrap.com/>. Acesso em: 20 nov. 2021.
- BRASIL. **Princípios da LGPD**. 2021a. Disponível em: <https://www.gov.br/cidadania/pt-br/aceso-a-informacao/lgpd/principios-da-lgpd>. Acesso em: 12 dez. 2021
- BRASIL. **Lei Geral de Proteção de Dados Pessoais (LGPD)**. 2021b. Disponível em: <https://www.gov.br/cidadania/pt-br/aceso-a-informacao/lgpd/principios-da-lgpd>. Acesso em: 12 dez. 2021.
- FLANAGAN, David. **JavaScript: o guia definitivo**. Porto Alegre Bookman 2014. *Ebook*
- HIRAMA, Kechi. **Engenharia de Software: Qualidade e Produtividade com Tecnologia**. Rio de Janeiro: Grupo GEN, 2011. *Ebook*
- LARAVEL. **Why Laravel?** 2021. Disponível em: <https://laravel.com/docs/8.x>. Acesso em: 20 jun. 2021
- MEUMEDICO. **O MeuMedico e a medicina digital**. 2020. Disponível em: <https://www.meumedico.org/como-funciona>. Acesso em: 22 ago. 2021.
- MILETTO, Evandro Manara. **Desenvolvimento de software ii: introdução ao desenvolvimento web com html, css, javascript e php**. Porto Alegre Bookman 2014. *Ebook*
- NEGRI, Fernanda. **As tecnologias da informação podem revolucionar o cuidado com a Saúde?** 2019. Disponível em: <https://www.ipea.gov.br/cts/pt/central-de-conteudo/artigos/artigos/107-as-tecnologias-da-informacao-podem-revolucionar-o-cuidado-com-a-saude>. Acesso em: 15 maio 2021.
- NEVES, Denise Lemes Fernandes. **PostgreSQL: conceitos e aplicações**. São Paulo: Érica, 2002.
- OLIVEIRA, Júlio. **A importância e o desafio da boa gestão na saúde pública**. 2018. Disponível em: <http://idisa.org.br/noticias/a-importancia-e-o-desafio-da-boa-gestao-na-saude-publica?lang=pt>. Acesso em: 01 jun. 2021.
- PÁDUA, P.F.W. D. **Engenharia de Software - Projetos e Processos - Vol. 2**. Rio de Janeiro: Grupo GEN, 2019. *Ebook*

PRESSMAN, Roger S. **Engenharia de software: uma abordagem profissional**. Porto Alegre: AMGH, 2016. *Ebook*.

PRESSMAN, Roger S. **Engenharia de software**. São Paulo: Makron Books, 1995. 1056 p.

TERUEL, Evandro Carlos. **HTML 5: Guia Prático**. São Paulo: Érica, 2014. *Ebook*.

TERADA, Routo. **Segurança de dados: criptografia em rede de computador**. São Paulo: Blucher 2008. *Ebook*.

WELLING, Luke; THOMSOM Laura. **PHP e MySQL: Desenvolvimento Web**. Rio de Janeiro: Elsevier Editora 2005. *Ebook*.