



UEPB

**UNIVERSIDADE ESTADUAL DA PARAÍBA
CAMPUS I
CENTRO DE CIÊNCIAS E TECNOLOGIA
DEPARTAMENTO DE COMPUTAÇÃO
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

JOANES MIRANDA DE SOUTO

**SISTEMA DE INFORMAÇÃO PARA GERENCIAMENTO DE CONSULTAS MÉDICAS
PARA AS UNIDADES BÁSICAS DE SAÚDE DA CIDADE DE CAMPINA GRANDE**

**CAMPINA GRANDE
2019**

JOANES MIRANDA DE SOUTO

SISTEMA DE INFORMAÇÃO PARA GERENCIAMENTO DE CONSULTAS MÉDICAS
PARA AS UNIDADES BÁSICAS DE SAÚDE DA CIDADE DE CAMPINA GRANDE

Trabalho de Conclusão de Curso de Graduação em Ciência da Computação da Universidade Estadual da Paraíba, como requisito à obtenção do título de bacharel em Ciência da Computação.

Área de concentração: Engenharia de Software.

Orientador: Prof. Dr. Vladimir Costa de Alencar.

CAMPINA GRANDE
2019

É expressamente proibido a comercialização deste documento, tanto na forma impressa como eletrônica. Sua reprodução total ou parcial é permitida exclusivamente para fins acadêmicos e científicos, desde que na reprodução figure a identificação do autor, título, instituição e ano do trabalho.

S728s Souto, Joanes Miranda de.
Sistema de informação para gerenciamento de consultas
médicas para as Unidades Básicas de Saúde da cidade de
Campina Grande [manuscrito] / Joanes Miranda de Souto. -
2019.
54 p. : il. colorido.
Digitado.
Trabalho de Conclusão de Curso (Graduação em
Computação) - Universidade Estadual da Paraíba, Centro de
Ciências e Tecnologia, 2019.
"Orientação : Prof. Dr. Vladimir Costa de Alencar ,
Coordenação do Curso de Computação - CCT."
1. Sistema de informação. 2. Linguagem PHP. 3. Sistema
Único de Saúde - SUS. I. Título

21. ed. CDD 005.13

JOANES MIRANDA DE SOUTO

**SISTEMA DE INFORMAÇÃO PARA GERENCIAMENTO DE
CONSULTAS MÉDICAS PARA AS UNIDADES BÁSICAS DE
SAÚDE DA CIDADE DE CAMPINA GRANDE**

Trabalho de Conclusão de Curso de Graduação
em Ciência da Computação da Universidade
Estadual da Paraíba, como requisito à
obtenção do título de Bacharel em Ciência da
Computação.

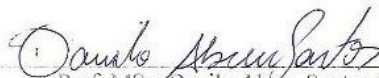
Aprovada em 6 de Dezembro de 2019.



Prof. Dr. Vladimir Costa de Alencar (DC - UEPB)
Orientador(a)



Prof. Dr. Alysson Filgueira Milanez (DC - UEPB)
Examinador(a)



Prof. MSc. Danilo Abreu Santos (DC - UEPB)
Examinador(a)

À Deus, meus familiares e amigos, pela
dedicação, apoio, companheirismo e amizade,
DEDICO.

AGRADECIMENTOS

À Deus por ter me dado saúde, força, perseverança e sabedoria para concluir este trabalho.

À Universidade Estadual da Paraíba por me conceder a oportunidade e privilégio de cursar e concluir o curso de Bacharelado em Ciência da Computação.

Ao orientador e Prof. Dr. Vladimir Costa de Alencar por ter me proporcionado oportunidades únicas na caminhada acadêmica, como também pela sua confiança, companheirismo e disponibilização das ferramentas necessárias para realização deste trabalho.

Por fim, mas não menos importante, agradeço a toda minha família, minha mãe, Maria José Miranda, ao meu pai, Teotônio Luiz de Souto, a minha esposa Joseane da Silva Miranda e ao amigo de longa data Douglas Rafael Oliveira dos Santos pelo apoio. Sem eles eu não teria a base necessária para concluir essa jornada.

RESUMO

O cuidado primário da saúde é o principal meio de tratamento da população mundial, porém existem déficits quanto ao acesso dos usuários aos postos de atendimento, principalmente em se tratando de atendimento público. No Brasil, o acesso a esses serviços é através do Sistema Único de Saúde – SUS, que atende a todos de forma integralizada com o intuito de promover a saúde e bem-estar populacional. O gerenciamento quanto a este atendimento possui lacunas que deixam a prestação do serviço público de saúde bastante precária, o que causa a insatisfação dos seus usuários que dependem únicos e exclusivamente deste atendimento. Diante disso, o presente trabalho tem como objetivo implementar o AgendaUBS, um sistema de informação para gerenciamento de consultas médicas para as unidades básicas de saúde da cidade de Campina Grande, promovendo um maior conforto aos seus usuários e profissionais de saúde nos itens com o maior índice problemático, como o controle de horários, registros de pacientes, controle de retorno de consultas, entre outros. O sistema foi desenvolvido utilizando a linguagem PHP, o sistema gerenciador de banco de dados MySQL para armazenar informações, GIT para o controle de versões e para disponibilizar o site na Internet foi utilizado o Heroku, que é uma plataforma de *cloud* que permite hospedar suas aplicações em um ambiente facilmente escalável e com suporte a várias tecnologias. Esse sistema permite aos usuários de maneira geral que consigam marcar, reagendar e visualizar consultas com maior comodidade, gasto mínimo de tempo e sem maiores transtornos. Além de promover maior interação e controle entre médico-paciente.

Palavras-Chave: SUS. Sistema de Informação. Cuidado Primário.

ABSTRACT

Primary health care is the main means of treatment of the world's population, but there are deficits in the access of users to the service centers, especially in the case of public care. In Brazil, the Unified Health System - SUS is used, which assists everyone in an integrated way with the aim of promoting health and population well-being. The management of this service has gaps that leave the public health service quite precarious, which causes the dissatisfaction of its users that depend solely and exclusively on this service. Given this, the present work aims to implement AgendaUBS, an information system for the management of medical consultations for the basic health units of the city of Campina Grande, promoting greater comfort to its users and health professionals in the items with the highest problem index, such as the control of schedules, patient records, control of return of consultations, among others. The system was developed using PHP language, MySQL database to store information, GIT for version control and to make the site available on the network was used Heroku a cloud platform that allows to host your applications in an easily scalable environment and with support for various technologies. This system allows users in general to be able to book, reschedule and view appointments with greater convenience, minimum time wasting and no hassle. In addition to promoting greater interaction and control between doctor-patient.

Keywords: SUS. Information System. Primary Care.

LISTA DE ILUSTRAÇÕES

Figura 1 – Arquitetura MVC.....	20
Figura 2 – Ferramenta MySQL Workbench.....	22
Figura 3 – Diagrama de casos de usos.....	23
Figura 4 – IDE PhpStorm.....	24
Figura 5 – Diagrama entidade relacionamento.....	26
Figura 6 – Validação JavaScript do campo de formulário de login.....	27
Figura 7 – Resultado da validação JavaScript no Front-End.....	28
Figura 8 – Função do PHP para autenticação de acesso do usuário no sistema.....	28
Figura 9 – Função do PHP para validação dos dados de login do usuário.....	29
Figura 10 – Código de verificação de e-mail logado na sessão.....	29
Figura 11 – Página de detalhes do GitLab.....	30
Figura 12 – Tela de acesso ao sistema.....	31
Figura 13 – Tela de login – escolhendo o tipo de usuário.....	32
Figura 14 – Tela de cadastro de novo usuário.....	32
Figura 15 – Tela de recuperação de senha.....	33
Figura 16 – Tela de boas-vindas do médico.....	34
Figura 17 – Tela de realização de consulta com paciente.....	36
Figura 18 – Tela de pesquisa sobre códigos de doenças (CID10).....	37
Figura 19 – Resultado da pesquisa por doença.....	37
Figura 20 – Tela de boas-vindas do atendente.....	38
Figura 21 – Tela de cadastro de médico.....	39
Figura 22 – Tela de criação de horário do atendimento médico.....	40
Figura 23 – Tela de unidades de saúde do atendente.....	41
Figura 24 – Tela de marcação de consulta do atendente.....	41
Figura 25 – Tela de escolha de horário da consulta do atendente.....	42
Figura 26 – Tela de escolha de médico com consulta agendada do atendente.....	42
Figura 27 – Tela de visualização de consultas agendadas.....	43
Figura 28 – Comprovante de consulta agendada pelo atendente.....	43
Figura 29 – Tela de boas-vindas do paciente.....	44
Figura 30 – Tela de listagem das unidades de saúde do paciente.....	45
Figura 31 – Tela de marcação de consulta do paciente.....	45

Figura 32 – Tela de escolha de horário da consulta do paciente.....	46
Figura 33 – Tela de escolha de horário com consulta marcada do paciente.....	46
Figura 34 – Tela de visualização de consultas agendadas do paciente.....	47
Figura 35 – Comprovante de consulta agendada pelo paciente.....	47

LISTA DE TABELAS

Tabela 1 – Comandos utilizados no GitLab.....	30
---	----

LISTA DE QUADROS

Quadro 1 – Campos do formulário de identificação do usuário.....	34
Quadro 2 – Campos do formulário de condições de saúde do paciente.....	35
Quadro 3 – Campos do formulário de classificação internacional de doenças.....	36
Quadro 4 – Campos do formulário de exames.....	36
Quadro 5 – Campos do formulário de prescrição de medicamentos.....	36

LISTA DE ABREVIATURAS E SIGLAS

CID10	Classificação Internacional de Doenças
CRUD	<i>Create, Read, Update e Delete</i>
HTML	<i>HyperText Markup Language</i>
HTTP	<i>Hypertext Transfer Protocol</i>
IDE	<i>Integrated Development Environment</i>
MVC	<i>Model, View, Controller</i>
PHP	<i>Personal Home Page</i>
RF	Requisito Funcional
RN	Requisito não Funcional
SUS	Sistema Único de Saúde
SGBD	Sistema de Gerenciamento de Banco de Dados
UBS	Unidade Básica de Saúde
UBSF	Unidade Básica de Saúde da Família
URL	<i>Uniform Resource Locator</i>

SUMÁRIO

1	INTRODUÇÃO.....	13
2	JUSTIFICATIVA.....	16
3	OBJETIVOS.....	17
3.1	Objetivo geral.....	17
3.2	Objetivos específicos.....	17
4	FUNDAMENTAÇÃO TEÓRICA.....	18
4.1	A linguagem PHP.....	18
4.2	O Sistema Gerenciador de Banco de Dados (SGBD) MySQL.....	18
4.3	O padrão de projetos MVC.....	19
4.4	O ambiente de deploy Heroku.....	20
5	METODOLOGIA.....	21
5.1	Modelo entidade relacionamento.....	21
5.2	Diagramas de casos de uso.....	22
5.3	IDE PhpStorm.....	24
6	RESULTADOS E DISCUSSÕES.....	25
6.1	Ferramentas utilizadas e trechos de códigos.....	25
6.1.1	<i>Banco de dados do sistema.....</i>	25
6.1.2	<i>Segurança do sistema.....</i>	26
6.1.3	<i>O uso do GitLab para controle de versões.....</i>	29
6.2	Funcionalidades do Sistema.....	31
7	CONCLUSÃO.....	48
	REFERÊNCIAS.....	49
	APÊNDICE A – REQUISITOS FUNCIONAIS DO SISTEMA.....	52
	APÊNDICE B – REQUISITOS NÃO FUNCIONAIS DO SISTEMA.....	54

1 INTRODUÇÃO

O Sistema Único de Saúde (SUS) é um dos maiores e mais complexos sistemas de saúde pública do mundo, abrangendo desde o simples atendimento para avaliação da pressão arterial, por meio da Atenção Básica, até o transplante de órgãos, garantindo acesso integral, universal e gratuito para toda a população do país. Com a sua criação, o SUS proporcionou o acesso universal ao sistema público de saúde, sem discriminação. A atenção integral à saúde, e não somente aos cuidados assistenciais, passou a ser um direito de todos os brasileiros, desde a gestação e por toda a vida, com foco na saúde com qualidade de vida, visando a prevenção e a promoção da saúde (MINISTERIO DA SAÚDE, 2018).

Ao analisar o sistema de saúde no Brasil percebe-se que um dos grandes desafios tem sido a redefinição, ao longo do tempo, das atribuições e das competências dos gestores das três esferas de governo. Para enfrentá-lo têm sido elaborados muitos dispositivos normativos demarcando os limites da tomada de decisão de cada gestor no seu âmbito de atuação, resultando na conformação de um modelo de gestão que tem como ponto de partida uma unidade de princípios, mas que tem de atuar de forma coerente com a diversidade operativa nos territórios em que está localizada a população com suas necessidades de saúde (CONASS, 2015). A insuficiência da infraestrutura pública, a falta de planejamento ascendente, as dificuldades com a montagem de redes na regionalização e os impasses para a mudança dos modelos de atenção e das práticas de saúde também comprometem o acesso universal e igualitário às ações e serviços de saúde (PAIM, 2018).

O SUS foi instituído pela a Lei 8.080/90, que descreve a saúde como direito fundamental do ser humano, devendo o Estado prover as condições indispensáveis ao seu pleno exercício, assegurando o acesso igualitário e universal ao sistema de saúde, visando a redução de riscos de doenças e de outros agravos. Para facilitar o acesso da população ao serviço de saúde é necessário implementar uma gestão de planejamento que permita a eficácia do atendimento de forma igualitária. No Sistema Único de Saúde, é utilizado como estratégia de atendimento o sistema de referencialização, que permite tratar cada tipo de enfermidade em um local de referência para o serviço que requer determinada especialidade. A lei ainda afirma que a porta de entrada ideal do cidadão na rede de saúde é pela atenção básica (postos de saúde, equipes de Saúde da Família, etc.) (MINISTÉRIO DA SAÚDE, 2006).

Uma característica fundamental de um sistema de saúde primário forte é o acesso aos cuidados. Dois elementos de um sistema de saúde forte são acessibilidade e tempos de espera curtos para serviços (FOURNIER et al, 2012). A facilidade de acesso propicia a satisfação do

usuário com o atendimento, determinando a escolha do serviço de saúde e estabelecendo, frequentemente, um bom vínculo (DALL' AGNOL et al, 2009).

A gestão no âmbito da saúde exige lidar com problemas de alta complexidade diariamente, assim como a adoção de medidas de alta relevância social. Dessa maneira, a informação deve ser tomada como um redutor de incertezas e possibilitar um planejamento mais próximo das necessidades de saúde para atingir uma situação futura desejada (GARCIA, P. T. et al, 2016). Prover cuidados com qualidade é um dos principais objetivos dos sistemas de saúde, mas a intenção nem sempre é suficiente. Equilibrar a demanda com capacidade ainda parece ser um sério problema a ser enfrentado em relação ao acesso na Atenção Primária à Saúde (ROCHA, S. A. et al, 2016). A oferta insuficiente ou inexistente de consultas em muitas especialidades faz com que os gestores encontrem muita dificuldade para garantir o acesso ao atendimento especializado aos seus municípios (SILVA, C.R. et al, 2016).

A atenção básica à saúde desempenha um papel estratégico no SUS, sendo o principal elo entre o sistema de saúde e a população. Mas não pode ser entendida apenas como porta de entrada do sistema, porque essa ideia caracteriza incapacidade de resolver problemas e implica desqualificação e isolamento. Não pode ser a única porta de entrada, porque as necessidades das pessoas se manifestam de maneira variável e precisam ser acolhidas. Nem pode ser porta obrigatória, porque isso burocratiza a relação das pessoas com o sistema. E não pode ser o único lugar de acolhimento, porque todas as partes do sistema precisam se responsabilizar pelo resultado das ações de saúde e pela vida das pessoas (VILELA, S. M. P., 2010).

Possíveis soluções para melhorar o acesso para os pacientes são numerosas e incluem a implementação de modelos de prática familiar que ofereçam cuidados abrangentes e contínuos com uma abordagem inter profissional e colaborativa. Esses modelos podem incluir maior acessibilidade aos cuidados pós-expediente, esforços para reduzir o tempo de espera por serviços de diagnóstico e modelos de remuneração que incentivem uma abordagem holística da prestação de cuidados de saúde primários (FOURNIER et al, 2012). Porém, esses modelos por muitas vezes, precisam de uma maior disponibilidade financeira para cumprir todas ou a maioria das necessidades dos usuários do SUS e acaba se tornando uma nova problemática para a gestão. Criar soluções para um sistema que se encontra com o uso de recursos limitados e poucos profissionais atuando na área somados a um grande público necessitando de atendimento, requer um enorme esforço e tempo para conseguir suprir essa demanda.

Os temas tecnologia e inovação tecnológica estão em pauta nos meios de comunicação e nas agendas de governos, empresas, agências de fomento a pesquisas e diversas organizações sociais, com forte influência no setor saúde (LORENZETTI, J. et al, 2012). A tipologia que aplica à tecnologia reitera e renova o valor da Teoria do Processo de Trabalho na compreensão das práticas de saúde (AYRES, J. R. C. M., 2000). Mais do que um desafio político, econômico ou administrativo, trata-se aqui também de um desafio tecnológico, de arranjar, criar e recriar ações de modo a produzir, de modo universal e equitativo, não apenas tratamento, prevenção ou recuperação da saúde, mas, por meio de todos e cada um destes recursos, produzir cuidado (AYRES, J. R. C. M., 2009).

A garantia de acesso ao serviço de saúde para os usuários representa a responsabilidade do serviço para com suas necessidades de saúde. Sem o acolhimento, o serviço não poderá garantir nem o acesso nem as prioridades de atendimento. Quando o usuário sente a necessidade de atendimento mais rápido, muitas vezes esta não está expressa fisicamente. Neste caso, somente com a escuta desse usuário é que se poderá saber de suas necessidades: Ter atendimento sem desmarcar e de qualidade. Com isso, deveria ter um modo de atender mais rápido as pessoas, pois quem marca a consulta deve ser atendido no dia marcado e não dois meses depois (COELHO, M. O. et al, 2009).

O meio de comunicação mais utilizado para agendamento em estabelecimentos de consultas privados é a telefonia. As outras opções são: agendar o procedimento pessoalmente e, recentemente, através da internet (LIMA, H. R. C., 2002). Porém, o meio mais comum de agendamento em unidades públicas é pessoalmente. Recentemente, a tecnologia vem trazendo uma nova perspectiva à marcação de consulta com o intuito de melhorar o atendimento, tornando este procedimento antes visto como desconfortante em um procedimento rápido com apenas um toque.

O que este trabalho propõe é seguir esta perspectiva de inovações tecnológicas tornando-a útil ao ramo que mais cresce e também o que mais necessita de atenção, que é o sistema de saúde. O desgaste físico e o estresse que é provocado aos usuários na tentativa de conseguir uma ficha de atendimento, seja em um posto de saúde ou UBSF, poderá ser minimizado ou extinto de maneira que o mesmo terá mais conforto ao poder marcar uma consulta ou exame de qualquer dispositivo em sua residência ou aonde estiver.

2 JUSTIFICATIVA

O presente trabalho foi elaborado visando expor as problemáticas do SUS em sua gestão, bem como suas dificuldades para atender a demanda contínua dos usuários que precisam de atendimento médico. Neste trabalho, o foco foi voltado para duas vertentes, que são a dificuldade de acesso da população ao serviço de saúde e a demora no atendimento resultante dessa alta demanda que, conseqüentemente, acaba gerando filas intermináveis.

A fila de espera é uma lista de pacientes que necessitam de um mesmo tratamento ou serviço médico cuja demanda é maior que a oferta. Metaforicamente, os pacientes na fila habitam uma sala de espera virtual, aguardando um mesmo procedimento, sendo chamados um por vez, em uma determinada ordem. A fila de espera para cirurgias eletivas é uma realidade em muitos hospitais gerais do país, com nuances regionais quanto aos procedimentos com fila maior ou menor, seja medida em número de pacientes ou em tempo de espera (SARMENTO JR et al, 2005).

Quem depende do SUS para realizar uma consulta ou exame sabe como a espera pode ser longa. Com equipes formadas por um número insuficiente de profissionais, que precisam trabalhar sobrecarregados para atender a demanda, a qualidade e a rapidez do atendimento ficam comprometidas (SIMERS, 2017). Uma forma de confirmar esses problemas é observar a marcação de consulta nos postos de saúde, as filas que são formadas diariamente por pessoas que saem de suas residências na madrugada para garantir o seu direito a consulta médica. Tudo isso causa um grande desconforto e insatisfação aos usuários do SUS e muitos deles como não possuem recursos financeiros para obter um plano de saúde ou até mesmo pagar uma consulta particular, permanecem sujeitos a este modelo de atendimento.

Diante disso, fica clara a necessidade de criar melhorias para o Sistema Único de Saúde (SUS) que visem facilitar o acesso ao atendimento e assim proporcionar a população maior conforto, tanto físico quanto psicológico, como também a satisfação de ser usuário de um sistema gratuito que funcione e que lhe traga benefícios.

3 OBJETIVOS

3.1 Objetivo Geral

O objetivo deste trabalho é construir uma ferramenta de apoio que ajude a atenção básica à saúde, de forma fácil e dinâmica, que solucione ou diminua os problemas de gerenciamento das unidades de atendimento, tornando o sistema padronizado e funcional, possibilitando ao usuário conseguir atendimento médico.

3.2 Objetivos Específicos

- Fornecer uma alternativa dinâmica à população favorecendo seu acesso a consultas médicas;
- Proporcionar um sistema de marcação de consultas que atenda às necessidades da população;
- Intervir de forma positiva no tocante ao controle dos profissionais de saúde à agendamento de consultas realizada pelos usuários.

4 FUNDAMENTAÇÃO TEÓRICA

Nesta seção será abordada uma breve introdução às tecnologias de desenvolvimento utilizadas no projeto como forma de simplificar o entendimento de como o sistema foi desenvolvido, na seção 4.1 foi realizado uma introdução à linguagem de programação PHP, na seção 4.2 foi mostrado o sistema gerenciador de banco de dados utilizado para guardar as informações dos usuários, na seção 4.3 foi mostrado o padrão MVC e seus benefícios ao desenvolvimento de um projeto, na seção 4.4 foi apresentado o ambiente de deploy Heroku.

4.1 A linguagem de programação PHP

PHP é a abreviação de “*Hypertext Preprocessor*” ou em tradução livre “Pré-processador de Hipertexto”. É uma linguagem de script de código aberto que tem como objetivo primário a geração de conteúdo dinâmico para páginas da Internet. Isto quer dizer que, ao invés de criar um programa para gerar e imprimir HTML, você pode escrever HTML com o código PHP embutido para gerar conteúdo dinâmico. Como as tags HTML são estáticas, cabe ao PHP ou outra linguagem como ASP ou JavaScript a criação do conteúdo dinâmico que vemos na web. Outra vantagem do PHP é que, por ser executado no lado do servidor, seu código-fonte não é exibido ao internauta, que apenas terá acesso ao seu HTML resultante (MELO, A.A de, NASCIMENTO, M G.F, 2007).

O PHP é uma das linguagens mais utilizadas na web. Milhões de sites no mundo inteiro utilizam PHP. A principal diferença em relação às outras linguagens é a capacidade que o PHP tem de interagir com o mundo web, transformando totalmente os websites que possuem páginas estáticas (NIEDERAUER, J. 2011).

4.2 O Sistema Gerenciador de Banco de Dados (SGBD) MySQL

O MySQL é um sistema gerenciador de banco de dados estruturado de código aberto conhecido por causa de seu desempenho, facilidade de uso e confiabilidade. Esta é a escolha mais comum de aplicativos da web para um banco de dados relacional. No mercado atual, milhares de aplicativos baseados na web usam MySQL incluindo indústrias gigantes como Facebook, Twitter e Wikipédia. Também provou ser a escolha do banco de dados para aplicativos baseados em Software como Serviço (SaaS), tais como Twitter, YouTube,

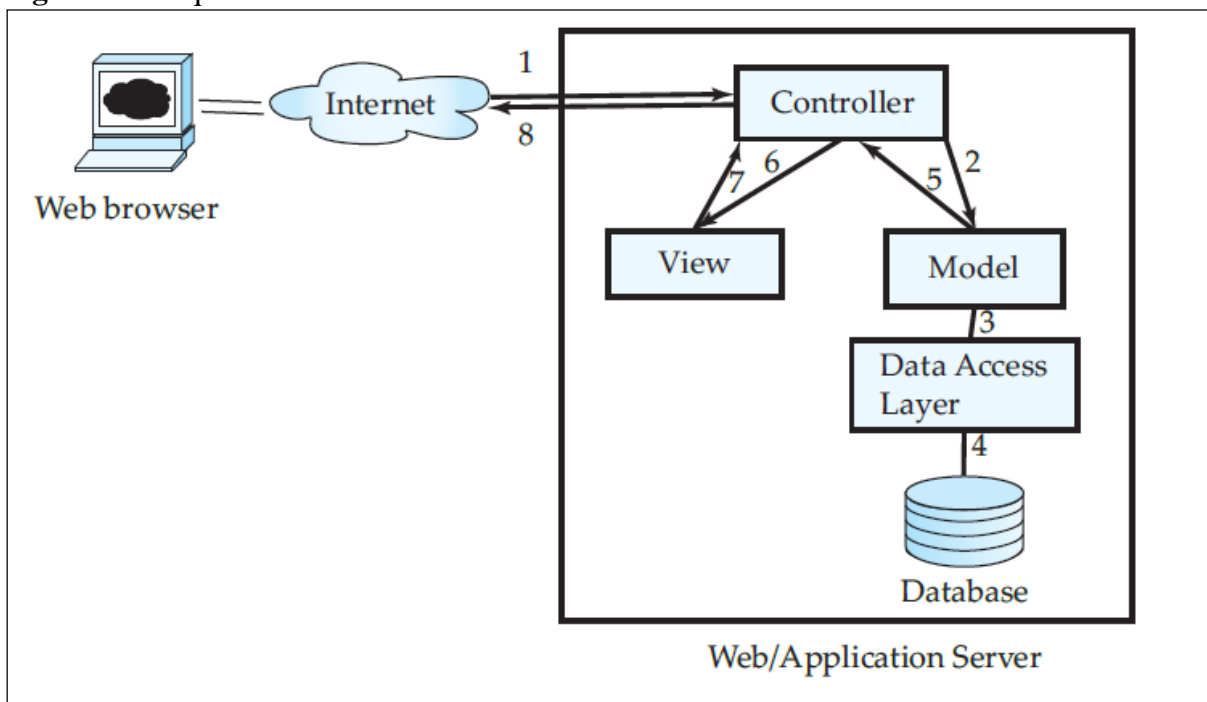
SugarCRM, Dinâmica de Suprimentos, Workday, RightNow, Omniture, Zimbra e muito mais (MEHTA, C. et al 2018).

Uma razão para seu sucesso deve ser que, assim como o PHP, é grátis para usar, mas também é extremamente poderoso e excepcionalmente rápido, pode funcionar até mesmo nos mais básicos hardwares, e isso dificilmente afeta os recursos do sistema (NIXON, R. 2018).

De acordo com (Silberschatz, A. 2010), um sistema de gerenciamento de banco de dados (SGBD) é uma coleção de dados inter-relacionados e um conjunto de programas para acessar esses dados. A coleção de dados, geralmente chamada de banco de dados, contém informações relevantes para uma empresa. O autor ainda fala que o objetivo principal de um SGBD é fornecer uma maneira de armazenar e recuperar informações do banco de dados que sejam convenientes e eficientes. Os sistemas de banco de dados são projetados para gerenciar grandes corpos de informações, o gerenciamento de dados envolve a definição de estruturas para armazenamento de informações e o fornecimento de mecanismos para a manipulação de informações, além disso, o sistema gerenciador de banco de dados deve garantir a segurança das informações armazenadas, apesar de falhas no sistema ou tentativas de acesso não autorizado.

4.3 O padrão de projeto MVC

O MVC (*Model-View-Controller*) é um padrão de design de software constituído de três tipos de componentes que são chamados de modelo, visão e controle. O modelo contém toda a lógica de negócios de uma aplicação. A visão é onde todos os elementos da interface do usuário são mantidos, pode incluir códigos HTML, folhas de estilo CSS e arquivos JavaScript. Tudo o que um usuário vê ou interage pode ser mantido na visão. O controle é o componente que conecta modelos e visualizações juntos, os controles isolam a lógica de negócio de um modelo a partir dos elementos da interface do usuário de uma exibição e manipula como o aplicativo responderá a interação do usuário com a visão. Controles são o primeiro ponto de entrada neste trio de componentes, porque o primeiro pedido é passado para um controlador, que instanciará os modelos e visualizações necessários para atender a uma solicitação da aplicação (PITT, 2012).

Figura 1 – Arquitetura MVC

Fonte: (Silberschatz, A. 2010).

4.4 O ambiente de deploy Heroku

Heroku é uma plataforma de desenvolvimento e fornecimento de aplicativos baseada em nuvem, ajuda os desenvolvedores na criação de aplicativos, removendo a necessidade de configuração e manutenção de servidores. A plataforma Heroku possui uma arquitetura construída em máquinas virtuais no Amazon EC2, seu gerenciamento e dimensionamento de aplicativos são feitos por meio de uma interface de linha de comando ou online e em dispositivos móveis, por meio de uma API. Os aplicativos criados no Heroku se beneficiam da arquitetura gerenciada de pilha e autocorreção, garantindo que eles não exijam que os administradores de sistemas gerenciem a plataforma subjacente para mantê-los seguros e confiáveis (KEMP, C.; GYGER, B. 2013).

O Heroku foi a plataforma de cloud utilizada no projeto para deploy da aplicação e disponibilidade de acesso para os usuários, pois permite uma facilidade de manuseio utilizando o GIT, gerenciamento através de interfaces gráficas, compatibilidade com vários SGBD como, por exemplo, o MySQL, além disso, permite o deploy de até três aplicações de forma gratuita.

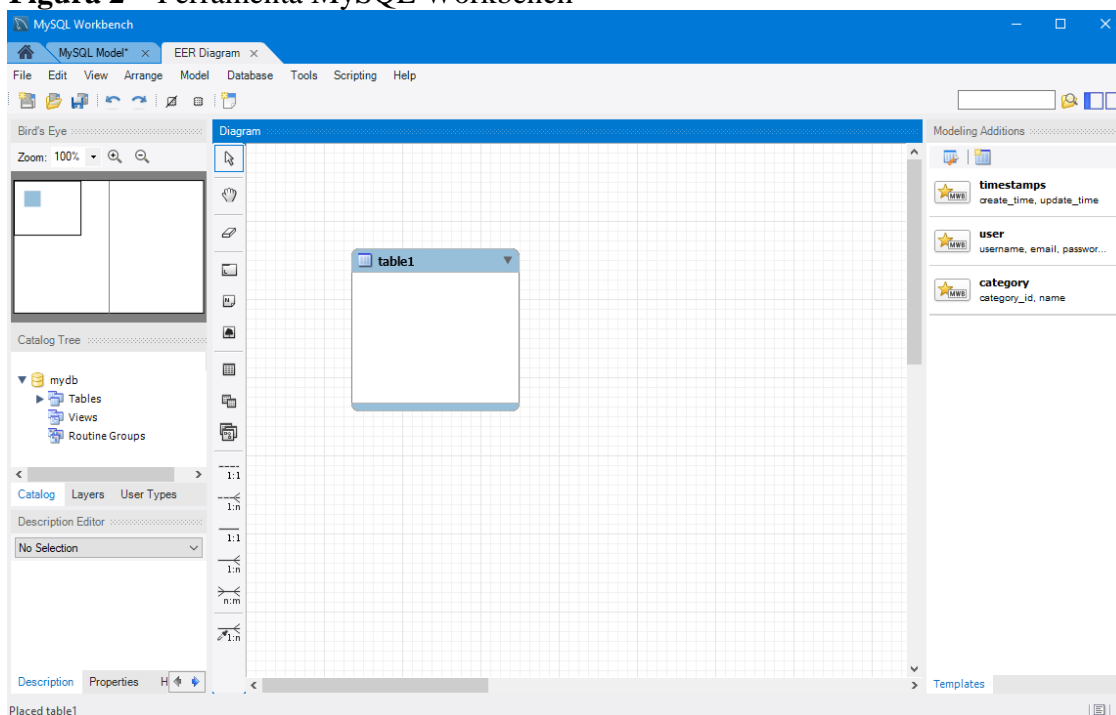
5 METODOLOGIA

A pesquisa pode ser classificada como exploratória, pois antes do processo de desenvolvimento do sistema web, foi necessário o estudo bibliográfico detalhado do panorama da saúde pública no Brasil, das tecnologias de desenvolvimento web como linguagem de Back-End PHP e JavaScript, do SGBD MySQL e da ferramenta visual MySQL Workbench para desenvolvimento do banco de dados, além da exploração do IDE PhpStorm, o controle de versão de cada etapa da codificação usando o GIT e deploy do produto final para uma plataforma de cloud, o Heroku.

5.1 O Modelo entidade relacionamento

Utilizando a ferramenta MySQL Workbench (Figura 2) foi criado o modelo MER, mostrando os relacionamentos utilizados no sistema e suas entidades. O modelo de dados de relacionamento de entidade (E-R) foi desenvolvido para facilitar o design do banco de dados, permitindo a especificação de um esquema corporativo que representa a estrutura lógica geral de um banco de dados.

O modelo E-R é muito útil no mapeamento dos significados e interações das empresas do mundo real em um esquema conceitual, devido a essa utilidade, muitas ferramentas de design de banco de dados se baseiam nos conceitos do modelo E-R, esse modelo E-R emprega três conceitos básicos: conjuntos de entidades, conjuntos de relacionamentos e atributos (Silberschatz, A. et al, 2010).

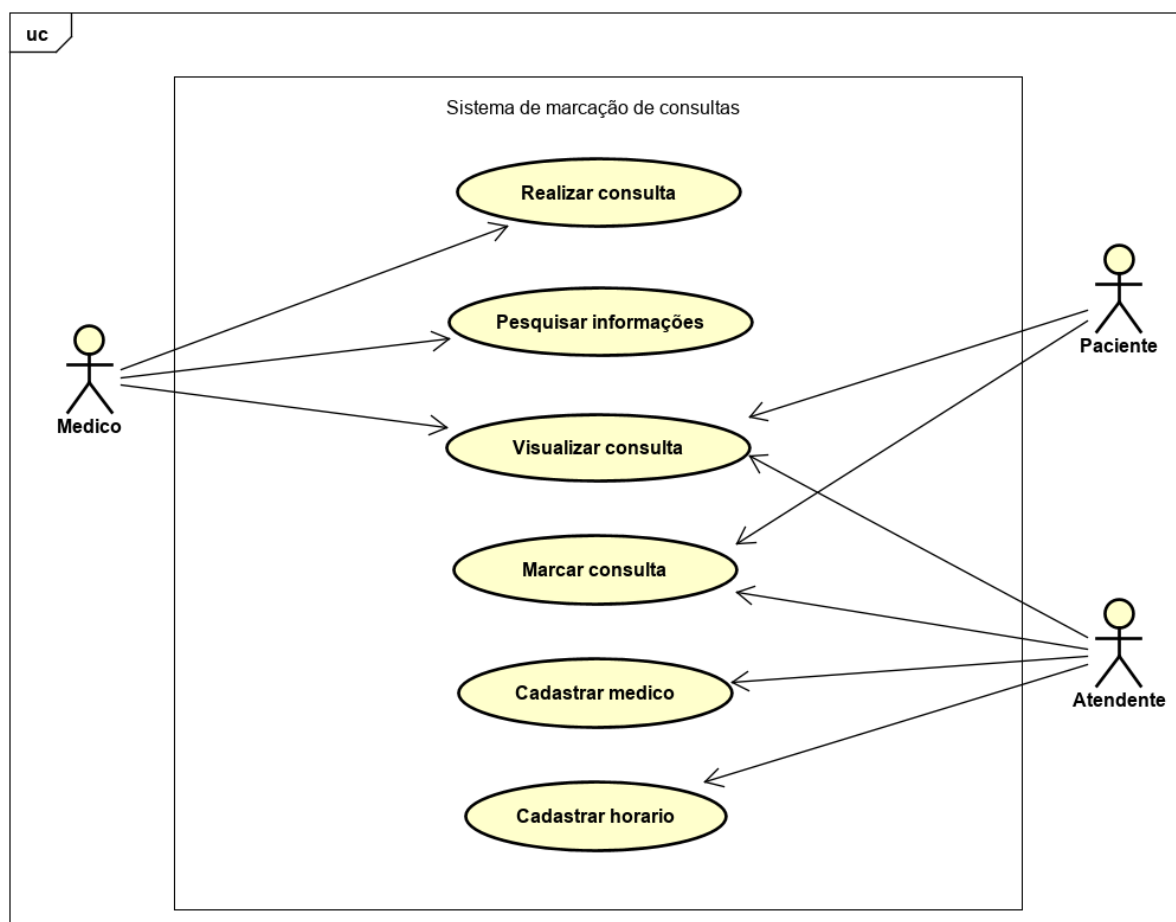
Figura 2 – Ferramenta MySQL Workbench

Fonte: <https://www.mysql.com/products/workbench/>

5.2 Diagramas de casos de uso

De acordo com Guedes (2018), o diagrama de casos de uso possibilita a compreensão do comportamento externo do sistema por qualquer pessoa com algum conhecimento sobre o problema, apresentando o sistema por intermédio de uma perspectiva dos usuários. Este diagrama tem por objetivo apresentar uma visão externa geral das funcionalidades que o sistema deverá oferecer aos usuários, sem se preocupar em profundidade com a questão de como tais funcionalidades serão implementadas. O diagrama de casos de uso é de grande auxílio para a identificação e compreensão dos requisitos funcionais do sistema, ajudando a especificar, visualizar e documentar as funções e serviços do software escolhidos pelos clientes e *stakeholders*, tentando identificar os tipos de usuários que interagirão com o sistema, quais papéis eles assumirão e quais funções um usuário específico pode requisitar (Figura 3).

Figura 3 – Diagrama de casos de usos



Fonte: Elaborada pelo autor, 2019.

O diagrama de casos de uso do sistema é composto de três atores que ilustram as ações de cada usuário no sistema de marcação de consultas:

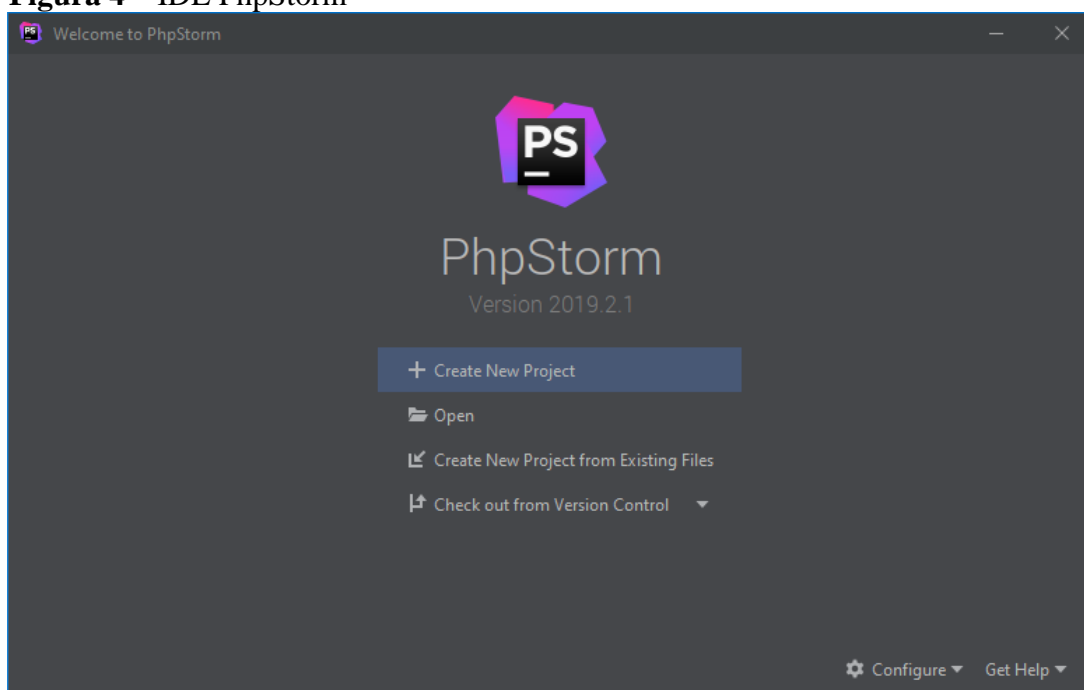
- **Médico:** Visualiza as consultas agendadas, realiza a consulta médica com diversos pacientes e pesquisa informações adicionais, como Código de Doenças (CID).
- **Secretária:** Marca consultas médicas de acordo com os horários e médicos disponíveis, cadastra médicos no sistema e seus respectivos horários de atendimentos, além de visualizar as consultas agendadas.
- **Paciente:** Marca consulta de acordo com os horários disponíveis, visualiza as consultas marcadas.

5.3 A IDE PhpStorm

Para a codificação e desenvolvimento de todo o sistema foi utilizada a IDE PhpStorm, é uma ferramenta desenvolvida pela empresa JetBrains que possui diversas funcionalidades para a linguagem PHP.

PhpStorm é uma IDE PHP inteligente que obtém seu código e entende suas estruturas, fornece auto completar de código, refatorar, realce de sintaxe e diversas outras funcionalidades (Figura 4). PhpStorm faz depuração e teste de forma simplificada, especialmente com validação de configuração do depurador (CHAUDHARY, M.et al 2014).

Figura 4 – IDE PhpStorm



Fonte: <https://www.jetbrains.com/phpstorm/>

6 RESULTADOS E DISCUSSÕES

O desenvolvimento do sistema foi dividido em três etapas principais: Desenvolvimento do *Front-End*, desenvolvimento do *Back-End* e desenvolvimento do banco de dados. Essas etapas serão apresentadas nesta seção, incluindo os resultados obtidos em cada uma destas.

6.1 Ferramentas utilizadas e trechos de códigos

Nesta seção são mostrados o desenvolvimento do banco de dados, bem como o processo de validação dos campos de formulário do Front-End, a validação das páginas web utilizando sessão do PHP, o uso do GIT como controle de versão das etapas do desenvolvimento, as funcionalidades que o sistema dispõe como também a sua utilização e trechos de códigos desenvolvidos.

6.1.1 Banco de dados do sistema

No diagrama MER do sistema de marcação de consultas foram criadas oito tabelas com algumas relações entre si (Figura 5), onde:

A tabela **medico** tem um relacionamento de **um para muitos** com a tabela **consulta**, onde um médico pode realizar várias consultas com pacientes diferentes;

A tabela **medico** tem um relacionamento de **um para muitos** com a tabela **horario**, onde um médico pode ter vários horários diferentes de atendimento durante uma semana;

A tabela **medico** tem um relacionamento **de um para um** com a tabela **diagnostico**, onde um médico poder realizar um diagnóstico para um paciente.

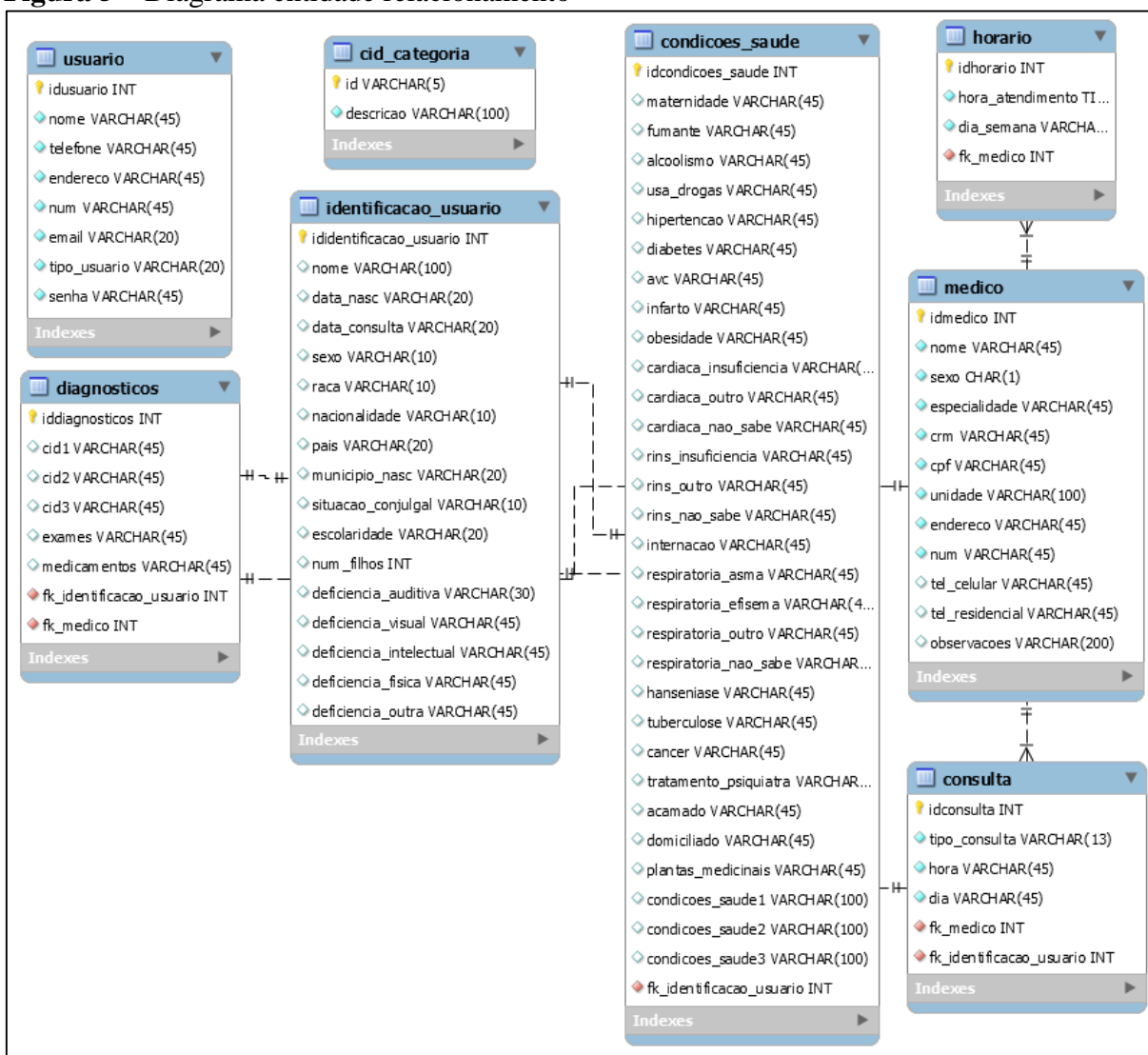
A tabela **identificacao_usuario** tem um relacionamento de **um para um** com a tabela **diagnostico**, onde um médico poderá definir um determinado diagnóstico para um paciente;

A tabela **identificacao_usuario** tem um relacionamento de **um para um** com a tabela, **condicoes_saude** onde durante a consulta, o médico irá relacionar as três tabelas: **identificacao_usuario**, **diagnostico** e **condicoes_saude**, para finalizar a consulta.

A tabela **usuario** ficou sem relacionamento com alguma outra tabela, pois foi utilizada de auxílio para criar um usuário para ter acesso às funcionalidades do sistema.

A tabela **cid_categoria** ficou sem relacionamento com outras tabelas, pois foi usada apenas para armazenar os códigos referentes aos tipos de doenças.

Figura 5 – Diagrama entidade relacionamento



Fonte: Elaborada pelo autor, 2019.

6.1.2 Segurança do sistema

Para maior segurança do sistema foi utilizado JavaScript para validação dos campos de formulário na parte do *Front-End*, como podemos ver na Figura 6, utilizamos alguns tipos de validações:

- Verificar se o tipo de USUÁRIO foi escolhido;
- Verificar se o campo E-MAIL está vazio;
- Validar o formato do E-MAIL;
- Verificar se o campo SENHA está vazio;
- Verificar se o campo SENHA possui no mínimo seis (6) caracteres;

Na imagem a seguir (Figura 6) é visualizado o código de validação JavaScript para a tela de login, o mesmo código foi utilizado para validar os demais formulários do sistema, apenas substituindo ou adicionado os campos do formulário ao qual foi validado.

Figura 6 – Validação JavaScript do campo de formulário de login

```

var form = document.getElementById("formLogin");

if (form.addEventListener) {
    form.addEventListener("submit", validaLogin);
} else if (form.attachEvent) {
    form.attachEvent("onsubmit", validaLogin);
}

function validaLogin(evt) {
    var email = document.getElementById('email');
    var senha = document.getElementById('senha');
    var tipoUsuario = document.getElementById('tipo_usuario');
    var filtro = /^[^\w-]+(?:\.[^\w-]+)*@((?:[\w-]+\.)*\w[\w-]{0,66})\.([a-z]{2,6}(?:\.[a-z]{2})?)$/i;
    var contErro = 0;
    /* Validação do campo email */
    caixa_email = document.querySelector('.msg-email');
    if (email.value == "") {
        caixa_email.innerHTML = "Favor preencher o e-mail";
        caixa_email.style.display = 'block';
        contErro += 1;
    } else if (filtro.test(email.value)) {
        caixa_email.style.display = 'none';
    } else {
        caixa_email.innerHTML = "Formato do e-mail inválido";
        caixa_email.style.display = 'block';
        contErro += 1;
    }
    /* Validação do campo senha */
    caixa_senha = document.querySelector('.msg-senha');
    if (senha.value == "") {
        caixa_senha.innerHTML = "Favor preencher a senha";
        caixa_senha.style.display = 'block';
        contErro += 1;
    } else if (senha.value.length < 6) {
        caixa_senha.innerHTML = "Favor preencher a senha com o mínimo de 6 caracteres";
        caixa_senha.style.display = 'block';
        contErro += 1;
    } else {
        caixa_senha.style.display = 'none';
    }
    /* Validação do campo tipo-usuario */
    caixa_tipoUsuario = document.querySelector('.msg-tipo-usuario');
    if (tipoUsuario.value == "selecione...") {
        caixa_tipoUsuario.innerHTML = "Favor selecionar o tipo de usuario";
        caixa_tipoUsuario.style.display = 'block';
        contErro += 1;
    } else {
        caixa_tipoUsuario.style.display = 'none';
    }
    if (contErro > 0) {
        evt.preventDefault();
    }
}

```

Fonte: Elaborada pelo autor, 2019.

Na figura a seguir podemos ver a validação realizada pelo código JavaScript (Figura 6) da tela de login ao deixar os campos tipo de usuário, e-mail e senha em branco.

Figura 7 – Resultado da validação JavaScript no Front-End

The figure shows two states of the login page for 'AgendaUBS Sistema Gerenciador de Consultas'. The left panel shows the form with a dropdown menu for 'selecione o usuario...', an email field, a password field, and an 'Entrar' button. Below the fields are links for 'Criar uma conta' and 'Esqueceu a senha?'. The right panel shows the same form but with a yellow warning box at the top that says 'Atenção' and 'Usuário ou Senha incorretos!'. The dropdown menu now shows 'Atendente', the email field contains 'atendente@gmail.com', and the password field is masked with dots.

Fonte: Elaborada pelo autor, 2019.

Na segurança do *Back-End* foi utilizado autenticação no login através de sessões do PHP, para isso foram capturadas as sessões de e-mail e tipo de usuário (Figura 8), caso as informações digitadas nos campos de e-mail e senha sejam as mesmas cadastradas no banco de dados o sistema valida as informações e direciona para a página principal, caso contrário, a mensagem “Usuário ou Senha incorretos!” é sinalizada na tela e o usuário permanece na tela de login como mostra a Figura 7.

Figura 8 – Função do PHP para autenticação de acesso do usuário no sistema

```
/**
 * Autenticar usuário.
 */
public function autenticar($email, $senha, $tipoUsuario)
{
    try {
        // Se o retorno for null é porque o usuário não foi encontrado!
        if (empty($this->dao->login($email, $senha, $tipoUsuario))) {
            Messenger::printMessageWarning("Usuário ou Senha incorretos!");
        } else {
            session_start();
            $_SESSION["email"] = $email;
            $_SESSION["tipo_usuario"] = $tipoUsuario;
        }
    } catch (PDOException $e) {
        Messenger::printMessageDanger($e->getMessage());
    }
}
```

Fonte: Elaborada pelo autor, 2019.

Figura 9 – Função do PHP para validação dos dados de login do usuário

```

public function login($email, $senha, $tipoUsuario)
{
    try {
        $db = Conexao::conecta();
        // tenta achar usuário pelo email e tipo
        $sql = "SELECT * FROM usuario WHERE email = :email AND tipo_usuario = :tipo_usuario LIMIT 1";
        $rs = $db->prepare($sql);
        $rs->bindValue(":email", $email);
        $rs->bindValue(":tipo_usuario", $tipoUsuario);

        $rs->execute();
        $result = $rs->fetch(PDO::FETCH_OBJ);

        // Verifica se o usuario foi encontrado pelo email e tipo
        // Verifica se a senha fornecida bate com o hash da senha salva no banco
        if (!empty($result) && password_verify($senha, $result->senha)) {
            return $result; // usuario autenticado!
        }
        return NULL; // usuario não autenticado!
    } catch (PDOException $e) {
        throw new DAOException($e->getMessage());
    } finally {
        $db = NULL;
        if (!empty($rs)) $rs->closeCursor();
    }
}

```

Fonte: Elaborada pelo autor, 2019.

Para impedir o acesso às páginas PHP na web não autorizadas foram feitas as verificações pela sessão de e-mail do usuário logado, como podemos ver na Figura 10 a seguir.

Figura 10 – Código de verificação de e-mail logado na sessão

```

<?php
session_start();
if (!isset($_SESSION["email"])) {
    echo "<script> document.location = './login.php'; </script>";
    exit();
} else {
    $email = $_SESSION['email'];
    $usuarioDAO = new UsuarioDAO();
}
if (isset($_REQUEST["logout"]) && $_REQUEST["logout"] == true) {
    unset($_SESSION["email"]);
    echo "<script> document.location = './login.php'; </script>";
    exit();
}
?>

```

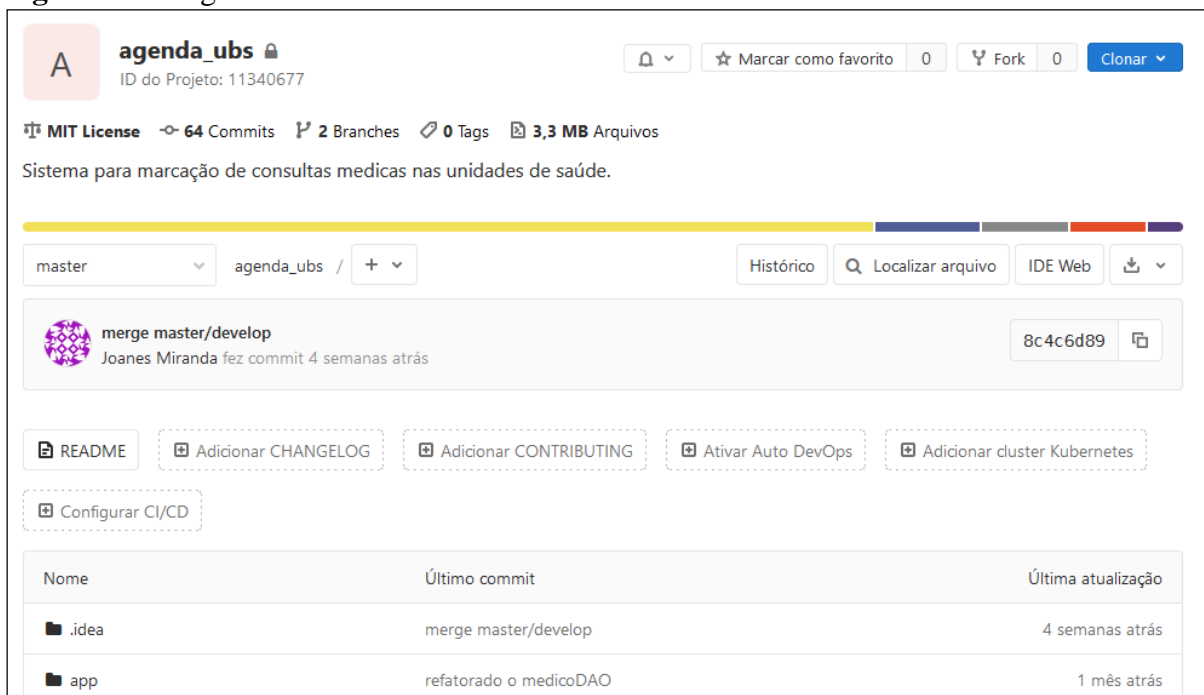
Fonte: Elaborada pelo autor, 2019.

6.1.3 O uso do GitLab para controle de versões

Para cada etapa de codificação realizada durante a evolução do sistema de marcação de consultas, foi efetuado o *commit* para o gerenciador de repositório de software baseado em GIT, o **GitLab**. O sistema de marcação de consultas está com seu repositório criado no

seguinte endereço: https://gitlab.com/joanesMiranda/agenda_ubs.git. Ao total foram realizados 64 commits em 2 branches (master/develop) durante o desenvolvimento desse sistema, como ilustrado na Figura 11.

Figura 11 – Página de detalhes do GitLab



Fonte: https://gitlab.com/joanesMiranda/agenda_ubs

Na tabela 1, são mostrados alguns comandos utilizados para manter o controle de versão de todas as etapas de codificação.

Tabela 1 – Comandos utilizados no GitLab

Comandos	Descrição
git clone	Clona um projeto disponível no GitLab
git init	Inicializa o git no seu projeto
git checkout master/develop	Alterna entre as branches disponíveis
git add .	Adiciona todas as alterações para o stage
git commit -m "comentário"	Faz o commit dos arquivos
git push origin develop	Envia os arquivos para a branch de origem

Fonte: Elaborada pelo autor, 2019.

6.2 Funcionalidades do sistema

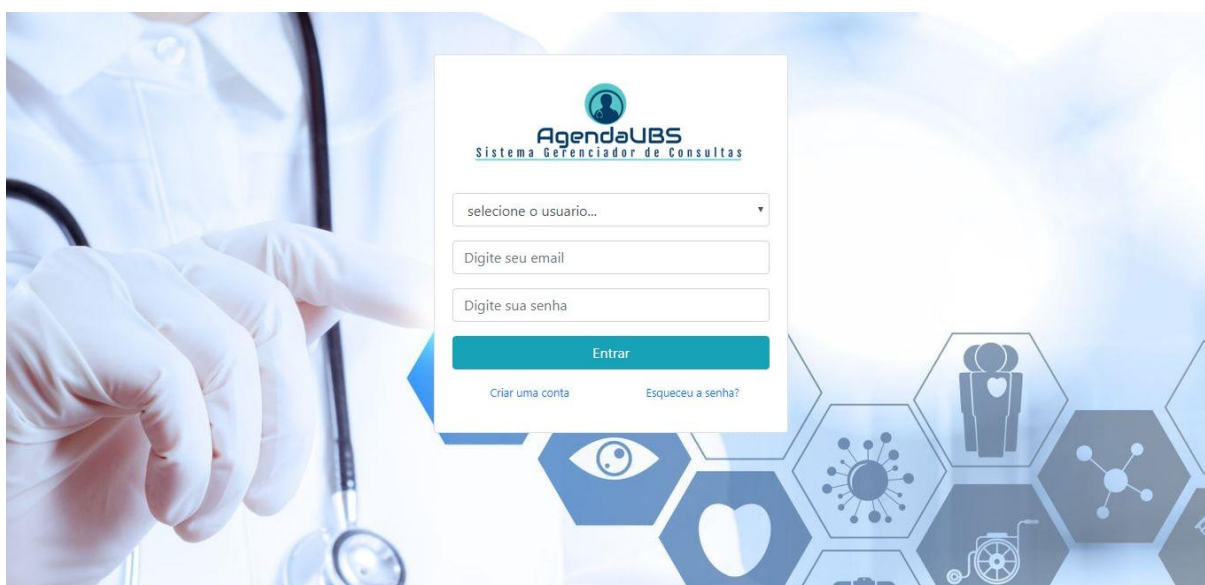
O sistema de marcação de consultas foi desenvolvido de acordo com os requisitos funcionais, descritos no APÊNDICE A, e não-funcionais, descritos no APÊNDICE B.

Foram definidos três tipos de usuários (paciente, atendente e médico), cada um com suas respectivas ações (Figura 13), e três ambientes distintos foram criados com suas funcionalidades: para o **usuário paciente** é permitido marcar uma consulta médica e visualizar as consultas já marcadas por ele; para o **usuário atendente** é possível cadastrar um médico no sistema, cadastrar um horário semanal para o atendimento de um médico, marcar uma consulta para um determinado paciente e visualizar as consultas previamente marcadas; para o **usuário médico** é possível realizar o atendimento de uma consulta, visualizar os pacientes que vão ser atendidos no dia e pesquisar informações sobre os códigos (CID) de doenças. Ambas as funcionalidades do sistema vão ser descritas abaixo.

Tela de Login:

Tela de acesso ao sistema, para chegar a essa tela foi definido um endereço no ambiente Heroku: <http://agendaubs.herokuapp.com>, a partir dela é possível realizar uma autenticação escolhendo o tipo de usuário (paciente, médico ou atendente), digitar uma senha e ter acesso as funcionalidades do sistema, como também realizar um cadastro prévio para o acesso no caso de ser usuário novo e a recuperação de senha.

Figura 12 – Tela de acesso ao sistema



Fonte: Elaborada pelo autor, 2019.

Figura 13 – Tela de login – escolhendo o tipo de usuário

selecione o usuario... ▼

selecione o usuario...

Atendente

Paciente

Médico

Digite sua senha

Entrar

[Criar uma conta](#) [Esqueceu a senha?](#)

Fonte: Elaborada pelo autor, 2019.

Cadastro de novo usuário:

Esta tela é bem simples, onde o usuário faz um cadastro digitando algumas informações obrigatórias que são seguidas pelo símbolo (*), são enviadas para o sistema, em seguida o usuário será cadastrado, como campos obrigatórios foram definidos: nome completo, e-mail, tipo de usuário e senha.

Figura 14 – Tela de cadastro de novo usuário

Criar uma conta

Nome completo * Celular

Digite seu nome Digite seu telefone

Endereço Numero

Nome da rua

Email * Tipo Usuário *

Digite seu email selecione...

Senha * Confirmar senha *

Digite sua senha Confirme sua senha

Registrar-se

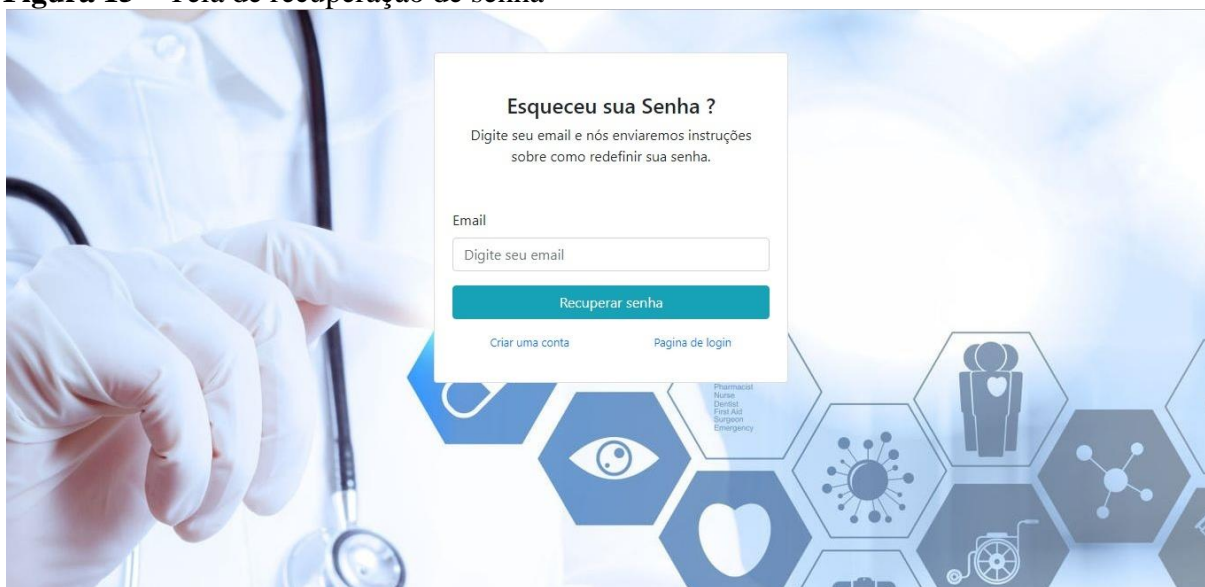
[Pagina de login](#) [Esqueceu a senha?](#)

Fonte: Elaborada pelo autor, 2019.

Tela de recuperação de senha:

Caso o usuário, previamente cadastrado, esqueça sua senha de acesso, o sistema oferece uma página de recuperação, nela é solicitado digitar o e-mail que foi cadastrado pelo usuário onde, em seguida, receberá uma nova senha de acesso.

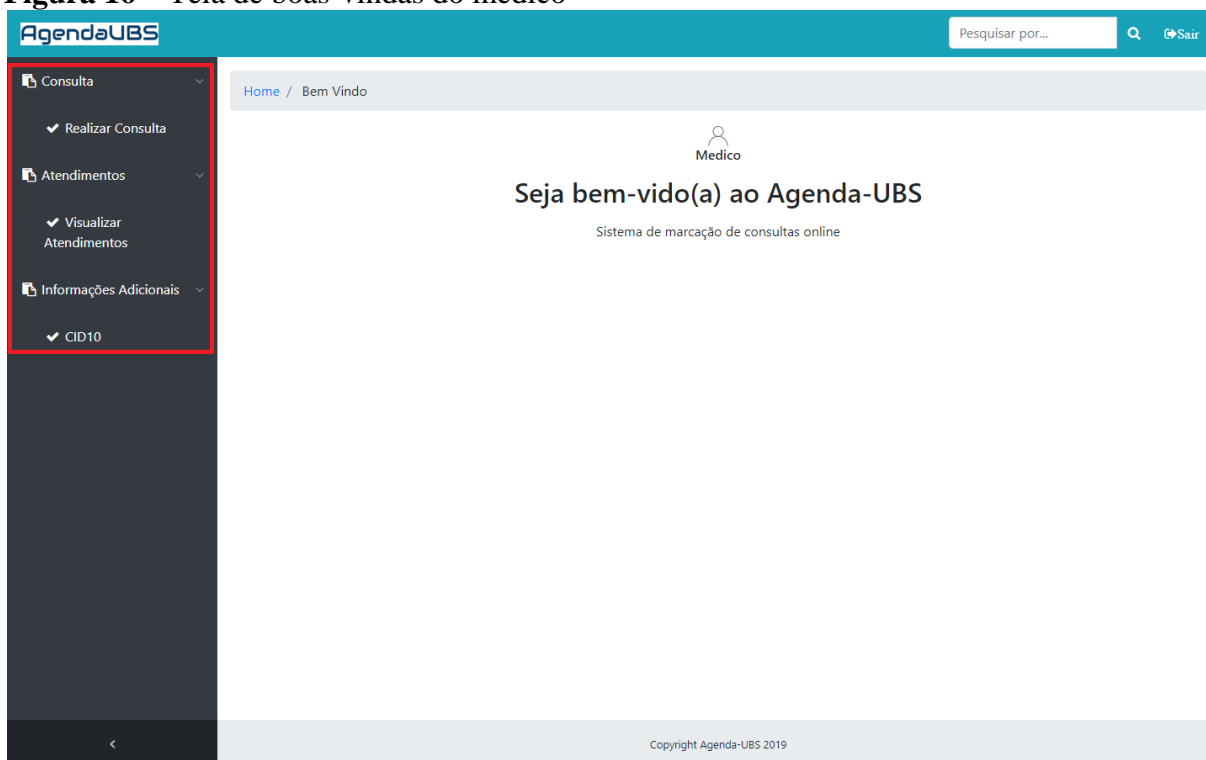
Figura 15 – Tela de recuperação de senha



Fonte: Elaborada pelo autor, 2019.

Área do médico:

A partir do login no sistema de marcação de consultas, escolhendo o tipo de usuário médico, o sistema disponibiliza algumas funcionalidades exclusivas para o médico: o médico pode realizar uma consulta preenchendo informações sobre as condições de saúde do paciente, pode diagnosticar determinadas doenças, pesquisar informações sobre códigos de doenças e visualizar os atendimentos de pacientes que irá realizar diariamente.

Figura 16 – Tela de boas-vindas do médico

Fonte: Elaborada pelo autor, 2019.

Tela de realização de consulta:

Permite ao médico realizar uma consulta de um paciente preenchendo informações importantes que foram divididas em cinco partes seguindo um formulário com critérios com base nas diretrizes estabelecidas pelo SUS:

Quadro 1 – Campos do formulário de identificação do usuário

Identificação do usuário	
Nome completo	
Data de nascimento	
Data de consulta	
Sexo	Masculino / Feminino
Raça	branco/preto/pardo/amarelo/indígena
Nacionalidade	brasileira/naturalizado/estrangeiro
País de nascimento	
Município de nascimento	
Situação conjugal	Solteiro/casado/divorciado/viúvo/outra

Escolaridade	Ensino fundamental/Ensino médio/Superior incompleto/Superior completo
Número de filhos	Não possui filhos/1...4/mais de 5 filhos
Tem alguma deficiência?	SIM/NÃO
Se SIM qual?	Auditiva/visual/intelectual/física/outra

Fonte: Elaborada pelo autor, 2019.

Quadro 2 – Campos do formulário de condições de saúde do paciente

Condições gerais de saúde	
Está gestante?	SIM / NÃO
Se SIM qual a maternidade de referência	
Está fumante?	
Está dependente ou abusa de outras drogas?	
Tem hipertensão arterial?	
Tem diabetes?	
Teve AVC ou derrame?	
Teve Infarto?	
Sobre seu peso você se considera?	Abaixo do peso / peso adequado / acima do peso
Tem doença cardíaca?	Insuficiência cardíaca/ Outro/ não sabe
Teve problemas nos rins?	Insuficiência renal / Outro/ Não sabe
Teve doença respiratória?	Asma/DPOC/Enfisema/outro/não sabe
Está com hanseníase?	
Está com tuberculose?	
Teve ou tem câncer?	
Teve alguma internação nos últimos 12 meses?	SIM/NÃO/ se SIM qual a causa?
Fez ou faz tratamento psiquiatra ou teve internação por problemas de saúde mental?	SIM/NÃO
Está acamado?	SIM/NÃO
Está domiciliado?	SIM/NÃO
Usa outras práticas interativas e complementares?	SIM/NÃO
Outras condições de saúde	
	1 – Qual? / 2 – Qual?/ 3 – Qual?

Fonte: Elaborada pelo autor, 2019.

Quadro 3 – Campos do formulário de classificação internacional de doenças

Diagnósticos CID10		
1?	2?	3?

Fonte: Elaborada pelo autor, 2019.

Quadro 4 – Campos do formulário de exames

Exames

Fonte: Elaborada pelo autor, 2019.

Quadro 5 – Campos do formulário de prescrição de medicamentos

Medicamentos

Fonte: Elaborada pelo autor, 2019.

Figura 17 – Tela de realização de consulta com paciente

AgendaUBS

Pesquisar por... Sair

Home / Realizar Consulta

1. Identificação do Usuário 2. Condições Gerais de Saúde 3. Diagnostico - CID10 4. Exames 5. Medicamentos 6. Concluir

Nome* Data de nascimento*
 dd/mm/aaaa

Data de consulta* Sexo*
 dd/mm/aaaa seleccione...

Raça* Nacionalidade:
 seleccione... Brasileira Naturalizado Estrangeiro

País de nascimento* Município de nascimento*

Situação conjugal:
 Solteiro (a) Casado(a)/Convívio com parceiro Divorciado (a) / Separado (a) Viúvo (a) Outra

Escolaridade: Numero de filhos
 Ensino fundamental Ensino médio Superior incompleto Superior completo não possui filhos

Fonte: Elaborada pelo autor, 2019.

Esta tela permite ao médico realizar uma busca por códigos de doenças relacionadas à saúde para facilitar no diagnóstico, seguindo a classificação internacional de doenças e problemas relacionados à saúde.

Figura 18 – Tela de pesquisar sobre códigos de doenças (CID10)

AgendaUBS

Pesquisar por... Q Sair

Home / Codigos CID10

Diagnosticos - CID 10

Digite abaixo o seu codigo de CID*

F41

Digite a LETRA e depois a numeração. Ex. F41

Copyright Agenda-UBS 2019

Fonte: Elaborada pelo autor, 2019.

Figura 19 – Resultado da pesquisa por doença

AgendaUBS

Pesquisar por... Q Sair

codigo	descrição
F00	Demência na doença de Alzheimer
F01	Demência vascular
F02	Demência em outras doenças classificadas em outra parte
F03	Demência não especificada
F04	Síndrome amnésica orgânica não induzida pelo álcool ou por outras substâncias psicoativas
F05	Delirium não induzido pelo álcool ou por outras substâncias psicoativas
F06	Outros transtornos mentais devidos a lesão e disfunção cerebral e a doença física
F07	Transtornos de personalidade e do comportamento devidos a doença
F09	Transtorno mental orgânico ou sintomático não especificado
F10	Transtornos mentais e comportamentais devidos ao uso de álcool
F11	Transtornos mentais e comportamentais devidos ao uso de opiáceos
F12	Transtornos mentais e comportamentais devidos ao uso de canabinóides
F13	Transtornos mentais e comportamentais devidos ao uso de sedativos e hipnóticos
F14	Transtornos mentais e comportamentais devidos ao uso da cocaína
F15	Transtornos mentais e comportamentais devidos ao uso de outros estimulantes
F16	Transtornos mentais e comportamentais devidos ao uso de alucinógenos

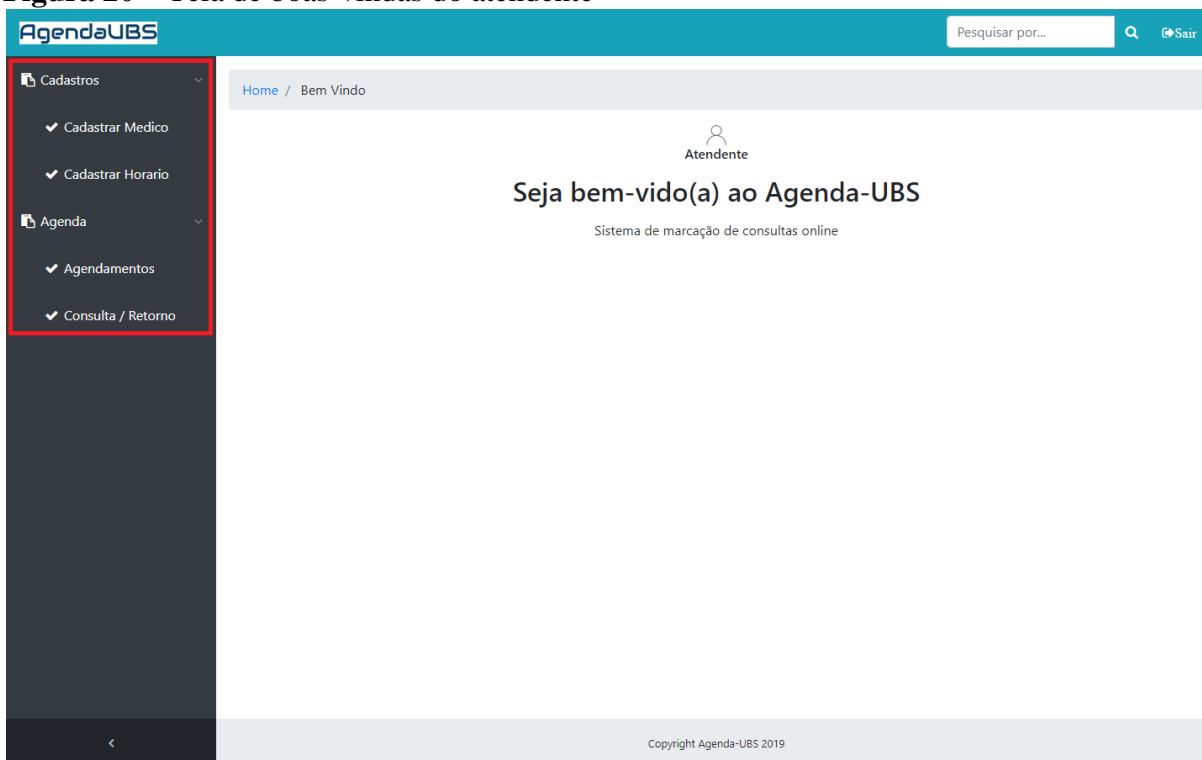
Copyright Agenda-UBS 2019

Fonte: Elaborada pelo autor, 2019.

Área do atendente:

A partir do login no sistema de marcação de consultas, escolhendo o tipo de usuário atendente, o sistema disponibiliza algumas funcionalidades exclusivas: no menu lateral **cadastros** o atendente dispõe de duas funcionalidades que é o **cadastro de médico**, onde é possível cadastrar vários médicos e alocá-los em unidades de saúde específicas e o **cadastrar horário** onde é cadastrado o horário semanal que o médico irá atender em determinada UBS. No menu agenda o atendente pode visualizar as consultas agendadas pelos atendentes e pacientes em **agendamentos**, além de poder agendar uma consulta para um paciente qualquer em **consulta/retorno**.

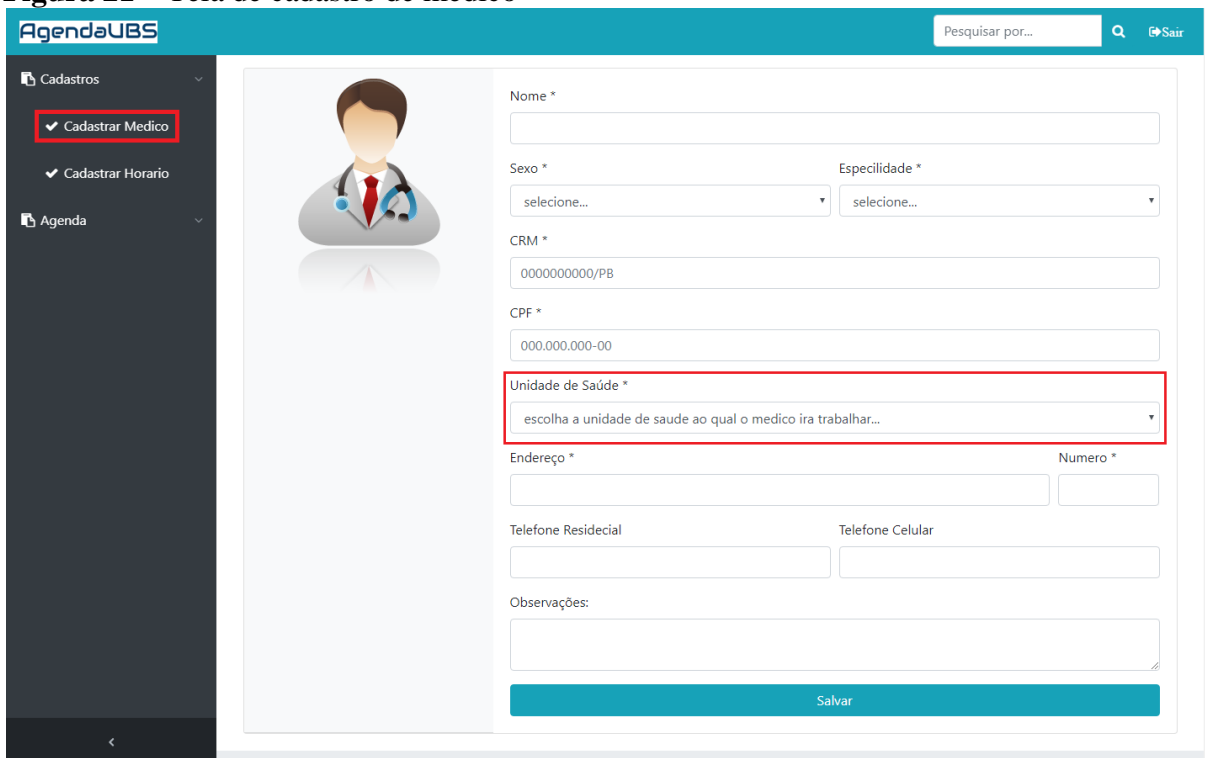
Figura 20 – Tela de boas-vindas do atendente



Fonte: Elaborada pelo autor, 2019.

Tela de cadastro médico:

O atendente poderá cadastrar vários médicos e alocá-los em unidades de saúde em diferentes bairros da cidade de Campina Grande.

Figura 21 – Tela de cadastro de médico

The screenshot displays the 'AgendaUBS' web application interface. On the left, a dark sidebar contains a menu with 'Cadastrados' expanded, showing 'Cadastrar Medico' (highlighted with a red box) and 'Cadastrar Horario'. Below the sidebar is a doctor icon. The main content area is a form for registering a doctor. The form fields are: 'Nome *' (text input), 'Sexo *' (dropdown menu), 'Especialidade *' (dropdown menu), 'CRM *' (text input with '0000000000/PB'), 'CPF *' (text input with '000.000.000-00'), 'Unidade de Saúde *' (dropdown menu with 'escolha a unidade de saude ao qual o medico ira trabalhar...' highlighted by a red box), 'Endereço *' (text input), 'Numero *' (text input), 'Telefone Residencial' (text input), 'Telefone Celular' (text input), and 'Observações:' (text area). A teal 'Salvar' button is at the bottom right.

Fonte: Elaborada pelo autor, 2019.

Página de cadastro de horários:

O atendente pode cadastrar o horário que um determinado médico está de plantão semanalmente, para isso deve escolher o dia da semana (de segunda à domingo), escolher um entre os vários médicos cadastrados no sistema, e definir um horário inicial de atendimento, também é disponibilizado a opção de excluir um horário.

Figura 22 – Tela de criação de horário do atendimento médico

The screenshot shows the 'Cadastro de Horário' (Appointment Registration) page in the AgendaUBS system. The page has a teal header with the logo and a search bar. A dark sidebar on the left contains navigation options: 'Cadastros' (with sub-options 'Cadastrar Medico' and 'Cadastrar Horário', the latter being highlighted with a red box), and 'Agenda'. The main content area is titled 'Cadastro de Horário' and contains three dropdown menus: 'Dia da Semana' (with the placeholder 'selecione o dia...'), 'Medico de Plantão' (with 'selecione o medico...'), and 'Horario de atendimento' (with '---'). A teal 'Salvar' button is positioned below these fields. At the bottom of the form area is a table with the following data:

Medico de Plantão	Horario de Atendimento	Dia da Semana	Excluir
Vladimir Costa de Alencar	09:00hs	Segunda	
Vladimir Costa de Alencar	09:00hs	Terça	
Vladimir Costa de Alencar	12:00hs	Domingo	

The footer of the page contains the text 'Copyright Agenda-UBS 2019'.

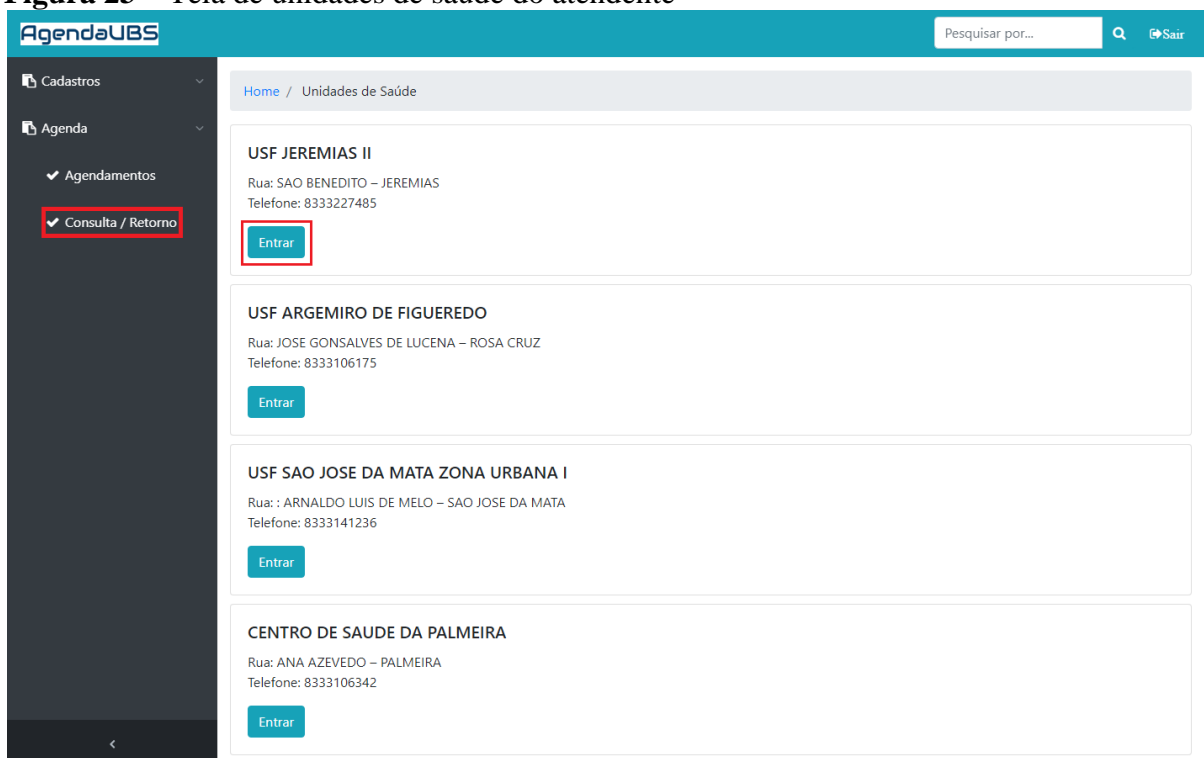
Fonte: Elaborada pelo autor, 2019.

Marcando uma consulta médica para um paciente:

O sistema disponibiliza a opção do atendente, marcar uma consulta para um paciente, para isso é necessário seguir alguns passos:

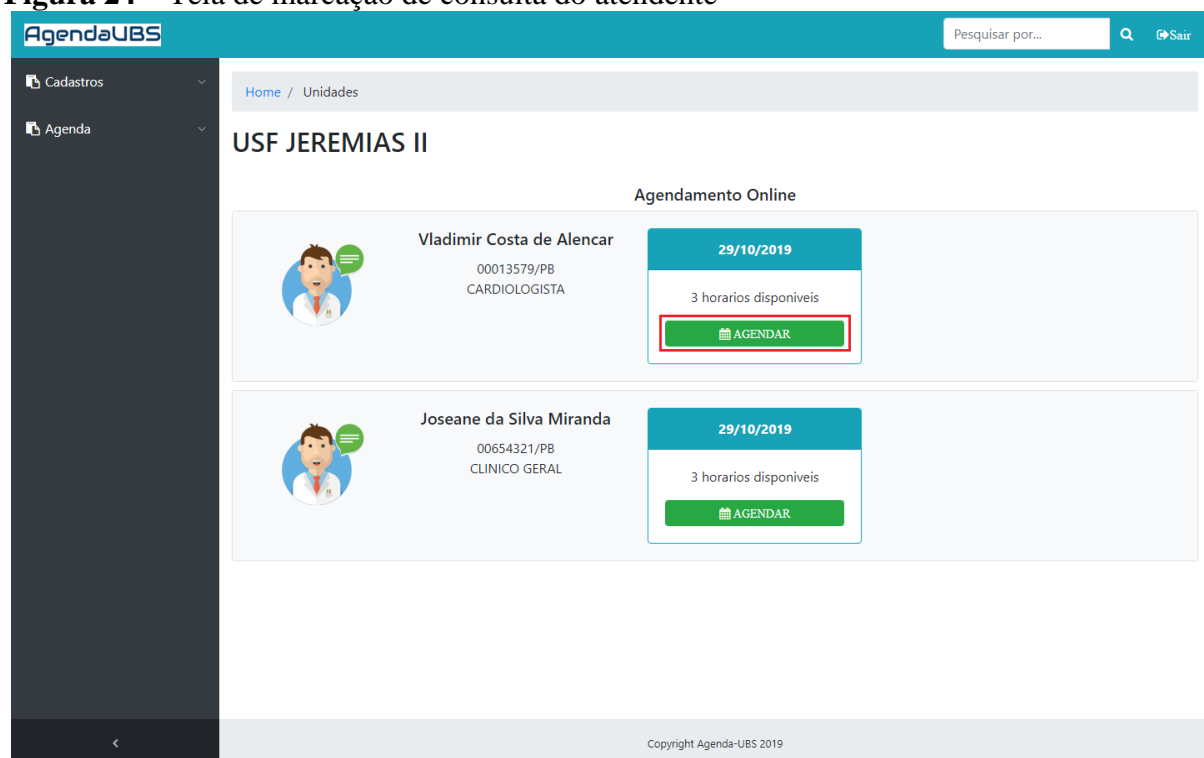
- Primeiro no menu lateral **agenda** escolher a opção **consulta/retorno**, será mostrado a tela de unidades de saúde (Figura 23), nessa tela são exibidas todas as unidades de saúde cadastradas;
- Segundo passo é clicar em **entrar** em uma das unidades de saúde ao qual deseja marcar uma consulta;
- Terceiro passo (Figura 24), mostra os médicos e horários disponíveis de atendimento, é só escolher um médico e clicar em **agendar**;
- Quarto e último passo é aberta uma janela modal (Figura 25) com as opções de colocar o nome do paciente, e então escolhe o tipo de consulta (nova ou retorno) e clicar em **marcar**. Por fim na tela o horário do agendamento é confirmado (Figura 26).

Figura 23 – Tela de unidades de saúde do atendente



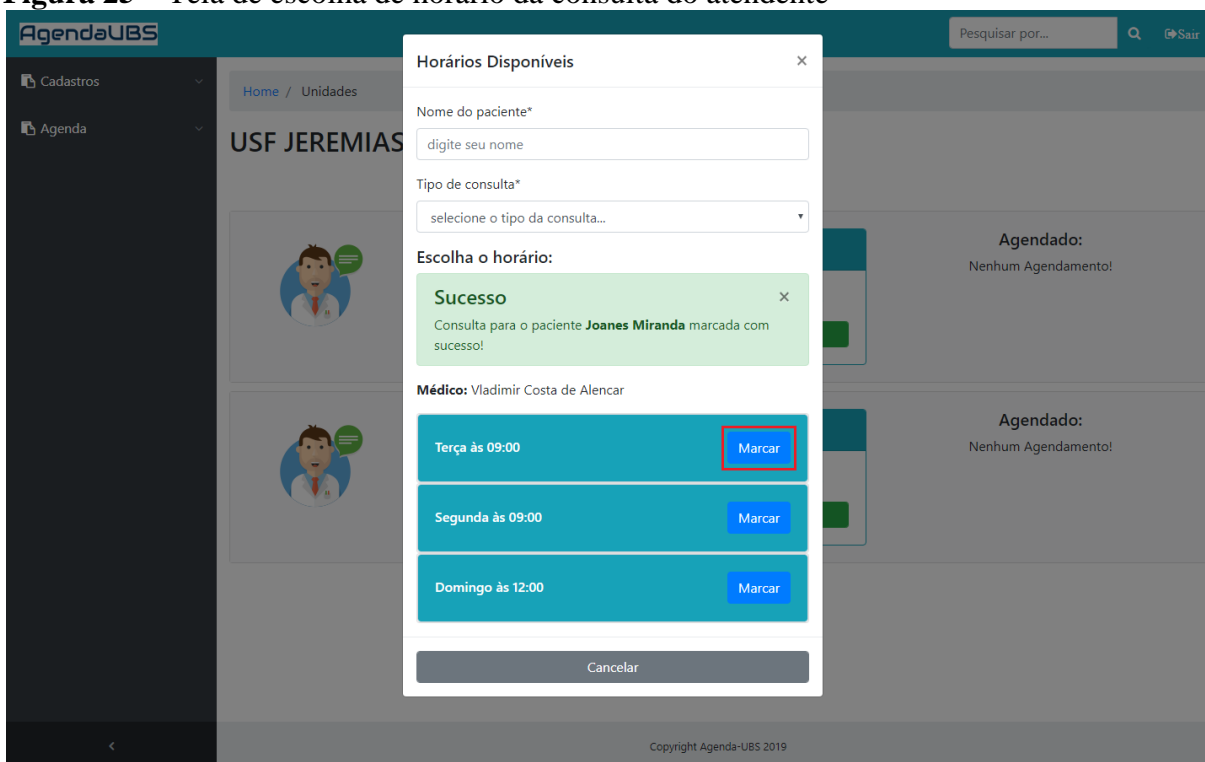
Fonte: Elaborada pelo autor, 2019.

Figura 24 – Tela de marcação de consulta do atendente



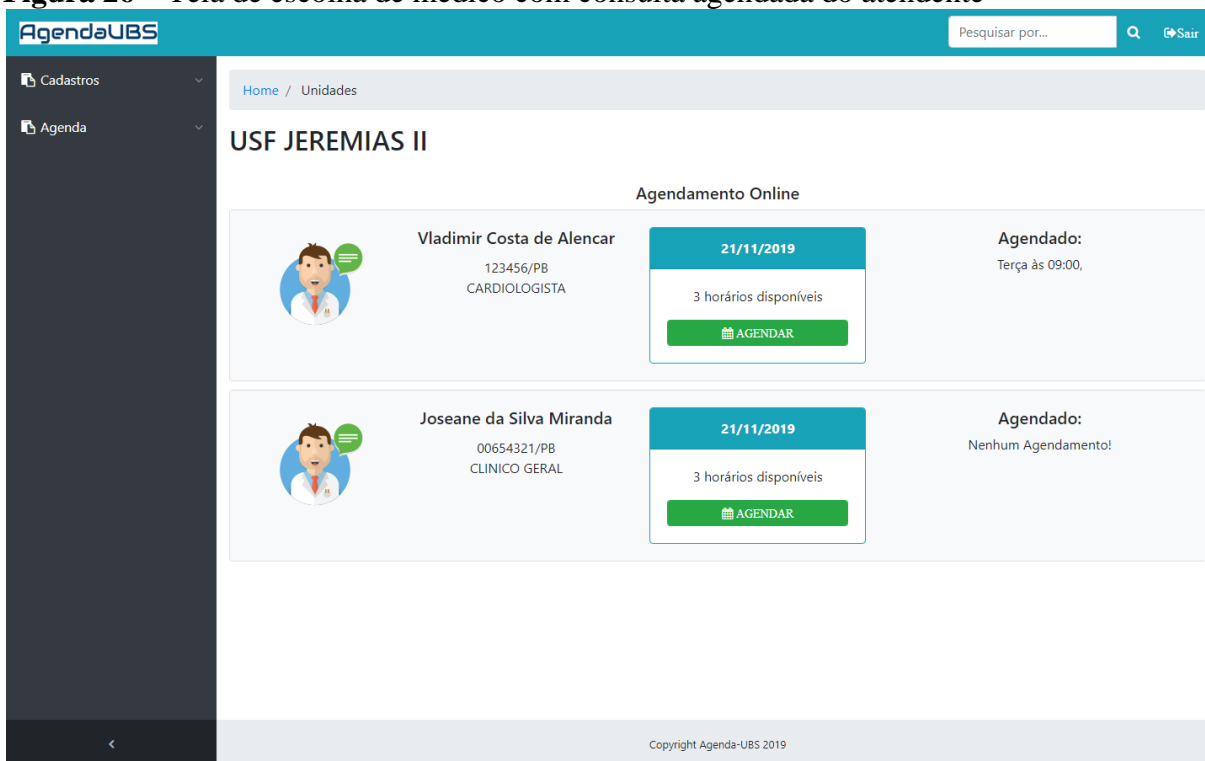
Fonte: Elaborada pelo autor, 2019.

Figura 25 – Tela de escolha de horário da consulta do atendente



Fonte: Elaborada pelo autor, 2019.

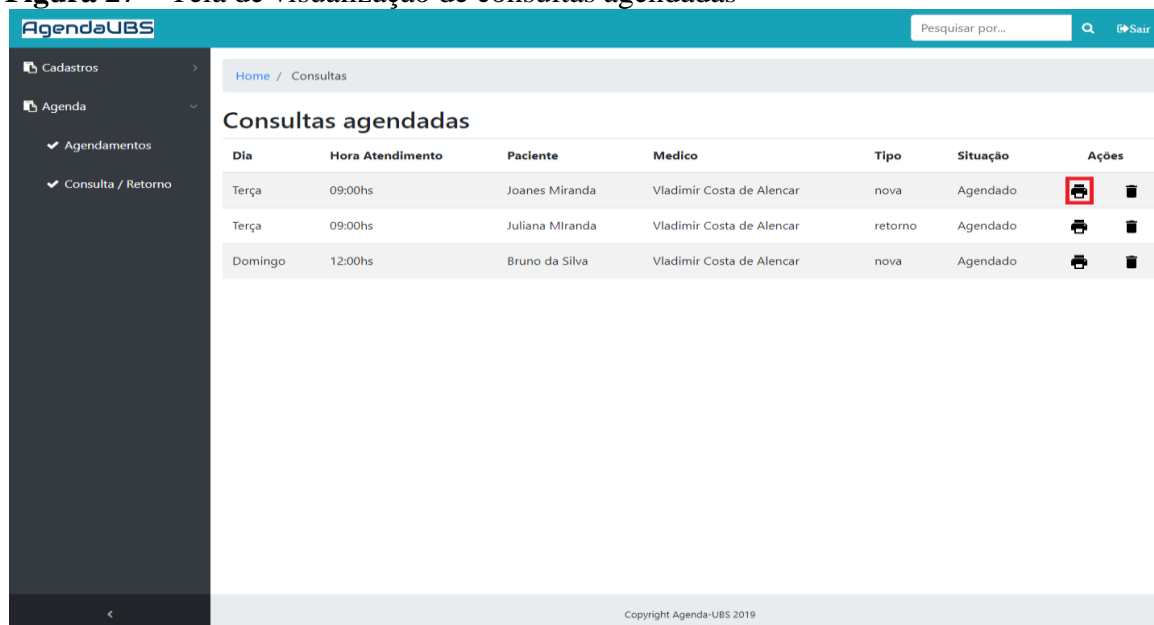
Figura 26 – Tela de escolha de médico com consulta agendada do atendente









Fonte: Elaborada pelo autor, 2019.

O atendente também pode visualizar as consultas já agendadas tanto pelo paciente quando por ele mesmo, para isso basta clicar no menu lateral **Agenda** e escolher a opção **Agendamentos** conforme Figura 27.

Figura 27 – Tela de visualização de consultas agendadas



The screenshot shows the AgendaUBS web application. On the left is a dark sidebar menu with options: 'Cadastros', 'Agenda' (expanded to show 'Agendamentos' and 'Consulta / Retorno'), and a back arrow at the bottom. The main content area has a teal header with 'AgendaUBS', a search bar, and a 'Sair' button. Below the header is a breadcrumb 'Home / Consultas' and the title 'Consultas agendadas'. A table lists three consultations with columns for Day, Time, Patient, Doctor, Type, Status, and Actions. The first row is highlighted, and a red box highlights the printer icon in the 'Ações' column.

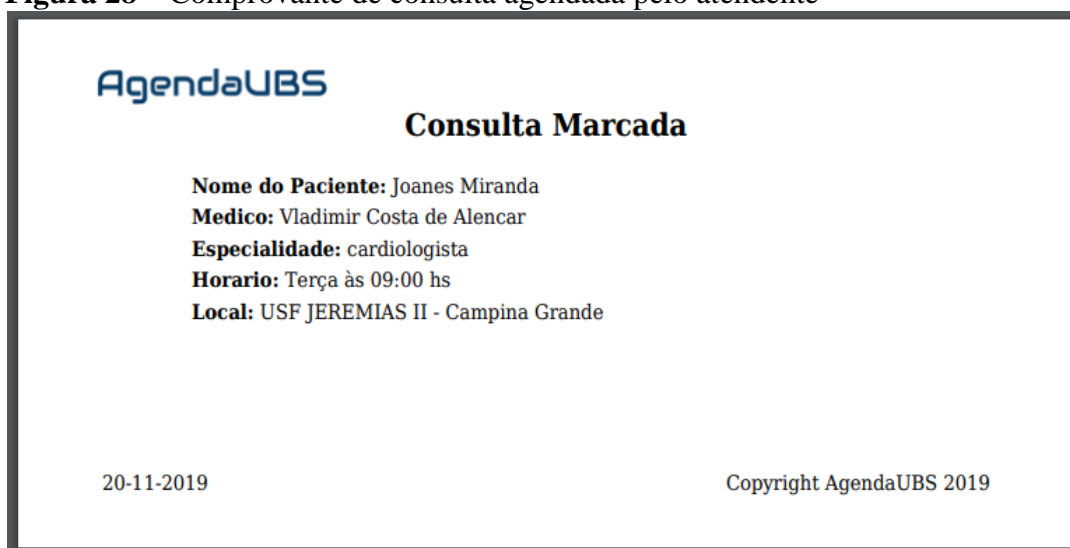
Dia	Hora Atendimento	Paciente	Medico	Tipo	Situação	Ações
Terça	09:00hs	Joanes Miranda	Vladimir Costa de Alencar	nova	Agendado	 
Terça	09:00hs	Juliana Miranda	Vladimir Costa de Alencar	retorno	Agendado	 
Domingo	12:00hs	Bruno da Silva	Vladimir Costa de Alencar	nova	Agendado	 

Copyright Agenda-UBS 2019

Fonte: Elaborada pelo autor, 2019.

Ao clicar no ícone (Figura 27) da impressora na tela visualização de consultas agendadas é disponibilizado um comprovante em pdf com o horário escolhido, nome do médico e local de atendimento do paciente.

Figura 28 – Comprovante de consulta agendada pelo atendente



The screenshot shows a PDF receipt titled 'Consulta Marcada' from AgendaUBS. It contains the following information:

Nome do Paciente: Joanes Miranda
Medico: Vladimir Costa de Alencar
Especialidade: cardiologista
Horario: Terça às 09:00 hs
Local: USF JEREMIAS II - Campina Grande

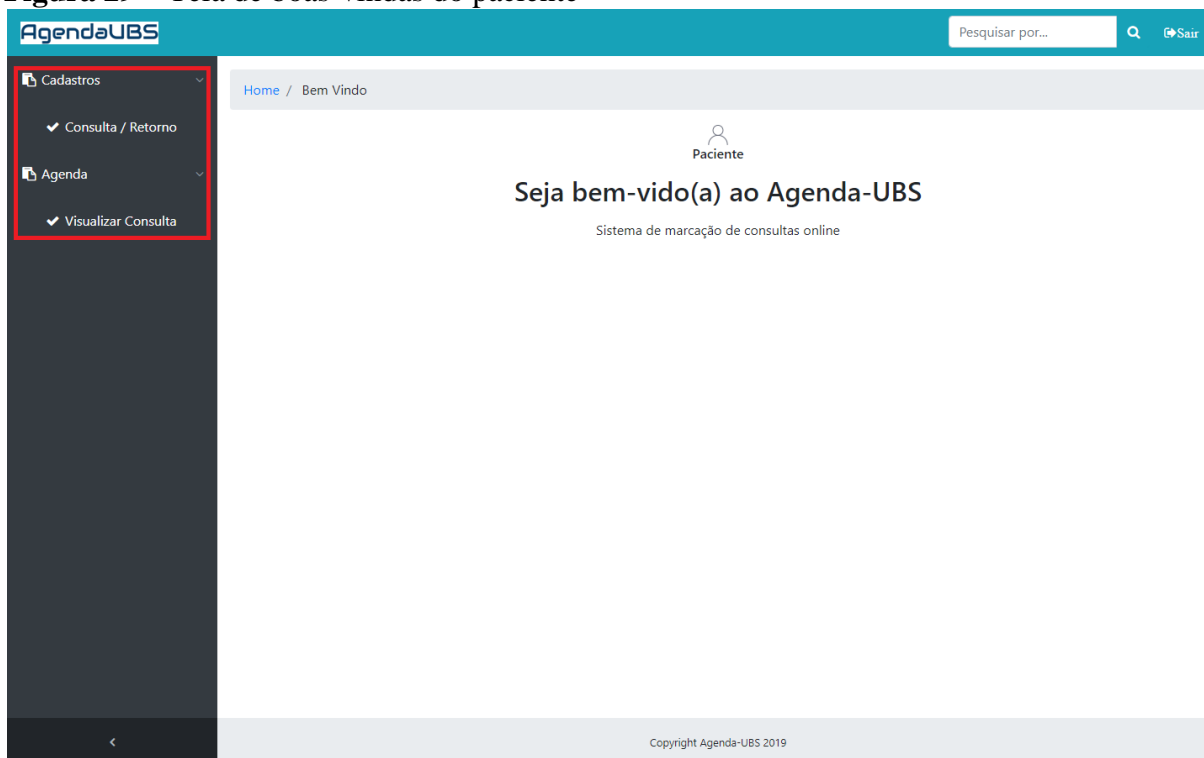
At the bottom, it shows the date '20-11-2019' and the copyright notice 'Copyright AgendaUBS 2019'.

Fonte: Elaborada pelo autor, 2019.

Área do Paciente:

A partir do login no sistema de marcação de consultas, escolhendo o tipo de usuário paciente, o sistema disponibiliza as funcionalidades a partir do menu lateral de agendar uma consulta médica, escolhendo a unidade de saúde ao qual ele pertence e o médico que está disponível no horário; além disso, o sistema dispõe da opção de visualizar a consulta agendada. Os passos para marcar uma consulta e visualizar as consultas agendas são os mesmos descritos na área do atendente (pagina 40 até a 43) conforme as figuras abaixo (Figura 29 até Figura 35).

Figura 29 – Tela de boas-vindas do paciente



Fonte: Elaborada pelo autor, 2019.

Figura 30 – Tela de listagem das unidades de saúde do paciente

The screenshot shows the 'AgendaUBS' web application interface. At the top, there is a teal header with the logo and a search bar. A dark sidebar on the left contains navigation options: 'Cadastros' and 'Agenda' (with sub-items 'Agendamentos' and 'Consulta / Retorno'). The main content area has a breadcrumb 'Home / Unidades de Saúde' and a list of four health units. Each unit entry includes its name, address, telephone number, and an 'Entrar' button.

Unidade	Rua	Telefone
USF JEREMIAS II	Rua: SAO BENEDITO – JEREMIAS	8333227485
USF ARGEMIRO DE FIGUEREDO	Rua: JOSE GONSALVES DE LUCENA – ROSA CRUZ	8333106175
USF SAO JOSE DA MATA ZONA URBANA I	Rua: : ARNALDO LUIS DE MELO – SAO JOSE DA MATA	8333141236
CENTRO DE SAUDE DA PALMEIRA	Rua: ANA AZEVEDO – PALMEIRA	8333106342

Fonte: Elaborada pelo autor, 2019.

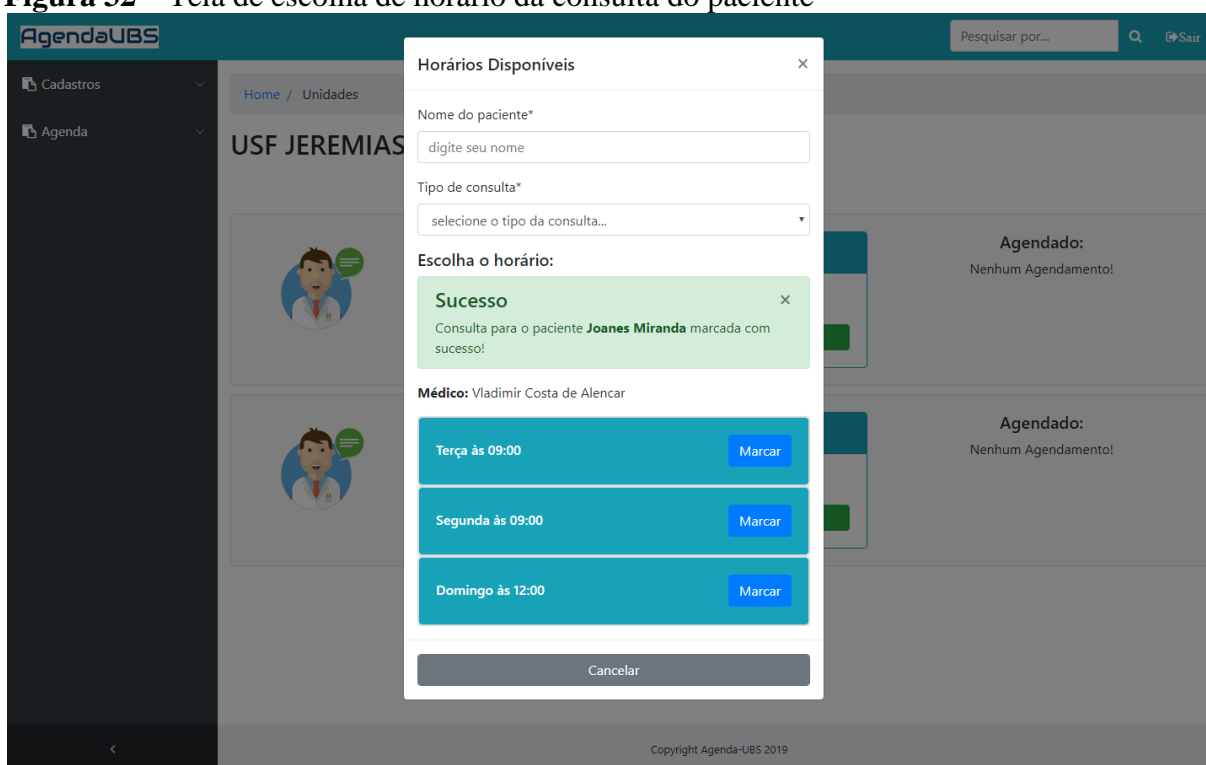
Figura 31 – Tela de marcação de consulta do paciente

The screenshot shows the 'AgendaUBS' web application interface for appointment scheduling. The header and sidebar are consistent with the previous figure. The main content area is titled 'USF JEREMIAS II' and 'Agendamento Online'. It displays two doctor profiles, each with a profile picture, name, specialty, and contact information. For each doctor, the date '21/11/2019' is shown, along with '3 horários disponíveis' and an 'AGENDAR' button. The status 'Agendado: Nenhum Agendamento!' is displayed for both.

Nome	Especialidade	Data	Horários Disponíveis	Status
Vladimir Costa de Alencar	CARDIOLOGISTA	21/11/2019	3 horários disponíveis	Nenhum Agendamento!
Joseane da Silva Miranda	CLINICO GERAL	21/11/2019	3 horários disponíveis	Nenhum Agendamento!

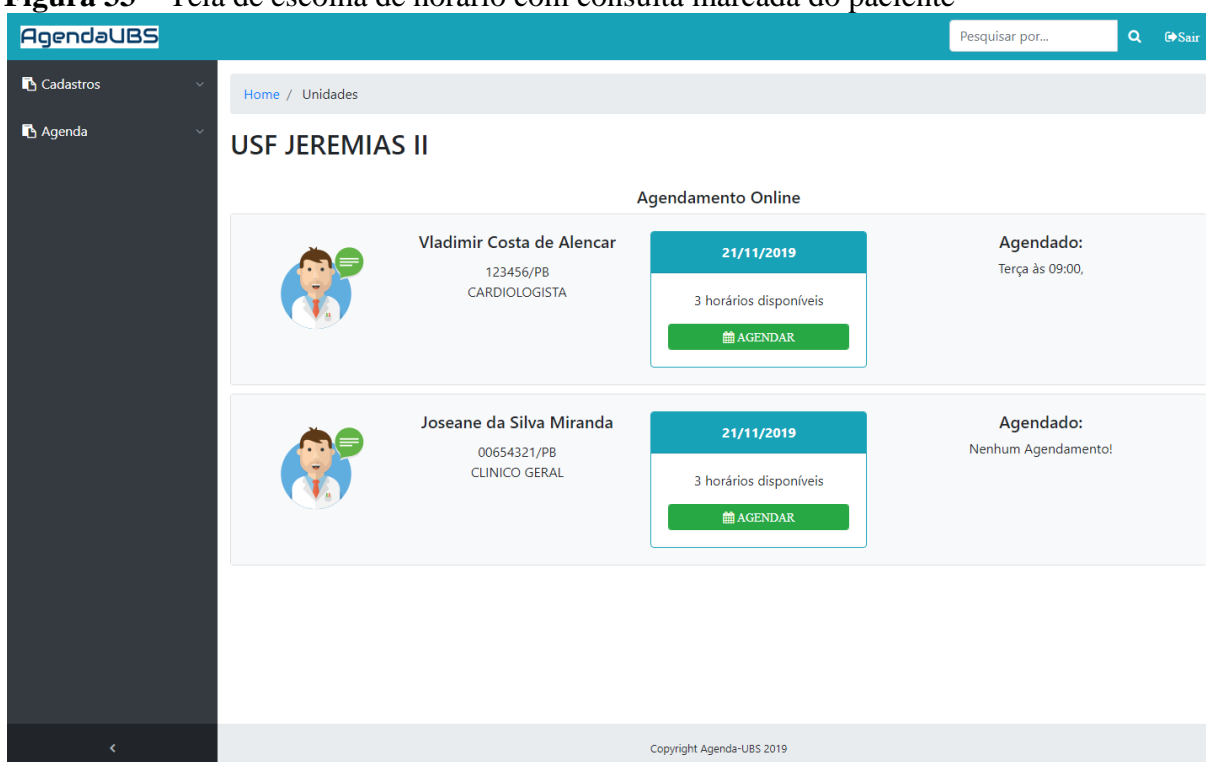
Fonte: Elaborada pelo autor, 2019.

Figura 32 – Tela de escolha de horário da consulta do paciente



Fonte: Elaborada pelo autor, 2019.

Figura 33 – Tela de escolha de horário com consulta marcada do paciente



Fonte: Elaborada pelo autor, 2019.

Figura 34 – Tela de visualização de consultas agendadas do paciente

The screenshot shows the AgendaUBS interface. At the top, there is a search bar with the text 'Pesquisar por...' and a 'Sair' button. A sidebar on the left contains navigation options: 'Cadastros', 'Agenda', 'Agendamentos', and 'Consulta / Retorno'. The main content area is titled 'Consultas agendadas' and displays a table with the following data:

Dia	Hora Atendimento	Paciente	Medico	Tipo	Situação	Ações
Terça	09:00hs	Joanes Miranda	Vladimir Costa de Alencar	nova	Agendado	
Terça	09:00hs	Juliana Miranda	Vladimir Costa de Alencar	retorno	Agendado	
Domingo	12:00hs	Bruno da Silva	Vladimir Costa de Alencar	nova	Agendado	

At the bottom of the page, there is a copyright notice: 'Copyright Agenda-UBS 2019'.

Fonte: Elaborada pelo autor, 2019.

Figura 35 – Comprovante de consulta agendada pelo paciente

The screenshot shows a printed receipt from AgendaUBS. The receipt is titled 'Consulta Marcada' and contains the following information:

AgendaUBS

Consulta Marcada

Nome do Paciente: Joanes Miranda
Medico: Vladimir Costa de Alencar
Especialidade: cardiologista
Horario: Terça às 09:00 hs
Local: USF JEREMIAS II - Campina Grande

At the bottom of the receipt, there is a date '20-11-2019' on the left and a copyright notice 'Copyright AgendaUBS 2019' on the right.

Fonte: Elaborada pelo autor, 2019.

7 CONCLUSÃO

O presente trabalho tem por objetivo, mediante as situações observadas no âmbito da assistência pública a saúde, oferecer a população conforto e praticidades através do sistema de informação criado para marcação de consultas, entre outros fins, proporcionando rapidez e eficácia ao acesso dos usuários em atendimentos de baixa complexidade do sistema único de saúde. Como foi visto, apesar dos avanços obtidos ao longo dos anos para os serviços nessa área, ainda é precária quando se trata em dar assistência às pessoas que a procuram. É preocupante que um país em desenvolvimento ainda tenha grandes obstáculos no âmbito da assistência, seja por motivos de desvios financeiros, má gestão da sistematização do atendimento ou até mesmo pela falta de profissionais de saúde, são fatores relevantes que impossibilitam o atendimento à população. Contudo, é dever de todos ajudar de algum modo o sistema único de saúde para que este serviço não se torne ainda mais comprometido.

Dessa forma, o AgendaUBS (Sistema Gerenciador de Consultas) foi criado para auxiliar na marcação de consultas e controle de atendimento ao usuário. Sendo assim, não se faz necessário o seu comparecimento na unidade de atendimento apenas para marcar uma consulta, somente será necessário a sua presença no dia agendado no sistema, caso não tenha agendamento disponível para a especialidade desejada, é informado ao solicitante dessa indisponibilidade e assim o mesmo não sofrerá com o tempo de espera de uma situação presencial, que causa desgaste físico e emocional, evitando que o mesmo permaneça em uma fila por várias horas aguardando sua vez no atendimento.

Este sistema foi elaborado diante da observação e pesquisas que comprovam a preocupação e desconforto da população que por vezes optam por desistir de um atendimento por ser demorado e estressante, esperando um momento com probabilidade mais favorável. Por isso, o objetivo deste trabalho é facilitar a marcação de consultas no SUS e enfatizar a importância de ter mais estudos nessa área para desenvolver outras formas ou métodos que auxiliem na melhoria da assistência mais humanizada.

Para trabalhos futuros, espera-se que haja melhorias no sistema de acordo com a necessidade dos usuários do SUS, que possibilite facilitar sua utilização promovendo cada vez mais a satisfação da população através da otimização da ferramenta.

REFERÊNCIAS

- AYRES, J. R. de C. M. **Cuidado: tecnologia ou sabedoria prática?** USP, 2000.
- AYRES, J. R. de C. M. **Os sentidos da integralidade na atenção e no cuidado a saúde.** Prefácio à 8ª ed. Rio de Janeiro, 2009.
- BRASIL. CONSELHO NACIONAL DE SECRETÁRIOS DE SAÚDE. **A gestão do SUS/Conselho nacional de secretários de saúde.** – Brasília. 1ª ed. CONASS, 2015.
- CHAUNDHARY, M. **PhpStorm Cookbook. Discover over 80 recipes to learn how to build and test PHP applications efficiently using the PhpStorm IDE.** 1ª ed. Editora Packt Publishing, 2014.
- COELHO, M. O.; JORGE, M. S. B. **Tecnologia das relações como dispositivo do atendimento humanizado na atenção básica à saúde na perspectiva do acesso, do acolhimento e do vínculo.** Revista ciência e saúde coletiva. Ceará, 2009.
- DALL'AGNOL, C. M. et al. **Fatores que interferem no acesso de usuários a um ambulatório básico de saúde.** Revista eletrônica de enfermagem. 2009;11(3):674-80.
- FOURNIER, J. et al. **I can't wait. Advanced acces decreases wait times in primary healthcare.** Revista healthcare quartely vol. 15 No.1, 2012.
- GARCIA, P.; REIS, R. S. **Gestão pública em saúde: sistemas de informação de apoio à gestão em saúde.** Editora EDUFMA. São Luís, 2016.
- GUEDES, G.T.A. **UML 2. Uma abordagem pratica.** 3ª ed. Editora novatec, São Paulo, 2018.
- KEMP, C.; GYGER, B. **Professional heroku programming.** 1ª ed. Editora Wrox, 2013.

LIMA, H. R. C. **Agendamento de consultas pela internet: proposta de um aplicativo de conectividade para a saúde.** Fundação Getúlio Vargas – Escola de administração de empresas – São Paulo, 2002.

LORENZETTI, J. et al. **Tecnologia, inovação tecnológica e saúde: uma reflexão necessária.** Texto & Contexto Enfermagem. Santa Catarina – vol. 21, núm. 2, abril-junho, 2012, pp. 432-439.

MEHTA, C. et al. **MySQL 8 Administrator's Guide. Effective guide to administering high-performance MySQL 8 solutions.** 1ª ed. Editora Packt Publishing Ltd. Birmingham, 2018.

MELO, A. A. de, NASCIMENTO, M. G. F. **PHP Profissional. Aprenda a desenvolver sistemas profissionais orientados a objetos com padrões de projeto.** 1ª ed. Novatec editora, 2007.

Na fila de espera: a realidade enfrentada pela população que busca atendimento.

Disponível em: < <http://www.simers.org.br/noticia/na-fila-de-espera-realidade-enfrentada-pela-populacao-que-busca-atendimento> >. Acesso em: 10 de set. 2019.

NIEDERAUER, J. **Desenvolvimento websites com PHP.** 2ª ed. Novatec editora, São Paulo 2011.

MINISTÉRIO DA SAÚDE. **Cartilha Entendendo o SUS.** Brasília/DF, 2006.

PAIM, J. S. **Sistema único de saúde (SUS) aos 30 anos.** Revista Ciência e saúde coletiva. Bahia, 2018.

PITT, C. **Pro PHP MVC. Everything you need to know about using mvc with php in a single.** 1ª ed. Editora Apress, 2012.

ROCHA, S. A. et al. **Acesso aos cuidados primários de saúde: revisão integrativa.** Physis. Revista de saúde coletiva, Rio de Janeiro, 26(1):87-111,2016.

SARMENTO JR, K. M. de A. et al. **O problema da fila de espera para cirurgias otorrinolaringológicas em serviços públicos.** Revista brasileira de otorrinolaringologia, 2005.

SILVA. C. R. et al. **Dificuldade de acesso a serviços de média complexidade em municípios de pequeno porte: um estudo de caso.** Revista em ciência e saúde coletiva. Paraná, 2016.

Sistema Único de Saúde (SUS): estrutura, princípios e como funciona. Ministério da saúde, 2018. Disponível em: <<https://http://www.saude.gov.br/sistema-unico-de-saude/>>. Acesso em: 16 de ago. de 2019.

VILELA, S. M. P. **Relato de experiência: o problema da fila numa unidade de saúde.** Centro de pesquisas Aggeu Magalhães, Fundação Oswaldo Cruz. Pernambuco, 2010.

APÊNDICE A – REQUISITOS FUNCIONAIS DO SISTEMA

RF001 - Tela de Login

Permite a validação dos dados de acesso do usuário para acesso às funcionalidades do sistema.

RF002 - Tela de Cadastro

Permite ao usuário realizar um cadastro simples para ter acesso às funcionalidades do sistema.

RF003 - Tela de recuperação de senha

Permite ao usuário do sistema recuperar a senha de acesso.

RF004 - Tela Inicial

Esta tela mostrará o usuário logado no sistema e as funcionalidades disponíveis.

RF005 - Área do Paciente

Permite ao paciente visualizar as consultas marcadas, como também agendar uma consulta médica.

RF006 - Área do Atendente

Permite o atendente visualizar os agendamentos, cadastrar um médico cadastrar um horário de atendimento médico e agendar a consulta para um paciente.

RF007 – Área do Médico

Permite ao médico visualizar os atendimentos diários, realizar uma consulta e pesquisar informações sobre a classificação internacional de doenças (cid10).

RF008 – Cadastrar Médico

Permite ao usuário atendente cadastrar um médico no sistema.

RF009 – Cadastrar horário

Permite ao usuário atendente cadastrar um horário de atendimento de um médico.

RF010 - Marcar Consulta Médica

Permite aos usuários atendente e paciente agendar uma consulta médica de acordo com os horários disponíveis nas unidades de saúde.

RF011 – Excluir consultas agendadas

Permite ao usuário atendente excluir uma ou mais consultas agendadas no sistema.

RF012 – Imprimir comprovante de marcação de consulta.

Permite aos usuários atendente e paciente imprimir um comprovante de marcação de consulta, informando o nome do paciente, do médico, local, dia e hora do atendimento.

RF013 – Realizar consulta médica

Permite ao médico preencher um questionário com informações sobre a identificação do paciente, as condições de saúde e definir um diagnóstico ao paciente.

RF014 – Sair

Permite ao usuário sair do sistema encerrando a sessão.

APÊNDICE B – REQUISITOS NÃO FUNCIONAIS DO SISTEMA

RN001 - Fácil Usabilidade

O sistema deve ser de fácil manuseio, permitindo que pessoas possam operá-lo sem treinamento.

RN002 - Utilização do padrão MVC no projeto

O projeto será desenvolvido utilizando o padrão de projeto MVC.

RN003 - Desenvolvimento Back-End

O sistema será desenvolvido utilizando a linguagem de programação PHP e o banco de dados MySQL.

RN004 - Desenvolvimento Front-End

As telas do sistema utilizaram de HTML5, CSS3, bootstrap 4 e JavaScript para validação dos campos de formulários.