

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE GOIÁS –
CÂMPUS DE URUAÇU
DEPARTAMENTO DE ÁREAS ACADÊMICAS
CURSO DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

CHARD WILLIAN NOVAIS MENDONÇA

**DESENVOLVIMENTO DE APLICAÇÃO PARA O GERENCIAMENTO DE
AGENDAMENTOS EM BARBEARIAS**

URUAÇU

2022



INSTITUTO FEDERAL
Goiás

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
SISTEMA INTEGRADO DE BIBLIOTECAS

**TERMO DE AUTORIZAÇÃO PARA DISPONIBILIZAÇÃO
NO REPOSITÓRIO DIGITAL DO IFG - ReDi IFG**

Com base no disposto na Lei Federal nº 9.610/98, AUTORIZO o Instituto Federal de Educação, Ciência e Tecnologia de Goiás, a disponibilizar gratuitamente o documento no Repositório Digital (ReDi IFG), sem ressarcimento de direitos autorais, conforme permissão assinada abaixo, em formato digital para fins de leitura, download e impressão, a título de divulgação da produção técnico-científica no IFG.

Identificação da Produção Técnico-Científica

- | | |
|--|---|
| ☐ Tese | ☐ Artigo Científico |
| ☐ Dissertação | ☐ Capítulo de Livro |
| ☐ Monografia – Especialização | ☐ Livro |
| ☒ TCC - Graduação | ☐ Trabalho Apresentado em Evento |
| ☐ Produto Técnico e Educacional - Tipo: _____ | |

Nome Completo do Autor: Chard Willian Novais Mendonça

Matrícula: 20181050100132

Título do Trabalho: Desenvolvimento de aplicação para o gerenciamento de agendamentos em barbearias

Autorização - Marque uma das opções

1. (x) Autorizo disponibilizar meu trabalho no Repositório Digital do IFG (acesso aberto);
2. () Autorizo disponibilizar meu trabalho no Repositório Digital do IFG somente após a data ____/____/____ (Embargo);
3. () Não autorizo disponibilizar meu trabalho no Repositório Digital do IFG (acesso restrito).

Ao indicar a opção **2 ou 3**, marque a justificativa:

- () O documento está sujeito a registro de patente.
() O documento pode vir a ser publicado como livro, capítulo de livro ou artigo.
() Outra justificativa: _____

DECLARAÇÃO DE DISTRIBUIÇÃO NÃO-EXCLUSIVA

O/A referido/a autor/a declara que:

- i. o documento é seu trabalho original, detém os direitos autorais da produção técnico-científica e não infringe os direitos de qualquer outra pessoa ou entidade;
- ii. obteve autorização de quaisquer materiais inclusos no documento do qual não detém os direitos de autor/a, para conceder ao Instituto Federal de Educação, Ciência e Tecnologia de Goiás os direitos requeridos e que este material cujos direitos autorais são de terceiros, estão claramente identificados e reconhecidos no texto ou conteúdo do documento entregue;
- iii. cumpriu quaisquer obrigações exigidas por contrato ou acordo, caso o documento entregue seja baseado em trabalho financiado ou apoiado por outra instituição que não o Instituto Federal de Educação, Ciência e Tecnologia de Goiás.

Uruaçu, 25/11/2022.
Local Data

Chard Willian Novais Mendonça

Assinatura do Autor e/ou Detentor dos Direitos Autorais

CHARD WILLIAN NOVAIS MENDONÇA

**DESENVOLVIMENTO DE APLICAÇÃO PARA O GERENCIAMENTO DE
AGENDAMENTOS EM BARBEARIAS**

Projeto de implementação apresentado como requisito parcial para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas ao Departamento de Áreas Acadêmicas do Instituto Federal de Educação, Ciência e Tecnologia de Goiás – *Câmpus Uruaçu*.

Orientador: Prof. Me. Alexandre Martins
Ferreira Bueno

URUAÇU

2022

Dados Internacionais de Catalogação na Publicação (CIP)

M539d Mendonça, Chard Willian Novais.
Desenvolvimento de aplicação para o gerenciamento de agendamentos em
barbearias [manuscrito]. / Chard Willian Novais Mendonça. – – Uruaçu:
IFG, 2022.
48 f. : il.
Bibliografia.

Orientador: Prof. Me. Alexandre Martins Ferreira Bueno.

Trabalho de conclusão de curso de graduação em Tecnologia em
Análise e Desenvolvimento de Sistemas – Instituto Federal de Educação,
Ciência e Tecnologia de Goiás, Câmpus Uruaçu, 2022.

1. Sistemas de informação – administração de empresa. 2. Sistemas de
informação – aplicação web. 3. Gestão de agendamentos – barbearia. I.
Bueno, Alexandre Martins Ferreira (orientador). II. Instituto Federal de
Educação, Ciência e Tecnologia de Goiás. III. Título.

CDD 005.7406



INSTITUTO FEDERAL
Goiás

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE GOIÁS
CÂMPUS URUAÇU

Chard Willian Novais Mendonça

DESENVOLVIMENTO DE APLICAÇÃO PARA O GERENCIAMENTO DE AGENDAMENTOS EM BARBEARIAS

Trabalho de Conclusão de Curso apresentado como requisito parcial para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas ao Departamento de Áreas Acadêmicas do Instituto Federal de Educação, Ciência e Tecnologia de Goiás - Campus Uruaçu.

Aprovado em 10 de novembro de 2022.

(Assinado eletronicamente)

Prof. Me. Alexandre Martins Ferreira Bueno
Instituto Federal de Goiás - Campus Uruaçu - Orientador

(Assinado eletronicamente)

**Prof. Me. Maurílio Humberto Rodrigues
Miranda**
Instituto Federal de Goiás - Campus Uruaçu

(Assinado eletronicamente)

Prof. Dr. Lynwood Livi de Souza
Instituto Federal de Goiás - Campus Uruaçu

Documento assinado eletronicamente por:

- Lynwood Livi de Souza, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 30/11/2022 23:53:46.
- Maurílio Humberto Rodrigues Miranda, CHEFE - CD4 - URU-DAA, em 28/11/2022 15:57:50.
- Alexandre Martins Ferreira Bueno, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 26/11/2022 11:46:18.

Este documento foi emitido pelo SUAP em 26/11/2022. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifg.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código Verificador: 349611
Código de Autenticação: f1e194a7c1



Agradeço à Deus, minha família,
professores e amigos.

AGRADECIMENTOS

Agradeço primeiramente à Deus, por me proporcionar todas as forças necessárias para concluir mais uma etapa em minha vida. Agradeço a minha família, em especial minha mãe e meu pai, pelo apoio e incentivos durante essa caminhada. Ao Instituto Federal de Goiás, docentes e servidores que me acompanharam desde o início até agora, pela contribuição à minha formação. Aos amigos e colegas pela amizade, troca de experiências em estudos e apoio. Ao meu orientador Prof. Me. Alexandre Martins, pela orientação, dedicação, motivação, confiança e seriedade para transmitir seu conhecimento.

“No dia em que eu temer, hei de confiar em
ti.”

Salmos 56:3

RESUMO

No presente trabalho, se propõe o projeto de desenvolvimento de uma aplicação *web* para auxiliar na gestão de agendamentos dos serviços prestados em uma barbearia. Atualmente, grande parte das vezes o atendimento ao cliente neste tipo de estabelecimento não é eficaz, haja visto que há uma perda de tempo e/ou demora para que o serviço seja prestado. Isso ocorre por diversos motivos, como: o barbeiro não se encontra no estabelecimento ou já está atendendo outro cliente. O sistema proposto surge com a finalidade de tornar o processo de atendimento e sua gestão mais rápido e eficaz. Para isso, foram feitas observações em estabelecimentos e conversas com os consumidores do serviço prestado. Para o desenvolvimento e implantação da aplicação foi feito o estudo de diversas tecnologias, sendo o método Scrum o escolhido como método de desenvolvimento desse projeto. O desenvolvimento das principais funcionalidades da aplicação foi feito e ela está em fase de testes. Durante todo o processo surgiram algumas dificuldades, como a complexidade em lidar com as diversas tecnologias utilizadas. Como trabalhos futuros, pretende-se incluir meios de pagamento na aplicação para que o cliente já faça o agendamento e o pagamento na mesma e só vá ao estabelecimento receber o serviço que adquiriu.

Palavras-chave: aplicação web, gestão de agendamentos, barbearia.

ABSTRACT

In the present work, we propose a project to develop a web application to assist in the scheduling management for services provided in a barbershop. Currently, most of the time customer service in this type of establishment is not effective, given that there is a waste of time and/or delay for the service to be provided. This occurs for several reasons, such as: the barber is not at the establishment or is already serving another customer. The proposed system arises with the purpose of making the process faster and more efficient. For this, observations were made at establishments and conversations with consumers of the service provided. For the development and implementation of the application, several technologies were studied, and the Scrum method was chosen as the development method for this project. The development of the main features of the application was done and it is in the testing phase. During the whole process some difficulties arose, such as the complexity in dealing with the different Technologies used. As future work, it is intended to include payment methods in the application so that the customer already makes the scheduling and payment in the application and only goes to the establishment to receive the service that he purchased.

Keywords: web application, scheduling management, barbershop.

LISTA DE FIGURAS

Figura 1 – Fluxo de funcionamento do NodeJs.	17
Figura 2 – Fluxo de funcionamento do ReactJs	18
Figura 3 – Fases do Scrum.	21
Figura 4 – Visão geral da arquitetura do sistema	25
Figura 5 – Caso de Uso com as principais funcionalidades.	26
Figura 6 – Fluxograma do processo feito pelo cliente.	29
Figura 7 – Fluxograma do processo feito pelo barbeiro.	30
Figura 8 – Diagrama entidade relacionamento.	31
Figura 9 – Diagrama de Classes.	31
Figura 10 – Wireframe da tela de login do cliente e do barbeiro	32
Figura 11 – Wireframe da tela de cadastro do cliente e do barbeiro	33
Figura 12 – Wireframe da tela de listagem dos agendamentos para o barbeiro	33
Figura 13 – Wireframe da tela de perfil cliente e do barbeiro	34
Figura 14 – Wireframe da tela de listagem dos barbeiros.	35
Figura 15 – Wireframe da tela de agendamento do horário.	36
Figura 16 – Página inicial do sistema	37
Figura 17 – Página de formulário de cadastro do cliente	38
Figura 18 – Página de formulário de cadastro do barbeiro.	38
Figura 19 – Página de edição de perfil	39
Figura 20 – Página inicial do cliente.	39
Figura 21 – Página de agendamento do cliente	40
Figura 22 – Página de sucesso da criação do agendamento.	41
Figura 23 – Página inicial do cliente com opção de exclusão do agendamento.	41
Figura 24 – Página inicial do cliente com opção de edição do agendamento.	42
Figura 25 – Segunda parte da página de edição do agendamento.	43
Figura 26 – Página inicial do barbeiro.	43
Página 27 – Página inicial do barbeiro com opção de cancelamento do agendamento.	44
Figura 28 – Página inicial do barbeiro com o motivo do cancelamento do agendamento.	45
Figura 29 – Página inicial do barbeiro com um cancelamento feito com sucesso.	45

LISTA DE QUADROS

Quadro 1 - Documentação do caso de uso Login.	26
Quadro 2 - Documentação do caso de uso Cadastrar usuário	27
Quadro 3 - Documentação do caso de uso Alterar perfil/cadastro.....	27
Quadro 4 - Documentação do caso de uso Acompanhamento de agendamentos.	27
Quadro 5 - Documentação do caso de uso Agendamento do horário.	28

LISTA DE ABREVIATURAS E SIGLAS

API – Application Programming Interface – Interface de Programação de Aplicação.

CSS – Cascading Style Sheets – Folha de Estilo em Camadas.

HTML – Hyper Text Markup Language – Linguagem de Marcação de Hipertexto.

JSON - JavaScript Object Notation – Notação de objeto JavaScript.

MER – Modelo Entidade Relacionamento.

SGBD – Sistema de Gerenciamento de Banco de Dados.

SQL – Structured Query Language – Linguagem de Consulta Estruturada.

UI – User Interface – Interface de Usuário.

UX – User Experience – Experiência do Usuário.

SUMÁRIO

1 INTRODUÇÃO	13
1.1 OBJETIVOS	13
1.1.1 Objetivo Geral	13
1.1.2 Objetivos Específicos.....	14
1.2 JUSTIFICATIVA	14
2 REFERENCIAL TEÓRICO	15
2.1 HTML.....	15
2.2 CSS	15
2.3 JAVASCRIPT.	16
2.4 NODEJS	16
2.5 REACTJS.	17
2.6 POSTGRESQL	19
2.7 METODOLOGIAS ÁGEIS	19
2.8 SCRUM.	20
3 MÉTODOS E MATERIAIS.....	22
3.1 MÉTODOS	22
3.2 MATERIAIS	22
4 SOFTWARE	23
4.1 ARQUITETURA.....	25
4.2 CASOS DE USO.....	25
4.3 FLUXOGRAMAS.....	29
4.4 MODELO ENTIDADE RELACIONAMENTO.....	30
4.5 DIAGRAMA DE CLASSES	31
4.6 WIREFRAMES.....	31
4.7 INTERFACES DO SISTEMA	37
5 CONCLUSÃO E CONSIDERAÇÕES FINAIS	46
6 REFERENCIAL BIBLIOGRÁFICO	47

1 INTRODUÇÃO

Segundo dados da agência Euromonitor¹, o mercado brasileiro de beleza masculina arrecadou R\$ 19,8 bilhões entre 2012 e 2017. Uma pesquisa feita pelo Instituto Qualibest² em 2017 apontou que 43% dos homens no Brasil se consideram vaidosos e 54% frequentam regularmente salões e barbearias.

Em 2019, a vaidade masculina impulsionou o mercado de barbearias e cosméticos e o Brasil já representava cerca de 13% das vendas mundiais desse mercado, atrás somente dos EUA que detêm 18% de participação. A Euromonitor estima que em 2021 o valor arrecadado ultrapasse os R\$ 26 bilhões no Brasil, e alcance US\$ 78,6 bilhões em 2023 no mundo todo, com o Brasil continuando na segunda posição do ranking mundial.

É sabido que as atuais gerações são marcadas pelo crescente e constante uso da tecnologia e que é comum que as tarefas a serem realizadas no dia-a-dia estejam sendo feitas cada vez mais em questão de alguns cliques na tela. Com a facilidade de um *smartphone* na palma da mão a todo instante, viu-se a oportunidade de se resolver problemas a quilômetros de distância e de forma quase que instantânea, sem a necessidade de deslocamento. Esse que, por muitas vezes, era feito e no final a pessoa acabava não recebendo o atendimento necessário, gerando uma frustração. Tal fato é comum de acontecer em barbearias.

Portanto, entendendo essa realidade desagradável dos atendimentos, o crescimento do mercado de barbearias e pensando em uma otimização e processos simples e rápidos, foi verificada a necessidade de se criar uma aplicação *web* para realizar os agendamentos de serviços dessas empresas, para que tanto o consumidor do serviço quanto o prestador possam ter uma experiência cada vez mais positiva e sem aquela velha e longa corriqueira espera para o atendimento.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Diante da expansão do mercado de beleza masculino e da presença das adversidades apresentadas, este trabalho tem como objetivo geral desenvolver uma aplicação *web* que auxilie

¹ Disponível em: <https://www.terra.com.br/noticias/dino/com-clientes-mais-exigentes-mercado-de-barbearias-segue-em-ascensao,a18706e3d2c19aec6085f85818ac6d20aob99hep.html>.

² Disponível em: <https://www.brazilbeautynews.com/quebra-de-preconceitos-e-busca-por-novidades,1598>.

no gerenciamento dos agendamentos realizados em uma barbearia.

1.1.2 Objetivos Específicos

Tendo como referência o objetivo geral apresentado, este trabalho visa alcançar os seguintes objetivos específicos:

- Reduzir o processo de deslocamento e espera às vezes desagradável do consumidor no local de prestação do serviço;
- Solucionar o problema do consumidor poder fazer seu agendamento e comparecer na hora marcada, já obtendo rapidamente o serviço agendado;
- Facilitar o processo, fazendo com que o consumidor possa realizar o agendamento a qualquer hora do dia ou da semana através da aplicação;
- Fazer com que os conflitos de horários não aconteçam pois, após um agendamento ser feito, outro não poderá ser realizado no mesmo dia e horário para o mesmo prestador do serviço;
- Melhorar a experiência existente entre prestadores de serviço e seus clientes.

1.2 JUSTIFICATIVA

Diante das diversas atividades a serem feitas durante o dia-a-dia, como emprego, faculdade, escola, atividade física ou quaisquer que sejam as preocupações de cada indivíduo, o tempo é valioso para todos. Então, não é agradável gastá-lo em uma fila de espera para receber um serviço de corte de cabelo, por exemplo.

A partir das experiências do autor deste trabalho em diversas barbearias, em especial na Brother's Barbearia, é evidente os problemas relacionados aos atendimentos que são feitos. É comum o cliente ter experiências desagradáveis em uma barbearia como, por exemplo, quando por muitas vezes ele cliente chega no estabelecimento e o barbeiro de sua confiança está livre, porém, aguardando outro cliente que informou que está chegando, sendo preciso esperar. Por algumas vezes, o barbeiro começa o corte de um cliente e o cliente que estava atrasado chega, ficando aquela sensação desagradável para todos os envolvidos.

A proposta de desenvolvimento de uma aplicação *web* justifica-se por motivos de praticidade, uma vez que o consumidor poderá realizar o agendamento em qualquer lugar e a qualquer hora, independentemente do dia da semana; evitar que horários conflituosos aconteçam e a ocorrência de situações desagradáveis (o que gera grande descontentamento); proporcionar satisfação ao consumidor, que ao chegar ao estabelecimento vai ter seu atendimento realizado

da forma esperada; facilitar ao barbeiro o controle de sua agenda e uma melhor organização e estimativa de seus atendimentos.

Diante das justificativas apresentadas, espera-se que através do uso da aplicação proposta ocorra uma maior satisfação de todos os consumidores dos serviços, e que a experiência realizada nestes estabelecimentos seja agradável.

2 REFERENCIAL TEÓRICO

2.1 HTML

A sigla HTML significa Hypertext Markup Language (em português, Linguagem de Marcação de Hipertexto). Ela é uma linguagem na qual são escritas as páginas *web*, utilizada para realizar as marcações de toda a estrutura que está sendo desenvolvida e que será interpretada por navegadores. Ela que irá fazer a conexão de imagens, vídeos, documentos, palavras e etc. Esses elementos não estão conectados linearmente como em um livro ou um passo-a-passo de qualquer tarefa, o que permite essa comunicação de dados é o hipertexto.

O HTML foi criado para ser uma linguagem independente de plataformas, browsers e outros meios de acesso. Interoperabilidade significava menos custo. Você cria apenas um código e este código pode ser lido por diversos meios, ao invés de versões diferentes para diversos dispositivos. Dessa forma, evitou-se que a Web fosse desenvolvida em uma base proprietária, com formatos incompatíveis e limitada. (Ferreira, Eis, [200-?], p.8).

2.2 CSS

A sigla CSS significa, em português, Folhas de Estilo em Cascata. É uma linguagem de estilos que formata a informação entregue pelo HTML. Essa formatação na maioria das vezes é visual, o CSS prepara essa informação para que ela seja consumida da melhor forma possível. Com ela é possível alterar a cor do texto, sua fonte e espaçamento, criar tabelas, dentre várias outras possibilidades.

Essa informação pode ser qualquer coisa: imagem, texto, vídeo, áudio ou qualquer outro elemento criado. Grave isso: CSS formata a informação. Essa formatação na maioria das vezes é visual, mas não necessariamente. (Ferreira, Eis, [200-?], cap.7).

2.3 JAVASCRIPT

O JavaScript também foi utilizado no desenvolvimento do sistema, garantindo dinamicidade na página web, entre outros. É uma linguagem muito popular no desenvolvimento web, suportada em todos os navegadores e responsável por possibilitar qualquer tipo de dinamismo que queira numa página.

Como o próprio nome já diz, o JavaScript é uma linguagem de script, ele é uma linguagem de programação que permite ao programador implementar funcionalidades mais complexas em páginas web. Além disso, está em constante evolução e desde sua criação possui vários avanços, não sendo mais uma linguagem somente de scripts, mas também uma linguagem de consulta em bancos de dados por exemplo (Flanagan, 2011).

JavaScript permite criar pequenos programas embutidos no próprio código de uma página HTML e capazes de gerar números, processar alguns dados, verificar formulários, alterar valor de elementos HTML e criar elementos HTML. Tudo isso diretamente no computador cliente, evitando a troca de informações com o servidor e o tempo passa a depender somente do processamento local do cliente, não mais da latência da rede. (Grillo, Fortes, 2008, p. 4)

Outra característica do JavaScript é que ele não depende de compilações para ser executado, ou seja, o código é interpretado e executado conforme lido pelo navegador, assim como o HTML.

2.4 NODE.JS

O Node.js³ é um ambiente de tempo de execução assíncrono orientado a eventos e foi projetado para construir aplicativos de rede escalonáveis. Isso é possível graças à união do ambiente de execução de JavaScript fornecido pelo próprio Node.js e o motor de interpretação e execução de JavaScript presente no Google Chrome, chamado de V8.

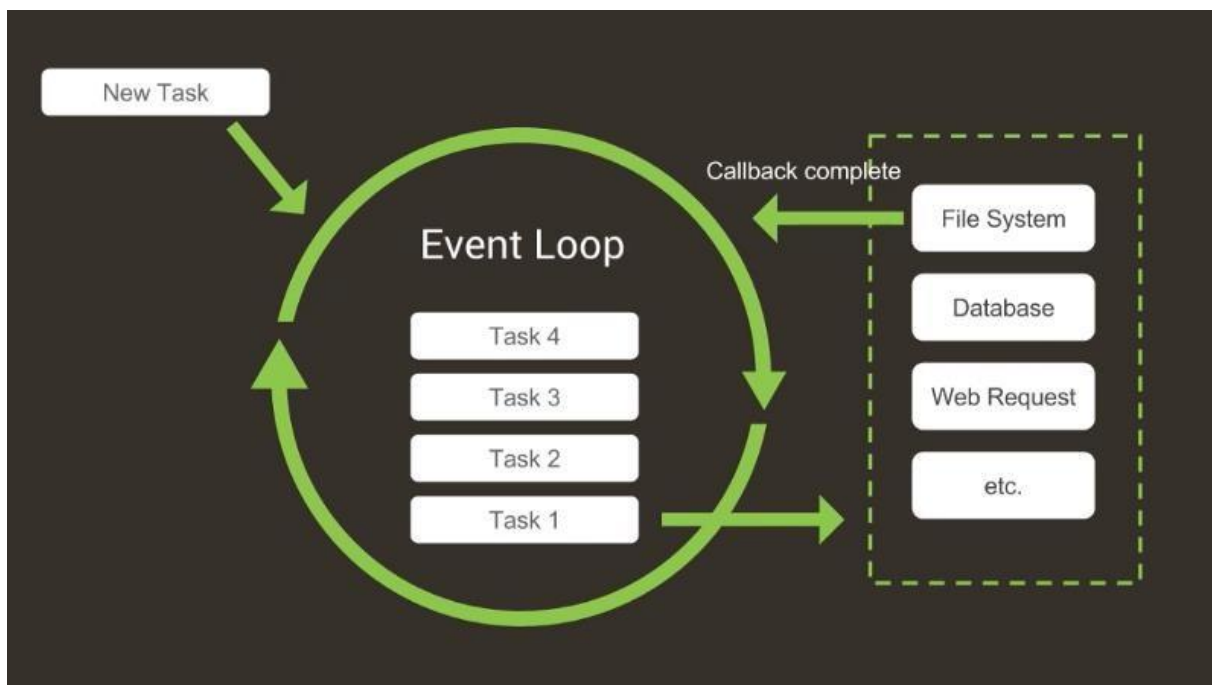
É uma plataforma *open-source* que permite a execução de código JavaScript a nível de *front-end* e *back-end*. Neste trabalho, ele será utilizado na API que será utilizada pelo *front-end* das aplicações.

O que torna o Node.js uma ferramenta poderosa é que ele é simples, uma vez que a linguagem usada é popular e você consegue levar essa simplicidade ao servidor, além de não possuir dependências e em poucos minutos se consegue ter o ambiente para uso da linguagem configurado e pronto. Neste contexto, um conceito importante é o de Non-blocking I/O, ou seja, pode-se realizar tarefas em concorrência, estando-se livre da preocupação de travar o processo

³ Disponível em: <https://nodejs.org/en/about/>

utilizado e, uma vez que não há bloqueios, sistemas escaláveis são muito razoáveis para se desenvolver em Node.Js.

Figura 1 – Fluxo de funcionamento do NodeJs.



Fonte: https://developer.mozilla.org/pt-BR/docs/Learn/Server-side/Express_Nodejs/Introduction

Na Figura 1 é possível verificar o funcionamento dos ciclos no Node.Js, toda vez que uma nova tarefa é iniciada, ela entra então nos ciclos de eventos e aguarda ser “solicitada”, ou seja, aguarda sua vez, e assim que chega esse momento a tarefa é executada e entrega o retorno necessário.

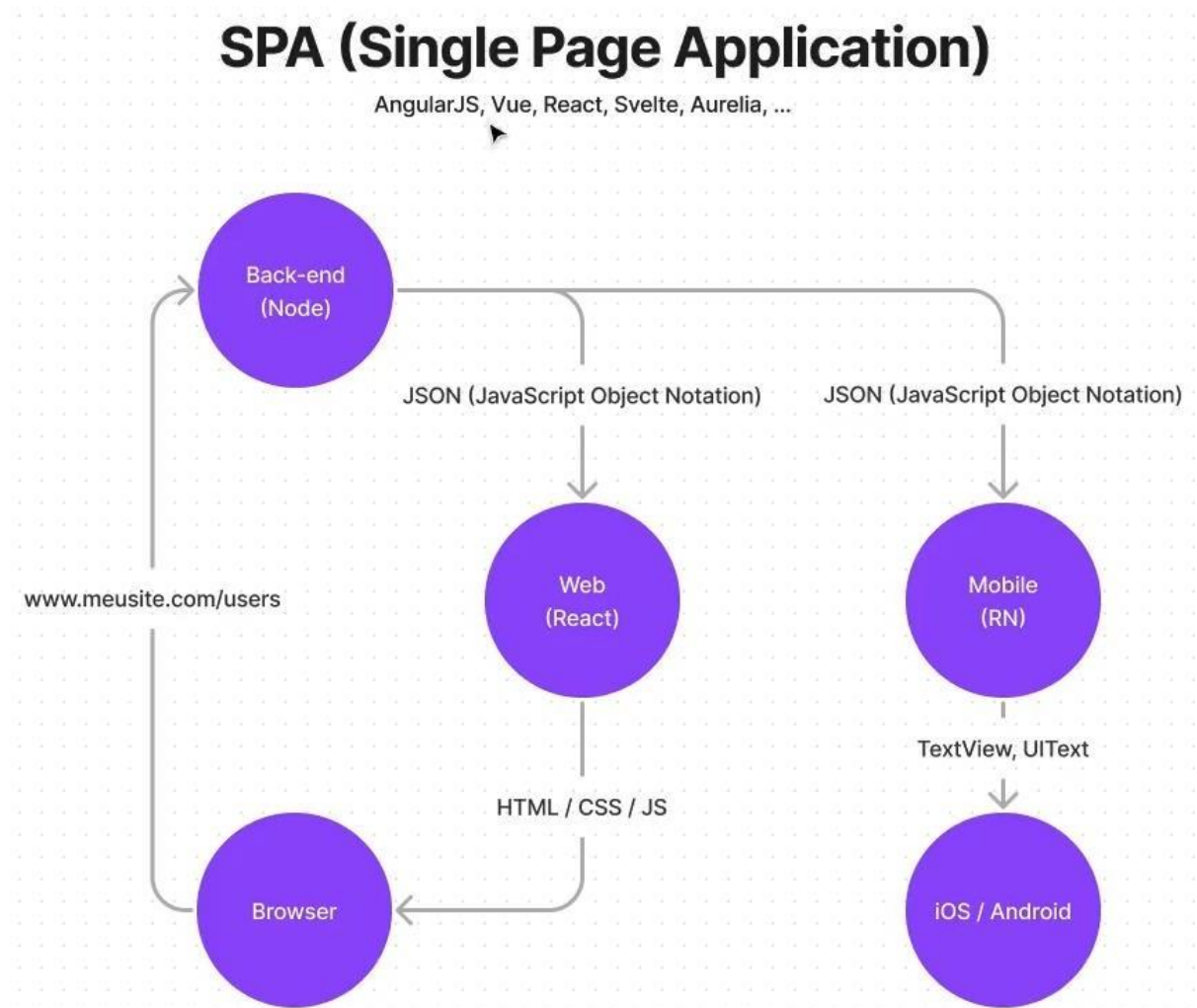
2.5 REACT JS

React JS⁴ é uma biblioteca JavaScript para a criação de interfaces de usuário. Foi criada em 2011 pelo time do Facebook e surgiu com o objetivo de otimizar a atualização e a sincronização de atividades simultâneas no *feed* de notícias da rede social, como chat, status e listagem de contatos. Todas essas atividades, chamadas de estados, tinham uma descrição complexa e com o React, tornaram-se mais simples, como também a conexão entre HTML, CSS e JavaScript.

⁴ Disponível em: <https://pt-br.reactjs.org/>

Devido ao seu sucesso, ele foi incorporado à interface do Instagram, que é outra rede social do grupo, e em 2013 seu código foi aberto a comunidade, dando início à sua popularização. No React tudo é componente e pode-se trabalhar em cima de cada um deles de maneira individual, além do que se pode reaproveitar esse componente e padronizar a interface de uma maneira muito simples.

Figura 2 – Fluxo de funcionamento do ReactJs.



Fonte: <https://pt-br.reactjs.org/docs/getting-started.html>

Na figura acima é possível verificar como funciona a biblioteca do ReactJs, através do browser vai ser solicitado um endereço web que irá chamar um endpoint do *back-end*, nesse exemplo o *back-end* é construído em NodeJs, o mesmo irá retornar os dados solicitados na forma de JSON(JavaScript Object Notation), por fim o ReactJs irá pegar esses dados que chegou e através do Html, Css e Javascript irá criar a interface para o usuário.

2.6 POSTGRESQL

O banco de dados escolhido para ser utilizado no desenvolvimento da aplicação foi o PostgreSQL⁵.

Existem diversos bancos de dados no mercado cuidando de informações e, para que eles funcionem conforme necessário, têm-se os chamados SGBDs (Sistemas de Gerenciamento de Banco de Dados). Um SGBD é uma coleção de programas que permite ao usuário manipular banco de dados para as mais diversas finalidades. O PostgreSQL é um SGBD relacional, uma vez que seus dados são organizados em formas de tabelas (linhas e colunas).

Além de ser *open-source*, seguro e ágil, ele também é simples e robusto. Entre suas características principais estão:

- Sistema de controle de acesso robusto e segurança em nível de coluna e linha;
- Confiabilidade e recuperação a desastres;
- Compatibilidade com várias linguagens, entre elas Java, PHP, Python, Ruby e C/C++;
- Tabelas com tamanho de até 32 TB e campos de até 1 GB;
- Suporte a recursos como *triggers*, *views*, *stored procedures*, *SSL*, *MVCC*, *schemas*, *transactions*, *savepoints*, *referential integrity* e expressões regulares.
- Instruções em SQL, como indica o nome do próprio SGBD.

2.7 METODOLOGIAS ÁGEIS

No desenvolvimento dos sistemas foi utilizada uma metodologia ágil, visto que o mercado está dinâmico e exigindo muita agilidade. Sommerville (2011) ressalta:

Os métodos ágeis são métodos de desenvolvimento incremental em que os incrementos são pequenos e, normalmente, as novas versões do sistema são criadas e disponibilizadas aos clientes a cada duas ou três semanas. Elas envolvem os clientes no processo de desenvolvimento para obter feedback rápido sobre a evolução dos requisitos. Assim, minimiza-se a documentação, pois se utiliza mais a comunicação informal do que reuniões formais com documentos escritos. (Sommerville, 2011, p. 39)

A metodologia ágil se encaixa muito bem neste trabalho, pois é visto com bons olhos uma entrega rápida e com eficácia o que, por consequência, trará uma maior satisfação do cliente/stakeholder.

⁵ Disponível em: <https://www.postgresql.org/>

2.8 SCRUM

Existem diversas metodologias ágeis disponíveis no mercado atualmente, como XP, Crystal, MSF, entre outras. Neste projeto será utilizada a metodologia Scrum⁶, que tem por proposta que um projeto seja dividido em pequenos ciclos de atividades, com reuniões frequentes para que a equipe possa alinhar as tarefas que estão sendo feitas, além de pensar formas de melhorar o processo com agilidade. A Figura 3 ilustra as principais fases deste método.

Existem alguns termos presentes no vocabulário de desenvolvimento do Scrum necessários para se entender suas fases. São eles:

- **Sprints:** É o nome dado para os ciclos de cada projeto, em geral são ciclos mensais e são determinados para que as tarefas sejam realizadas;
- **Product Backlog:** É o nome dado para o conjunto de objetivos de um projeto. É o nome dado ao pacote de funcionalidades a serem desenvolvidas;
- **Sprint Planning Meeting:** São as reuniões periódicas que acontecem no início de cada sprint, ou ciclo, para planejar e priorizar os itens do Product Backlog que serão desenvolvidos naquele período;
- **Sprint Backlog:** São as tarefas específicas que serão desenvolvidas em cada ciclo;
- **Daily Scrum:** É a reunião para acompanhamento do projeto, a ideia é que toda a equipe se reúna diariamente para discutir as atividades desenvolvidas e priorizar o trabalho naquele dia;
- **Sprint Review Meeting:** É a reunião que acontece no final de cada ciclo para que a equipe apresente o que foi realizado e os resultados daquele ciclo.

Uma das regras mais importantes da metodologia é que todos no projeto saibam o que está sendo feito e que as atividades de cada ciclo sejam mostradas para toda a equipe de forma visual. Existem três papéis fundamentais no gerenciamento de projetos com Scrum, são eles:

- **Product Owner:** É o ponto central de liderança sobre o produto, é o responsável por decidir o que será feito e em qual ordem de prioridade. É de sua responsabilidade

⁶ Disponível em: <https://www.treasy.com.br/blog/scrum/>

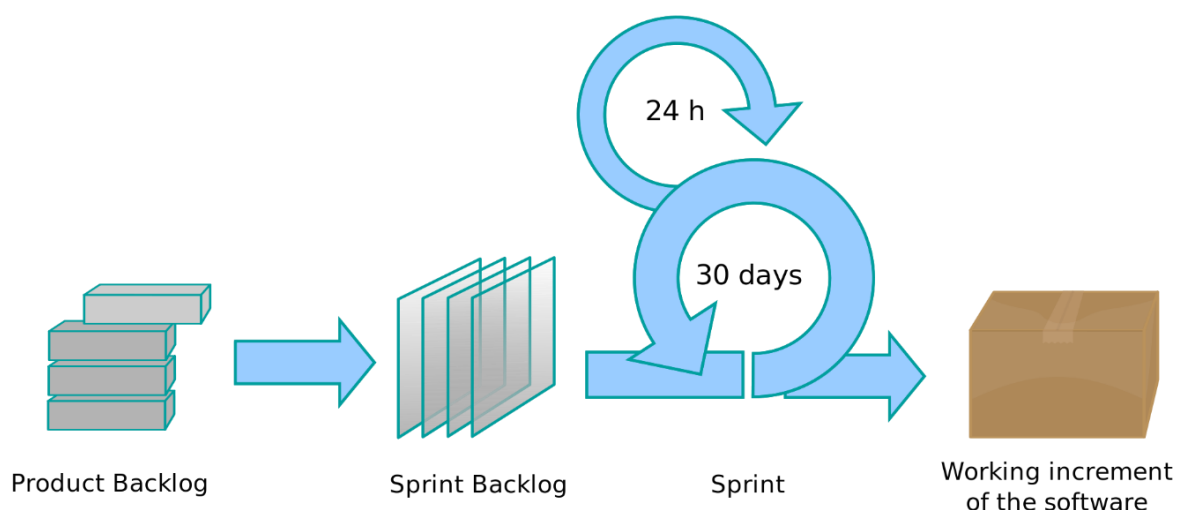
comunicar todos os outros participantes, uma visão clara do que a equipe está buscando alcançar. Ele é responsável pelo sucesso global do projeto;

- **Scrum Master:** É responsável por ajudar todos os envolvidos a entender e abraçar os valores, princípios e práticas do Scrum. Tem um importante papel para ajudar a resolver problemas e proteger a equipe de interferências externas que possam atrapalhar a produtividade;
- **Time Scrum:** São as pessoas que formam aquela equipe, geralmente entre 5 e 10 pessoas.

Dentre os principais pontos da metodologia pode-se destacar:

- **As reuniões diárias:** Rápidas e objetivas, em apenas 15 minutos é possível repassar os principais pontos;
- **Mostrar o trabalho de forma visual usando o quadro Kanban Scrum:** Um enorme painel com todas as atividades e o status de cada uma no qual todos podem ver as tarefas;
- **Equipes enxutas:** Estudos comprovam que equipes menores trabalham mais rápido e melhor do que grandes times.

Figura 3 – Fases do Scrum.



Fonte: <https://www.treasy.com.br/blog/scrum/>

3 MÉTODOS E MATERIAIS

3.1 MÉTODOS

Para o desenvolvimento do sistema foi utilizada a metodologia ágil Scrum. Essa metodologia facilitou o desenvolvimento do sistema e as alterações que ocorreram no decorrer de seu desenvolvimento.

Para o início do desenvolvimento, foi realizada uma análise de como funciona todo o processo do atendimento numa barbearia, desde a etapa de marcação do horário de atendimento até o serviço ser realizado. Para tal, foram feitas observações na barbearia Brother's, entrevistas informais com os prestadores de serviço e também com clientes do estabelecimento.

Diante dessas informações, começou-se a criação da especificação do sistema com os diagramas de casos de uso, fluxograma de como serão as atividades e também o Modelo de Entidade Relacionamento (MER) para a criação do banco de dados do sistema.

A próxima etapa foi o estudo das tecnologias HTML, CSS, JAVASCRIPT, REACT JS e NODE JS que seriam necessárias durante todo o processo de desenvolvimento do software.

A partir daí começou-se a programação das funcionalidades, utilizando-se das fases do Scrum. Foram definidas algumas funcionalidades a serem feitas, começando pelo *back-end* com o NodeJs: criação dos usuários, a parte de autenticação dos mesmos e a criação do banco de dados. Logo após, foi iniciada a parte do *front-end* com o ReactJs, criando as primeiras telas de login e cadastro integradas com o *back-end* e, em seguida, uma rodada de testes para verificação se o funcionamento do que foi programado estava de acordo com o esperado. A próxima etapa foi criar a parte do agendamento em si, alteração do perfil no *back-end* e, novamente, integração ao *front-end* e mais uma rodada de testes. Surgiram propostas de melhorias na aplicação, como: envio de e-mails para os clientes, para isso foram feitas novas alterações na documentação e após a implementação da funcionalidade, também foram feitos testes para garantir o funcionamento integral do software. O procedimento de alterações e testes foi repetido a cada nova necessidade de criação/alteração das funcionalidades.

3.2 MATERIAIS

Para o desenvolvimento do sistema, foram utilizados os seguintes recursos:

- Um computador, com processador Intel Core i5 6400, memória RAM HyperX DDR4

16GB, armazenamento interno de 1TB e sistema operacional Windows 10 64 bits. Este computador foi usado para o desenvolvimento dos sistemas e para o desenvolvimento da monografia exigida por esse projeto.

- Um smartphone IOS, modelo Iphone 7, 128GB de armazenamento interno, 2GB de memória RAM e tela de 4,7 polegadas.

Para o desenvolvimento do sistema também foi preciso atender às necessidades das tecnologias descritas anteriormente. Para isso, o ambiente de desenvolvimento precisou ter instalado e configurado nele os seguintes recursos: React, Node.js, Insomnia e o SGBD PostgreSQL, todos podendo ser encontrados em suas respectivas documentações oficiais. Foi utilizada a IDE Visual Studio Code para edição dos códigos do sistema e o framework Insomnia para os testes do código *back-end*.

Foi de suma importância o uso desses recursos para que o sistema fosse desenvolvido e testado. Ademais, esses recursos foram inteiramente próprios.

4 SOFTWARE

O sistema que foi desenvolvido neste trabalho irá gerenciar o fluxo de trabalho na barbearia Brother's. As informações e dados cadastrados referentes aos agendamentos serão armazenados em um banco de dados PostgreSQL. O objetivo desse trabalho é permitir uma melhor experiência para os clientes da barbearia.

Na aplicação será possível o cliente se cadastrar; conferir os barbeiros que prestam serviços na barbearia, podendo então, escolher o de sua preferência e visualizar quais horários o mesmo tem disponível; realizar, alterar ou cancelar um agendamento; e visualizar seu próprio perfil. Também será permitido que o barbeiro acompanhe quais os horários já estão agendados e qual usuário agendou em determinado momento, além de visualizar e alterar os dados do seu próprio perfil. Os agendamentos poderão ser feitos de 30 em 30 minutos ou seja, às 08:00, 08:30, 09:00 e assim sucessivamente, até o horário de fechamento da barbearia.

Desta forma, o sistema permitirá que um cliente faça o agendamento de um horário na barbearia sem a necessidade de se deslocar ou ligar no estabelecimento. Ele também terá a liberdade de fazer o agendamento em qualquer dia e qualquer lugar.

Atendendo às necessidades citadas anteriormente, o sistema apresentará as seguintes funcionalidades:

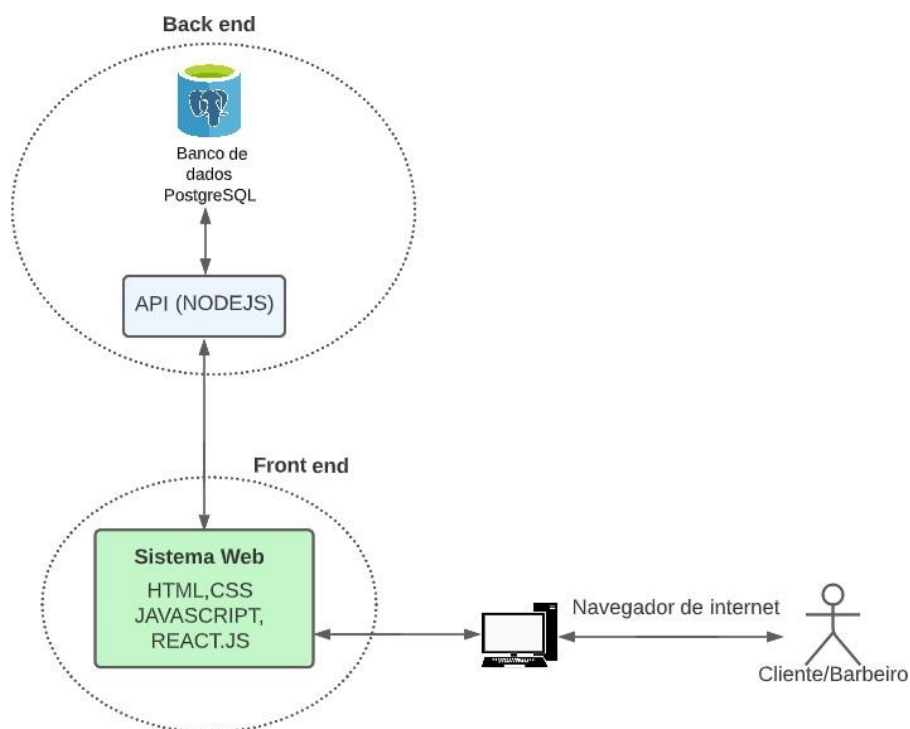
- Login para acesso ao sistema;
- Cadastro do cliente e do barbeiro;
- Acompanhamento dos agendamentos pelo barbeiro;
- Agendamento do horário para realização do serviço;
- Alteração dos dados do perfil/cadastro.

É importante ressaltar o uso da técnica de segurança e criptografia utilizada no desenvolvimento da aplicação, o JWT token, o que garante que o usuário com acesso seja de fato um usuário válido. Foi utilizada uma biblioteca chamada bcrypt para criptografar as senhas dos usuários, utilizando-se de métodos hash para criar sequências aleatórias e difíceis de “quebrar”. Para as validações de formulário foi utilizada uma biblioteca bastante conhecida e utilizada, o Yup, ela informa em tela erros de preenchimento de formulário, por exemplo. Foi utilizado da melhor forma possível as tags do HTML para garantir acessibilidade para usuários portadores de deficiência, que se utilizam de leitores de tela. Utilizando-se de boas práticas de usabilidade e fazendo uso de User Interface (UI) e User Experience (UX), foi definida uma fonte e tamanho adequados, a disposição dos botões em tela, o espaçamento entre os objetos em tela, as cores utilizadas e seus contrastes.

4.1 ARQUITETURA

Para uma melhor compreensão do funcionamento da aplicação, a Figura 4 ilustra uma visão geral do sistema.

Figura 4 – Visão geral da arquitetura do sistema.



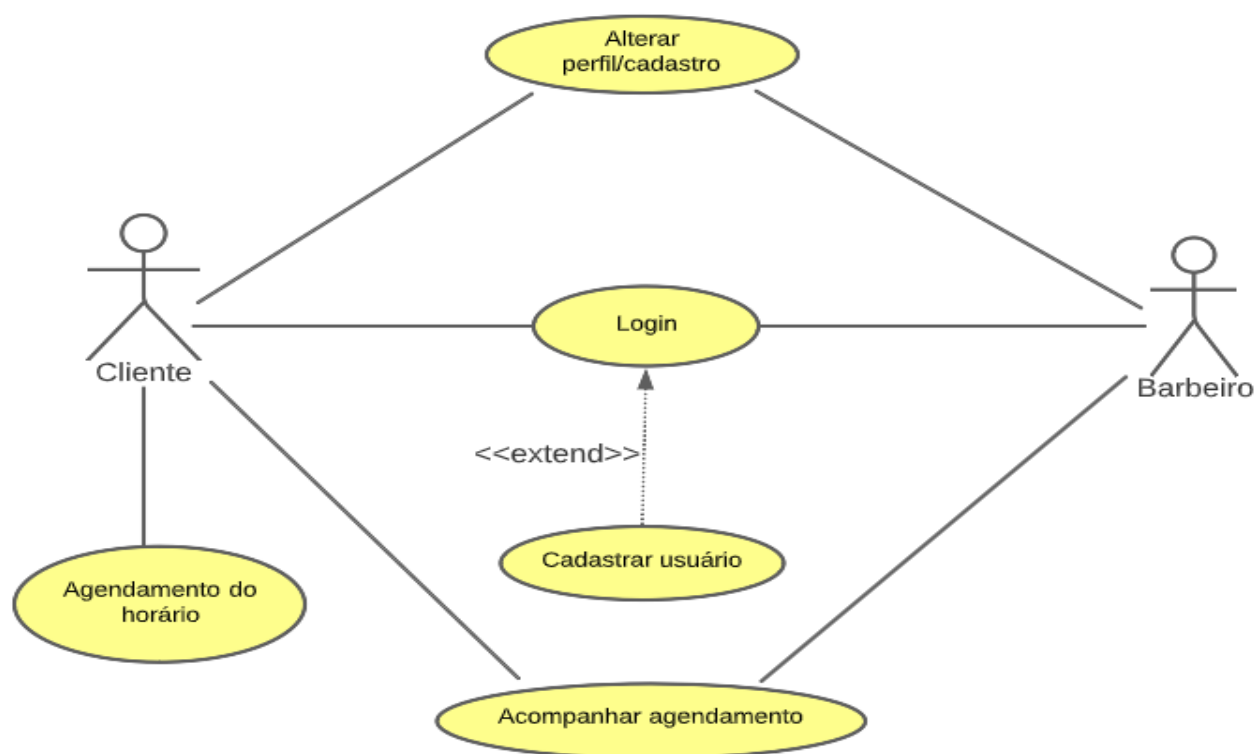
Fonte: Próprio autor

Como pode ser visualizado, os usuários irão se comunicar com a aplicação através de um navegador de Internet. De acordo com o que for solicitado na interface do sistema, será feito então as tratativas necessárias pela API do sistema (construída utilizando o Node.js), que também realizará toda a comunicação com o banco de dados PostgreSQL. Após isso, a API retornará os dados que serão organizados na interface do usuário.

4.2 CASOS DE USO

O diagrama de casos de uso basicamente documenta as funcionalidades que o sistema possui do ponto de vista do usuário. Ele é escrito usando atores externos, casos de uso e os relacionamentos. Seu objetivo geral é apresentar uma visão ampla das funcionalidades que o sistema oferecerá aos usuários e representar como os casos de uso interagem entre si no sistema. (LUCIDCHART, [200-?]).

Figura 5 - Caso de Uso com as principais funcionalidades



Fonte: Próprio autor.

No diagrama apresentado na Figura 5 pode-se visualizar as principais atividades que o sistema oferece aos usuários, mostrando as participações dos envolvidos no processo. Os quadros 1, 2, 3, 4 e 5 apresentam a documentação referente a essas funcionalidades.

Quadro 1 – Documentação do caso de uso Login.

Nome do caso de uso	Login
Caso de uso geral	
Ator principal	Cliente, Barbeiro
Atores secundários	
Resumo	Este caso de uso é responsável pelo Login dos usuários no sistema, para terem acesso ao mesmo.
Pré-condições	Estar cadastrado no sistema.
Pós-condições	
Ações do ator	Ações do sistema
1 – Inserir e-mail e senha.	
	2 – Dar acesso ao usuário caso os dados informados sejam válidos.
Fluxo Alternativo I – Cadastrar usuário	
Caso o usuário não esteja cadastrado, executar o caso de uso “Cadastrar usuário”.	
Restrições/Validações	

Quadro 2 – Documentação do caso de uso Cadastrar usuário.

Nome do caso de uso	Cadastrar usuário
Caso de uso geral	
Ator principal	Cliente, Barbeiro
Atores secundários	
Resumo	Este caso de uso é responsável pelo cadastro dos usuários no sistema, para terem acesso ao mesmo.
Pré-condições	
Pós-condições	
Ações do ator	Ações do sistema
1 – Para barbeiro, inserir nome, e-mail, senha e tipo barbeiro (administrador ou não). Para cliente, inserir nome, e-mail, telefone, data de aniversário e senha.	
	2 – Realizar as verificações para o cadastramento na base de dados.
	3 – Cadastrar o usuário na base de dados.
Restrições/Validações	1 – Não permitir cadastro de um e-mail já cadastrado anteriormente. 2 – Apenas um barbeiro com perfil de administrador pode cadastrar outros barbeiros.

Quadro 3 – Documentação do caso de uso Alterar perfil/cadastro.

Nome do caso de uso	Alterar perfil/cadastro
Caso de uso geral	
Ator principal	Cliente, Barbeiro
Atores secundários	
Resumo	Este caso de uso é responsável pela alteração dos dados dos perfis dos usuários.
Pré-condições	Estar logado no sistema.
Pós-condições	
Ações do ator	Ações do sistema
1 – Preencher os campos que se deseja fazer alterações: nome, e-mail, a senha e/ou avatar.	
	2 - Realizar as alterações e atualizar o perfil do usuário.
Restrições/Validações	

Quadro 4 – Documentação do caso de uso Acompanhar agendamento.

Nome do caso de uso	Acompanhar agendamento
Caso de uso geral	
Ator principal	Barbeiro
Atores secundários	

Resumo	Este caso de uso é responsável pela listagem dos agendamentos que estão marcados para que o barbeiro saiba os horários de seu atendimento.
Pré-condições	Estar logado no sistema.
Pós-condições	
Ações do ator	Ações do sistema
1 – Selecionar o dia desejado para que a listagem seja feita.	
	2 – Mostrar os agendamentos, horários e qual cliente agendou aquele serviço.
Fluxo Alternativo I – Cancelar agendamento	
1 – O barbeiro seleciona o dia.	
	2 – O sistema faz a listagem dos agendamentos marcados.
3 – O barbeiro solicita o cancelamento de um agendamento, informando o motivo para o cancelamento.	
	4 – O sistema realiza o cancelamento do agendamento.
Restrições/Validações	

Quadro 5 – Documentação do caso de uso Agendamento do horário.

Nome do caso de uso	Agendamento do horário
Caso de uso geral	
Ator principal	Cliente.
Atores secundários	
Resumo	Este caso de uso é responsável pela listagem dos barbeiros que estão disponíveis e de quais horários e dias eles prestam serviço.
Pré-condições	Estar logado no sistema.
Pós-condições	
Ações do ator	Ações do sistema
1 – Seleciona o barbeiro desejado.	
	2 – Exibir a lista de horários e calendário dos dias.
3 – Seleciona o dia e o horário.	
	4 – Realizar o agendamento, mostrar uma confirmação do horário e dia marcado para aquele usuário e enviar um e-mail para o usuário com os detalhes do agendamento.
Fluxo Alternativo I – Cancelar agendamento	
1 – O cliente seleciona o dia e horário agendado e solicita seu cancelamento	

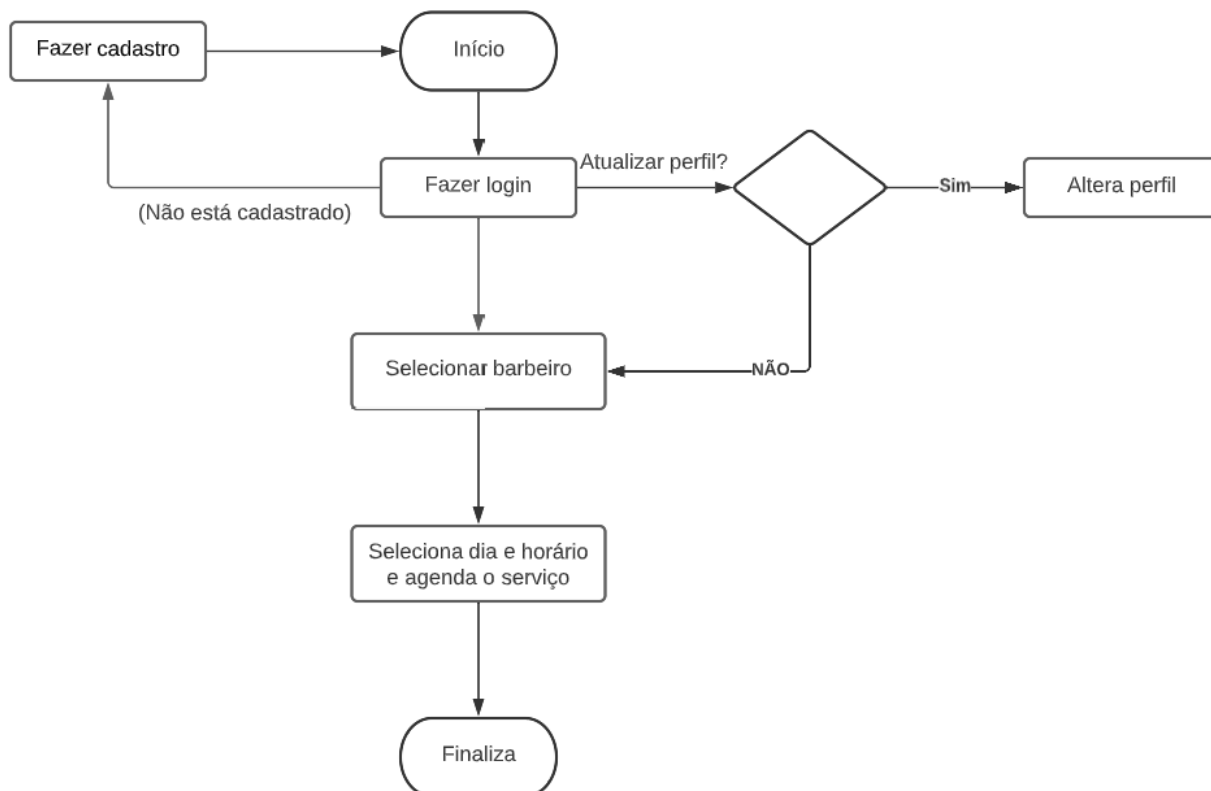
	2 – O sistema realiza o cancelamento e permite que o horário anteriormente agendado esteja disponível novamente.
Fluxo Alternativo II – Editar agendamento	
1 – O cliente seleciona o dia e horário agendado e solicita a alteração para novo horário e dia.	
	2 – O sistema permite que o horário inicial esteja disponível novamente e realiza o novo agendamento.
Restrições/Validações	

4.3 FLUXOGRAMAS

Um fluxograma é um diagrama que descreve um processo, um sistema, ou até mesmo um algoritmo de computador. Eles são usados em diversas ocasiões e em várias áreas de estudo, seja para documentação, planejamento, estudos, entre outros. Suas etapas são ilustradas por meios de figuras geométricas ligadas entre si, para sua representação (LUCIDCHART, [200-?]).

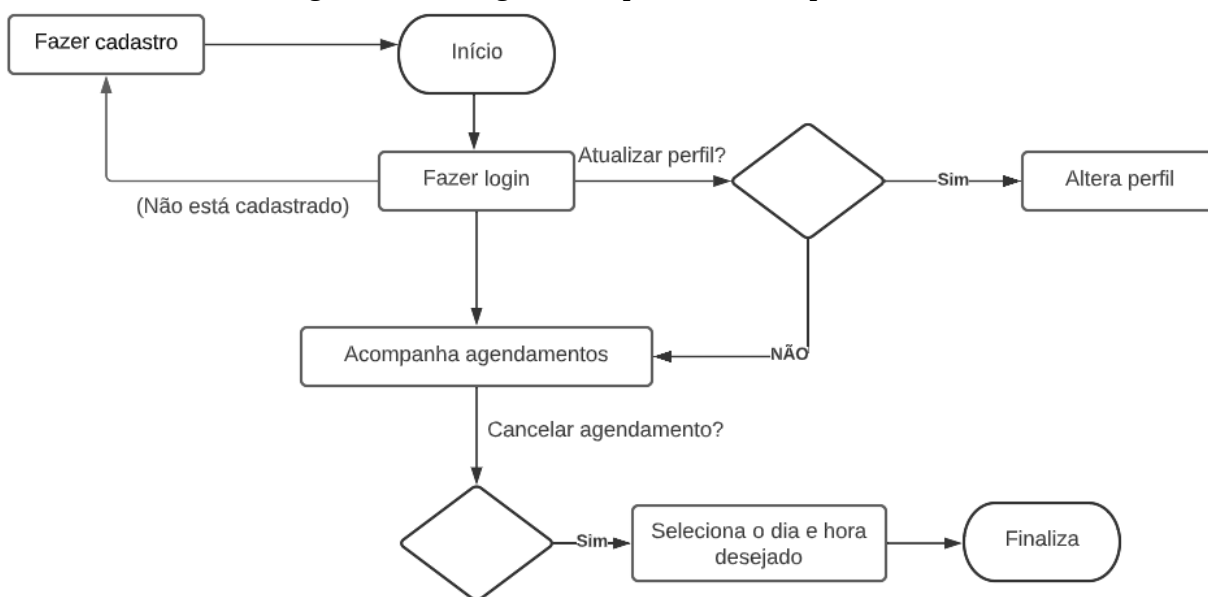
Para entender melhor como o software irá funcionar, foi criado um fluxograma comum demonstrando o processo de agendamento feito pelo cliente e outro com o processo feito pelo barbeiro.

Figura 6 – Fluxograma do processo feito pelo cliente.



Fonte: Próprio autor.

Figura 7 – Fluxograma do processo feito pelo barbeiro.



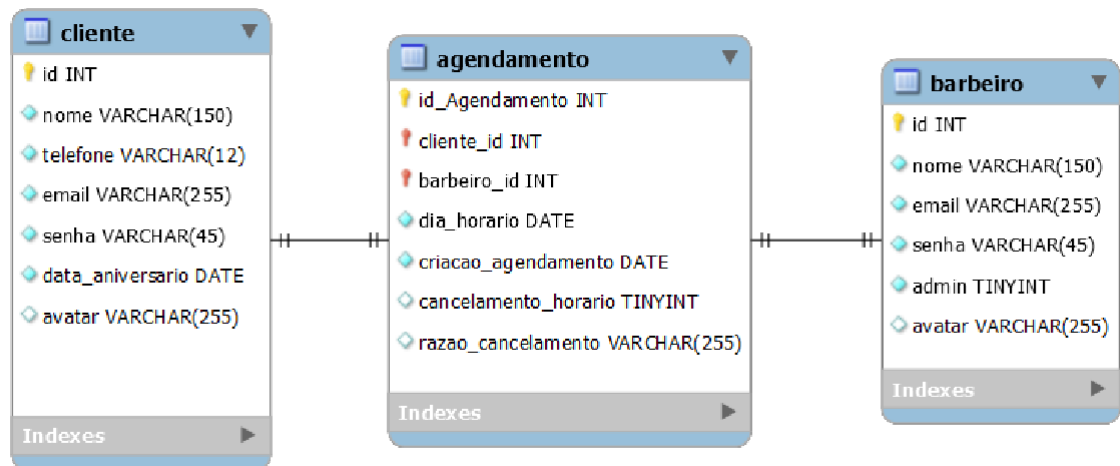
Fonte: Próprio autor.

4.4 MODELO DE ENTIDADE RELACIONAMENTO

O Modelo de Entidade Relacionamento (MER)⁷ serve para facilitar o entendimento de como o sistema irá funcionar, ilustrando como pessoas, objetos e o próprio sistema interagem entre si. Com esse modelo fica simples e visível as tabelas, relacionamentos e seus atributos. Em outras palavras, ele mostra a estrutura que possuirá o banco de dados das aplicações.

⁷ Disponível em: <https://www.devmedia.com.br/modelo-entidade-relacionamento-mer-e-diagrama-entidade-relacionamento-der/14332>

Figura 8 – Diagrama entidade relacionamento.



Fonte: Próprio autor.

4.5 DIAGRAMA DE CLASSES

O diagrama de classes tem como objetivo principal a especificação dos componentes do software e como eles se relacionam, ou seja, sua estrutura. A Figura 6 mostra o diagrama de classes dos principais componentes do sistema que foi desenvolvido.

Figura 9 – Diagrama de Classes



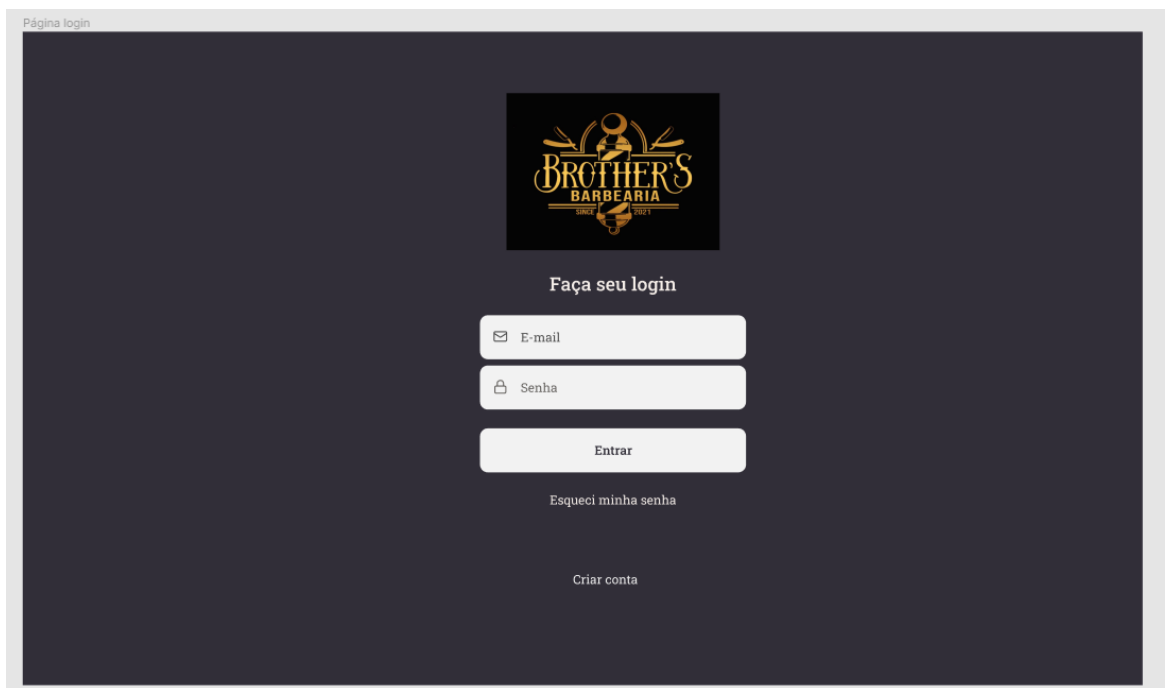
Fonte: Próprio autor.

4.6 WIREFRAMES

Segundo Becker (2019), wireframes são esqueletos ou protótipos de um projeto cuja intenção é demonstrar a estrutura e layout da página e como os elementos se encontram nela.

Ele não possui cores, podendo ser feito com lápis e borracha ou com algumas ferramentas que oferecem esse recurso. São muito úteis para validar conceitos com usuários realizando testes de usabilidade. As figuras 10 a 15 apresentam algumas telas do sistema proposto, criadas com o auxílio da ferramenta Figma.

Figura 10 – Wireframe da tela de login do cliente e do barbeiro.



Fonte: Próprio autor (2021).

Figura 11 – Wireframe da tela de cadastro do cliente e do barbeiro.

Cadastro

BROTHER'S BARBEARIA
SINCE 1985

Faça seu cadastro

Nome

E-mail

Senha

Cadastrar

[Voltar para o login](#)

Fonte: Próprio autor (2021).

Figura 12 – Wireframe da tela de listagem dos agendamentos para o barbeiro.

Dashboard

BROTHER'S BARBEARIA

Bem vindo

Julho de 2021

Horários agendados

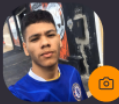
	Chard Willian	08:00
	Bheatriz M.	08:30
	Luciana Novais	11:00
	Novais M.	13:00
	B. Mendonça	15:00
	C. Mendonça	17:30

Fonte: Próprio autor (2021).

Figura 13 – Wireframe da tela de perfil do cliente e do barbeiro.

Perfil

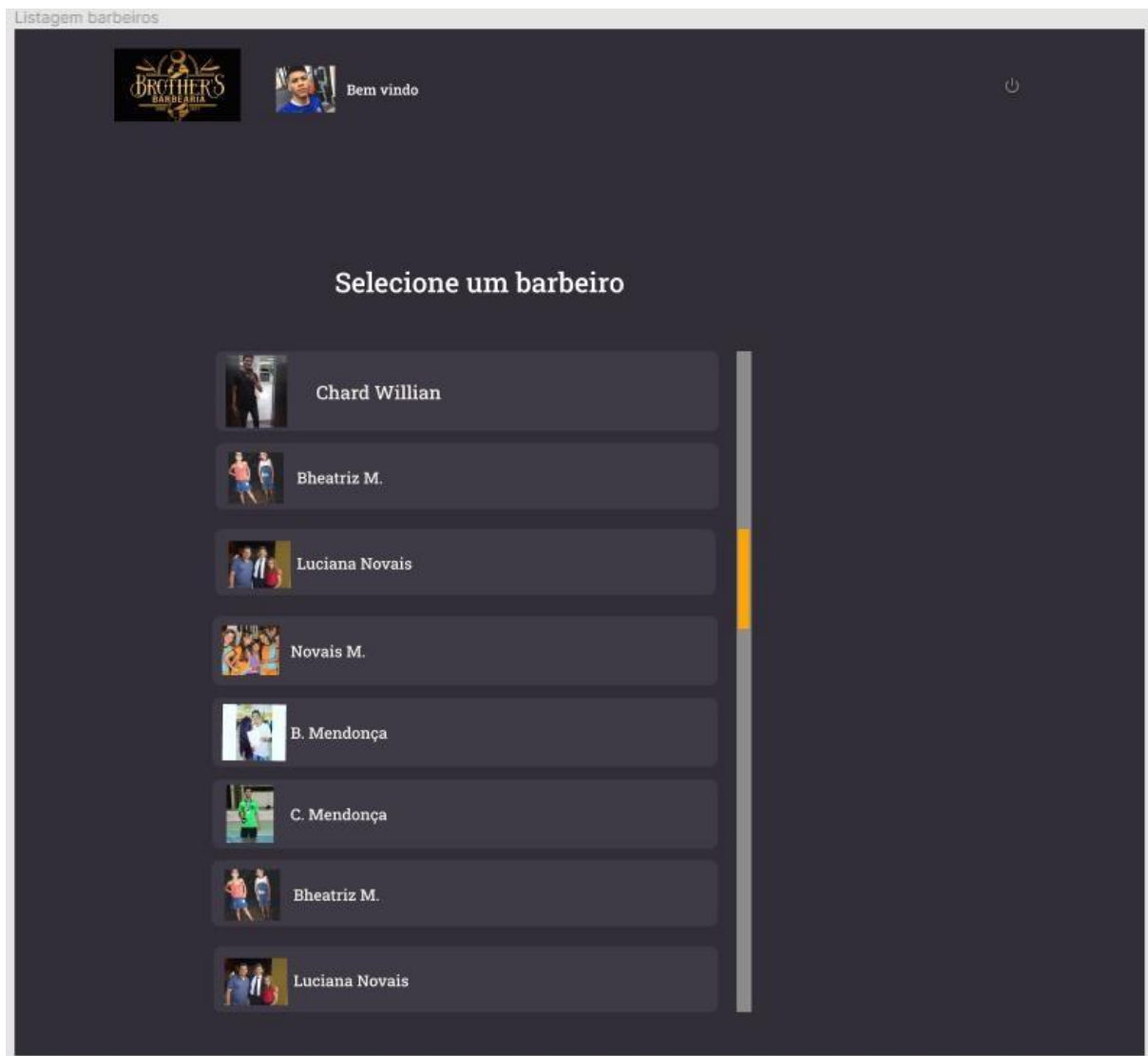
←



Meu perfil

Fonte: Próprio autor (2021).

Figura 14 – Wireframe da tela de listagem dos barbeiros.



Fonte: Próprio autor (2021).

Figura 15 – Wireframe da tela de agendamento do horário.

agendamento

 Bem vindo



[← Voltar](#)

Escolha o horário

Manhã

09:00

11:30

12:00

Tarde

12:00

13:30

14:00

15:00

17:30

Noite

19:00

19:30

Escolha a data

←

Julho de 2021

→

D	S	T	Q	Q	S	S
			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	

Agendar

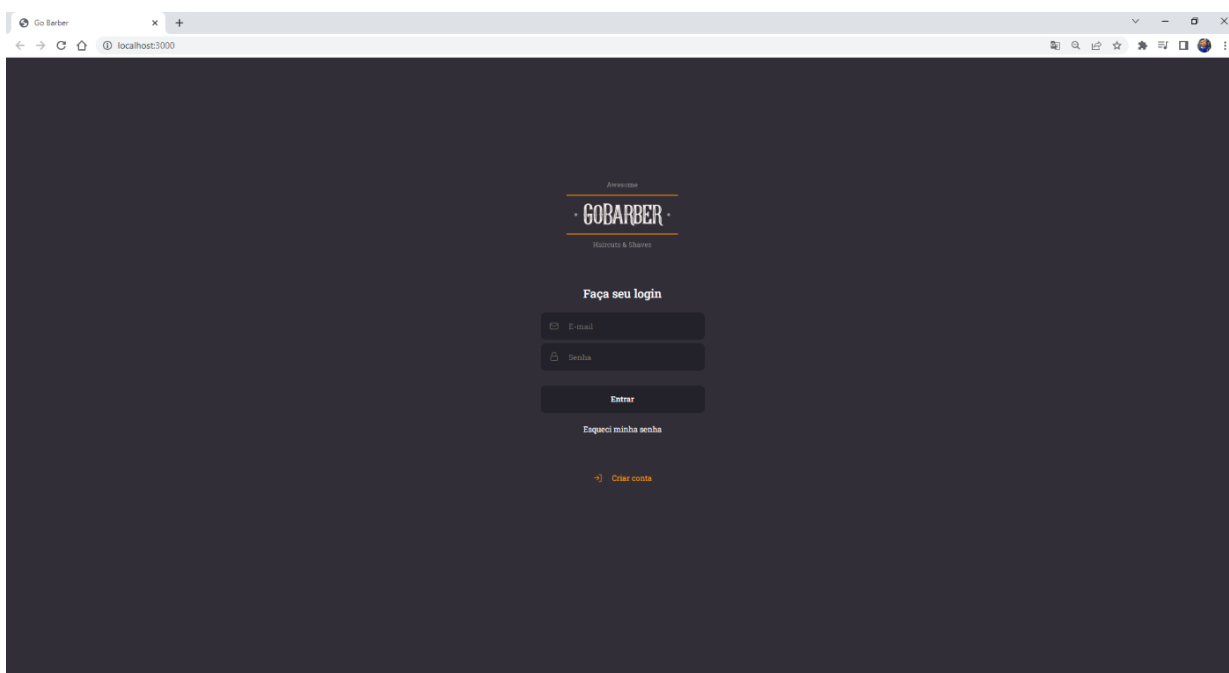
Fonte: Próprio autor (2021).

4.7 INTERFACES DO SISTEMA

Tendo como base os wireframes criados e apresentados na seção anterior, foram criadas as interfaces do sistema (foram realizadas mudanças e adaptações em relação aos protótipos no decorrer do desenvolvimento da aplicação). As telas a seguir mostram as interfaces do sistema construídas, apresentando as principais funcionalidades nele presentes.

A Figura 16 mostra a página inicial do sistema, que possui os links para o usuário fazer login ou se registrar no sistema. Nesta tela também há um link para recuperação de senha, na qual o usuário pode clicar em “Esqueci minha senha”, solicitar a recuperação e lhe será enviado um e-mail para que crie uma nova senha.

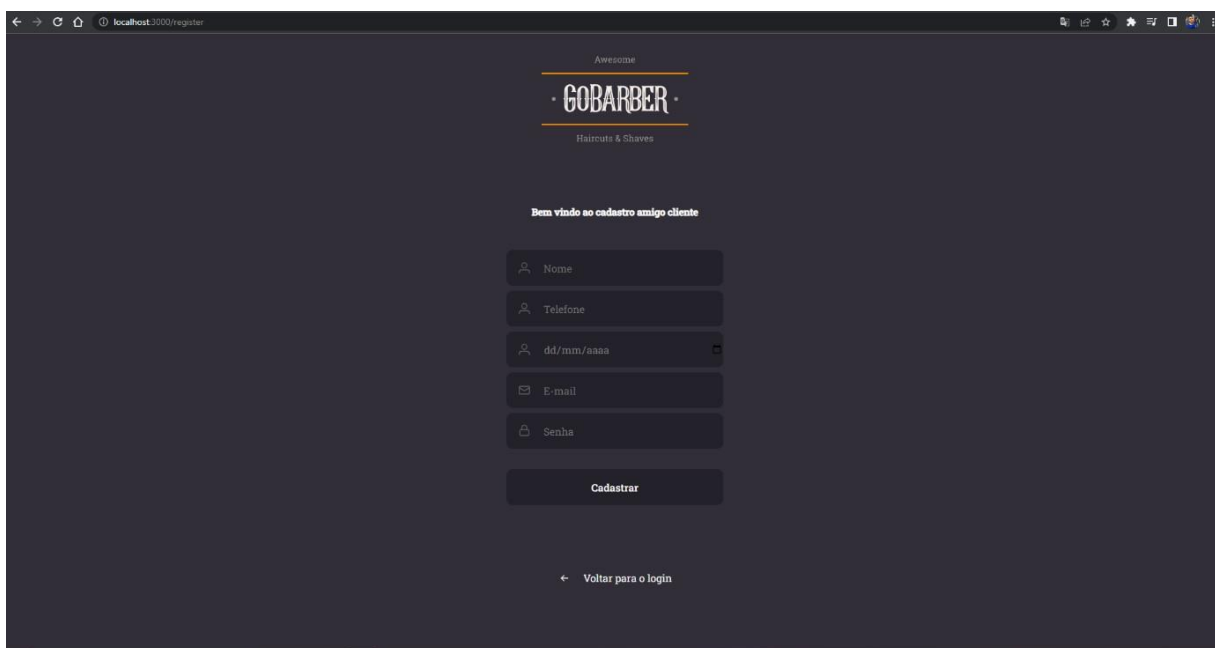
Figura 16 – Página inicial do sistema.



Fonte: Próprio autor (2022).

A Figura 17 mostra a página de formulário de cadastro do cliente, onde ele preencherá seus dados e submeterá o formulário para se registrar no sistema. Após isso, ele receberá um e-mail informando sobre a criação de sua conta.

Figura 17 – Página de formulário de cadastro do cliente.

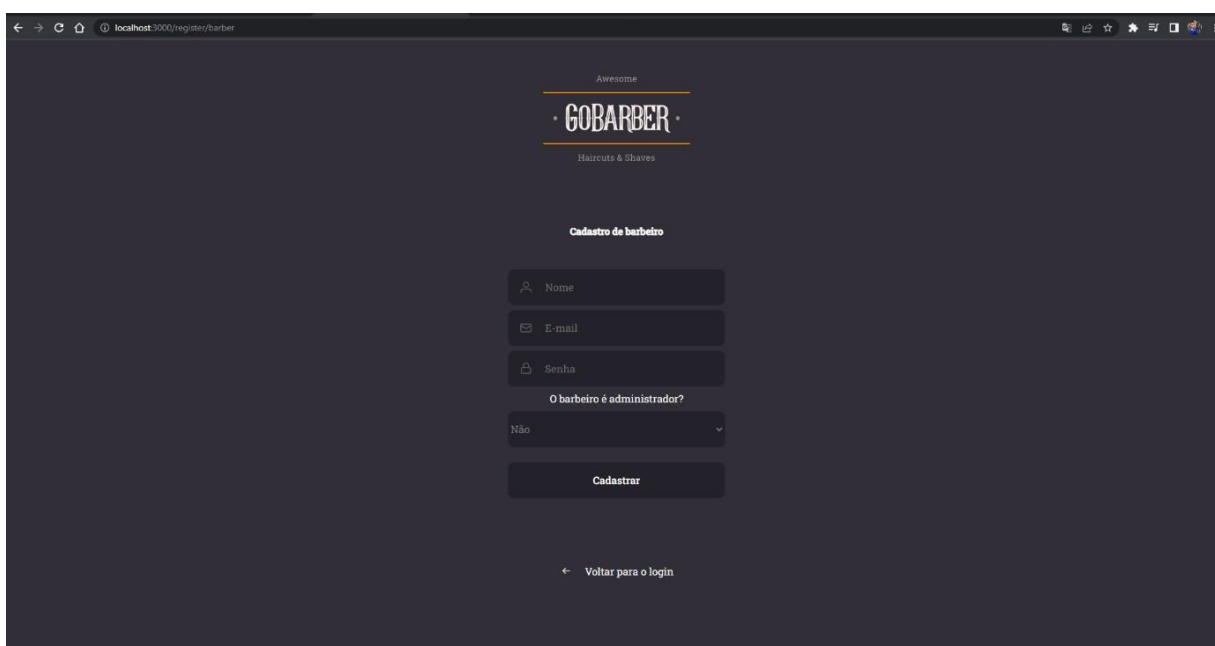


The screenshot shows a web browser window with the URL `localhost:3000/register`. The page has a dark theme and features the GOBARBER logo at the top, which includes the text "Amescom" above "GOBARBER" and "Haircuts & Shaves" below it. The main heading is "Bem vindo ao cadastro amigo cliente". The form contains five input fields: "Nome" (with a person icon), "Telefone" (with a phone icon), "dd/mm/aaaa" (with a calendar icon and a date picker), "E-mail" (with an envelope icon), and "Senha" (with a lock icon). A "Cadastrar" button is positioned below the fields. At the bottom, there is a link "← Voltar para o login".

Fonte: Próprio autor (2022).

A Figura 18 mostra o formulário de cadastro do barbeiro, no qual um barbeiro administrador poderá cadastrar outro barbeiro para trabalhar em seu estabelecimento, podendo inclusive dar acesso de administrador para o mesmo. Após isso, ele receberá um e-mail informando sobre a criação de sua conta.

Figura 18 – Página de formulário de cadastro do barbeiro.



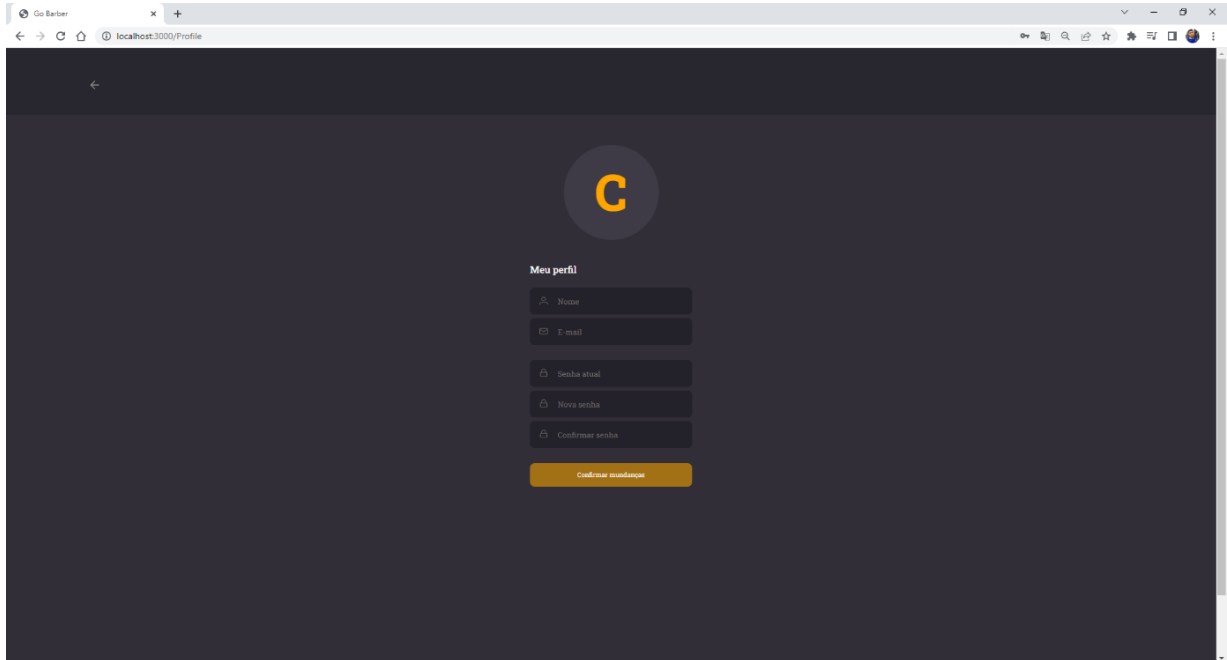
The screenshot shows a web browser window with the URL `localhost:3000/register/barber`. The page has a dark theme and features the GOBARBER logo at the top, which includes the text "Amescom" above "GOBARBER" and "Haircuts & Shaves" below it. The main heading is "Cadastro de barbeiro". The form contains three input fields: "Nome" (with a person icon), "E-mail" (with an envelope icon), and "Senha" (with a lock icon). Below these fields is a dropdown menu labeled "O barbeiro é administrador?" with "Não" selected. A "Cadastrar" button is positioned below the dropdown. At the bottom, there is a link "← Voltar para o login".

Fonte: Próprio autor (2022).

A Figura 19 mostra a página de edição do perfil, na qual o cliente ou o barbeiro podem

fazer alterações como, por exemplo, a mudança de senha.

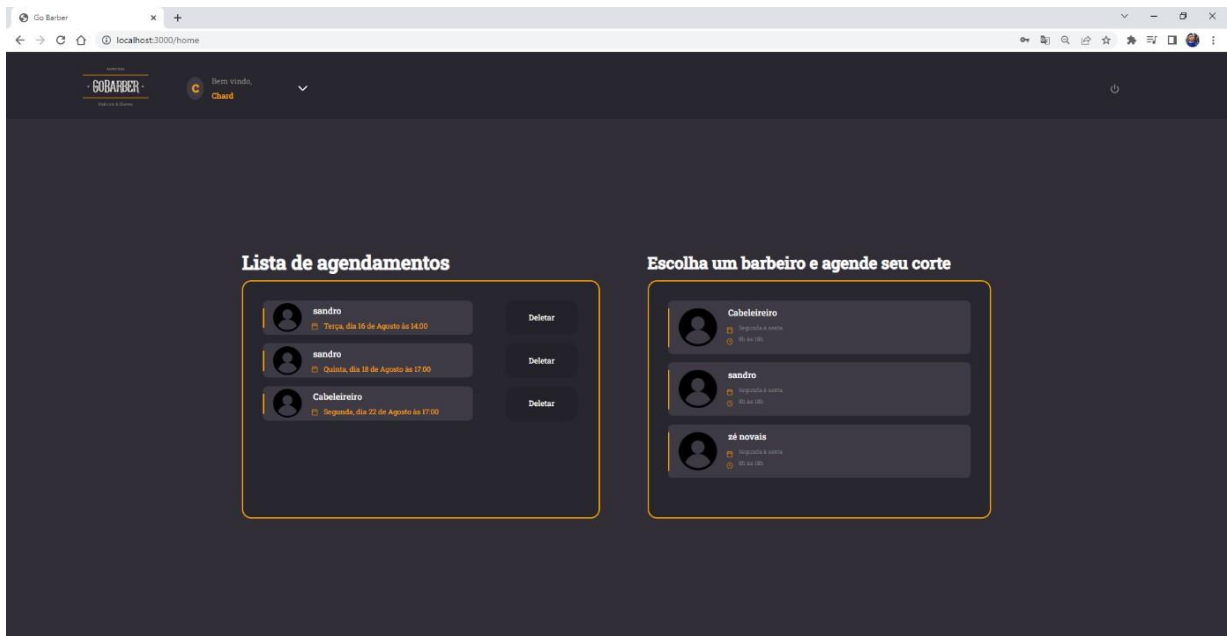
Figura 19 – Página de edição de perfil.



Fonte: Próprio autor (2022).

A Figura 20 mostra a página inicial do cliente logado, na qual ele pode ver os agendamentos que realizou e pode escolher um barbeiro para ir à página de agendamento.

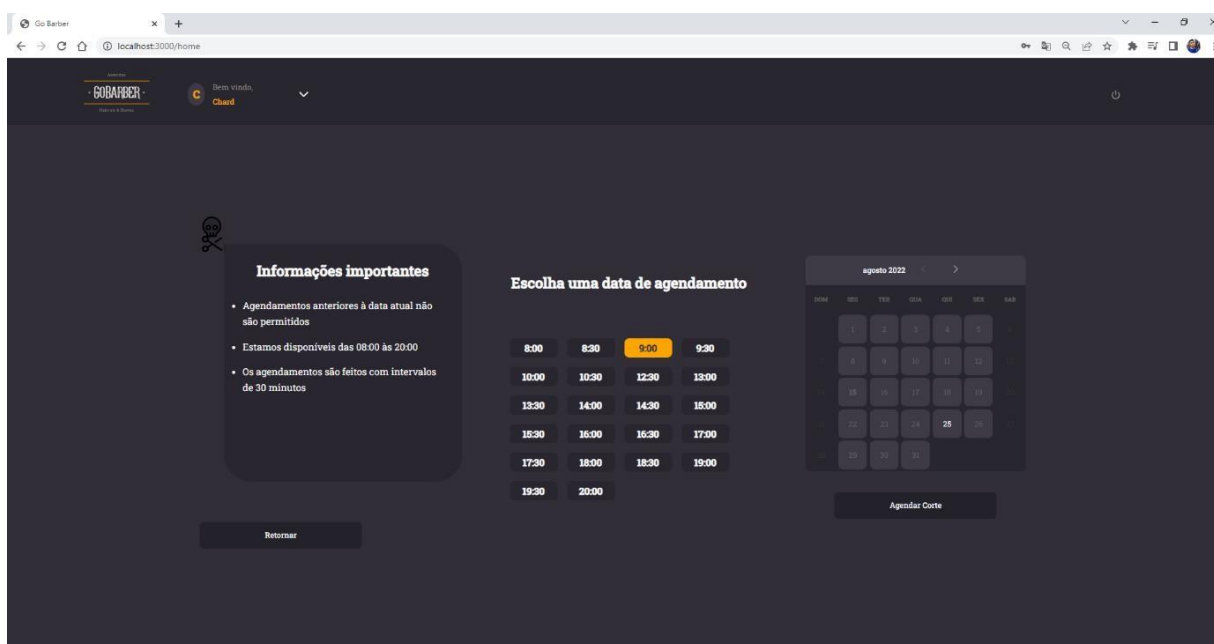
Figura 20 – Página inicial do cliente.



Fonte: Próprio autor (2022).

A Figura 21 mostra a página do agendamento em si (após o cliente ter selecionado o barbeiro na sua página inicial), onde o cliente escolherá o horário e o dia em que deseja realizar o serviço. Caso ele tente criar um agendamento com um barbeiro em uma data e horário já preenchidos, o sistema não permitirá a criação do agendamento e irá aparecer uma mensagem informando que a data já foi reservada. Após o agendamento ser feito, será enviado um e-mail informando os dados da criação desse agendamento.

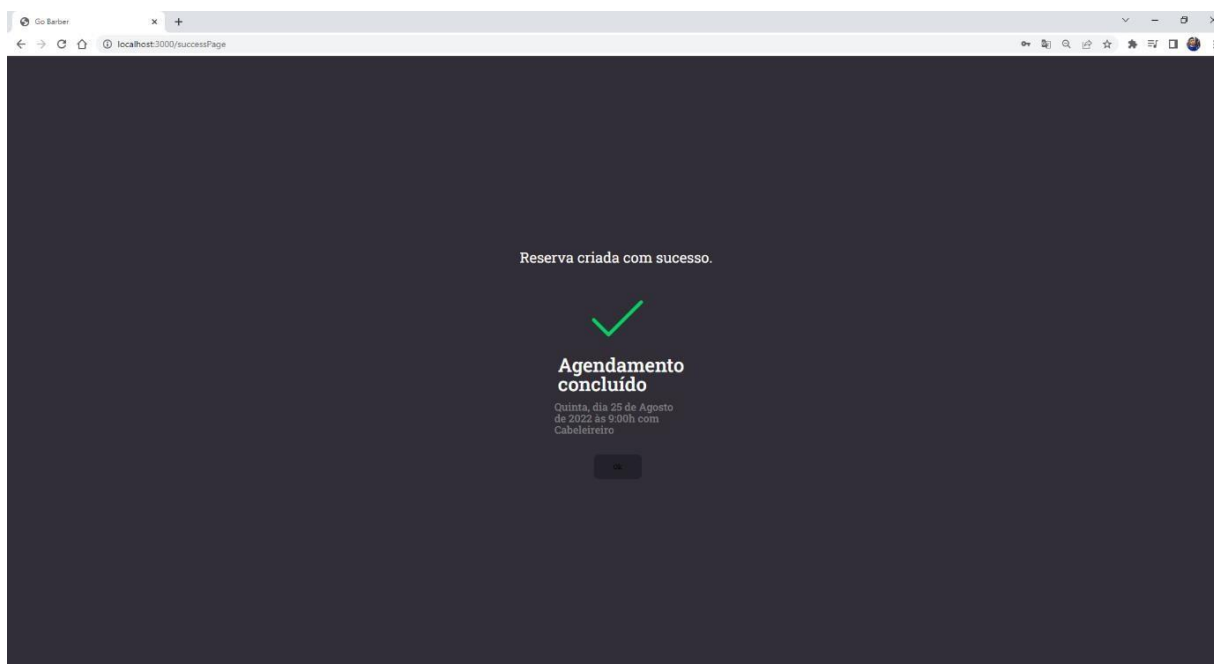
Figura 21 – Página de agendamento do cliente.



Fonte: Próprio autor (2022).

A Figura 22 mostra a página de confirmação, informando que o agendamento foi feito com sucesso, além de mostrar a data e o horário para os quais o agendamento foi criado.

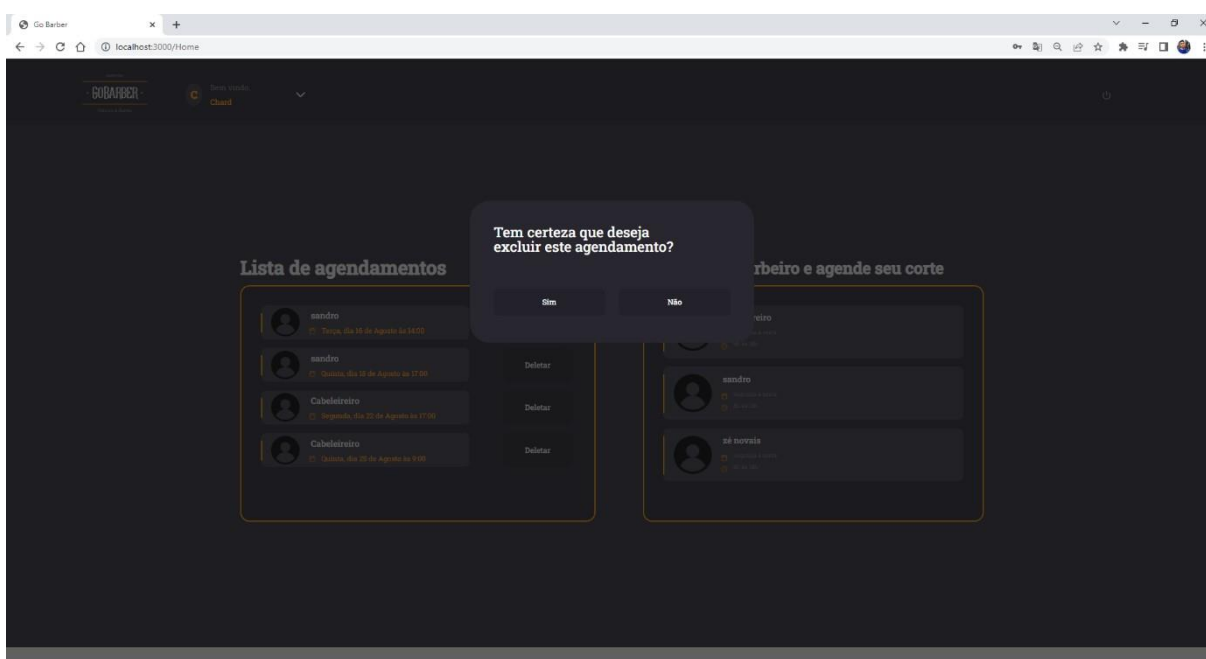
Figura 22 – Página de sucesso da criação do agendamento.



Fonte: Próprio autor (2022).

A Figura 23 mostra a página inicial do cliente na qual, a partir da listagem dos seus agendamentos, ele pode excluir o agendamento desejado ainda em aberto ao clicar no botão “Deletar” e, em seguida, em “Sim” no modal de confirmação de exclusão que irá ser aberto (caso o cliente tenha clicado por engano basta clicar em “Não” e o cancelamento será descartado).

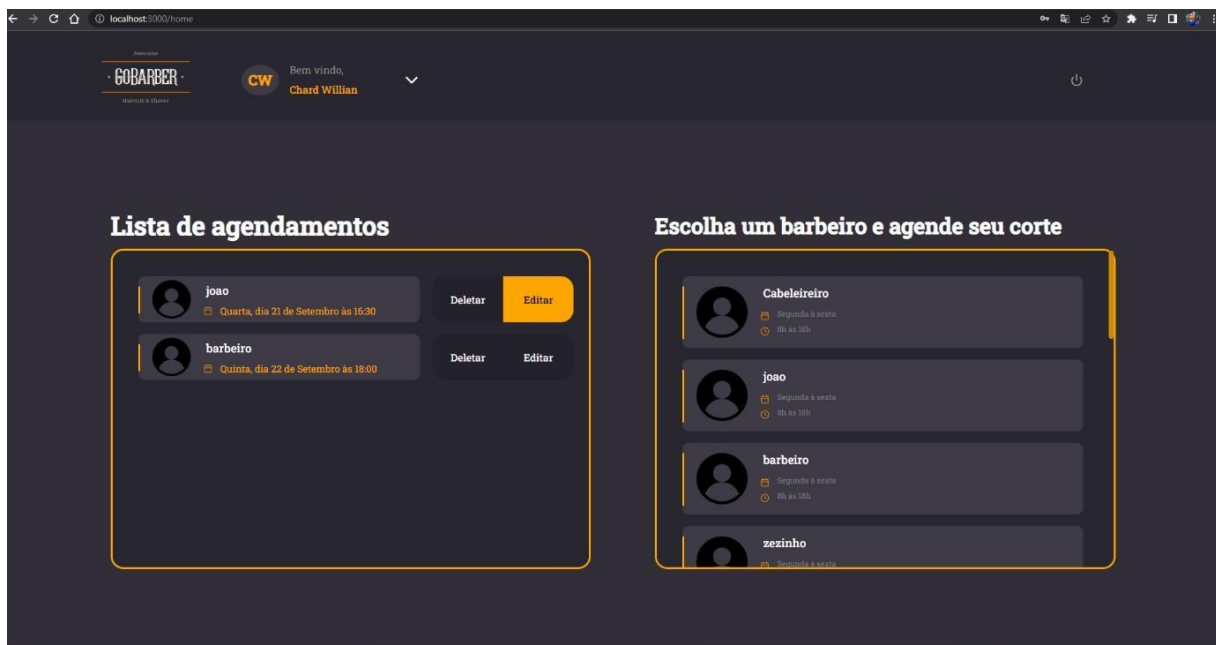
Figura 23 – Página inicial do cliente com opção de exclusão do agendamento.



Fonte: Próprio autor (2022).

A Figura 24 mostra a página inicial do cliente na qual, a partir da listagem dos seus agendamentos, ele pode editar um agendamento em aberto, clicando no botão “Editar” e, então, será redirecionado para a página de edição.

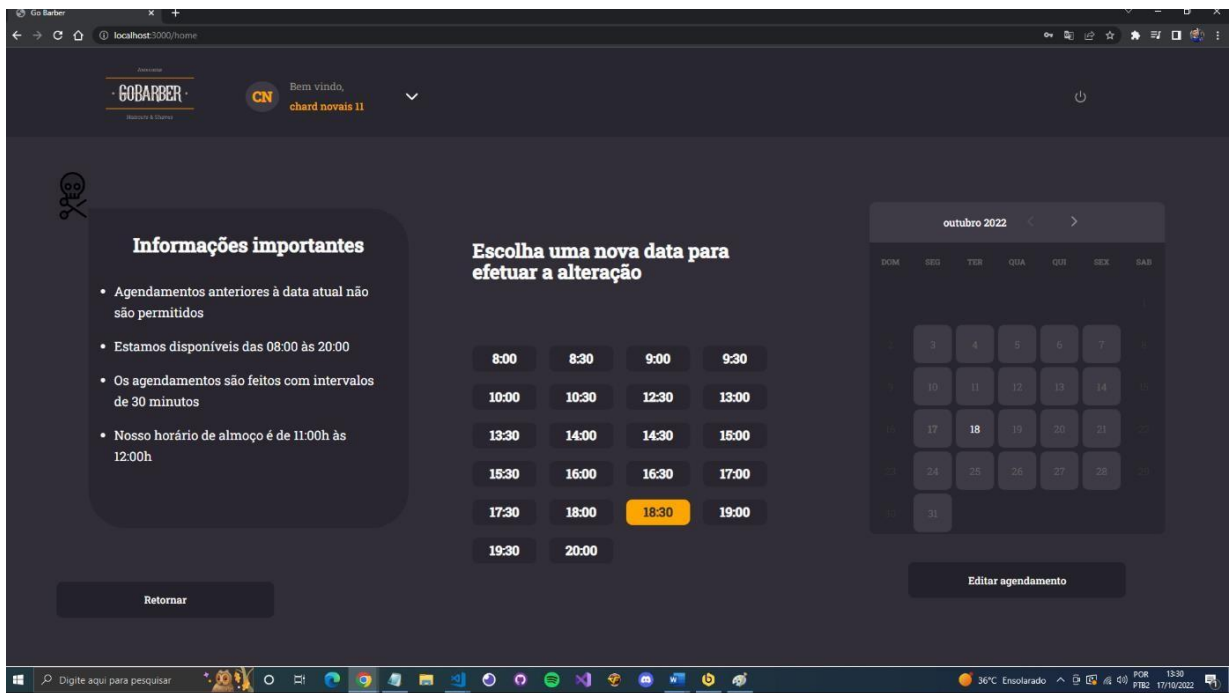
Figura 24 – Página inicial do cliente com opção de edição do agendamento.



Fonte: Próprio autor (2022).

A Figura 25 mostra a segunda parte da edição do agendamento na qual o cliente pode selecionar um novo horário e/ou data e, em seguida, clicar no botão “Editar agendamento” localizado abaixo do calendário.

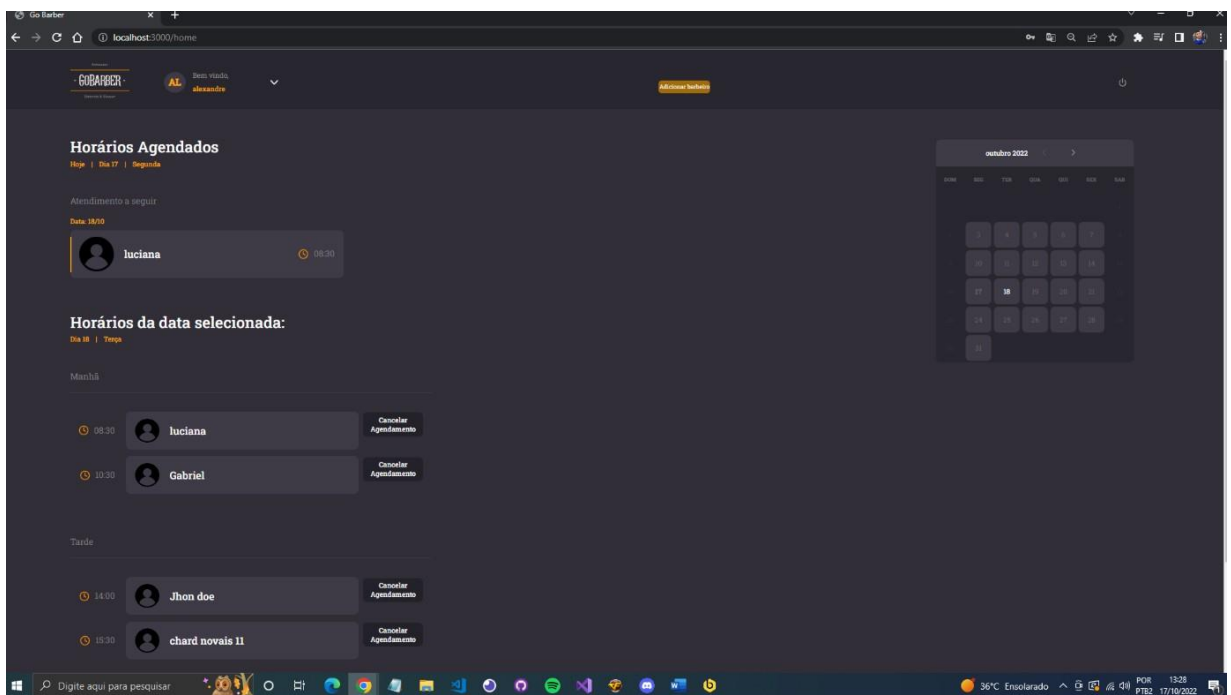
Figura 25 – Segunda parte da página de edição do agendamento.



Fonte: Próprio autor (2022).

A Figura 26 mostra a página inicial do barbeiro, que contém a listagem dos agendamentos que foram feitos com ele no dia atual e a opção de ele acessar o calendário do lado direito da tela e escolher outro dia para que a listagem de agendamentos seja carregada. Em caso de barbeiro administrador, é habilitado o botão para ele adicionar um novo barbeiro.

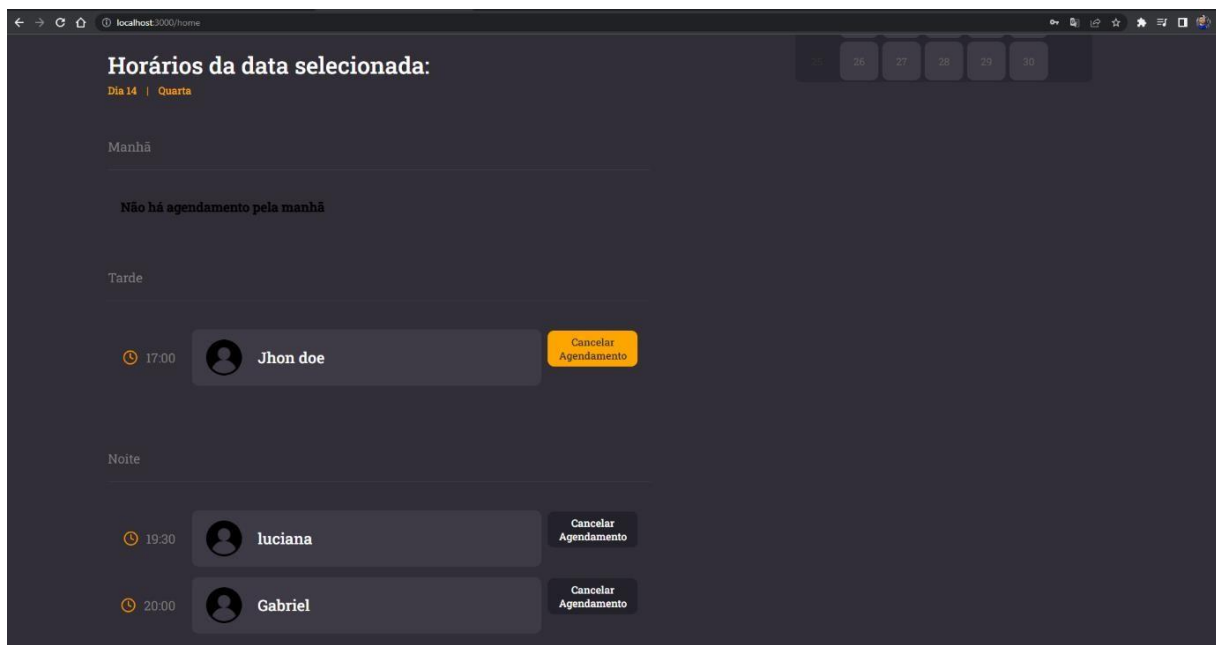
Figura 26 – Página inicial do barbeiro.



Fonte: Próprio autor (2022).

A Figura 27 mostra a opção onde o barbeiro pode cancelar um agendamento feito com ele, bastando a ele clicar no botão “Cancelar Agendamento”.

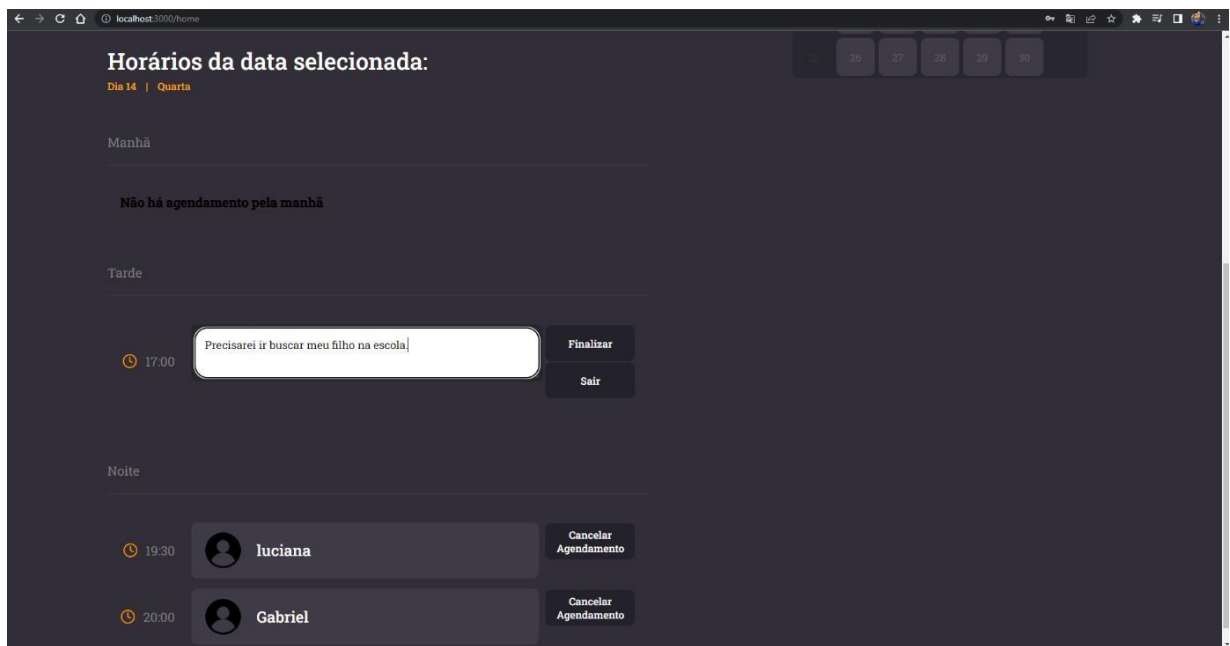
Figura 27 – Página inicial do barbeiro com opção de cancelamento do agendamento.



Fonte: Próprio autor (2022).

A Figura 28 mostra o local que o barbeiro deve colocar o motivo do cancelamento do agendamento, informando na mensagem o motivo e clicando no botão “Finalizar” para que o cancelamento seja efetuado com sucesso ou clicando em “Sair” para que o processo de cancelamento seja descartado.

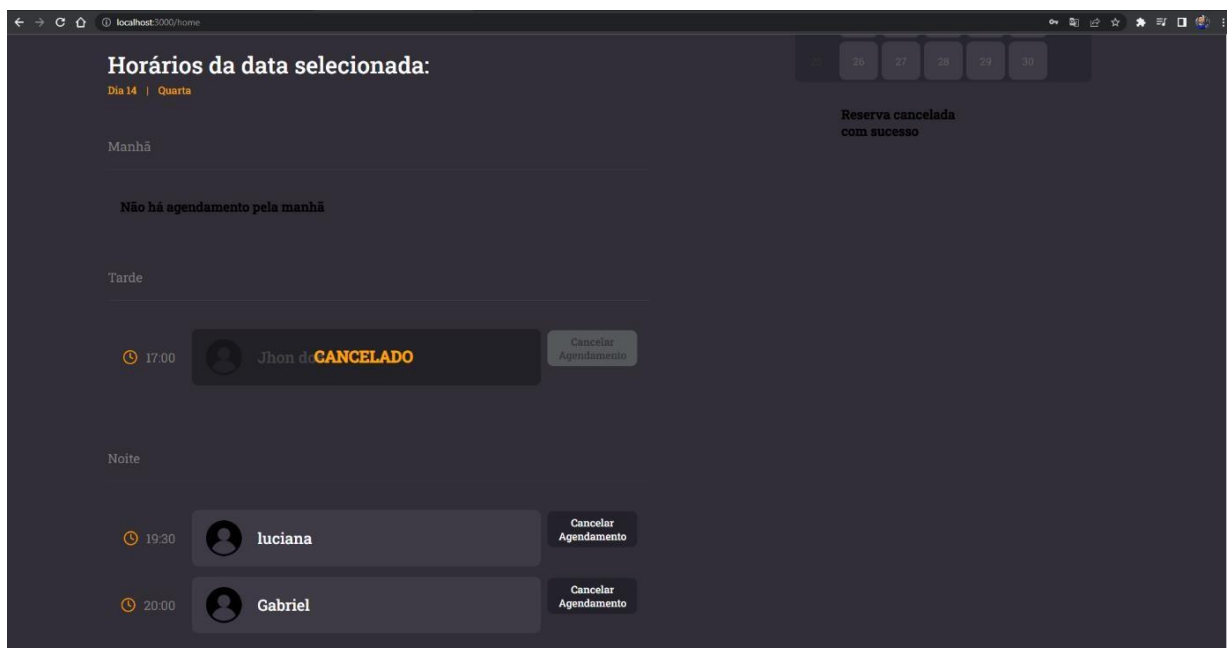
Figura 28 – Página inicial do barbeiro com o motivo do cancelamento do agendamento.



Fonte: Próprio autor (2022).

A Figura 29 mostra um agendamento que foi cancelado com sucesso.

Figura 29 – Página inicial do barbeiro com um cancelamento feito com sucesso.



Fonte: Próprio autor (2022).

5 CONCLUSÃO E CONSIDERAÇÕES FINAIS

Neste trabalho foi discutido sobre o processo por vezes ineficaz dos atendimentos prestados principalmente em barbearias, com os clientes despendendo de tempo para não serem atendidos ou de tempo extra para iniciarem o serviço desejado. Estes aspectos podem causar incômodos, estresse e uma péssima experiência que pode acarretar, inclusive, com que os clientes não voltem a frequentar o estabelecimento para uma nova prestação do serviço.

Tendo como objetivo resolver essas adversidades na experiência do cliente, foi proposta a criação de uma aplicação *web* para a realização online dos agendamentos em uma barbearia. Foi feito um levantamento de como funcionava o processo dos atendimentos/agendamentos neste tipo de estabelecimento e, então, se mostrou possível construir uma solução viável e eficaz para o problema.

O desenvolvimento da aplicação foi desafiador, foi necessário o aprendizado de novas tecnologias. O sistema feito em etapas, começando pela especificação do sistema, modelagem do banco de dados, criação dos wireframes etc., até se conseguir a primeira versão funcional da aplicação. Por muitas vezes, este autor dispendeu de tempo e esforço tentando resolver os bugs e erros apresentados. Foi um processo difícil, porém, bastante proveitoso e desafiador.

A aplicação atualmente está em fase de testes, em vias de implantação em uma barbearia. Como trabalhos futuros, pretende-se incluir meios de pagamento dentro da plataforma e realizar a sua implantação, para que ela possa atender à barbearia e/ou estabelecimentos semelhantes; permitir que um mesmo cliente faça apenas dois agendamentos por dia para um mesmo barbeiro, evitando-se problemas de uma pessoa má intencionada agendar todos os horários livres; não permitir que um horário já agendado seja selecionado e agendado por outro cliente; e melhorar o contraste de cores de algumas telas, facilitando assim a experiência dos usuários.

A realização deste trabalho possibilitou o uso do conhecimento adquirido no curso de Tecnologia Em Análise e Desenvolvimento de Sistemas para o cumprimento deste desafio, além de permitir ganhos expressivos em conhecimento e aprendizagem.

6 REFERENCIAL BIBLIOGRÁFICO

BECKER, Lauro. **Wireframes, o que são e por que os utilizamos?**. Disponível em: <<https://www.organicadigital.com/blog/o-que-sao-wireframes-e-por-que-os-utilizamos/>>. Acesso em: 03/04/2022.

DEVMEDIA. **Guia de Node.js**. Disponível em: <<https://www.devmedia.com.br/guia/node-js/40312>> Acesso em: 30/03/2021.

DEVMEDIA. **Modelo Entidade Relacionamento (MER) e Diagrama Entidade-Relacionamento (DER)**. Disponível em: <<https://www.devmedia.com.br/modelo-entidade-relacionamento-mer-e-diagrama-entidade-relacionamento-der/14332>> Acesso em: 15/05/2021.

FERREIRA, E.; EIS, D. **HTML5 e CSS3. com farinha e pimenta**. Diego Eis Elcio Ferreira. Disponível em: <<https://docplayer.com.br/1431492-Html5-e-css3-com-farinha-e-pimenta-diego-eis-elcio-ferreira.html>> Acesso em: 29/03/2021.

GRILLO, F. D. N.; FORTES R. P. M. **Aprendendo JavaScript**. Disponível em: <http://conteudo.icmc.usp.br/CMS/Arquivos/arquivos_enviados/BIBLIOTECA_113_ND_72.pdf>. Acesso em: 05/03/2022.

Kenzie. **React: O que é, como funciona e porque usar e como aprender**. Disponível em: <<https://kenzie.com.br/blog/react/#:~:text=O%20React%20%C3%A9%20uma%20biblioteca%20criada%20para%20construir%20telas%20de,mobile%20ou%20qualquer%20outro%20dispositivo.>> Acesso em: 11/05/2021.

Lucidchart. **Diagrama de caso de uso UML: O que é, como fazer e exemplos**. Disponível em: <<https://www.lucidchart.com/pages/pt/diagrama-de-caso-de-uso-uml>> Acesso em 11/05/2021.

Lucidchart. **O que é um fluxograma?**. Disponível em: <<https://www.lucidchart.com/pages/pt/o-que-e-um-fluxograma>>. Acesso em: 20/11/2021.

Mozilla. **Introdução Express/Node**. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Learn/Server-side/Express_Nodejs/Introduction> Acesso em: 07/10/2022.

NODEJS. Disponível em: <<https://nodejs.org/en/about/>> Acesso em: 30/03/2021.

PRESSMAN, R. S. **Engenharia de software: Uma Abordagem Profissional**. 7. ed. Porto Alegre: AMGH, 2011.

PostgreSQL: O banco de dados relacional de código aberto mais avançado do mundo.

Disponível em: <<https://www.postgresql.org/>> Acesso em: 11/05/2021.

React. Disponível em: <<https://pt-br.reactjs.org/>> Acesso em 17/05/2021.

ReactJs. **Introdução**. Disponível em: <<https://pt-br.reactjs.org/docs/getting-started.html>> Acesso em: 07/10/2022.

Rockcontent. **PostgreSQL: saiba o que é, para que serve e como instalar.** Disponível em: < <https://rockcontent.com/br/blog/postgresql/> > Acesso em: 11/05/2021.

Rocketseat **NodeJS: Vale a pena? Vantagens, vagas e salário.** Disponível em: < <https://blog.rocketseat.com.br/nodejs-vale-a-pena-vantagens/> > Acesso em: 30/03/2021.

SOMMERVILLE, L. **Engenharia de Software.** 9. ed. São Paulo: Pearson Prentice Hall. 2011.

Treasy. **Tudo sobre Metodologia Scrum: O que é e como essa ferramenta pode te ajudar a poupar tempo e gerir melhor seus projetos.** Disponível em: < <https://www.treasy.com.br/blog/scrum/> > Acesso em: 11/07/2021.