



Instituto Federal de Educação, Ciência e Tecnologia de Brasília  
*Campus Brasília*  
Tecnologia em Sistemas para Internet

JOÃO AUGUSTO DE CASTRO CRUZ  
PEDRO HENRIQUE DA SILVA SANTOS

## **SISTEMA DE AGENDAMENTO ONLINE PARA UMA BARBEARIA**

Brasília,  
2025

JOÃO AUGUSTO DE CASTRO CRUZ

PEDRO HENRIQUE DA SILVA SANTOS

## **SISTEMA DE AGENDAMENTO ONLINE PARA UMA BARBEARIA**

Trabalho de conclusão de curso de graduação apresentado à Coordenação do Curso Superior de Tecnologia em Sistemas para Internet do Instituto Federal de Brasília – Campus Brasília, como requisito parcial para a obtenção do título de Tecnólogo em Sistemas para Internet.

Orientador: Alisson Wilker Andrade Silva

Brasília,  
2025

C957 Cruz, João Augusto de Castro.

Sistema de agendamento online para uma barbearia / João Augusto de Castro Cruz, Pedro Henrique da Silva Santos. — Brasília, 2025.

63 f. : il.

Orientador: Alisson Wilker Andrade Silva.

Trabalho de conclusão de curso (Graduação) — Instituto Federal de Educação, Ciência e Tecnologia de Brasília, Tecnologia em Sistemas para Internet, 2025.

1. Sistema web. 2. Sistema de gerenciamento.  
I. Santos, Pedro Henrique da Silva. II. Silva, Alisson Wilker Andrade (orient.). III. Título.

CDU: 004.42

Elaborada com os dados fornecidos pelos autores.

JOÃO AUGUSTO DE CASTRO CRUZ

PEDRO HENRIQUE DA SILVA SANTOS

## **SISTEMA DE AGENDAMENTO ONLINE PARA UMA BARBEARIA**

Trabalho de conclusão de curso de graduação apresentado à Coordenação do Curso Superior de Tecnologia em Sistemas para Internet do Instituto Federal de Brasília – Campus Brasília, como requisito parcial para a obtenção do título de Tecnólogo em Sistemas para Internet.

Aprovado em: \_\_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_.

### **BANCA EXAMINADORA**

---

Prof. Me. Alisson Wilker Andrade Silva  
Orientador

---

Prof. Me. Carlos Roberto Gonçalves Selva

---

Prof. Me. Fábio Ferraz Fernandez

Brasília,  
2025

## RESUMO

Cruz, João Augusto de Castro; Santos, Pedro Henrique da Silva. **Sistema de Agendamento Online para uma Barbearia**. Trabalho de Conclusão de Curso (Graduação) – Tecnólogo em Sistemas para Internet. Instituto Federal de Brasília – *Campus* Brasília. Brasília/DF, 2025.

Com o avanço da tecnologia e sua crescente utilidade no cotidiano, é natural que novas demandas surjam ao longo do tempo. A busca por agilidade, praticidade e economia de tempo é um dos principais fatores que impulsionam as adaptações às novas tecnologias no mercado. O uso de smartphones e outros sistemas web transformou a maneira como as pessoas percebem o comércio e a competitividade no setor. Antigamente, as informações eram gerenciadas por meio de anotações em cadernetas; hoje, já é possível contar com softwares e aplicativos web que facilitam esse processo. Este trabalho de conclusão de curso tem como foco a implementação de um sistema sob demanda, com a finalidade principal de desenvolver uma plataforma online para uma barbearia. O objetivo é otimizar o processo de agendamento, e aumentar a satisfação tanto dos clientes quanto da empresa. A proposta era criar uma solução que atendesse às expectativas dos clientes, ao mesmo tempo em que melhorasse a eficiência dos serviços oferecidos pela barbearia. Para isso, foi desenvolvido um sistema que inclui funcionalidades como agendamento online e exibição dos trabalhos realizados pelos barbeiros. A implementação foi feita utilizando o framework Django, em Python, com ênfase na usabilidade e em uma interface intuitiva para o público-alvo. Além disso, foram utilizadas linguagens como HTML e CSS, juntamente com a integração a um banco de dados SQLite3.

**Palavras-chave:** sistema web; sistema de agendamento.

## ABSTRACT

With the advancement of technology and its increasing utility in everyday life, it is natural that new demands will arise over time. The search for speed, practicality, and time-saving is one of the main factors driving adaptations to new technologies in the market. The use of smartphones and other web systems has transformed the way people perceive commerce and competitiveness in the sector. In the past, information was managed through notes in notebooks; today, it is possible to rely on software and web applications that simplify this process. This final project focuses on the implementation of a custom system, with the main goal of developing an online platform for a barbershop. The objective is to optimize the appointment scheduling process, and increase the satisfaction of both customers and the business. The proposal was to create a solution that would meet customers' expectations while improving the efficiency of the services offered by the barbershop. To achieve this, a system was developed that includes features such as online booking and the display of the barbers' work. The implementation was carried out using the Django framework in Python, with a focus on usability and an intuitive interface for the target audience. Additionally, languages such as HTML and CSS were used, along with integration to an SQLite3 database.

**Keywords:** web system; scheduling system.

## LISTA DE ILUSTRAÇÕES

### Figuras

Figura 1 – Diagrama de Caso de Uso .....	24
Figura 2 – Modelo de Classes .....	25
Figura 3 - Model-View-Controller .....	26
Figura 4 – Organização da Aplicação em Camadas .....	27
Figura 5 – Diagrama de Caso de Uso .....	34
Figura 6 – Diagrama de Classes .....	36
Figura 7 – Diagrama de Entidade e Relacionamento .....	37
Figura 8 – Tela de login .....	41
Figura 9 – Tela inicial .....	42
Figura 10 - Tela Cadastrar Novo Cliente .....	43
Figura 11 - Tela Adicionar Barbearia .....	44
Figura 12 - Tela Cadastrar Novo Barbeiro .....	45
Figura 13 – Tela Cadastrar Nova Agenda .....	46
Figura 14 – Tela Lista de Agendas .....	47
Figura 15 - Tela Detalhes do Agendamento.....	48
Figura 16 – Tela Lista de agendamentos .....	49
Figura 17 - Tela Dashboard.....	50
Figura 18 – Tela inicial (cliente) .....	51
Figura 19 – Tela Lista de Barbeiros .....	51
Figura 20 - Tela Detalhes do Barbeiro.....	52
Figura 21 - Tela Lista de Clientes.....	53
Figura 22 – Tela Cadastrar Novo Agendamento .....	54
Figura 23 – Tela Detalhes do Agendamento .....	55

### Quadros

Quadro 1 - Comparativo - Sistemas Similares .....	20
Quadra 2 – Comparativo – Sistemas Similares.....	20
Quadro 3 – Requisitos Funcionais .....	33
Quadro 4 – Requisitos não-funcionais .....	34
Quadro 5 - Caso de uso 1 .....	35
Quadro 6 - Caso de uso 2 .....	36

## LISTA DE TABELAS

### **Tabela**

Tabela 1 – Pesquisa de satisfação do cliente .....	57
--	----

# SUMÁRIO

## Sumário

1. INTRODUÇÃO .....	10
1.1 - Definição do tema e problemática .....	11
1.3 - Objetivo Geral .....	12
1.4 - Objetivos específicos.....	12
1.5 - Estrutura.....	13
1.6 - Classificação da Pesquisa.....	13
2. SISTEMAS SIMILARES.....	15
2.1 - Considerações sobre a revisão de sistemas similares.....	18
3. REFERENCIAL TEÓRICO.....	21
3.1 - Engenharia de Software .....	21
3.1.1 - Linguagem UML.....	21
3.1.2 - Model-View-Controller MVC.....	25
3.2 - Desenvolvimento Web: os desenvolvimentos de sistemas-web .....	26
3.3 - JavaScript .....	28
3.4 - Django.....	28
3.5 - HTML .....	29
3.6 - CSS.....	29
3.7 - Python.....	29
3.8 - Banco de dados .....	30
3.9 - SQLite3.....	30
3.10 - Microsoft Visual Studio Code.....	31
3.11 - Lucidchart.....	31
3.12 - BRmodelo .....	31
4 - METODOLOGIAS UTILIZADAS .....	32
4.1 - Levantamento de Requisitos .....	33
4.1.2 - Requisitos Funcionais .....	33
4.1.3 - Requisitos não-funcionais .....	34
4.2 - Casos de Uso.....	34
4.3 - Diagrama de Classes .....	36
4.4 Diagrama de entidade e relacionamento - DER.....	37
5. APRESENTAÇÃO E ANÁLISE DOS RESULTADOS.....	39
5.1 - A solução para o cliente .....	39
Gestão de Funcionários e Serviços.....	39

5.2. Desenvolvimento do Sistema .....	40
5.3 - Apresentação do sistema .....	41
5.3.1 - Telas dos sistemas e os cenários de uso.....	42
5.4 - Testes e Validação.....	56
6. AVALIAÇÃO E CONSIDERAÇÕES FINAIS.....	60
7. REFERÊNCIAS BIBLIOGRÁFICAS .....	61

## 1.INTRODUÇÃO

A tecnologia tem tomado espaço na vida em sociedade. Sem ela é até difícil imaginar alguns contextos sociais. O uso imediato da internet, a facilidade em se ter um equipamento em mãos a todo momento, como é o caso do smartphone, tudo isso contribui para a emergência em se adequar às novas demandas tecnológicas.

Segundo Kohn & Moraes (2007), a tecnologia se tornou uma ferramenta que demonstra desenvolvimento, riqueza, poderio e conhecimento. Obter tecnologia no cotidiano em sociedade é um bem a ser conquistado e implementado nas práticas sociais. Essas práticas levam ao avanço do compartilhamento de informações, da competição comercial, na mudança nas formas de se relacionar, do avanço da internet na vida em sociedade.

O uso da Internet também tem seu papel importante no que se diz respeito ao marketing digital, por meio dela se anuncia, se negocia, se precifica um produto, se conhece a demanda e a competitividade do mercado. Esse avanço no mercado comercial do uso das tecnologias, fez com que as empresas procurassem cada vez mais o serviço. A tecnologia não só contribuiu para a competitividade, bem como abriu portas para a inovação e a atualização das empresas, além de facilitar a vida cotidiana de quem a procura (Kohn & Moraes, 2007). A procura pela criação de aplicativos, de sistemas-web ou até mesmo do gerenciamento das redes sociais tem aumentado, pois por meio dessas ferramentas amplia-se a interação com o cliente, além de atender parte das exigências da clientela.

O comportamento de consumo torna-se mais informado, o que conduz a que os indivíduos sejam cada vez mais exigentes. Por este motivo, para que as marcas e organizações se consigam destacar da concorrência é fulcral que se consigam adaptar e reajam rapidamente às exigências que são feitas pelos consumidores. É este o motivo que leva as empresas e organizações a apostarem na construção de estratégias de comunicação digital. (Faria, 2022, p. 04)

A internet tornou-se, assim, quase uma obrigação para garantir uma comunicação eficaz e ágil com os clientes, para organizar a empresa e até mesmo para atrair novos consumidores. Os benefícios que essa ferramenta trouxe para as empresas são gigantescos. A comunicação, que antes ocorria de forma presencial, com horário marcado, agora possui acessibilidade mais extensiva, o que agrega tempo à empresa e poder de escolha ao cliente.

Sendo assim, os usos dessas tecnologias contribuíram e ainda contribuem nas mais variadas formas com a empresa. Um site ou uma rede social, pode se tornar a vitrine daquele

comércio, onde seu produto pode ser exposto, organizado, avaliado e por fim comprado (Santos, Camargo, Sarti & Novelli, 2004). Além disso, pode auxiliar no gerenciamento dos dados da empresa.

Existem várias formas de se trabalhar o gerenciamento dentro da área tecnológica, entre elas está justamente a incorporação de novas tecnologias no processo de organização da empresa. Um sistema de agendamento online, por exemplo, pode se tornar uma ferramenta inovadora para um negócio. Por meio dela é possível não somente fazer um agendamento, mas verificar os profissionais disponíveis, quais serviços a empresa disponibiliza, os valores, as formas de pagamento, além de gerar um banco de dados para a gestão da empresa.

Para tanto, este projeto teve como proposta a construção de um sistema de agendamento online na sua versão web para uma barbearia que precisava organizar a agenda dos seus funcionários e clientes no dia-a-dia, para facilitar o serviço e poupar tempo de ambas as partes.

## **1.1 - Definição do tema e problemática**

Atualmente alguns empreendimentos necessitam de sistemas de agendamento para melhor conduzir os atendimentos de seus empreendimentos. A proposta deste trabalho se baseou na construção de um sistema de agendamento online por meio de um sistema-web.

O interesse pelo tema veio da inquietação de um cabeleireiro do qual um dos autores deste trabalho é cliente. Nem toda empresa possui um sistema on-line que gera e guarda dados. Essa barbearia tem apresentado problemas com relação ao agendamento de seus clientes, justamente por ainda usar um método manual, a agenda física. Além disso, foi possível observar que esse tipo de gerenciamento adotado pela empresa toma muito tempo de quem de fato vai mexer com a agenda manual, tempo este em que o profissional poderia investir no trabalho com cortes. Os clientes também reclamaram do choque de horários, o que gera afastamento e desconforto para os mesmos. O dono reclamou ainda de que não possuía ferramentas para divulgar o trabalho de seus funcionários, o que abriria espaço para novas clientelas.

Por esse motivo, numa conversa sobre o gerenciamento dos agendamentos da empresa, identificou-se uma problemática e o dono da barbearia demandou possíveis soluções para o seu negócio. Com isso, notou-se a necessidade de sistema-web. Diante disso, a problemática do

trabalho consiste em: Quais ferramentas utilizar para desenvolver um sistema de agendamento online para uma barbearia que não só resolva os problemas dos agendamentos, mas também melhore a experiência do cliente? Tal problemática engloba tanto os desafios metodológicos do desenvolvimento do sistema quanto às necessidades específicas do cliente em relação às funcionalidades que poderão ser oferecidas.

Como justificativa, este trabalho visou desenvolver um sistema de agendamento para uma barbearia a fim de contribuir para melhoria no atendimento dos clientes e no trabalho dos profissionais quanto à organização da empresa e de seu tempo, assegurando ambos em sua livre demanda.

Além disso, o sistema surgiu como um novo método adotado pela empresa, para substituir o agendamento de caderneta que vinha ocasionando problemas. O site teve a intenção de inovar também o marketing da empresa, por meio dele seria possível divulgar o trabalho dos profissionais, gerando uma melhor visibilidade do que os mesmos oferecem.

### **1.3 - Objetivo Geral**

O objetivo geral deste trabalho consistiu em desenvolver um sistema de agendamento eficiente para uma barbearia, que simplifique o processo de marcação de horários, melhore a gestão de clientes, diminua os erros e conflitos de agenda, e propicie uma experiência clara e acessível tanto para os clientes quanto para os profissionais, tornando o agendamento mais prático e eficaz.

### **1.4 - Objetivos específicos**

Entre os objetivos específicos, foi observado:

- Implementar uma interface intuitiva para os clientes, permitindo que eles realizem agendamentos de forma simples e rápida via site.

- Desenvolver funcionalidades de controle de agenda em tempo real para os profissionais da barbearia, garantindo que os horários disponíveis e já agendados sejam atualizados automaticamente.

- Integrar uma ferramenta de gestão de clientes, que armazene dados como histórico de serviços e preferências.

Esses objetivos específicos visam resolver a problemática de maneira eficiente, aprimorando tanto a experiência do cliente quanto a gestão interna da barbearia.

## **1.5 - Estrutura**

No Capítulo 1, intitulado como introdução foi descrita toda a problemática do trabalho, bem como o surgimento da ideia de construir um sistema web para uma clientela específica. Além disso, foram apontados os caminhos seguidos para responder a essa problemática.

No Capítulo 2, são abordados os trabalhos similares a este, os quais justificam e fundamentam a consistência e a importância de um sistema web para o gerenciamento de empresas, nas diversas modalidades atuais, seja para o aumento de vendas, seja para a organização financeira, seja para a melhoria dos atendimentos.

No Capítulo 3, desenvolve-se a discussão teórica deste trabalho, com uma revisão da literatura que explica quais caminhos e quais ferramentas são possíveis de utilizar para desenvolver o sistema proposto.

No capítulo 4 aborda-se as metodologias utilizadas neste trabalho, bem como os levantamentos de requisitos necessários para a implementação do sistema. Já no capítulo 5, apresenta-se a aplicação do sistema com seus cenários de uso, bem como os resultados alcançados por meio do sistema.

## **1.6 - Classificação da Pesquisa**

A pesquisa desenvolvida neste projeto de conclusão de curso é classificada como exploratória, o que permite ao pesquisador compreender melhor o funcionamento das barbearias e identificar oportunidades de melhoria no processo de agendamento online. Além

disso, a pesquisa possui uma abordagem qualitativa e quantitativa, combinando diferentes métodos para obter uma visão mais completa do problema e da solução proposta.

De acordo com Gil (2019), esse tipo de pesquisa auxilia na compreensão do campo investigado, criando familiaridade com o tema, o que contribui para a definição dos objetivos a serem alcançados ou até mesmo para a formulação de hipóteses. Este trabalho surgiu a partir de uma necessidade específica: melhorar a experiência do cliente e otimizar a gestão de agendamentos. Para isso, foi realizada uma pesquisa bibliográfica sobre tecnologias e sistemas de agendamento em outras áreas de serviços, além de uma análise das metodologias empregadas por plataformas já existentes.

A abordagem qualitativa foi utilizada para entender as necessidades de fato da barbearia em questão, por meio de entrevistas e levantamento de requisitos com o proprietário da barbearia. Essa abordagem permitiu uma análise descritiva do problema, identificando as principais dificuldades enfrentadas no agendamento manual e as expectativas em relação ao sistema proposto. Além disso, na pesquisa qualitativa, os dados descritivos do problema são essenciais para uma análise empírica da situação estudada.

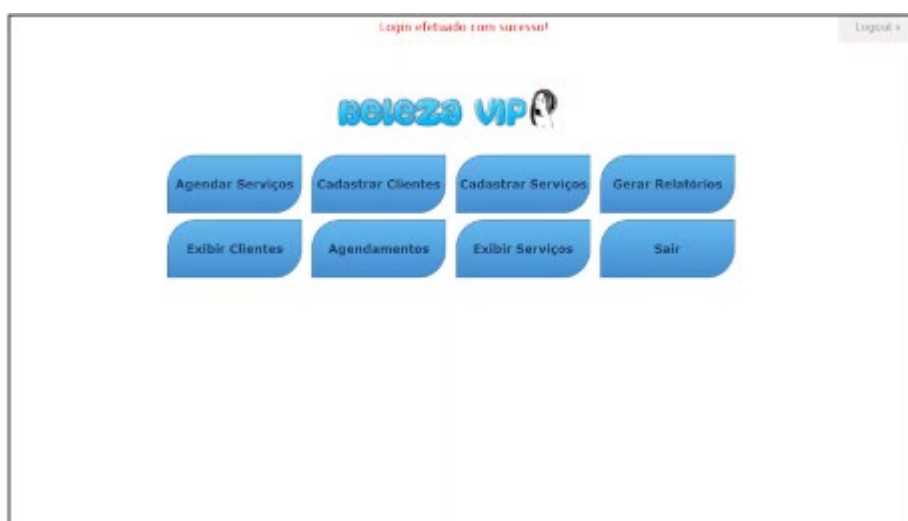
Já a abordagem quantitativa foi aplicada durante a fase de testes e validação do sistema, onde foram coletados dados numéricos relacionados ao desempenho do sistema, como tempo de resposta, taxa de erros e resultados da pesquisa de satisfação do cliente. Esses dados permitiram uma avaliação em relação a eficácia do sistema e identificação de ferramentas que precisam melhorar.

A combinação dessas abordagens (exploratória, qualitativa e quantitativa) proporcionou uma visão abrangente do problema e uma solução mais alinhada com as necessidades do cliente.

## 2. SISTEMAS SIMILARES

Ao pesquisar sobre outros trabalhos que envolvem a criação de sistemas de agendamento, foi possível encontrar alguns que se assemelham ao proposto nesse projeto.

Ávila (2013) traz em seu trabalho a criação de um sistema de informações com intuito de gerenciar algumas informações para um salão de beleza em Santa Catarina. Entendendo a necessidade de um sistema de gerenciamento de informações para a cliente, ela propôs a criação de um projeto que gerenciasse o cadastro de clientes e os serviços prestados, além de fornecer a opção de agendamento para esses clientes. Junto a essas funções, o sistema possibilita gerar relatórios dos atendimentos para a dona do estabelecimento. Ela utilizou a linguagem PHP, junto ao banco de dados MySQL.



Fonte: Ávila, 2013.

Nascimento (2022) tomando como inspiração plataformas como Ifood e UberEats como sistemas que agregam serviços no campo alimentício, pensou num sistema para gerenciamento de barbearias. O autor utilizou duas interfaces, a web e a mobile. O diferencial do seu projeto está justamente no processo agregador em que o usuário pode visualizar diversas barbearias, além de poder agendar uma visita a qualquer uma delas para tomar conhecimento da sua forma de trabalho. Esse mesmo sistema também proporciona agendamento para um melhor controle da agenda dos barbeiros. Além de um gerenciador, sua proposta foi a criação de um software.



Fonte: Nascimento, 2022.

Kieras (2019) também tem como proposta um sistema de agendamento, nesse caso para serviços de profissionais que necessitam dessa ferramenta, como médicos, psicólogos, cabeleireiros, etc. O diferencial desse projeto é que ele utiliza um sistema para uma cidade muito bem delimitada, além de fazer comparações de preços entre os serviços ali disponibilizados. O autor também utilizou duas soluções, a web e a mobile.

Fonte: Kieras, 2019.

Por fim, Martins & Fernandes (2019) a partir das necessidades de uma barbearia, criou-se um software desktop *Two Style*. A proposta está em auxiliar na organização do serviço administrativo da barbearia, oferecendo segurança e comodidade. Além de possibilitar o agendamento de serviço, inclui como finalidade ofertar um suporte como cadastro de clientes e serviços.



Fonte: Martins & Fernandes, 2019.

E o que diferencia este trabalho dos demais? A diferença está em como ele foi construído, no caso a partir de código aberto, assim a comunidade técnica pode ajudar na melhoria do desenvolvimento o tornando mais robusto a depender das necessidades de

melhoria de cada usuário. Além disso, o sistema oferece ferramentas vinculadas ao marketing da empresa.

## **2.1 - Considerações sobre a revisão de sistemas similares**

Com base na análise dos sistemas similares, percebe-se que várias funcionalidades são recorrentes em sistemas de agendamento e gestão de barbearias. As características mais comuns incluem o cadastro e gerenciamento de clientes, a oferta de serviços e a funcionalidade de agendamento, que é o foco principal em todos os projetos analisados. A utilização de tecnologias web e mobile também é uma abordagem comum, proporcionando mais acessibilidade e facilidade no uso.

Entretanto, algumas propostas trazem diferenciais importantes, como o gerenciamento agregador de barbearias, como no sistema de Nascimento (2022); a comparação de preços entre prestadores de serviços, como proposto por Kieras (2019); ou a inclusão de funcionalidades administrativas mais complexas, como o controle financeiro e a geração de relatórios, como no exemplo de Ávila (2013). Esses diferenciais visam atender demandas específicas de cada tipo de negócio.

Em contraste, o sistema proposto neste trabalho foca em um gerenciamento mais eficiente e direto para uma única barbearia, oferecendo controle completo sobre a agenda dos barbeiros, e a possibilidade de gerenciar promoções ou pacotes de serviços. Além disso, o sistema contará com um módulo de feedback, onde os clientes poderão avaliar os serviços e barbeiros, criando uma camada de interação e melhoria contínua.

Com essa abordagem, o "Sistema Online para uma Barbearia" trará soluções que atendem diretamente as necessidades do negócio, sem a complexidade de sistemas agregadores, porém oferecendo funcionalidades que ampliam a eficiência e a experiência do cliente, posicionando-o como uma solução mais prática e personalizada em relação aos sistemas analisados.

Quadro 1 - Comparativo - Sistemas Similares

Quadro Comparativo de linguagens - Sistemas Similares				
Sistema	PHP	Java	JavaScript	Python
Sistema Gerenciador para um Salão De Beleza	x			
Sistemas de Agendamento de Serviço	x			
Two Style: Um Software de Agendamento e Gerenciamento		x		
Desenvolvimento de um Sistema Agregador de Barbearias			x	
Sistema de Agendamento Online para uma Barbearia				x

Quadro 2 - Comparativo - Sistemas Similares

Sistema	Agendamento	Web	Desktop	Dashboard
Sistema Gerenciador para um Salão De Beleza	x	x		
Sistemas de Agendamento de Serviço	x	x		
Two Style: Um Software de Agendamento e Gerenciamento	x		x	
Desenvolvimento de um Sistema				

<b>Agregador de Barbearias</b>	X	X		
<b>Sistema de Agendamento Online para uma Barbearia</b>	X	X		X

Comparado a outros sistemas, o sistema aqui desenvolvido tem diferenciais, entre eles a linguagem utilizada e uma ferramenta para colhimento de dados. Além das funcionalidades, o sistema também contará com um dashboard interativo desenvolvido com Chart.js, permitindo a visualização de dados por meio de gráficos dinâmicos. Com isso, os barbeiros poderão acompanhar o número de atendimentos, os horários de maior demanda e o desempenho individual dos barbeiros.

Dashboard, também conhecido como painel interativo, são painéis que apresentam informações e dados de uma empresa de forma gráfica, facilitando o entendimento dos dados apresentados.

A parte de agendamento foi desenvolvida utilizando Python, proporcionando uma base robusta para o gerenciamento de horários e reservas. Essa escolha tecnológica garante escalabilidade e flexibilidade, além de facilitar futuras integrações com outras ferramentas ou melhorias no sistema.

Dessa forma, o Sistema Online para uma Barbearia combina funcionalidades essenciais com tecnologias modernas para oferecer um gerenciamento completo e acessível, garantindo não apenas a organização das agendas, mas também uma boa experiência para clientes e profissionais.

### 3. REFERENCIAL TEÓRICO

Para entender o desenrolar deste trabalho, é preciso definir algumas categorias e métodos importantes para o desenvolvimento do sistema proposto, entre estes estão: a linguagem de modelagem UML e Desenvolvimento Web.

#### 3.1 - Engenharia de Software

A engenharia de software é um campo importante para a definição dos aspectos de produção de um software. Pressman (1995) define a engenharia de software como:

Um rebento da engenharia de sistemas e de hardware. Ela abrange um conjunto de três elementos fundamentais – métodos, ferramentas e procedimentos – que possibilita ao gerente o controle do processo de desenvolvimento de software e oferece ao profissional uma base para a construção de software de alta qualidade produtivamente. (p.31)

Os métodos, segundo o autor, envolvem o planejamento de como fazer o software; as ferramentas dão suporte automatizado a criação do mesmo; e os procedimentos que são aquilo que mantém os métodos e as ferramentas juntos. Além disso, os procedimentos

definem a sequência em que os métodos serão aplicados, os produtos (deliverables) que se exige que sejam entregues (documentos, relatórios, formulários, etc.), os controles que ajudam a assegurar a qualidade e a coordenar as mudanças, e os marcos de referência que possibilitam aos gerentes de software avaliar o progresso. (Pressman, 1995, p.32)

Entende-se que a engenharia de software como disciplina trabalha com processos de desenvolvimento, o que é fundamental para a criação do sistema proposto. Além disso, esses processos perpassam desde a criação do sistema, o seu processo durante e o pós uso, dando espaços à manutenções quando necessário. Dentro dessa mesma disciplina situa-se vários modelos do processo de software. Nesse processo, temos a UML (Linguagem de Modelo Unificada) na qual auxilia na modelação dos softwares, e que será explicada abaixo.

##### 3.1.1 - Linguagem UML

A linguagem UML é uma linguagem de modelagem unificada utilizada para modelar softwares. Com ela se define quais são as características do sistema, “seus requisitos, seu comportamento, sua estrutura lógica, a dinâmica de seus processos e até mesmo suas necessidades físicas em relação ao equipamento sobre o qual o sistema deverá ser implantado” (Guedes, 2011, p. 19). Sendo uma linguagem independente, pode ser utilizada em diversos

desenvolvimentos. Através do uso de diagramas, essa linguagem permitiu a melhor visualização e entendimento dos clientes nos projetos de software.

#### *3.1.1.1 - Levantamento de requisitos*

Os requisitos documentam aquilo que o sistema precisa, ou seja, o que ele deve fazer. Sommerville (2011) traz a definição de requisitos e o separa em duas categorias: requisitos de usuário e requisitos de sistema. Ambas possuem detalhes referentes ao objetivo final, ou seja, a criação do sistema.

1. Requisitos de usuário são declarações, em uma linguagem natural com diagramas, de quais serviços o sistema deverá fornecer a seus usuários e as restrições com as quais este deve operar.

2. Requisitos de sistema são descrições mais detalhadas das funções, serviços e restrições operacionais do sistema de software. O documento de requisitos do sistema (às vezes, chamado especificação funcional) deve definir exatamente o que deve ser implementado. Pode ser parte do contrato entre o comprador do sistema e os desenvolvedores de software. (Sommerville, 2011, p. 58)

Alguns autores mencionam a dificuldade da comunicação nesse processo, pois muitas vezes o profissional terá de se atentar aos detalhes exigidos pelo cliente e transformar esses detalhes numa linguagem mais técnica e inteligível para o próprio sistema (Sommerville, 2011; Guedes, 2018).

Sendo assim, é possível observar a importância dos requisitos para a estruturação do sistema. Arriscaríamos dizer que é a etapa mais importante do desenvolvimento de um sistema, pois antecede toda a elaboração do mesmo, ou seja, a averiguação de detalhes, das exigências precisam ser feitas antes mesmo da sua estruturação. É preciso que o desenvolvedor se atente a essa fase, para que o trabalho seja seguro e confiante. A fase do levantamento de requisitos exige a identificação de requisitos: os funcionais e os não-funcionais.

#### *3.1.1.2 - Requisitos Funcionais*

Os requisitos funcionais estão relacionados às funcionalidades que o sistema deverá fazer de acordo com suas funções ou serviços. Correspondem aquilo que o cliente quer fazer, ou seja, as funcionalidades que o sistema permitirá (Guedes, 2018). Um exemplo é a possibilidade do próprio agendamento para a barbearia.

### *3.1.1.3 - Requisitos não-Funcionais*

Os requisitos não funcionais correspondem às restrições do sistema. Está vinculada a confiabilidade, a proteção e a validação do sistema (Guedes, 2018). Esses requisitos são características mínimas de um software de qualidade.

Os requisitos não funcionais surgem por meio das necessidades dos usuários, devido a restrições de orçamento, políticas organizacionais, necessidade de interoperabilidade com outros sistemas de software ou hardware, ou a partir de fatores externos, como regulamentos de segurança ou legislações de privacidade. (Sommerville, 2011, p.61)

Os requisitos não-funcionais têm ligação direta com o funcionamento do sistema, sendo uma das partes de suma importância. Um exemplo de requisito não-funcional é o tipo de linguagem utilizada na construção do sistema, como JAVASCRIPT. Ou mesmo a parte que é restrita ao usuário.

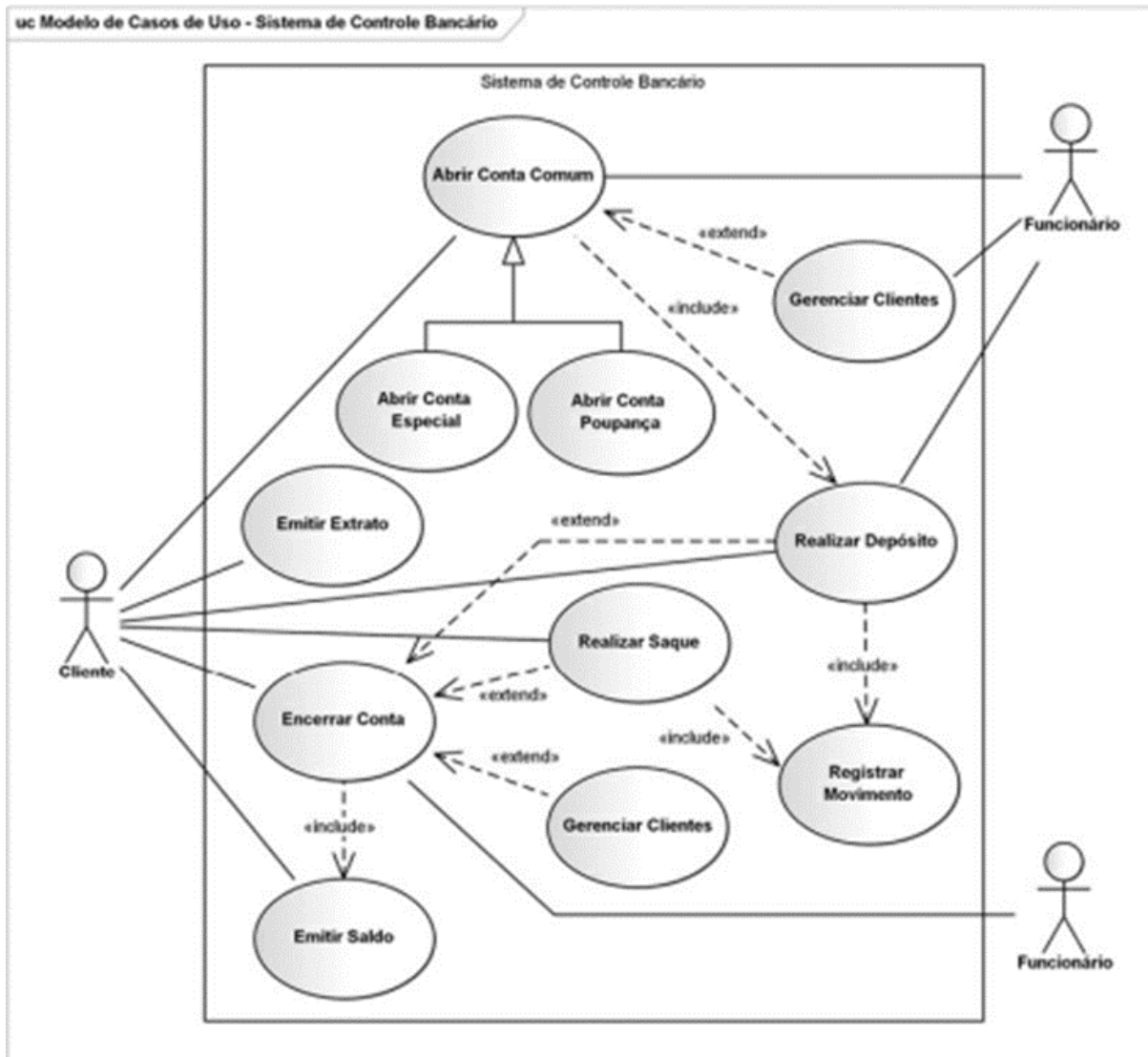
### *3.1.1.4 - Diagrama de Casos de Uso*

Na linguagem UML, o diagrama de casos de uso descreve as principais funcionalidades do sistema, uma visão geral que ele deverá oferecer aos usuários (Guedes, 2018). É um diagrama comportamental<sup>1</sup>. Esse tipo de diagrama demonstra como os casos de uso interagem entre si e com os atores. Guedes (2018), exemplifica utilizando o modelo de um sistema de controle bancário:

---

<sup>1</sup> <https://www.ateomomento.com.br/o-que-e-caso-de-uso/>

Figura 1 – Diagrama de Caso de Uso

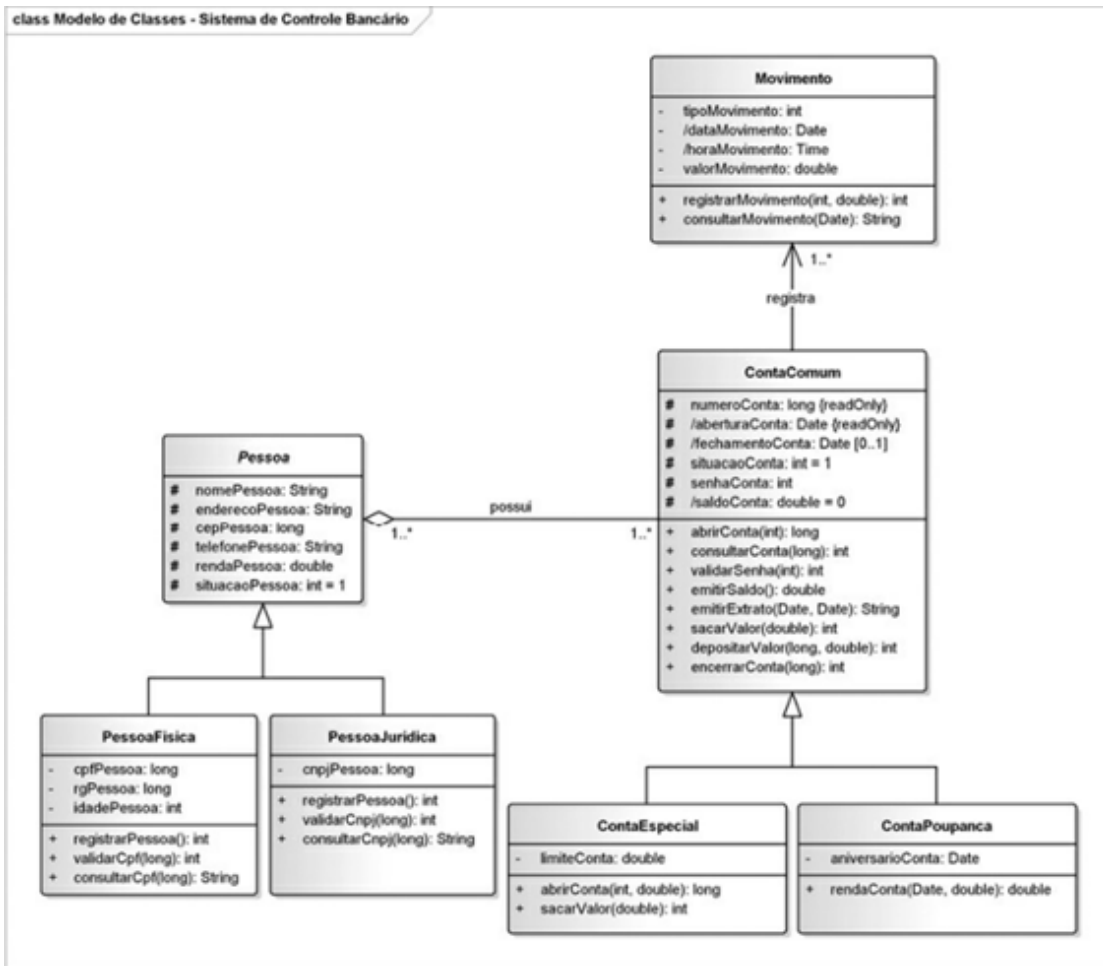


Fonte: Guedes, 2018.

### 3.1.1.5 - Diagrama de Classes

Esse diagrama é o mais utilizado na linguagem UML. A partir dele define-se categorias/classes que vão compor o sistema, formando uma representação gráfica desse conjunto. Normalmente esse diagrama é desenhado utilizando retângulos e setas e dentro deles estão as classes, os atributos e os métodos. Além disso, também demonstra como essas classes se relacionam entre si, indicação importante para o diagrama representado pelas setas (Guedes, 2018). Dentro desses retângulos é preciso conter 3 divisórias. A primeira seria o nome da classe, a segunda os atributos dessa classe e a terceira os métodos que a classe possui (Nascimento, 2022).

Figura 2 – Modelo de Classes

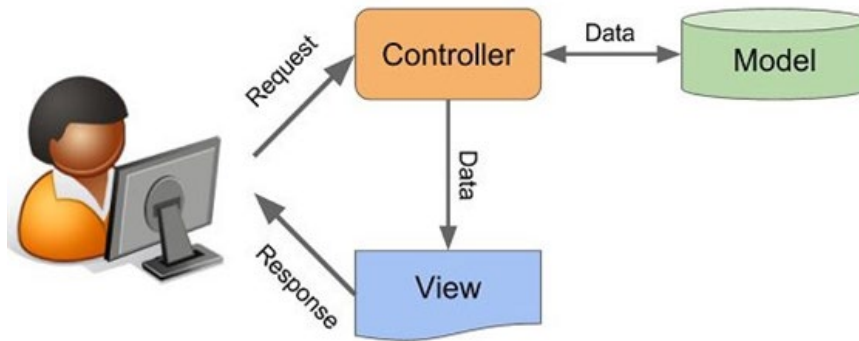


Fonte: Guedes, 2018.

### 3.1.2 - Model-View-Controller MVC

O MVC (modelo-visão-controlador) é um tipo de padrão de arquitetura, muito utilizado em vários sistemas baseados em web (Sommerville, 2011). O componente modelo é responsável por gerenciar as formas como os dados se comportam e as operações associadas a esses dados. A visão é responsável pela forma em que esses dados aparecem para os usuários, ou seja, sua interface, sendo ela a camada de interação com o usuário. (Nascimento, 2022). Já o controlador é responsável pelo fluxo de dados, responsável por ligar o Model e o View, fazendo com que as ações do usuário sejam enviadas para os Models e/ou os Views.

**Figura 3 - Model-View-Controller**



Fonte: Nascimento, 2022.

### **3.2 - Desenvolvimento Web: os desenvolvimentos de sistemas-web**

O avanço da internet é um dos aspectos que levam ao avanço de tecnologias web. Atualmente vemos diversos softwares gerindo várias áreas da sociedade, um estacionamento de shopping, um sistema de monitoramento de um condomínio, um sistema de vendas online, todos estes são sistemas encontrados no nosso cotidiano. Ou seja, essas ferramentas se tornaram essenciais quando nos referimos a diversos aspectos, como: segurança, agilidade, comércio e até mesmo política.

Algumas modificações ocorreram durante esses anos de uso da internet. O uso de Websites, um espaço virtual com modelo estático e exposição de conteúdo (textos, links, vídeos e imagens) não era suficiente para novas modalidades exigidas no avanço da internet. Essa modalidade até atendia ao público que a procurava, principalmente aqueles que queriam focar no marketing, mas para quem precisava da melhoria do sistema interno de gerenciamento ou de funções como cadastramento de produtos, não era suficiente. Com isso, avançou-se nos novos recursos da internet, dando espaço para os Softwares Web. Com essa nova modalidade, os sistemas poderiam ser acessados por qualquer dispositivo que tivesse acesso a um navegador, sem a necessidade de instalar um software num computador local. (Paula, Almeida, Gorges, Fernandes & Sanjulião, 2020)

Sendo assim, um sistema-web possui novas adesões tecnológicas, na qual a web não possui. Santos (2018 apud Paula, Almeida, Gorges, Fernandes & Sanjulião) afirma que os sistemas web são dinâmicos, onde a “experiência do usuário é personalizada” (p.04). Tal

sistema dá abertura para criação de algumas funções importantes para esse trabalho, como: um login e o gerenciamento de dados, a dinamicidade dos dados, o poder de atualização e manutenção do sistema. Além disso, um sistema-web é acompanhado por um banco de dados próprio, o que facilita tal gerenciamento.

Miletto & Bertagnolli (2014) afirmam que dentro das aplicações web é preciso considerar alguns aspectos como: “a) quais tecnologias podem ser utilizadas; b) como é realizado o processamento do sistema; c) quais são o perfil do usuário e suas tarefas na interface gráfica da aplicação; d) como armazenar os dados” (p. 05). Além disso, para que um sistema web seja considerado robusto e consistente, ou seja, seguro, dinâmico, rápido, é necessário que ele possua quatro aspectos, que estão relacionados “com as boas práticas de desenvolvimento de software” (p. 05), permitindo que ao atualizar uma interface, as outras não sejam prejudicadas. São estes:

**Figura 4 – Organização da Aplicação em Camadas<sup>2</sup>.**



Fonte: Miletto & Bertagnolli, 2014.

Atualmente a requisição do sistema web também se dá pelo que ele mesmo oferece. Diferentemente dos websites, que não permitiam tanta mobilidade nas suas configurações, o sistema web permite que softwares sejam instalados e utilizados num servidor sem ter que

---

<sup>2</sup> Fonte: Desenvolvimento de software II: introdução ao desenvolvimento web com HTML, CSS, JavaScript e PHP/ Organizadores: Evandro Manara Miletto, Silvia de Castro Bertagnolli – Porto Alegre: Bookman, 2014.

alterar o software do outro lado da conexão, do cliente. O mesmo pode ser utilizado em muitos navegadores, sem que se sofra alterações (Groef, 2016 apud Kulesza, Sousa, Lima, Araujo, Filho, 2018).

A importância da discussão acima, está na finalidade desses sistemas. No caso, atender ao pedido do cliente de maneira a ser um sistema confiável e adaptável. Para a criação desses sistemas, utiliza-se linguagens como HTML, CSS, JavaScript e Python, na qual será explicada nos tópicos abaixo.

### **3.3 - JavaScript**

JavaScript é uma linguagem de programação da Web e uma das mais populares. Essa linguagem pode ser executada diretamente em navegadores web e tem como característica a sua tipagem dinâmica (Miletto & Bertagnolli, 2014). A mesma “adiciona funcionalidades interativas em uma página web por meio de scripts que são executados no lado cliente e são capazes de alterar a página sem ter que passar pelo servidor” (Yoshiura, 2015, p. 59)

### **3.4 - Django**

Django é um framework web de alto nível para Python, conhecido por facilitar o desenvolvimento rápido e limpo de aplicações web. Criado com o objetivo de ajudar os desenvolvedores a construir aplicações web complexas de forma simples e eficiente, o Django segue o princípio de "Don't Repeat Yourself" (DRY), o que incentiva a reutilização de código e a redução de redundâncias.

Django é um framework de código aberto que oferece uma série de ferramentas e recursos prontos para uso, como um ORM (Object-Relational Mapping) poderoso, um sistema de administração automatizado e um sistema de roteamento flexível. Além disso, ele é conhecido por sua segurança robusta, com proteções contra ataques comuns como SQL injection e cross-site scripting (XSS).

Por ser altamente escalável e fácil de integrar com outras tecnologias, Django é amplamente utilizado para construir desde pequenos sites até grandes aplicações web. Sua

comunidade ativa e a vasta documentação tornam-no uma escolha popular entre desenvolvedores que procuram uma solução completa para o desenvolvimento de aplicações web com Python.

### **3.5 - HTML**

Enquanto a linguagem JavaScript auxilia na criação de páginas interativas, o HTML (Hypertext Markup Language - Linguagem de Marcação de Hipertexto) é uma linguagem padrão para acessar as páginas da internet (Costa, 2007 apud Kieras, 2019) responsável pela estrutura base das páginas. Uma de suas características principais é o uso de Hipertextos. Essa linguagem também conhecida como linguagem de marcação é composta por textos e códigos conhecidos como tags. Essas tags iniciam e terminam a marcação e usam símbolos como “menor que” (<) e “maior que” (>) (Miletto & Bertagnolli, 2014). Sua composição é para publicação de conteúdo (texto, imagem, vídeo, áudio e etc) na Web. Sendo assim, para visualização e interpretação do código HTML é necessário um navegador.

### **3.6 - CSS**

CSS (Folha de Estilo em Cascata) é uma linguagem de estilo, como o próprio nome diz, responsável pela formatação e apresentação do conteúdo: suas bordas, cores, fontes, fundo e layouts. Ela surge para complementar a linguagem HTML, já que é responsável pela estrutura visual da página. Ou seja, é ele que define a aparência em páginas web, que utilizam a linguagem de marcação HTML. Essa linguagem permite ainda que em um único lugar se defina a formatação que será utilizada por cada TAG, “com isso, apenas um arquivo é alterado, sendo que a mudança é automaticamente propagada a todas as páginas que compõe o site”. (Miletto & Bertagnolli, 2014, p.70)

### **3.7 - Python**

Python é uma linguagem de programação de alto nível, desenvolvida por Guido van Rossum e lançada pela primeira vez em 1991. Conhecida por sua sintaxe simples e legível, Python permite aos desenvolvedores escrever código de forma mais rápida e com menos esforço, o que a torna ideal tanto para iniciantes quanto para profissionais experientes. Python pode ser utilizada em uma variedade de ambientes, incluindo computadores, servidores e dispositivos móveis, e é capaz de criar uma ampla gama de aplicações, desde scripts simples e

automação até software robusto e completo, aplicativos web, análise de dados e inteligência artificial.

Uma das principais características de Python é sua vasta biblioteca padrão e a enorme quantidade de pacotes de terceiros, que facilitam a adição de funcionalidades ao desenvolvimento de software. Além disso, Python é uma linguagem multiplataforma, o que significa que o código pode ser executado em diferentes sistemas operacionais, como Windows, macOS e Linux, sem a necessidade de modificações significativas.

A linguagem é amplamente utilizada em diversas indústrias e setores, devido à sua flexibilidade, comunidade ativa e suporte contínuo, o que a torna uma das linguagens de programação mais populares no mundo atualmente.

### **3.8 - Banco de dados**

Banco de Dados é uma ferramenta que será utilizada para armazenar arquivos e informações importantes para uma empresa. Essas informações se relacionam entre si. A ferramenta possui três características principais: Armazenar os dados; relacionar esses dados armazenados; recuperar os dados rapidamente (Angelotti, 2010).

O Sistema Gerenciador de Banco de Dados (SGBD) é um conjunto de software utilizado para gerenciar dados. É uma ferramenta fácil de ser manuseada e controlada (Yoshiura, 2015). Suas principais características são: controlar, acessar, organizar e proteger as funções de uma aplicação.

“o SGBD facilita o processo de definir (especificação de dados a serem armazenados), construir (processo de armazenar os dados em algum meio controlado pelo SGBD) e manipular (inserção, atualização, eliminação e recuperação de dados) um banco de dados de diversas aplicações”. (Yoshiura, 2015, p.54)

Essa ferramenta tem sido uma das mais utilizadas nos últimos tempos por empresas, justamente pela segurança que a mesma fornece.

### **3.9 - SQLite3**

SQLite3 é um sistema de gerenciamento de banco de dados do tipo relacional, que é implementado como uma biblioteca em código aberto, disponível em várias plataformas, incluindo UNIX e Windows. Diferente de muitos outros bancos de dados relacionais, o SQLite3 é um banco de dados leve e auto-contido, que armazena todo o banco de dados em

um único arquivo no disco. Ele é compatível com as propriedades ACID (Atomicidade, Consistência, Isolamento e Durabilidade), que são fundamentais para garantir a integridade das transações em um banco de dados relacional. Devido à sua simplicidade e facilidade de uso, o SQLite3 é amplamente utilizado em aplicações de software embarcado, aplicativos móveis e outros sistemas onde um banco de dados leve e sem servidor é necessário.

### **3.10 - Microsoft Visual Studio Code**

O Visual Studio Code ou VSCode, é um editor de código desenvolvido pela Microsoft. Esse editor comporta mais linguagens de programação e pode ser utilizado em vários sistemas operacionais, como Windows, MacOS e Linux. Ainda vem com um suporte integrado para a linguagem JavaScript, TypeScript e Node.js. Além do mais, o editor possui sintaxe de diversas linguagens como Python, Ruby, C ++, PHP. Outra curiosidade sobre o VSCode é que ele é um editor de código aberto e seu código está disponível no GitHub, assim a comunidade técnica pode ajudar na melhoria do seu desenvolvimento.

### **3.11 - Lucidchart**

Lucidchart é um aplicativo com foco na diagramação e wireframes. Diversas áreas que vão além da TI utilizam o Lucidchart, pois é uma ferramenta com suas facilidades em se trabalhar e aprender.

### **3.12 - BRmodelo**

O BRmodelo foi criado para a modelagem de Banco de Dados relacionais. Sua interface web faz com que qualquer pessoa tenha acesso ao BRmodelo. Além do seu código ser aberto e totalmente gratuito.

#### **4 - METODOLOGIAS UTILIZADAS**

Como mencionado, este trabalho consiste em um projeto de conclusão de curso, com foco na implementação de um sistema sob demanda. A partir disso, foi desenvolvido um sistema web com funcionalidades específicas para agendamento. Este capítulo descreve os métodos e as ferramentas utilizadas no desenvolvimento do sistema. Uma das primeiras ferramentas adotadas foi o levantamento de requisitos, essencial para a compreensão das demandas do cliente.

Para compreender as necessidades do cliente e definir os requisitos do sistema, foi realizada uma entrevista inicial com o proprietário da barbearia, que também é o administrador do sistema. Essa entrevista ocorreu no início do projeto e teve como objetivo identificar os principais problemas enfrentados com o agendamento manual, bem como as expectativas em relação ao sistema online. Durante essa entrevista, foram levantados pontos como a necessidade de uma agenda em tempo real, a facilidade de uso para os clientes e a possibilidade de divulgar os trabalhos dos barbeiros.

Além da entrevista inicial, houve encontros informais regulares com o proprietário ao longo do desenvolvimento do sistema. Um dos autores do trabalho era frequente na barbearia por ser cliente da mesma. Acabava que os encontros com o proprietário eram mensais durante os cortes de cabelo. Nessas ocasiões, ele fornecia feedbacks e sugestões sobre o sistema, compartilhando ideias sobre funcionalidades que gostaria de ver implementadas, como a possibilidade de cancelar ou reagendar agendamentos diretamente pelo sistema, e a exibição dos horários disponíveis de forma clara.

Essas interações informais foram fundamentais para ajustar o sistema às necessidades reais da barbearia, garantindo que o desenvolvimento estivesse alinhado com as expectativas do cliente. Além disso, elas permitiram uma comunicação contínua e eficiente, onde dúvidas e sugestões podiam ser discutidas de forma ligeira, sem a necessidade de agendamentos formais.

Por fim, ao término do desenvolvimento do sistema, foi realizada uma entrevista final com o proprietário para coletar seu feedback sobre o sistema implementado. Nessa entrevista, ele pôde avaliar as funcionalidades, a usabilidade e a eficácia do sistema em resolver os problemas de agendamento que enfrentava anteriormente. O feedback nessa etapa foi essencial para validar o sistema e identificar possíveis melhorias para versões futuras.

## 4.1 - Levantamento de Requisitos

O levantamento de requisitos é uma etapa inicial crucial para o desenvolvimento de software, pois é fundamental para a compreensão do problema que o sistema se propõe a resolver. Para isso, foram realizadas diversas reuniões e entrevistas com o proprietário da barbearia, a fim de entender suas necessidades e expectativas. Essa fase incluiu a coleta de requisitos funcionais (como o sistema de agendamento online) e não funcionais (como a segurança dos dados e a usabilidade). Além disso, foram analisados sistemas similares disponíveis no mercado para identificar boas práticas e funcionalidades que poderiam ser incorporadas ao nosso sistema.

### 4.1.2 - Requisitos Funcionais

O sistema desenvolvido apresenta os seguintes requisitos funcionais:

**Quadro 3 – Requisitos Funcionais**

<b>Nº Requisito</b>	<b>Descrição</b>
RF 01	Permitir fazer o cadastro dos profissionais da barbearia
RF 02	Permitir que o perfil dos profissionais possa gerenciar suas agendas
RF03	Permite que o profissional veja os clientes que estão agendados
RF04	Cadastrar clientes no sistema da barbearia
RF05	Cliente poderá editar seu perfil dentro do sistema
RF06	Cliente poderá acessar o perfil dos profissionais e ver seu contato

RF07	Cliente poderá cancelar ou remarcar seu agendamento
------	---

Fonte: Elaborado pelos autores.

#### 4.1.3 - Requisitos não-funcionais

Como atributos de qualidade ao nosso sistema, foram definidos os seguintes requisitos não-funcionais:

**Quadro 4 – Requisitos não-funcionais**

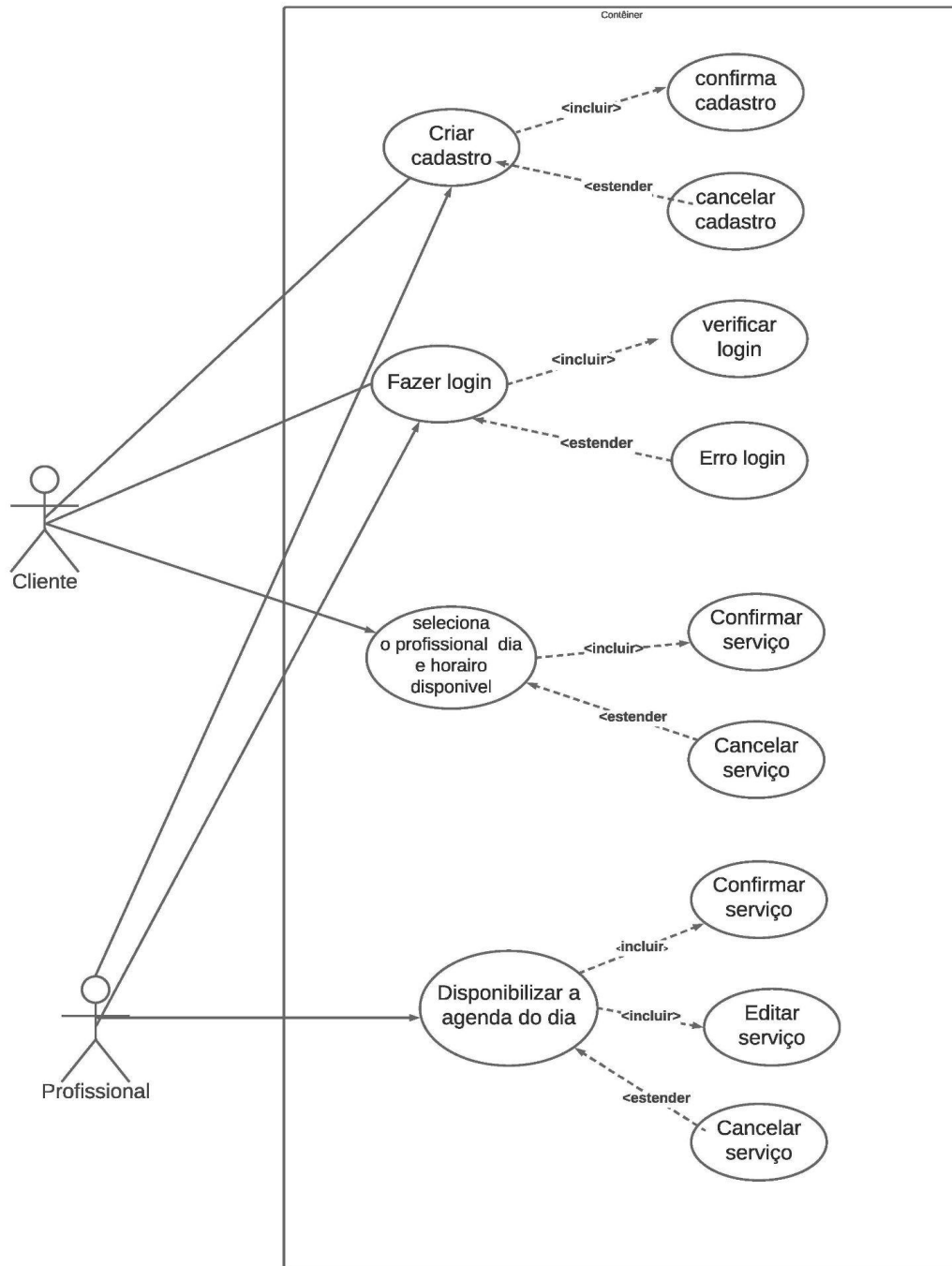
Nº Requisito	Descrição
RNF01	O sistema deve ser desenvolvido a linguagem de programação python
RNF02	O sistema deve ter uma boa usabilidade com uma interface intuitiva
RNF03	O sistema deve ser seguro, usando sistema de autenticação do usuário
RNF03	O sistema deve ser web
RNF05	O sistema deverá ser compatível com diversos navegadores

Fonte: Elaborado pelos autores.

#### 4.2 - Casos de Uso

Considerando a importância dos Casos de Uso na Engenharia de Software, especialmente no que diz respeito à relação entre o usuário e o sistema, além de sua função na documentação das funcionalidades do software, apresentamos a seguir o diagrama de Casos de Uso do sistema.

**Figura 5 – Diagrama de Caso de Uso**



Fonte: Elaborado pelos autores.

Quadro 5 - Caso de uso 1

**Gerenciamento da Agenda do Barbeiro**

Descrição: O objetivo deste caso de uso é o sistema permite que o barbeiro adicione, modifica e exclua horários disponíveis
Pré condição: O barbeiro deve estar cadastrado no sistema
Pós Condição: O usuário terá disponibilizado os dias e horários conforme a sua disponibilidade
Requisitos Funcionais: RF01 - O sistema deverá permitir a inserção de dados solicitados somente para profissionais cadastrado
Fluxo Principal: 1. O barbeiro faz login no sistema 2. O Barbeiro pode visualizar a agenda, adicionar horários, editar horários, excluir horários 3. O barbeiro confirmar agendamento dos clientes 4. O sistema atualiza a agenda em tempo real

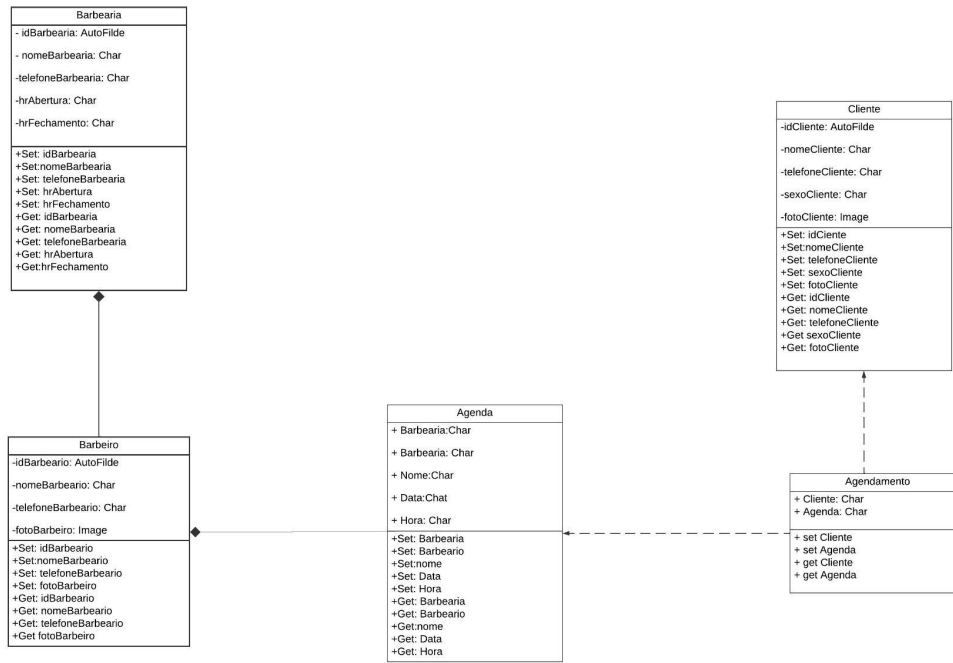
**Quadro 6 - Caso de uso 2**

<b>Cliente agendando horário com o barbeiro</b>
Descrição: O objetivo deste caso de uso é o sistema deixa disponível os dias e horários disponíveis com os dados fornecidos pelo usuário “admin” barbeiro”
Pré-condição: Os clientes precisam estar logados no sistema O Barbeiro deve ter horário disponível
Pós-Condição: O agendamento deve aparecer na agenda do barbeiro e do cliente
Requisitos Funcionais: RF01 - O sistema deverá permitir a inserção de dados solicitados somente para profissionais cadastrado
Fluxo Principal: 1. O Cliente acessa o sistema de agendamento 2. O Cliente visualiza a agenda do barbeiro desejado, selecione uma data e horário disponível 3 O cliente confirma o agendamento com informações de contatos atualizados

### 4.3 - Diagrama de Classes

Dentro da Linguagem UML, para um mapeamento mais eficaz da estrutura do sistema por meio das classes e objetos, foi utilizado o diagrama de classes, que está apresentado a seguir.

**Figura 6 – Diagrama de Classes**

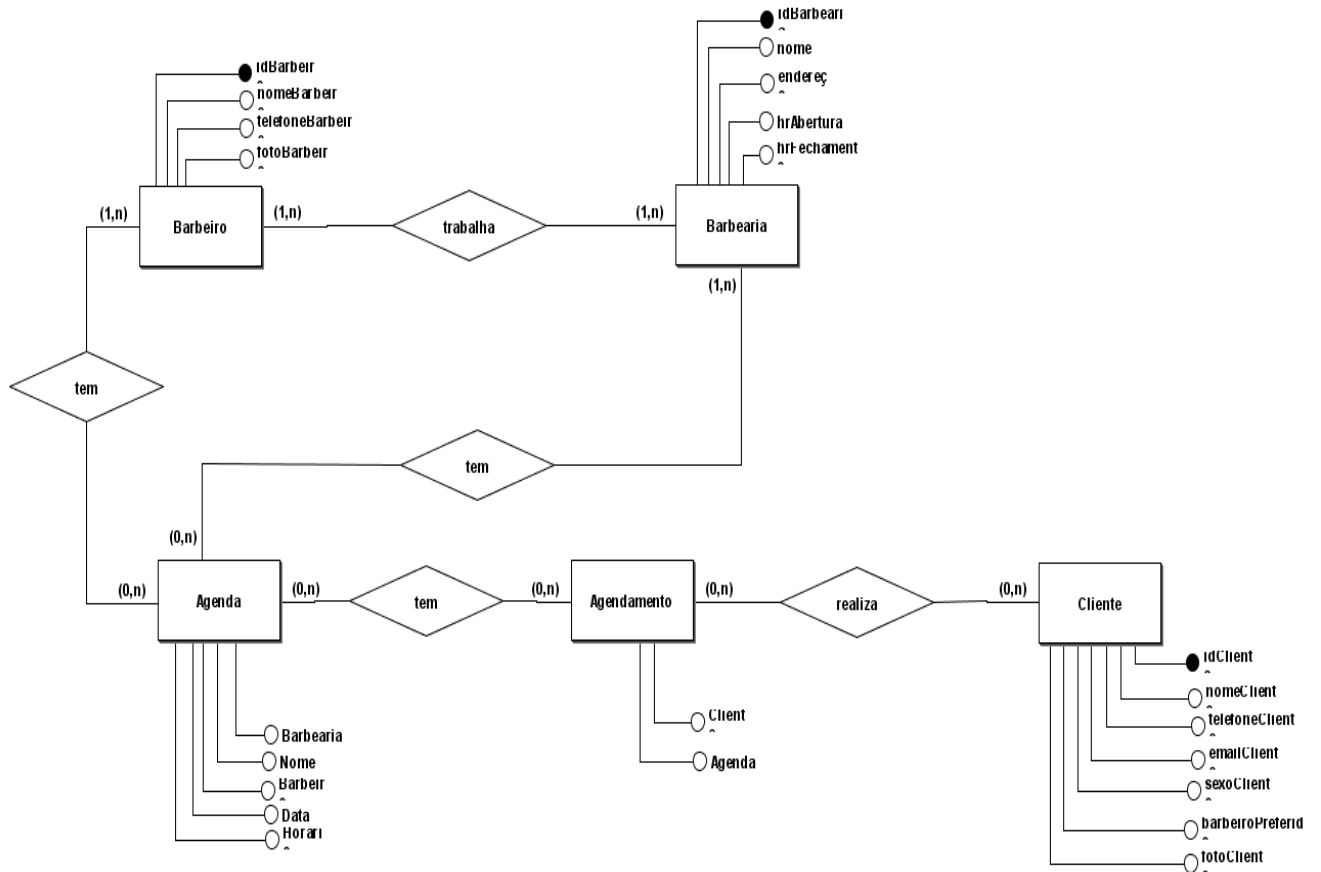


Fonte: Elaborado pelos autores.

#### 4.4 Diagrama de entidade e relacionamento - DER

Abaixo será apresentado o diagrama de entidade de relacionamento a fim de apresentar um fluxo relacional:

**Figura 7 – Diagrama de Entidade e Relacionamento**



Fonte: Elaborado pelos autores.

## 5. APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

Neste capítulo, apresentamos o sistema desenvolvido, seus resultados, funcionalidades e os desafios enfrentados durante o seu desenvolvimento.

### 5.1 - A solução para o cliente

Como abordado na apresentação da problemática que fundamenta este trabalho, a busca por uma solução envolveu, inicialmente, a compreensão das necessidades do cliente e das demandas de sua clientela. Sabendo que o foco principal era a organização dos horários e a facilidade de acesso para os clientes, foi idealizado um sistema intuitivo, de fácil compreensão para os usuários. Para isso, projetou-se uma interface interativa e simples, capaz de atender tanto aos comandos dos barbeiros quanto dos clientes.

O cliente destacou que a implementação de um modelo tecnológico para sua barbearia reduziria problemas relacionados à agenda, especialmente em relação aos conflitos de horários. Além disso, ressaltou que a coleta de dados sobre os atendimentos, por meio de uma agenda online, contribuiria significativamente para o controle financeiro e organizacional do estabelecimento. Com isso, a solução proposta consistiu em um sistema que não apenas organizasse a rotina dos trabalhadores, mas também gerasse dados úteis para o proprietário.

Sendo assim, foi apresentada as seguintes ferramentas dentro do sistema para o cliente:

Ferramentas para o agendamento:

- Funcionalidade para que os clientes possam fazer seus agendamentos.

Ferramenta de Cancelamento e Reagendamento:

- Funcionalidade para que os clientes possam cancelar ou reagendar seus agendamentos diretamente pelo sistema.

Gestão de Funcionários e Serviços

- Gerenciamento de Barbeiros:
  - Painel administrativo para controle de horários dos barbeiros, com visualização das agendas individuais.
- Gestão de Serviços:
  - Controle de disponibilidade de serviços com base nos horários dos barbeiros e na demanda, ajustando a agenda para otimizar o atendimento.

A partir das soluções oferecidas ao cliente, foi criado o sistema que será descrito nas próximas páginas.

## **5.2. Desenvolvimento do Sistema**

O sistema web desenvolvido tem como objetivo auxiliar na gestão de tempo de uma barbearia, otimizando a agenda tanto dos funcionários quanto dos clientes. Reconhecendo que a tecnologia está cada vez mais presente nos serviços disponíveis no mercado, o cliente optou por adaptar sua agenda cotidiana de forma mais moderna. Assim, o sistema apresentado neste trabalho funciona como uma ferramenta de apoio profissional, de manuseio simples, mas extremamente funcional.

Como mencionado anteriormente, algumas escolhas tecnológicas foram feitas para a construção do sistema, com base em uma revisão teórica que nos orientou para ferramentas mais seguras e adequadas ao desenvolvimento. No projeto inicial, optamos pelo uso de Java para os comandos principais do sistema, mas, após algumas tentativas frustradas, decidimos utilizar Python devido à sua fluidez e proximidade com as necessidades do projeto.

Dessa forma, o backend foi desenvolvido com Python e Django, seguindo o padrão arquitetural MVC (Model-View-Controller). Para a edição dos códigos, utilizamos o Visual Studio Code, que, além de ser um editor de código aberto, possui seu código disponível no GitHub, plataforma também utilizada para armazenar o código-fonte. As funcionalidades principais foram implementadas, incluindo o sistema de agendamento, gerenciamento de barbeiros e serviços, controle de promoções e envio de notificações. A segurança do sistema foi uma prioridade, com a implementação de medidas como autenticação de usuários.

Optou-se pelo Django como framework principal devido à sua robustez e capacidade de acelerar o desenvolvimento de aplicações web. Além disso, o Django oferece uma excelente integração com o banco de dados SQLite, escolhido por sua leveza e facilidade de configuração, ideal para um projeto de porte pequeno a médio.

A modelagem do banco de dados foi realizada com as ferramentas do Django, criando tabelas para representar barbeiros, clientes, serviços, agendamentos e promoções. Essa modelagem foi baseada nos requisitos levantados, garantindo que todas as informações necessárias estivessem devidamente estruturadas e inter-relacionadas.

Para o desenvolvimento do frontend, utilizamos HTML para agregar conteúdo à página, além de CSS e JavaScript para garantir uma interface responsiva e amigável ao usuário. Foram criadas páginas para agendamento de serviços, visualização de perfis de barbeiros com apresentação de sua foto no perfil. O JavaScript foi utilizado para funcionalidades dinâmicas no frontend, enquanto HTML e CSS foram usados para a estruturação e design da interface.

Para a diagramação e o desenho do sistema, utilizamos o Lucidchart. Além disso, o BRModelo foi empregado na modelagem dos dados.

Quanto à implantação do sistema, ele foi configurado para um ambiente de produção, tornando-se acessível online para os usuários finais. Um servidor web com Django foi configurado, e o banco de dados SQLite foi migrado para o ambiente de produção.

Para a hospedagem do sistema web, foi utilizada a plataforma PythonAnywhere, escolhida por permitir o desenvolvimento e a hospedagem de aplicações Python na nuvem. Como se trata de uma ferramenta online, ela possibilita a edição de códigos diretamente pelo navegador, oferecendo um servidor integrado, interface de linha de comando e um editor de código com realce de sintaxe. Além disso, a plataforma permite integração com o banco de dados utilizado no sistema.

Para manipulação de imagens, utilizou-se a biblioteca Pillow<sup>3</sup>, uma ferramenta versátil para processar imagens em Python, suportando diversos formatos. Essa biblioteca permite realizar edições e operações avançadas de maneira simplificada, graças à sua API intuitiva.

Por fim, para a criação de gráficos baseados nos dados gerados pelo sistema, como informações sobre agendamentos, foi utilizada a biblioteca Chart.js<sup>4</sup>. Essa ferramenta, oferece gráficos interativos e de fácil interpretação, utilizando JavaScript, além de disponibilizar uma ampla variedade de tipos de gráficos.

### **5.3 - Apresentação do sistema**

Nesse momento serão apresentadas as telas do sistema por meio de cenários de uso, na qual demonstrará como de fato o site se apresentará aos usuários.

---

<sup>3</sup> Pillow. Disponível em: <https://pillow.readthedocs.io/en/stable/>. Acessado em: 14 de janeiro de 2025.

<sup>4</sup> Chart.js. Disponível em: <https://www.chartjs.org/docs/latest/>. Acesso em: 14 de janeiro de 2025.

### 5.3.1 - Telas dos sistemas e os cenários de uso

- **Dos barbeiros**

Esta é a primeira tela (figura 8) visualizada pelos barbeiros que irão administrar suas agendas, a tela de login.

**Figura 8 – Tela de login**

Login

Usuário:

Senha:

Não tem uma conta? [Cadastre-se](#)

Fonte: arquivo pessoal.

#### **Cenário de uso 1: Login (Barbeiro)**

<b>Cenário:</b>	Login
<b>Usuário:</b>	Barbeiro, Cliente
<b>Objetivo:</b>	Logar no sistema de agendamento
<b>Entrada:</b>	Dados solicitado
<b>Ação:</b>	1 preencher dados solicitado 2 clicar em Login 3 Visualizar página Home
<b>Saída:</b>	Barbeiro e cliente autenticado, visualizar página home

Abaixo (figura 9) apresentamos a Página Inicial do sistema, com uma breve descrição da barbearia com os serviços ofertados e sua localização. Esta tela aparece para ambos, clientes e barbeiros.

**Figura 9 – Tela inicial**



Fonte: arquivo pessoal.

### Cenário de uso 2: Página inicial

<b>Cenário:</b>	Página Home
<b>Usuário:</b>	Barbeiro, Cliente
<b>Objetivo:</b>	Escolher Funcionalidade
<b>Entrada:</b>	Funcionalidade selecionada
<b>Ação:</b>	Clicar em funcionalidade desejada
<b>Saída:</b>	Abrir funcionalidade selecionada

A tela a seguir (figura 10) mostra o processo de cadastro no site feito pelos clientes. Além disso, os barbeiros podem realizar novos cadastros de clientes que desejem que a empresa realize o agendamento.

**Figura 10 - Tela Cadastrar Novo Cliente**

HOME CLIENTES BARBEARIAS BARBEIROS AGENDAS AGENDAMENTOS B & B

### Cadastrar Novo Cliente

NomeCliente:

TelefoneCliente:

EmailCliente:

SexoCliente:

BarbeiroPreferido:

FotoCliente:  
 Nenhum arquivo escolhido

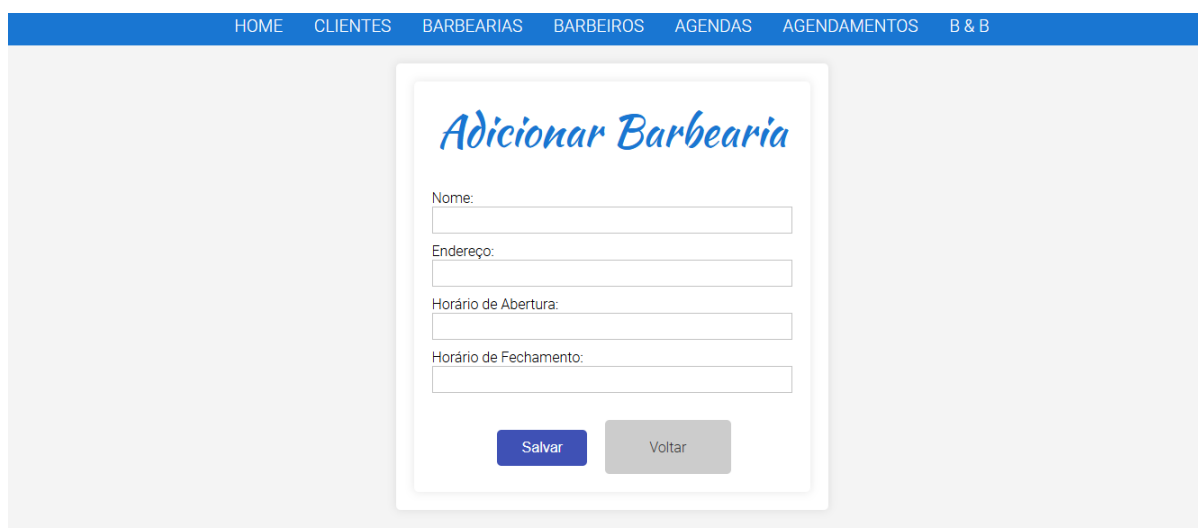
Fonte: arquivo pessoal.

#### **Cenário de uso 4: Cadastro de cliente novo**

<b>Cenário:</b>	Cadastrar novo Cliente
<b>Usuário:</b>	Barbeiro, Cliente
<b>Objetivo:</b>	Adicionar novo Cliente
<b>Entrada:</b>	Dados solicitados
<b>Ação:</b>	Preencher dados solicitados Clicar em botão Salvar Visualizar resultado
<b>Saída:</b>	Cliente adicionado

Neste cenário (figura 11), observa-se que o sistema apresentado também oferece a opção de cadastrar novas barbearias. Isso é especialmente útil caso o proprietário da empresa decida abrir novas franquias e queira centralizar as agendas em um único sistema web. Dessa forma, o sistema é capaz de gerenciar mais de uma agenda simultaneamente.

**Figura 11 - Tela Adicionar Barbearia**



Fonte: arquivo pessoal.

### **Cenário de uso 5: Adicionar novas barbearias**

<b>Cenário:</b>	Adicionar Barbearia
<b>Usuário:</b>	Barbeiro
<b>Objetivo:</b>	Adicionar nova barbearia
<b>Entrada:</b>	Dados solicitado
<b>Ação:</b>	Preencher dados solicitado Clicar no botão salvar Visualizar resultado
<b>Saída:</b>	Barbearia adicionada a lista

Nesta tela (figura 12), é possível cadastrar os barbeiros que trabalham na empresa, permitindo a criação e o gerenciamento de seus respectivos agendamentos.

**Figura 12 - Tela Cadastrar Novo Barbeiro**

HOME CLIENTES BARBEARIAS BARBEIROS AGENDAS AGENDAMENTOS B & B Logout

### Cadastrar Novo Barbeiro

NomeBarbeiro:

TelefoneBarbeiro:

FotoBarbeiro:  
 Nenhum arquivo escolhido

Fonte: arquivo pessoal.

#### **Cenário de uso 6: Cadastro de novos barbeiros no sistema**

<b>Cenário:</b>	Cadastrar novo Barbeiro
<b>Usuário:</b>	Barbeiro
<b>Objetivo:</b>	Adicionar novo Barbeiro
<b>Entrada:</b>	Dados solicitados
<b>Ação:</b>	Preencher com os dados solicitado Clicar no Botão Salvar Visualizar resultado
<b>Saída:</b>	Barbeiro adicionado à lista

Nesta tela (figura 13), é possível cadastrar novos horários e dias de disponibilidade dos barbeiros, ampliando as opções de atendimento na barbearia.

**Figura 13 – Tela Cadastrar Nova Agenda**

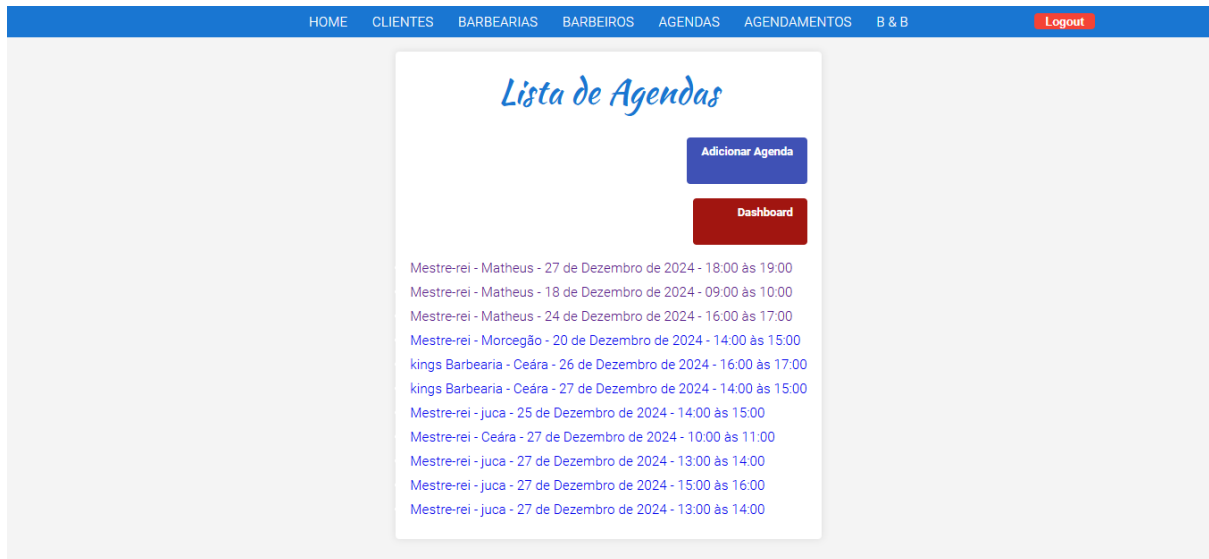
Fonte: arquivo pessoal.

### **Cenário de uso 7: Cadastrar nova agenda**

<b>Cenário:</b>	Cadastrar nova Agenda
<b>Usuário:</b>	Barbeiro
<b>Objetivo:</b>	Disponibilizar horários para atendimento ao cliente
<b>Entrada:</b>	Clicar em Adicionar Agenda
<b>Ação:</b>	Preencher dados solicitados Clicar em Salvar Visualizar resultado
<b>Saída:</b>	Lista de agendas

Já nesta tela (figura 14), a agenda é apresentada ao barbeiro, contendo informações como o nome do barbeiro, os dias e horários disponíveis, permitindo que o cliente escolha a opção mais conveniente.

**Figura 14 – Tela Lista de Agendas**



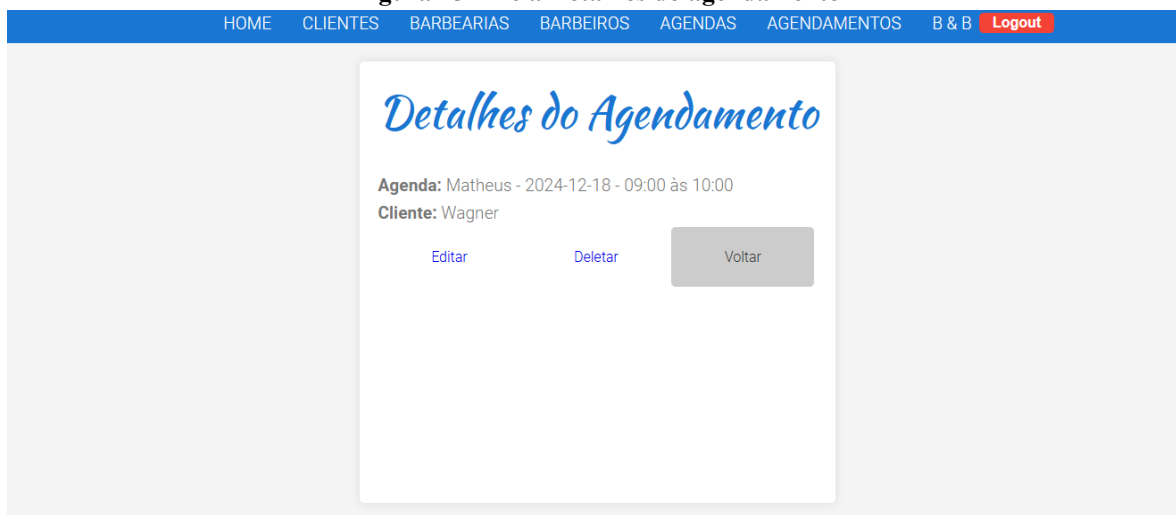
Fonte: arquivo pessoal.

### **Cenário de uso 8: A agenda**

<b>Cenário:</b>	Lista de Agendas
<b>Usuário:</b>	Barbeiro
<b>Objetivo:</b>	horários disponíveis com o barbeiro
<b>Entrada:</b>	Clicar em Agendas
<b>Ação:</b>	Clicar em agenda
<b>Saída:</b>	Horário disponível para agendamento

Nesta tela (figura 15), apresenta-se mais informações relacionada a agenda para o barbeiro, como o nome do barbeiro, o horário e a data.

**Figura 15 – Tela Detalhes do agendamento**



Fonte: arquivo pessoal.

### **Cenário de uso 9: Detalhamento da agenda**

<b>Cenário:</b>	Detalhes da Agenda
<b>Usuário:</b>	Barbeiro
<b>Objetivo:</b>	Verificar detalhes da agenda; editar ou deletar horário disponível
<b>Entrada:</b>	Clicar em horário disponível
<b>Ação:</b>	Clicar no horário disponível Verificar detalhes do horário Editar ou deletar horário do banco de dados
<b>Saída:</b>	verificar dados, editado ou deletado

Por fim, a agenda dos barbeiros (figura 16). Nela são apresentados dados como: o nome do barbeiro, a data do agendamento, o horário de início e término do corte, além do nome do cliente.

**Figura 16 – Tela Lista de agendamentos**



Fonte: arquivo pessoal.

### **Cenário de uso 10: Os agendamentos**

<b>Cenário:</b>	Lista de Agendamentos
<b>Usuário:</b>	Barbeiros
<b>Objetivo:</b>	Verificar clientes agendado
<b>Entrada:</b>	Clicar agendamentos
<b>Ação:</b>	Clicar em agendamentos Verificar horário marcado com o cliente
<b>Saída:</b>	atender o cliente no horário marcado

Na tela a seguir é apresentado um Dashboard (figura 17) onde contém informações de forma rápida e intuitiva. Essas informações geram dados a empresa, para que mantenha o controle e a gerência de seus atendimentos.

**Figura 17 – Tela Dashboard**



Fonte: arquivo pessoal.

**Cenário de uso 11: Dashboard**

<b>Cenário:</b>	Dashboard
<b>Usuário:</b>	Barbeiro
<b>Objetivo:</b>	Visualizar informações de forma rápida e intuitiva
<b>Entrada:</b>	Clicar em Agendas e no botão Dashboard
<b>Ação:</b>	Clicar em agenda e depois no botão Dashboard Visualizar informação no Dashboard
<b>Saída:</b>	informações de forma rápida e intuitiva

- **Do cliente:**

Ao acessar o sistema, o cliente será direcionado à Página Inicial (figura 18), que apresenta as informações gerais sobre o funcionamento da empresa. Em seguida, o cliente poderá navegar pelas abas disponíveis para realizar seu agendamento.

**Figura 18 – Tela inicial (cliente)**



Fonte: arquivo pessoal.

Na aba “Barbeiros”, aparece ao cliente a lista de barbeiros cadastrados para seu agendamento (figura 19).

**Figura 19 – Tela Lista de Barbeiros**



Fonte: arquivo pessoal.

### Cenário de uso 1 para clientes: Lista de barbeiros cadastrados no sistema

<b>Cenário:</b>	Lista de Barbeiros
<b>Usuário:</b>	Cliente
<b>Objetivo:</b>	Verificar lista de Barbeiros
<b>Entrada:</b>	Clicar em Barbeiros
<b>Ação:</b>	Clicar em barbeiro desejado Verificar dados do barbeiro desejado
<b>Saída:</b>	Dados do Barbeiro

Na tela abaixo (figura 20), apresentam-se os dados referentes ao barbeiro, caso o cliente deseje conhecer melhor quem irá atendê-lo, ou precise entrar em contato direto com o barbeiro.

**Figura 20 – Tela Detalhes do Barbeiro**



Fonte: arquivo pessoal.

### Cenário de uso 2 para clientes: Detalhamento do barbeiro

<b>Cenário:</b>	Detalhes do Barbeiro
<b>Usuário:</b>	Cliente, barbeiro
<b>Objetivo:</b>	Verificar detalhes dos dados do Barbeiro; Editar ou excluir.
<b>Entrada:</b>	Clicar no nome do barbeiro na lista de barbeiros

<b>Ação:</b>	Clicar no nome do barbeiro na lista de Barbeiros Verificar detalhes do barbeiro Editar ou deletar barbeiro do banco de dados
<b>Saída:</b>	verificar dados, editado ou deletado

Nas telas a seguir (figura 21 e 22), são listados os clientes que possuem cadastro na plataforma. Além disso, os barbeiros podem realizar novos cadastros de clientes que desejem que a empresa realize o agendamento.

**Figura 21 – Tela Lista de Clientes**



Fonte: arquivo pessoal.

### **Cenário de uso 3 para clientes: Lista de clientes**

<b>Cenário:</b>	Lista de Clientes
<b>Usuário:</b>	Clientes
<b>Objetivo:</b>	Verificar clientes e dados dos clientes
<b>Entrada:</b>	clicar em cliente
<b>Ação:</b>	Clicar no nome do cliente desejado verificar dados do cliente desejado
<b>Saída:</b>	Lista de clientes e dados do cliente

Tela de novos agendamentos realizados pelos clientes.

**Figura 22 – Tela Cadastrar Novo Agendamento**

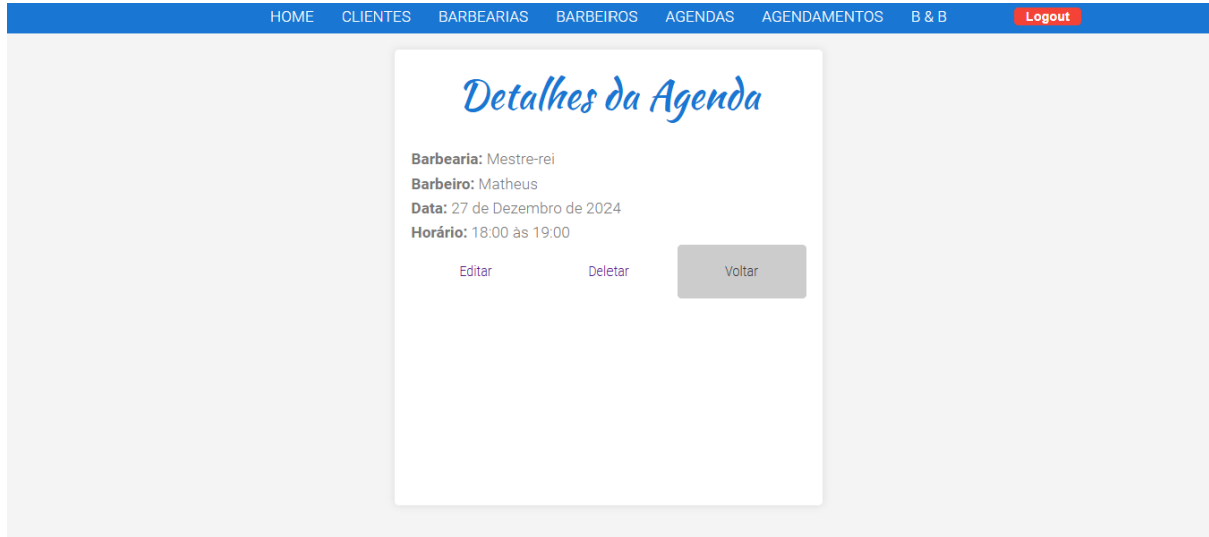
Fonte: arquivo pessoal.

#### **Cenário de uso 4 para clientes: Cadastrar novo agendamento**

<b>Cenário:</b>	Cadastrar novo agendamento
<b>Usuário:</b>	Cliente
<b>Objetivo:</b>	Agendar horário com o barbeiro
<b>Entrada:</b>	Clicar em novo agendamento
<b>Ação:</b>	Clicar em novo agendamento Clicar em barbeiro e horário disponível
<b>Saída:</b>	Horário agendado

Nesta tela (figura 23), o cliente poderá verificar os detalhes referentes ao seu agendamento, bem como editar ou fazer o cancelamento para reagendamento futuro.

**Figura 23 – Tela Detalhes do Agendamento**



Fonte: arquivo pessoal.

### **Cenário de uso 5 para clientes: Editar ou Cancelar o agendamento**

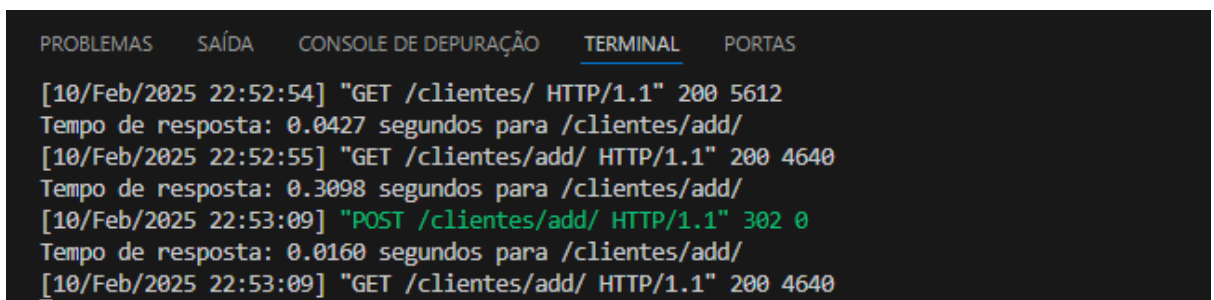
<b>Cenário:</b>	Detalhes do Agendamento
<b>Usuário:</b>	Cliente
<b>Objetivo:</b>	Verificar detalhes do agendamento; Editar ou cancelar agendamento.
<b>Entrada:</b>	Clicar no seu agendamento
<b>Ação:</b>	Clicar no agendamento desejado na lista de agendamento Verificar detalhes do agendamento Editar ou deletar agendamento
<b>Saída:</b>	agendamento editado ou deletado

### **5.4 - Testes e Validação**

Após a implantação, o desempenho do sistema foi monitorado para garantir que ele estivesse funcionando conforme o esperado. Foram analisados indicadores como o tempo de resposta, a taxa de erros e a satisfação dos usuários. Com base nesses dados, foi possível avaliar a eficácia do sistema e possíveis áreas de melhoria para futuras versões.

Para avaliar o tempo de resposta das requisições HTTP, implementamos um *middleware de logging* no Django, que registra o tempo de execução de cada requisição. Segundo Jakob Nielsen, especialista em usabilidade, a velocidade de resposta de um sistema é essencial para a experiência do usuário. De acordo com suas diretrizes:

- **Até 0,1 segundo:** O sistema é percebido como instantâneo e altamente responsivo.
- **Entre 0,1 e 1 segundo:** O usuário sente que a interação é rápida e eficiente.
- **De 1 a 10 segundos:** O tempo de espera se torna perceptível e pode gerar leve frustração.
- **Acima de 10 segundos:** A demora excessiva pode causar insatisfação e levar o usuário a abandonar a interação.



```
PROBLEMAS  SAÍDA  CONSOLE DE DEPURAÇÃO  TERMINAL  PORTAS
[10/Feb/2025 22:52:54] "GET /clientes/ HTTP/1.1" 200 5612
Tempo de resposta: 0.0427 segundos para /clientes/add/
[10/Feb/2025 22:52:55] "GET /clientes/add/ HTTP/1.1" 200 4640
Tempo de resposta: 0.3098 segundos para /clientes/add/
[10/Feb/2025 22:53:09] "POST /clientes/add/ HTTP/1.1" 302 0
Tempo de resposta: 0.0160 segundos para /clientes/add/
[10/Feb/2025 22:53:09] "GET /clientes/add/ HTTP/1.1" 200 4640
```

Conforme demonstrado nos testes, o sistema desenvolvido responde em menos de 1 segundo, garantindo um ótimo desempenho e uma experiência rápida para o usuário

Além do tempo de resposta, também foi avaliada a taxa de erro do sistema, com base nos códigos de status HTTP retornados durante os testes. Esses erros foram identificados diretamente no terminal, onde o Django exibe mensagens detalhadas sobre falhas internas. O cálculo da taxa de erro foi realizado dividindo-se o número total de respostas com erro pelo total de requisições processadas.

Com esses resultados, foi possível concluir que o sistema atende aos requisitos de desempenho e estabilidade, proporcionando uma ótima experiência aos usuários e servindo como uma base sólida para futuras melhorias.

Durante as fases de teste foram realizados ainda testes unitários e testes de integração do sistema foram feitos testes unitários em todas as funcionalidades críticas do sistema a fim de garantir que cada parte estivesse funcionando conforme o esperado. Isso incluiu testes no sistema de agendamento.

Após a conclusão dos testes unitários, foram conduzidos testes de integração para verificar se as diferentes partes do sistema funcionam corretamente em conjunto. Os testes de integração focaram principalmente na interação entre o frontend e o backend, bem como na comunicação com o banco de dados.

Além disso, foi requisitado do cliente o feedback sobre o sistema na sua finalização. Para tal foi utilizado uma ferramenta de pesquisa de satisfação, geralmente utilizada nas áreas tecnológicas, baseadas principalmente em práticas de User Experience (UX) e Customer Feedback Analysis. Segue abaixo os resultados a partir de perguntas destinadas ao cliente:

**Tabela 1 – Pesquisa de satisfação do cliente**

<b>Tabela de Pesquisa de Feedback do Cliente</b>						
<b>ID</b>	<b>Aspecto Avaliado</b>	<b>Pergunta</b>	<b>Resposta do Cliente</b>	<b>*Pontuação (1-5)</b>	<b>Sugestão do Cliente</b>	<b>Ações Recomendadas</b>
1	Design	O design do site é agradável e moderno?	Sim, está de acordo com os padrões atuais.	4	Colocar o nome da barbearia	Melhorar a interface do sistema.
2	Facilidade de uso	Foi fácil encontrar o que você procurava?	Sim, está simples e fácil de mexer	5	Nenhuma	Nenhuma.
3	Velocidade	O site carregou rapidamente para você?	Sim.	5	Nenhuma	Manter o desempenho atual.
4	Conteúdo	O conteúdo do site foi claro e informativo?	Sim, mas falta mais detalhes	4	Colocar os preços dos cortes.	Acrescentar mais informações relacionadas ao serviço ofertado
5	Funcionalidades	As funcionalidades atenderam às suas expectativas?	Sim, ocorreu tudo bem.	5	Nenhuma.	Manter como está.

6	Acessibilidade	O site foi acessível para você?	Sim.	5	Nenhuma.	Manter como está.
7	Satisfação Geral	Qual foi sua experiência geral com o site?	Satisfatória.	4	Para o futuro, acrescentar mais informações de serviço.	Enriquecer o site com mais funcionalidades.

\*Pontuação: Escala de 1 a 5 para avaliar o nível de satisfação (1 = muito insatisfeito, 5 = muito satisfeito).

A tabela apresentada foi elaborada com base nas diretrizes de usabilidade definidas pela ISO 9241-11 (ISO, 2018) e nos princípios heurísticos de usabilidade descritos por Jakob Nielsen (NIELSEN, 1994). Essas abordagens são amplamente utilizadas para avaliar a experiência do usuário e coletar feedbacks sobre sistemas interativos.

De acordo com Jakob Nielsen (1994), a usabilidade de um sistema pode ser avaliada a partir de critérios como eficiência, eficácia e satisfação do usuário. A tabela apresentada neste trabalho segue esses princípios e foi estruturada para facilitar a análise do feedback dos usuários sobre a plataforma estudada.

Esta tabela (tabela 1) apresenta uma síntese de perguntas feitas ao cliente, a fim de apresentar de maneira sucinta e direta parte do processo satisfatório ou não do sistema aqui apresentado. Para fins de análise inicial, é possível concluir que o sistema foi bem avaliado pelo cliente, demonstrando bom desempenho. A queixa esteve em acrescentar algumas informações de serviço, o que pode ser levado a novas agendas de pesquisa futuras para acréscimo e melhoria do sistema.

## 6. AVALIAÇÃO E CONSIDERAÇÕES FINAIS

Diante da exposição acima, conclui-se que um sistema web pode auxiliar de forma efetiva na gestão de agendas empresariais e no aprimoramento do atendimento ao cliente. O objetivo geral deste trabalho foi desenvolver um sistema de agendamento eficiente para uma barbearia, capaz de simplificar o processo de marcação de horários e proporcionar uma experiência intuitiva e acessível tanto para os clientes quanto para os profissionais, tornando o gerenciamento mais prático e eficaz.

Para alcançar esse objetivo, foram analisadas as demandas específicas do cliente e as funcionalidades que o sistema poderia oferecer. A partir desse diagnóstico, identificaram-se recursos que poderiam facilitar a rotina da barbearia, como agendamento em tempo real e uma administração simplificada da plataforma, incluindo dados precisos sobre os agendamentos e a equipe.

Durante a implementação, constatou-se que o sistema funcionava conforme o planejado e de acordo com as expectativas apresentadas ao cliente, evidenciando a eficácia das ferramentas e linguagens empregadas. Mesmo diante da necessidade de migrar para o Python no decorrer do processo, foi possível manter o escopo inicial e entregar um sistema completo, com todas as funcionalidades previstas.

A metodologia aplicada foi revisada para identificar os pontos fortes e as dificuldades enfrentadas durante o desenvolvimento do projeto. Essa reflexão será útil para futuros projetos, tanto em termos de desenvolvimento técnico quanto de gestão de projetos.

Alguns pontos fortes foram identificados e contribuíram para o sucesso do projeto, bem como algumas dificuldades que precisaram ser superadas.

Pontos Fortes:

- O sistema foi desenvolvido com base em um levantamento detalhado de requisitos, incluindo uma entrevista inicial e interações informais regulares com o proprietário da barbearia. Isso garantiu que o sistema atendesse às expectativas do cliente e resolvesse os problemas reais de agendamento.
- A escolha do framework Django, em conjunto com Python, HTML, CSS e SQLite3, permitiu o desenvolvimento de um sistema robusto, seguro e de fácil

manutenção. A arquitetura MVC (Model-View-Controller) também facilitou a organização do código e a escalabilidade do sistema.

- O sistema foi projetado com foco na usabilidade, tanto para os clientes quanto para os barbeiros. A interface é simples e fácil de usar, o que foi validado por meio de testes e feedbacks do proprietário.
- As interações informais com o proprietário durante os cortes de cabelo permitiram ajustes iterativos no sistema, garantindo que ele estivesse sempre alinhado com as necessidades da barbearia.

#### Dificuldades Encontradas:

- Inicialmente, o sistema foi planejado para ser desenvolvido em Java, mas, devido a dificuldades técnicas, optou-se por migrar para Python e Django. Essa mudança exigiu um replanejamento do projeto, mas acabou sendo benéfica devido à facilidade de desenvolvimento oferecida pelo Django.
- A fase de testes exigiu um esforço significativo para garantir que todas as funcionalidades estivessem funcionando corretamente. A realização de testes unitários e de integração foi essencial para identificar e corrigir bugs antes da implantação do sistema.

Quanto aos trabalhos futuros, o sistema desenvolvido oferece uma base sólida para aprimoramentos e para a inclusão de novas funcionalidades, o que permitirá expandir ainda mais suas capacidades e atender às crescentes demandas da empresa. Entre as possíveis melhorias, destacam-se: especificação do serviço agendado, maior privacidade no acesso às diferentes áreas do sistema, separando o que é visível apenas para o cliente e o que é exclusivo para o barbeiro, e a criptografia da senha de acesso.

## 7. REFERÊNCIAS BIBLIOGRÁFICAS

ANGELOTTI, Elaine Simoni. Banco de dados. Curitiba: Editora do Livro Técnico, 2010.

ÁVILA, Daiana Paula de. Sistema Gerenciador para um salão de beleza. Trabalho de Conclusão de Curso. Universidade do Contestado, Concórdia, 2013.

CLARO D. B. e SOBRAL J. B. M. Programação em Java. Copyleft Pearson Education. Florianópolis, SC, 2008.

Desenvolvimento de Software II: Introdução ao desenvolvimento web com HTML, CSS, JavaScript e PHP. Organizadores, Evandro Manara Miletto, Silvia de Castro Bertagnolli – Porto Alegre: Bookman, 2014.

GIL, Antonio Carlos. Como elaborar projetos de pesquisa. 6 ed. São Paulo: Atlas, 2019.

GUEDES, Gilleanes T. A. UML 2 : uma abordagem prática / Gilleanes T. A. Guedes. -- 2. ed. -- São Paulo : Novatec Editora, 2011.

KIERAS, Ramon Wolski. Sistema para agendamento de serviços. Trabalho de Conclusão de Curso. Ponta Grossa, 2019.

KOHN, Karen; MORAES, Cláudia Herte de. O impacto das novas tecnologias na sociedade: conceitos e características da Sociedade da Informação e da Sociedade Digital. Santos. Intercom, 2007.

MARTINS, Elba Franciérika Rocha; FERNANDES, Taciliano Augusto. Two Style: um software de agendamento e gerenciamento. Trabalho de Conclusão de Curso. Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte. Lajes, 2019.

NASCIMENTO, Ewerson Vieira. Desenvolvimento de um sistema agregador de barbearias / Ewerson Vieira Nascimento. - 2022.

NIELSEN, Jakob. Usability Engineering. San Francisco: Morgan Kaufmann, 1994.

PAULA, Hygor Martins de; ALMEIDA, Luiz Fernando Ferreira de; GORGES, Vânia de Oliveira; FERNANDES, Carlos Henrique; SANJULIÃO, Lo-ruana K. A. Freire. Desenvolvimento de um sistema web para gerenciamento de trabalho de conclusão de curso. X Congresso Brasileiro de Engenharia de Produção, 2020.

PINHEIRO, Rafael. Vue.JS: tudo sobre o framework para trabalhar com mídias interativas. 2021. Disponível em: <https://rockcontent.com/br/talent-blog/vue-js/#:~:text=js%3F-.O%20Vue.,e%20proporcionar%20o%20desenvolvimento%20%C3%A1gil>. Acessado em: 15/01/2023.

PRESSMAN, Roger S. Engenharia de Software. Tradução: José Carlos Barbosa dos Santos; Revisão Técnica: José Carlos Maldonado, Paulo Cesar Masiero, Rosely Sanches. São Paulo, Makron Books, 1995.

SANTOS, Marilda Corrêa Leite dos; CAMARGO, Ricardo de Arruda; SARTI, Maria Regina Catarino; NOVELLI, Valéria Aparecida Moreira , “Web sites o espaço virtual para a organização da informação nas bibliotecas.” *Repositório – FEBAB, 2004.*

SOMMERVILLE, Ian. Engenharia de Software. tradução Ivan Bosnic e Kalinka G. de O. Gonçalves ; revisão técnica Kechi Hiramã. — 9. ed. — São Paulo : Pearson Prentice Hall, 2011.

YOSHIURA, Vinicius Tohoru. Desenvolvimento e Implantação de um Sistema Web para Monitoramento da Rede de Atenção em Saúde Mental / Vinicius Tohoru Yoshiura; orientador Domingos Alves. São Carlos, 2015

---

## Sites

[1] <https://www.ateomomento.com.br/o-que-e-caso-de-uso/>

[2] Fonte: Desenvolvimento de software II: introdução ao desenvolvimento web com HTML, CSS, JavaScript e PHP/ Organizadores: Evandro Manara Miletto, Silvia de Castro Bertagnolli – Porto Alegre: Bookman, 2014.

[3] Pinheiro, Rafael. Vue.JS: tudo sobre o framework para trabalhar com mídias interativas. 2021. Disponível em: <https://rockcontent.com.br/talent-blog/vue-js/#:~:text=js%3F-O%20Vue.,e%20proporcionar%20o%20desenvolvimento%20%C3%A1gil>. Acessado em: 15/01/2023.

[4] <https://natahouse.com/pt/saiba-as-diferencas-entre-java-e-javascript-de-uma-vez-por-todas>. Acessado em: 16/01/2023

[5] <https://www.devmedia.com.br/comparando-as-ides-netbeans-e-eclipse-revista-easy-java-magazine-22/25767>. Acessado em: 17/01/2023.

[6] <https://www.confrriadabarba.com.br/>

[7] Pillow. Disponível em: <https://pillow.readthedocs.io/en/stable/>. Acessado em: 14 de janeiro de 2025.

[8] Chart.js. Disponível em: <https://www.chartjs.org/docs/latest/>. Acesso em: 14 de janeiro de 2025.