



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

ALEX DOS SANTOS MOREIRA

GESTÃO DE CADASTRO E ATENDIMENTO AO CLIENTE

**Assis/SP
2024**



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

ALEX DOS SANTOS MOREIRA

GESTÃO DE CADASTRO E ATENDIMENTO AO CLIENTE

Projeto de desenvolvimento apresentado ao curso de Análise e Desenvolvimento de Sistemas do Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

Orientando(a): Alex Dos Santos Moreira

Orientador(a): Dr. Alex Sandro Romeo de Souza Poletto

**Assis/SP
2024**

Moreira, Alex dos Santos

M838g Gerenciamento de cadastro e atendimento ao cliente / Alex dos Santos Moreira. -- Assis, 2024.

32p.

Trabalho de Conclusão de Curso (Graduação em Análise e Desenvolvimento de Sistemas) -- Fundação Educacional do Município de Assis (FEMA), Instituto Municipal de Ensino Superior de Assis (IMESA), 2024.

Orientador: Prof. Dr. Alex Sandro Romeo de Souza Poletto.

1. Sistemas de gestão empresarial. 2. Sistemas de informação gerencial. I Poletto, Alex Sandro Romeo de Souza. II Título.

CDD 003

AGRADECIMENTO

Agradeço aos meus amigos de jornada da faculdade, que conheci durante este trajeto, porém irei levar toda a amizade, momentos e tudo que me agregaram para o resto de minha vida.

Aos meus pais que sempre me motivaram a continuar os estudos, e nunca desistir.

Também quero agradecer ao meu orientador Alex Sandro Romeo de Souza Poletto que, sempre se mostrou disponível, atendendo aos meus chamados realizando os auxílios com grande precisão e agilidade.

E por fim, sempre serei grato as pessoas mais importantes da minha vida, com seu zelo, suporte e auxílio constante.

LISTA DE ILUSTRAÇÕES

Figura 1 – Exemplo de aplicações para Java	16
Figura 2 – Exemplo de criação de um componente em React	17
Figura 3 – Diagrama de Caso de Uso - Administrador	19
Figura 4 – Diagrama de Caso de Uso - Sistema	20
Figura 5 – Diagrama de Caso de Uso Geral	20
Figura 6 – Diagrama de Caso de Uso - Funcionário	21
Figura 7 – Modelo de Entidade-Relacionamento	21
Figura 8 – Diagrama de Classe	22
Figura 9 – Painel do Administrador	24
Figura 10 – Tela para Login	24
Figura 11 -Tela para registro de cliente	25
Figura 12 -Tela para registro de empresa	25
Figura 13 -Tela para registro de funcionário	26
Figura 14 – Autenticação de Usuário	27

SUMÁRIO

1. INTRODUÇÃO	10
1.1 OBJETIVOS	11
1.2 OBJETIVO GERAL	11
1.3 OBJETIVOS ESPECÍFICOS	11
1.4 PÚBLICO ALVO	11
1.5 JUSTIFICATIVAS	12
1.6 ESTRUTURA DO TRABALHO	12
2. MÉTODO DE DESENVOLVIMENTO	14
2.2 JAVA	14
2.3 SPRING FRAMEWORK	15
2.4 REACT	16
2.5 TAILWIND	16
2.6 BYCRYPT PASSWORD ENCODER	17
2.7 INSOMNIA	17
2.7 POSTGRE SQL	17
3. ANÁLISE E ESPECIFICAÇÃO DO SISTEMA	18
3.1 LEVANTAMENTO DE REQUISITOS	18
3.2 DIAGRAMA DE CASOS DE USO GERAL	18
3.5 DIAGRAMA DE CASOS DE USO – FUNCIONÁRIO	20
3.6 DIAGRAMA DE ENTIDADE-RELACIONAMENTO	20
3.7 DIAGRAMA DE CLASSES	21
4. DESENVOLVIMENTO DO SISTEMA	22
4.1 FUNCIONAMENTO	22
4.2 PAINEL DO ADMINISTRADOR	22
4.3 TELA PARA REALIZAR O LOGIN	23
4.4 TELA PARA REALIZAR O REGISTRO DE CLIENTE	23
4.5 TELA PARA REALIZAR O REGISTRO DE UMA EMPRESA	24
4.6 TELA PARA REALIZAR O REGISTRO DE UM FUNCIONÁRIO	25
4.7 AUTENTICAÇÃO DE USUÁRIO POR MEIO DE LOGIN	25
5. CONCLUSÕES	27
5.1 TRABALHOS FUTUROS	27

RESUMO

Neste trabalho, apresenta-se um software inovador desenvolvido para potencializar e agilizar o atendimento ao cliente. A proposta visa oferecer soluções eficazes para o controle de atendimentos e cadastros, com foco na usabilidade e eficiência.

O sistema foi projetado com uma interface intuitiva, permitindo que os colaboradores realizem o cadastro de clientes e empresas de forma rápida e prática. Essa agilidade no processo não só melhora a experiência do usuário, mas também contribui para um atendimento de qualidade superior.

Ao longo deste trabalho, serão discutidas as funcionalidades do software, sua importância para a gestão de atendimentos e como ele pode transformar a interação entre empresas e clientes.

Palavras chave: *web*, atendimento, cliente.

ABSTRACT

In this work, we present an innovative software developed to enhance and streamline customer service. The proposal aims to provide effective solutions for managing service requests and registrations, focusing on usability and efficiency.

The system is designed with an intuitive interface, allowing employees to quickly and easily register clients and companies. This agility in the process not only improves the user experience but also contributes to superior quality service.

Throughout this work, we will discuss the software's functionalities, its importance for service management, and how it can transform the interaction between businesses and customers.

Key words: *web*, customer service, customer.

1. INTRODUÇÃO

O atendimento ao cliente é uma parte essencial de uma empresa, sendo um aspecto crítico e almejado que é a chave para o sucesso e o desenvolvimento e na construção de um bom relacionamento entre empresa e cliente. Empresas de todos os portes e todos os setores procuram uma maneira de atender de uma forma ágil e personalizada cada um de seus clientes, e é aí aonde o Logic Fly entra.

Logic Fly, é uma aplicação Web que soluciona os conflitos entre este relacionamento entre empresa e cliente, trazendo uma interface intuitiva e amigável para utilização favorecendo ambos os lados na hora de atender toda sua clientela.

Ao oferecer uma abordagem integrada, o Logic Fly visa não apenas simplificar a gestão de clientes, mas também aprimorar o atendimento, possibilitando uma compreensão mais profunda das preferências e necessidades individuais de cada cliente.

Em resumo, o Logic Fly se destaca como uma solução abrangente, visando centralizar informações, automatizar processos e aprimorar o relacionamento com o cliente, proporcionando um ambiente mais eficiente e orientado ao cliente para os estabelecimentos de qualquer setor.

Um exemplo de software focado no atendimento ao cliente, Zendesk(2007), é uma opção abrangente e moderna a para gestão de clientes. Ele oferece recursos como controle de cadastros, CRM, Telefonia, Calendários, Projetos, chats para atendimento, Integrações em diversos aplicativos e redes sociais e muito mais. Essa interface intuitiva, de fácil compreensão e acesso garantem o aumento do desempenho e fluxo de trabalho das empresas melhorando também principalmente o atendimento.

Logic Fly, traz a ideia de modernizar, agilizar e facilitar o processo de gestão de clientes no dia a dia de uma empresa que antes era precarizada por filas demoradas, imensidões de pastas e papéis que aumentavam ainda mais a lentidão na rotina. Agora com poucos cliques, é possível armazenar os dados de clientes e prestar o atendimento de forma rápida e dinâmica e para Paladini (1994, p. 16), ele afirma que “qualidade corretamente definida, é aquela que prioriza o consumidor”.

Com base no trabalho de AGUSTI(2022), o atendimento ao cliente não é um produto, mas sim um serviço. No entanto, um atendimento deficiente pode prejudicar a percepção que o cliente possui do produto, no caso, o software fornecido pela empresa, e a imagem desta como um todo.

Segundo EXAME(2012), as 4 coisas que os clientes mais esperam de você como atendente são: respeito, agilidade, criatividade e credibilidade.

1.1 OBJETIVOS

1.2 OBJETIVO GERAL

O objetivo principal deste presente trabalho é criar um software de gerenciamento para empresas que utilizam cadastros e atendimento ao cliente, tanto de forma presencial ou remota.

1.3 OBJETIVOS ESPECÍFICOS

Como objetivo específico, as seguintes funcionalidades responsáveis pela construção do aplicativo são:

- Utilizar um software Web, que será acessado pelo navegador do funcionário;
- Limite de acesso: o software terá acesso limitado para estagiários e funcionários terão o acesso completo a plataforma;
- Relatório para o funcionário: o funcionário terá acesso aos dados do atendimento, e do cliente;
- Cadastro Único: o cliente será cadastrado no sistema uma única vez, apenas tendo alterações posteriores caso necessário;

1.4 PÚBLICO- ALVO

O software de cadastro de clientes e gestão de atendimento ao cliente tem como público-alvo uma ampla gama de empresas de diversos setores e tamanhos. Desde pequenos empreendimentos até grandes corporações, todos aqueles que buscam aprimorar o relacionamento com seus clientes encontram nessa ferramenta uma solução valiosa.

Profissionais de vendas, atendimento ao cliente e gestão de relacionamento com clientes são os principais usuários desse software, aproveitando suas funcionalidades para acompanhar o ciclo de vida do cliente, fornecer suporte eficiente e personalizado. Empresas centradas no

cliente, que valorizam a satisfação e fidelização de seus clientes, bem como a eficiência operacional, são o público ideal para essa solução.

Ao investir nesse software, as empresas estão priorizando a excelência no atendimento, a automação de processos e a capacidade de tomar decisões estratégicas baseadas em dados sólidos, fortalecendo assim sua posição competitiva no mercado.

1.5 JUSTIFICATIVAS

Com a evolução da tecnologia, os estabelecimentos do ramo comercial de gestão empresarial e de clientes estão se adaptando à era moderna. A pesquisa da Economist Intelligence Unit(1946) revela que respostas imediatas são cruciais para uma experiência de atendimento de qualidade, com cerca de 50% dos consumidores considerando essa prontidão como um fator chave.

Além disso, segundo a empresa Dimension Data(1983) indica que 75% dos gestores reconhecem o serviço de atendimento como uma estratégia competitiva crucial, com a previsão de um aumento para mais de 90% nos próximos dois anos.

Ambas as pesquisas mostram um crescimento significativo na percepção da importância do atendimento ao cliente. Isso impulsiona a busca por soluções que automatizem processos, centralizem informações e proporcionem uma visão mais abrangente do atendimento.

E nesta solução defendida por meio deste trabalho, uma das principais soluções atendidas

é a agilidade no atendimento, excluindo a necessidade do cliente de repetir processos que são automatizáveis. Deixando assim, a fila de espera menor e o atendimento de qualidade excepcional.

Em resumo, as pesquisas evidenciam a relevância crescente da prontidão no atendimento e a importância da integração com softwares e plataformas SaaS para otimizar processos, reduzir custos operacionais e oferecer uma experiência de atendimento mais eficiente e satisfatória aos clientes.

O presente trabalho está dividido e estruturado na seguinte forma:

Capítulo 1 - Introdução

Capítulo 2 - Tecnologias e Ferramentas de Desenvolvimento

Capítulo 3 - Análise e Especificação do Sistema

Capítulo 4 - Desenvolvimento do Sistema

Capítulo 5 - Conclusão

Referências

2. MÉTODO DE DESENVOLVIMENTO

A metodologia empregada neste projeto será executada em etapas distintas:

Primeiramente, foi feita uma entrevista com o cliente para identificar quais os requisitos necessários para iniciar o mapeamento do sistema. Após essa entrevista inicial, foi decidido quais tecnologias serão utilizadas na etapa de desenvolvimento do sistema.

As tecnologias escolhidas foram: para o Front-end, ReactJs para a interface do usuário, onde entrará o TypeScript para adicionar a tipagem ao código. No Back-end, foi escolhido o Java com SpringBoot para a lógica e implementação.

Para o banco de dados, foi escolhido o PostgreSQL, como o sistema de gerenciador do banco de dados.

Após a finalização das escolhas das tecnologias, foi feito um protótipo no Figma, escolhendo as imagens, paleta de cor, tipografia que serão utilizadas no projeto. Novamente, foi feita outra entrevista com o cliente para avaliação e feedback, ajustando o protótipo até o cliente demonstrar satisfeito com o resultado.

Finalizada esta etapa, foi feito o desenvolvimento do sistema com as tecnologias escolhidas, implementando o SGBD até a execução em produção.

2.2 JAVA

A linguagem de programação Java foi escolhida para este trabalho devido às suas inúmeras qualidades. Como uma linguagem de alto nível, Java se assemelha à linguagem humana, facilitando a compreensão e o desenvolvimento.

Entre suas características destacam-se a grande estabilidade, robustez e segurança, além de sua natureza portátil, que permite a execução em diferentes plataformas sem a necessidade de adaptações significativas.

Java é uma escolha versátil, adequada para projetos de empresas de todos os tamanhos, desde pequenas startups até grandes corporações. Essa flexibilidade possibilita a aplicação em diversos setores do mercado, o que, por sua vez, contribui para uma alta empregabilidade dos profissionais que dominam essa linguagem

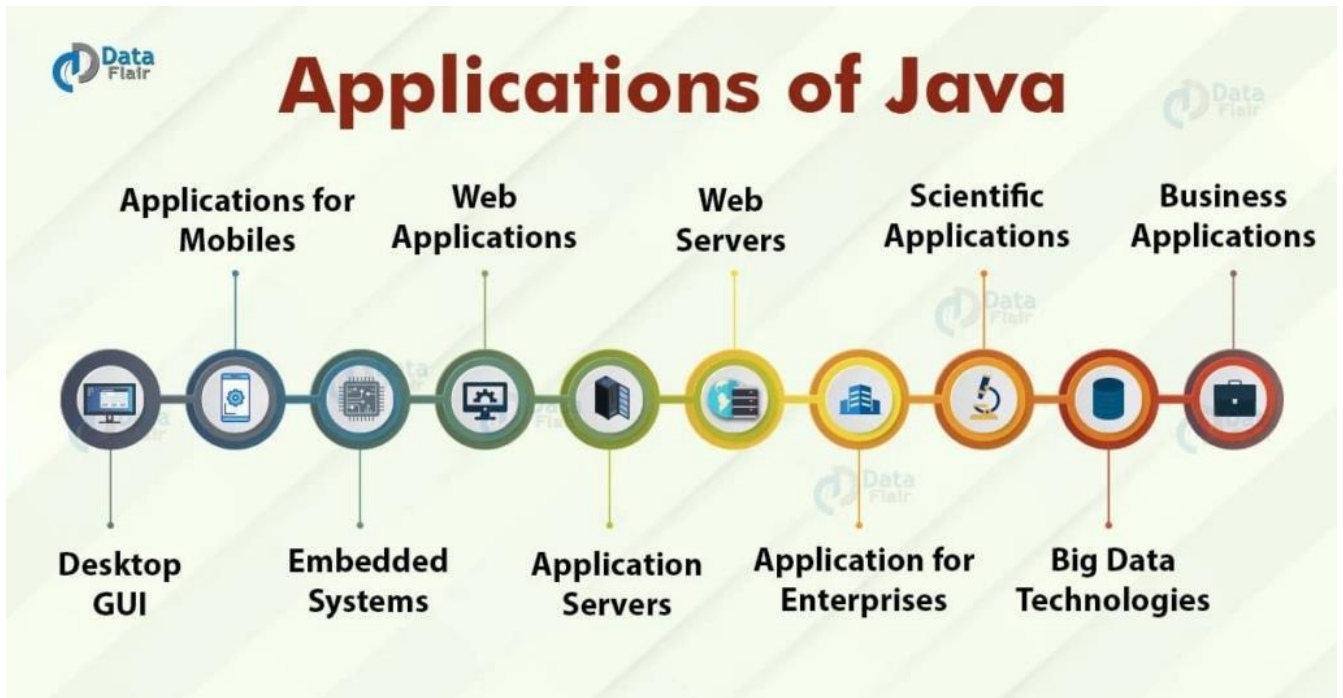


Figura 1 – Exemplo de aplicações para Java

2.3 SPRING FRAMEWORK

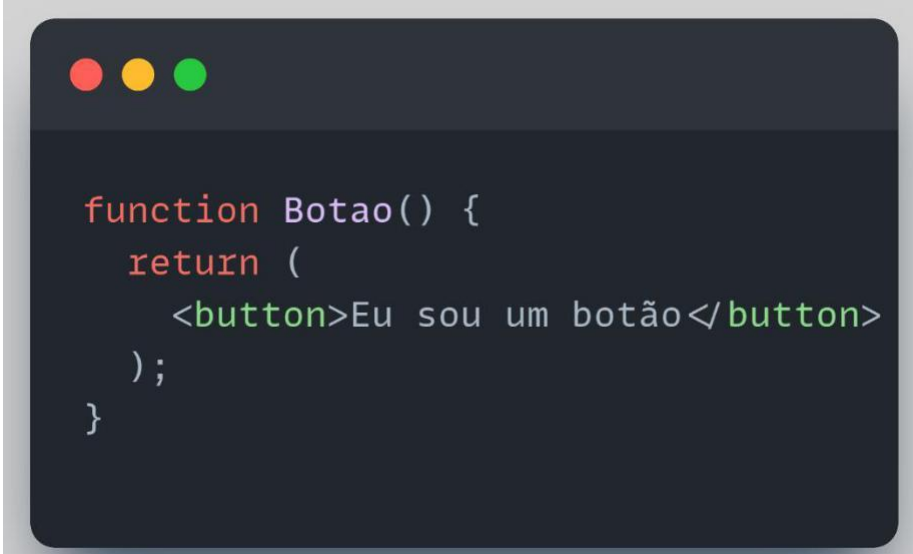
O Spring Framework, desenvolvido em Java, foi escolhido para este trabalho devido à sua capacidade de acelerar o desenvolvimento de aplicações web, oferecendo eficácia e versatilidade no código. Com uma ampla gama de bibliotecas e recursos disponíveis, o Spring se destaca como uma opção ideal para o desenvolvimento de aplicações web, garantindo confiabilidade e credibilidade.

Além disso, o Spring Framework é amplamente suportado e constantemente atualizado, o que assegura estabilidade e segurança ao longo do tempo. Sua habilidade em trabalhar com microsserviços e a integração com outras tecnologias open-source tornam-no uma solução versátil que atende às demandas modernas de desenvolvimento. Assim, utilizar o Spring Framework não apenas acelera o processo de desenvolvimento, mas também proporciona uma base sólida e confiável para a construção de aplicações escaláveis e de fácil manutenção.

2.4 REACT

React é uma biblioteca de JavaScript criada pelo Facebook (Meta) para desenvolver interfaces de usuários em aplicativos web, que foi aderida neste presente trabalho. Sua principal característica é a abordagem baseada em componentes, que permite construir interfaces de forma modular e reutilizável. O uso do Virtual DOM otimiza as atualizações da interface, melhorando o desempenho. O fluxo de dados unidirecional facilita o gerenciamento do estado da aplicação, tornando-o mais previsível.

Com um ecossistema rico em bibliotecas e ferramentas, como Redux e Next.js, React oferece uma ampla gama de recursos. A comunidade ativa proporciona suporte e recursos variados, e a curva de aprendizado é acessível para quem já conhece JavaScript. Além disso, aplicações React podem ser otimizadas para SEO, aumentando sua visibilidade em motores de busca. Em resumo, React se destaca como uma ferramenta eficaz para desenvolver aplicações web escaláveis e de alta performance.

A imagem mostra um editor de código com um fundo escuro e uma barra de título com três botões (vermelho, amarelo, verde). O código exibido é uma função JavaScript chamada 'Botao' que retorna um elemento HTML de botão.

```
function Botao() {  
  return (  
    <button>Eu sou um botão</button>  
  );  
}
```

Figura 2 – Exemplo de criação de um componente em React

2.5 TAILWIND

O Tailwind CSS é um framework CSS de código aberto que se destaca por sua abordagem inovadora e flexível de design utilitário. Ao invés de oferecer estilos pré-definidos para elementos específicos, o Tailwind disponibiliza uma vasta gama de classes utilitárias que podem ser aplicadas diretamente aos elementos HTML. Essas classes representam propriedades CSS individuais, como margens, preenchimentos, cores, tamanhos de fonte, alinhamentos e muito mais, permitindo que os desenvolvedores criem interfaces personalizadas de forma rápida e eficiente.

2.6 BYCRYPT PASSWORD ENCODER

Uma tecnologia presente no ecossistema Spring é comumente utilizada para mascarar e verificar senhas em aplicações web. Desenvolvida para ser segura e adaptativa, essa tecnologia emprega algoritmos robustos que ajudam a proteger contra tentativas de ataques de força bruta e quebras de senha. Seu design visa garantir a segurança das informações dos usuários, tornando-a uma solução eficaz para a proteção de dados sensíveis.

2.7 INSOMNIA

O Insomnia é uma ferramenta popular para desenvolvimento de APIs, que facilita a criação, teste e documentação de requisições HTTP. Com uma interface intuitiva, permite organizar requisições em projetos, suportando autenticação como OAuth 2.0 e Basic Auth. A ferramenta possibilita a importação e exportação de configurações, além de gerar documentação automática das APIs. Também oferece suporte a plugins, permitindo extensões conforme a necessidade. Em resumo, o Insomnia é uma solução eficiente para desenvolvedores que buscam otimizar o trabalho com APIs.

2.7 POSTGRE SQL

O PostgreSQL é um sistema de gerenciamento de banco de dados objeto-relacional de código aberto, conhecido por sua robustez, extensibilidade e suporte a padrões SQL. Ele oferece uma variedade de recursos avançados, como transações ACID, suporte a JSON e XML, e uma rica biblioteca de funções e operadores. O PostgreSQL é amplamente utilizado em aplicações web, sistemas de análise de dados e em ambientes corporativos devido à sua

confiabilidade e escalabilidade. Além disso, possui uma comunidade ativa que contribui para seu desenvolvimento contínuo e oferece suporte.

3. ANÁLISE E ESPECIFICAÇÃO DO SISTEMA

A análise realizada neste trabalho foi organizada em diferentes seções, incluindo diagramas de classe, diagramas de entidade-relacionamento (ER), diagramas de caso de uso e especificações do software. Essa estrutura permite uma compreensão clara e detalhada dos componentes e funcionalidades do sistema.

3.1 LEVANTAMENTO DE REQUISITOS

- Manter Clientes
- Manter Empresa
- Manter Funcionário
- Manter Estagiário
- Manter Usuários
- Manter funções
- Manter Atendimentos

Para entendimento e visualização das funcionalidades do trabalho, serão apresentados os diagramas de cada seção e fragmentos de pontos principais do software utilizando a ferramenta de apresentação de código Snappify. Na figura 3 pode-se ver o caso de uso geral do sistema como um todo.

3.2 DIAGRAMA DE CASOS DE USO GERAL

Este é o modelo principal dos casos de uso, onde mostra os atores que podem desempenhar funções distintas na aplicação, e também demonstrando seus fluxos onde representam ações que os dados atores podem desempenhar.

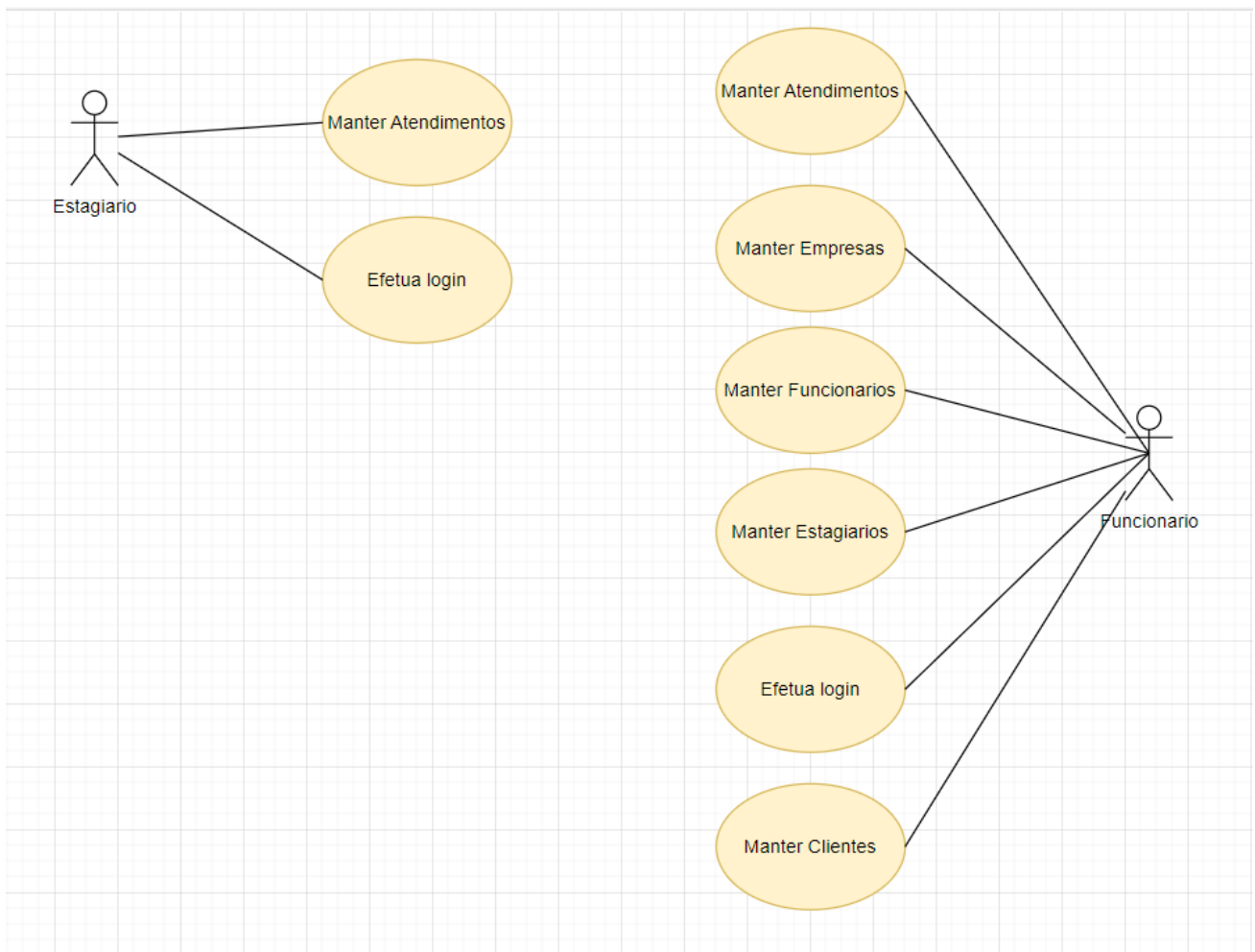


Figura 3 – Diagrama de Caso De Uso Geral

3.3 DIAGRAMA DE CASOS DE USO - FUNCIONÁRIO

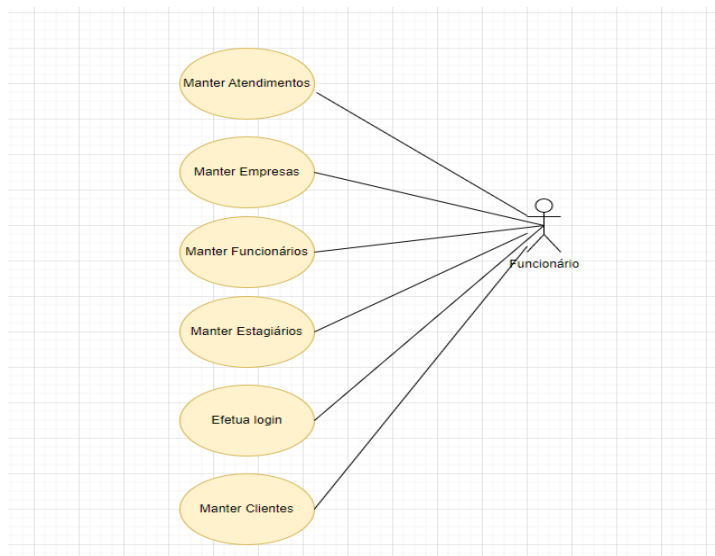


Figura 4 – Diagrama de Caso De Uso - Funcionário

3.6 DIAGRAMA DE ENTIDADE-RELACIONAMENTO

O diagrama de entidade relacionamento, é uma representação do sistema e organização do banco de dados da aplicação.

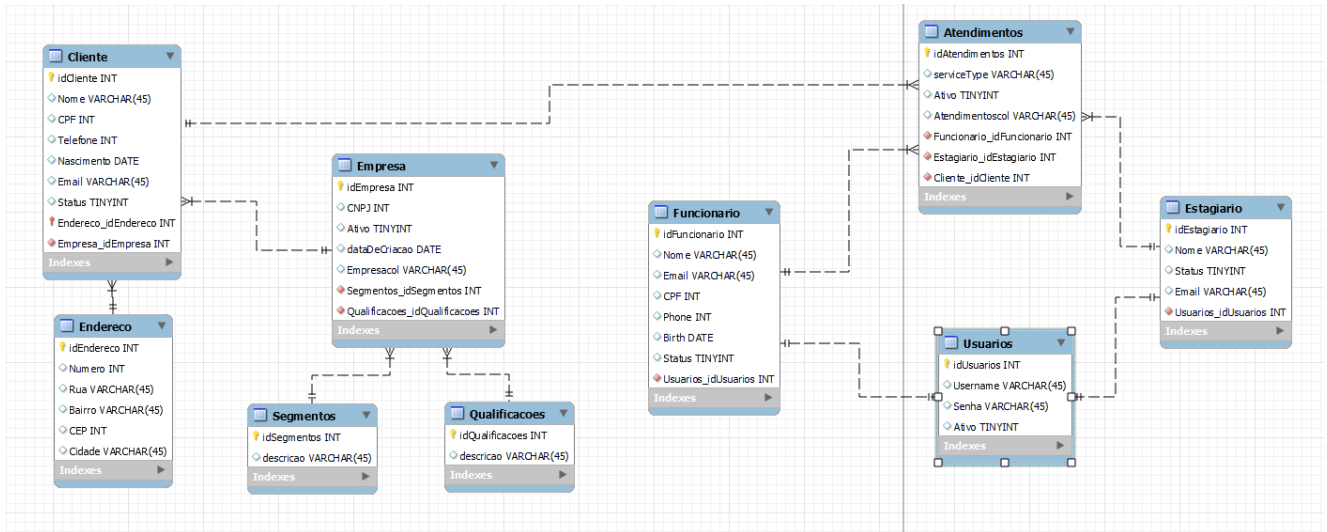


Figura 7 – Diagrama de Entidade Relacionamento

3.7 DIAGRAMA DE CLASSES

O diagrama de classes, tem a função de apresentar uma melhor visualização das classes, entidades, objetos e métodos desenvolvidos dentro da aplicação demonstrado na figura abaixo.

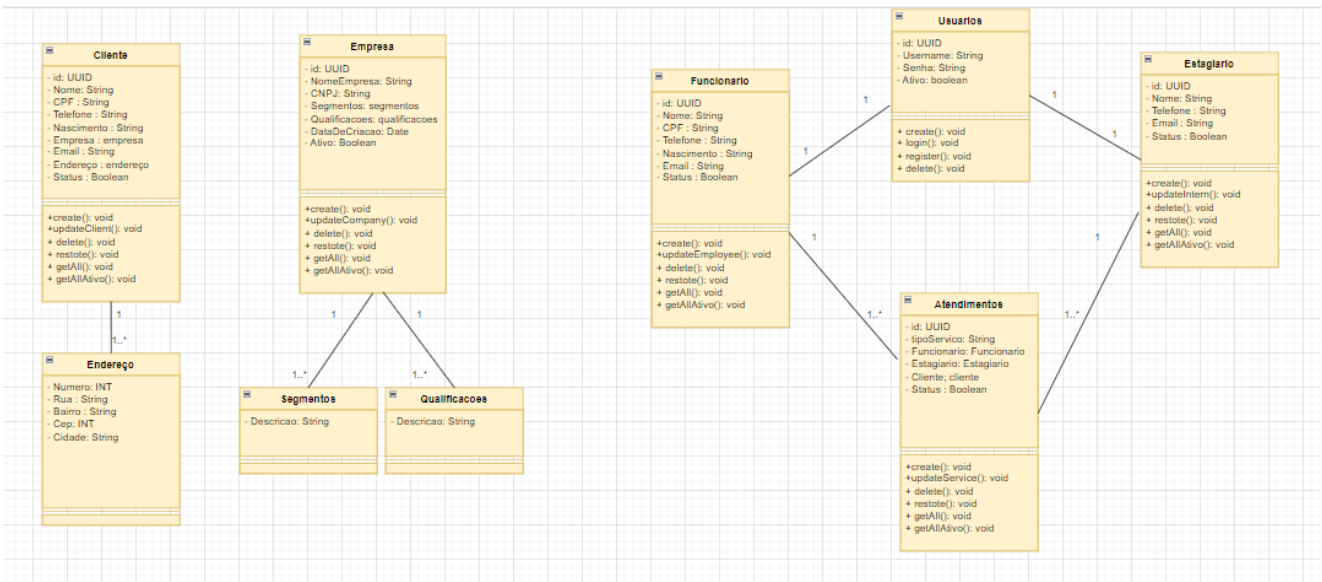


Figura 8 – Diagrama de classes

4. DESENVOLVIMENTO DO SISTEMA

4.1 FUNCIONAMENTO

Neste presente trabalho, foi desenvolvida uma API para ser integrada nas chamadas das camadas Front-End e Back-End da aplicação. A interface do front-end foi projetada para permitir que o usuário final realize chamadas à API através de métodos HTTP, como GET, POST, PUT e DELETE. Cada requisição é cuidadosamente validada mediante um processo de autenticação, que envolve o login do usuário e a verificação de um token de autenticação. Esse mecanismo garante a segurança e a integridade das operações realizadas na aplicação.

Os usuários com perfil de “Admin” ou “Funcionário” possuem um nível de acesso privilegiado à interface da aplicação. Esses usuários têm a capacidade de acessar todas as funcionalidades disponíveis, permitindo-lhes gerenciar dados e configurações sem restrições. Essa flexibilidade é essencial para a administração eficiente da aplicação.

Por outro lado, os usuários classificados como “Intern” ou estagiário têm um acesso mais restrito. Eles podem interagir apenas com rotas específicas da aplicação, conforme definido nas políticas de controle de acesso. Esse modelo de acesso limitado é fundamental para garantir que informações sensíveis sejam protegidas e que os estagiários possam operar dentro de um escopo definido, promovendo aprendizado e responsabilidade.

4.2 PAINEL DO ADMINISTRADOR

Ao realizar a autenticação por meio do login, o Admin ou Funcionário irá acessar este painel. Como mostra a Figura 9, ele terá acesso a todos os módulos de cadastro, detalhamento e atendimento da aplicação.

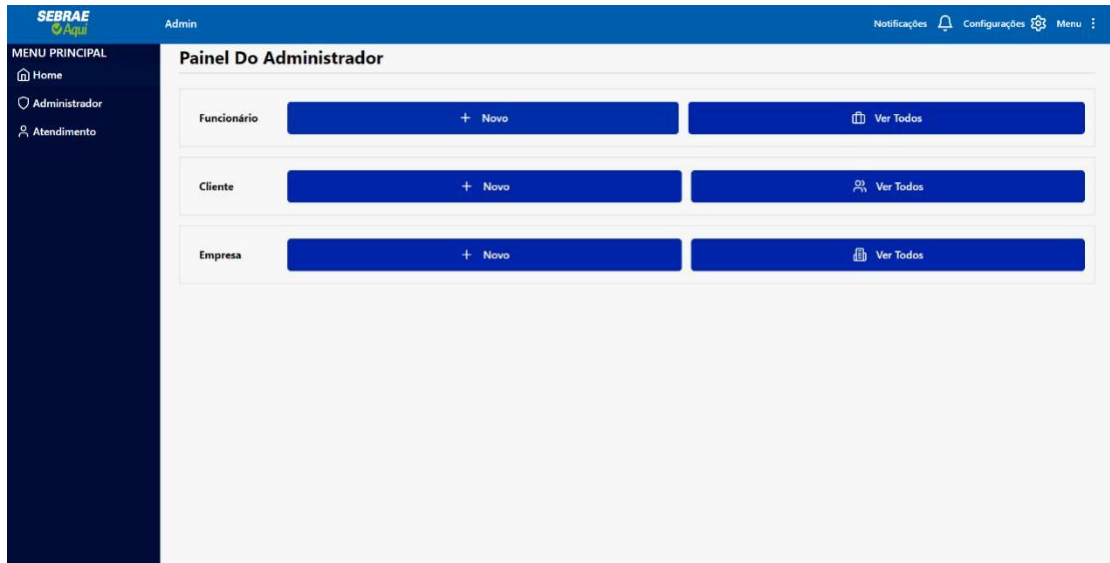


Figura 9 – Painel do administrador

4.3 TELA PARA REALIZAR O LOGIN

Na Figura 10, está a tela onde os usuários irão realizar o login e se autenticar baseado na *role* de seu usuário.

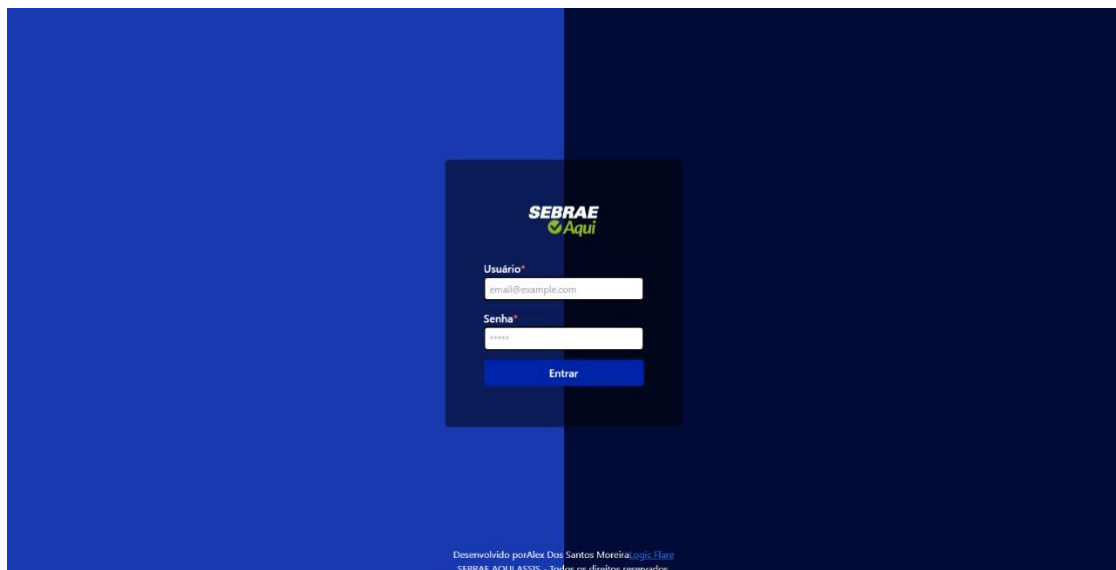


Figura 10 – Tela para Login

4.4 TELA PARA REALIZAR O REGISTRO DE CLIENTE

Dentro do painel do Administrador/Funcionário, existe a possibilidade de cadastrar, alterar, deletar e listar clientes. Na Figura 11, apresenta o formulário de cadastro onde é possível

preencher todas as informações devidas e necessárias

SEBRAE Admin

MENU PRINCIPAL: Home, Administrador, Atendimento

Cadastrar Cliente

DADOS PESSOAIS

Nome: CPF: Email: Celular:

Nascimento:

ENDEREÇO

CEP: Cidade: Rua: Número:

Bairro:

CNPJ RELACIONADO

CNPJ: Segmento: Qualificações:

SELECIONADOS

6 Por página

CNPJ	Nome Fantasia	Qualificação	Segmento	Status
No data				

Figura 11 – Tela para registro de clientes

4.5 TELA PARA REALIZAR O REGISTRO DE UMA EMPRESA

Dentro do painel do Administrador/Funcionário, também existe a tela para registrar uma empresa que pode solicitar atendimento, e ser relacionado a um cliente.

SEBRAE Admin

MENU PRINCIPAL: Home, Administrador, Atendimento

Cadastrar Empresa

DADOS DA EMPRESA

Nome Da Empresa: CNPJ: Segmento: Qualificação:

Figura 12 – Tela para registro de empresas

4.6 TELA PARA REALIZAR O REGISTRO DE UM FUNCIONÁRIO

No painel do administrador, ele poderá registrar um novo funcionário dentro da aplicação.

A imagem mostra a interface de usuário para o registro de um funcionário no sistema SEBRAE. O cabeçalho azul contém o logo 'SEBRAE App', o nome de usuário 'Admin', e links para 'Notificações', 'Configurações' e 'Menu'. À esquerda, há um menu principal com opções para 'Home', 'Administrador' e 'Atendimento'. O formulário principal, intitulado 'Cadastrar Funcionário', contém o seguinte conteúdo:

DADOS PESSOAIS			
Nome*	CPF*	Email*	Celular*
<input type="text" value="Rubens da Silva Santos"/>	<input type="text" value="12345678900"/>	<input type="text" value="email@exemplo.com"/>	<input type="text" value="1899999999"/>
Nascimento*	<input type="text" value="dd/mm/aaaa"/>		

Na base do formulário, há dois botões: 'Voltar' (em azul claro) e 'Cadastrar' (em azul escuro).

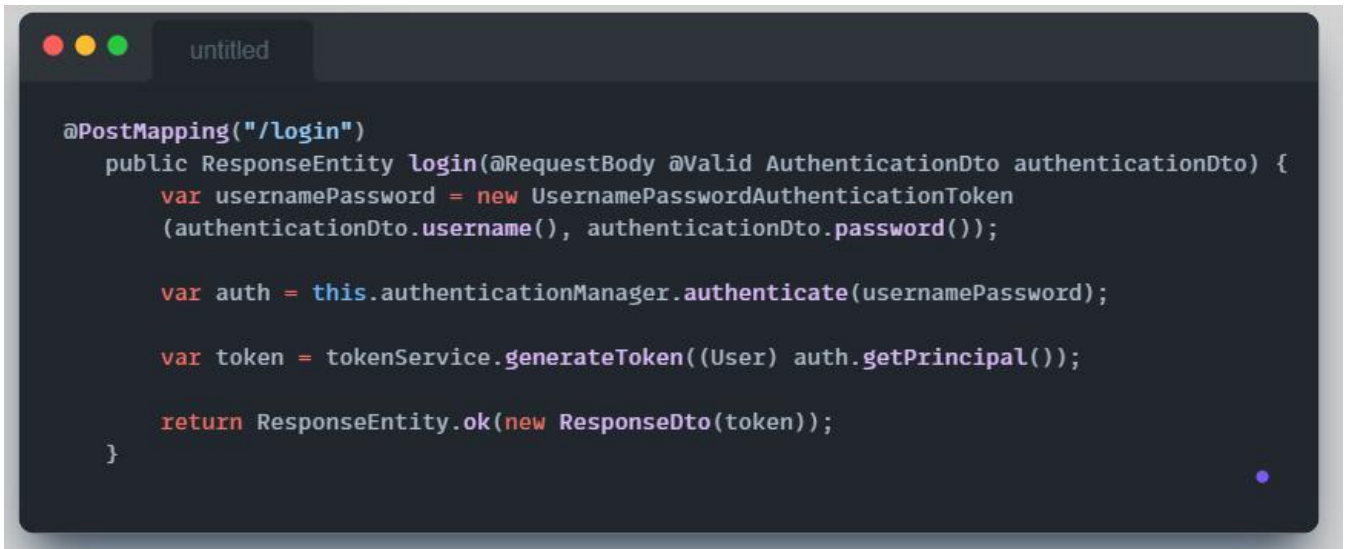
Figura 13 – Tela para registro de funcionários

4.7 AUTENTICAÇÃO DE USUÁRIO POR MEIO DE LOGIN

Para proteger o acesso às informações sensíveis da aplicação, o método de *login* gera um *token* de autenticação personalizado para cada usuário. Esse token é baseado nas permissões e roles atribuídas ao usuário, garantindo um acesso controlado e específico às funcionalidades da aplicação.

Quando um usuário realiza o login através da interface Front-End, uma requisição HTTP POST é enviada para o endpoint de login. Essa requisição inclui o nome de usuário e a senha. O servidor deserializa esses dados em um objeto `AuthenticationDto`, e utiliza o `UsernamePasswordAuthenticationToken` para autenticar as credenciais fornecidas.

O `authenticationManager`, um componente central do Spring Security, verifica a validade das credenciais. Se a autenticação for bem-sucedida, um token de autenticação é gerado pelo `tokenService`. Esse token é então retornado ao cliente dentro de um `ResponseDto`. O cliente usa esse token para acessar recursos protegidos da aplicação, permitindo um controle seguro e específico conforme as roles do usuário.

A screenshot of a code editor window with a dark theme. The window title is "untitled". The code is a Java Spring controller method for user login. It uses annotations like @PostMapping, @Valid, @RequestBody, and @Valid. The method logic involves creating a UsernamePasswordAuthenticationToken, authenticating it, generating a token, and returning a ResponseEntity with a ResponseDto containing the token. The code is as follows:

```
@PostMapping("/login")
public ResponseEntity login(@RequestBody @Valid AuthenticationDto authenticationDto) {
    var usernamePassword = new UsernamePasswordAuthenticationToken
        (authenticationDto.username(), authenticationDto.password());

    var auth = this.authenticationManager.authenticate(usernamePassword);

    var token = tokenService.generateToken((User) auth.getPrincipal());

    return ResponseEntity.ok(new ResponseDto(token));
}
```

Figura 14 – Autenticação de Usuário

5. CONCLUSÃO

Desde antes da revolução tecnológica, o atendimento ao cliente sempre foi um dos pilares fundamentais para avaliar a qualidade e a credibilidade de uma empresa. A satisfação do cliente é essencial para a missão e os valores de qualquer organização. Este software visa revolucionar o atendimento ao cliente, proporcionando uma solução moderna e ágil que reduz significativamente o tempo de espera, armazena os dados de maneira eficiente e elimina a necessidade de métodos obsoletos, como planilhas, papel e caneta, ou sistemas de senha.

A proposta desta aplicação é melhorar a produtividade e a eficiência no dia a dia dos funcionários, ao mesmo tempo que aumenta a atratividade da empresa para os clientes. Ao oferecer um suporte e atendimento de alta qualidade, a aplicação não apenas otimiza os processos internos, mas também promove um feedback positivo, refletido na satisfação dos clientes. Com isso, o software contribui para um atendimento mais eficaz e uma experiência mais gratificante para todos os envolvidos.

Além disso, ao implementar esta solução, a empresa posiciona-se na vanguarda da inovação no atendimento ao cliente, demonstrando um compromisso com a excelência e a modernização dos seus processos. Isso não só melhora a eficiência operacional, mas também reforça a imagem da empresa como uma organização que valoriza e prioriza a experiência do cliente. Com um atendimento mais eficiente e uma gestão de dados aprimorada, a empresa está preparada para enfrentar desafios futuros e manter-se competitiva em um mercado em constante evolução.

5.1 TRABALHOS FUTUROS

Com o progresso alcançado, os próximos passos para a evolução desta solução incluirão a adição de novas funcionalidades, visando otimizar ainda mais a agilidade no atendimento. Serão introduzidas ferramentas adicionais tanto para o painel dos Funcionários quanto para a interface dos Estagiários. Entre as melhorias planejadas, destacam-se a implementação de relatórios detalhados por data, semana, mês, trimestre e semestre, permitindo uma análise aprofundada do desempenho dos funcionários.

Além disso, será desenvolvido um sistema de agendamento antecipado de atendimentos, possibilitando aos clientes a criação de solicitações com datas e horários disponíveis. Para isso, será adicionado um novo tipo de usuário, "Cliente", que terá acesso exclusivo para

agendar atendimentos. Essas melhorias têm como objetivo aprimorar continuamente a experiência dos usuários e clientes, oferecendo um software de alta qualidade que atenda plenamente todas as suas necessidades por meio de um atendimento excepcional.

REFERENCIAS

Atende Simples. (2012). **Como as plataformas SaaS podem otimizar o serviço de atendimento ao cliente**. Disponível em: <https://www.atendesimples.com/blog/como-as-plataformas-saas-podem-otimizar-o-servico-de-atendimento-ao-cliente>. Acesso em: 23 de Nov, 2023.

TEx Tecnologia. (2009). **Gestão de Clientes**. Disponível em: <https://www.textecnologia.com.br/blog/gestao-de-clientes>. Acesso em: Nov. 2023.

LinkedIn. (2002). **Plataformas de CRM e gestão de relacionamentos com clientes na era digital**. Disponível em: https://www.linkedin.com/pulse/plataformas-de-crm-e-gestao-relacionamentos-com-clientes-na-era-1ffzf/?trk=organization_guest_main-feed-card_feed-article-content&originalSubdomain=pt. Acesso em: 23 de Nov, 2023.

AUGUSTI, N. C.; EMBALDE, I. L. P.; MENDES, O. L. PROJETO DE MELHORIA DO ATENDIMENTO, PRODUÇÃO E DISTRIBUIÇÃO DE UMA LANCHONETE UTILIZANDO O SOFTWARE DE SIMULAÇÃO ARENA. **Revista Interface Tecnológica**, [S. l.], v. 19, n. 1, p. 477–488, 2022. DOI: 10.31510/infa.v19i1.1415. Disponível em: <https://revista.fatectq.edu.br/interfacetecnologica/article/view/1415>. Acesso em: 13 mar. 2024.

PALADINI, Edson Pacheco. *Gestão da qualidade: teoria e prática*. 3. ed. São Paulo: Atlas, 2012.

EXAME. 4 Coisas que os clientes realmente querem de você. Disponível em: . Acesso em: 13 mar. 2024.

Microsoft Azure. O que é o Spring Boot? Disponível em: <https://azure.microsoft.com/pt-br/resources/cloud-computing-dictionary/what-is-java-spring-boot#:~:text=O%20que%20%C3%A9%20o%20Spring,criar%20microservi%C3%A7os%20e%20aplicativos%20Web>. Acesso em: 2 set. 2024.

IBM. Java Spring Boot. Disponível em: <https://www.ibm.com/br-pt/topics/java-spring-boot>. Acesso em: 2 set. 2024.

Brasil Code. Tailwind CSS: O que é e como usar? Disponível em: <https://www.brasilcode.com.br/tailwind-css-o-que-e-como-usar/>. Acesso em: 2 set. 2024.

React. Introducing React Dev. Disponível em: <https://pt-br.react.dev/blog/2023/03/16/introducing-react-dev>. Acesso em: 2 set. 2024.

Alura. Artigos sobre React JS. Disponível em: https://www.alura.com.br/artigos/react-js?srsltid=AfmBOoo9F21riEDBfj5c4LWZQ20Hv0p-Kqv6-h_1szY5Eb9Kz4lssQsf. Acesso em: 2 set. 2024.

Insomnia. Produtos Insomnia. Disponível em: <https://insomnia.rest/products/insomnia>. Acesso em: 2 set. 2024.

PostgreSQL. Sobre o PostgreSQL. Disponível em: <https://www.postgresql.org/about/>. Acesso em: 2 set. 2024.

JAVA. O que é Java? Disponível em: https://www.java.com/pt-BR/download/help/whatis_java.html. Acesso em: 2 set. 2024.

REACT. Página inicial do React. Disponível em: <https://pt-br.react.dev>. Acesso em: 2 set. 2024.

