**UNIP – Universidade Paulista**

**Projeto Integrado Multidisciplinar III**

**MARCUS VINICIUS LUCAS DE OLIVEIRA**

**RAFAEL MORANGON BESSA**

**RAPHAEL LIMA CAPRARA**

**RODRIGO DE TOLEDO RODRIGUES**

**Projeto de Levantamento e Análise do Sistema de Controle de Fazenda Urbana**

São Paulo

2024

**MARCUS VINICIUS LUCAS DE OLIVEIRA**

**RAFAEL MORANGON BESSA**

**RAPHAEL LIMA CAPRARA**

**RODRIGO DE TOLEDO RODRIGUES**

**Projeto de Levantamento e Análise do Sistema de Controle de Fazenda Urbana**

Projeto Integrado Multidisciplinar (PIM), apresentado como requisito parcial no segundo semestre para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas na Universidade Paulista do Polo Tatuapé.

Orientador(a): Prof. Salatiel Luz Marinho

São Paulo

2024

**RESUMO**

Este documento técnico descreve o desenvolvimento de um sistema de controle integrado para uma fazenda urbana para otimizar a produção e distribuição de alimentos em ambientes urbanos. O projeto não só envolve a implementação de tecnologias inovadoras, mas também a criação de uma infraestrutura robusta de TI e a análise detalhada de dados para garantir eficiência operacional e segurança alimentar. O sistema foi desenvolvido seguindo o modelo de engenharia de software ágil, para facilitar a rápida adaptação às necessidades dinâmicas de uma fazenda urbana. A linguagem C# foi escolhida por sua robustez e compatibilidade com sistemas baseados em .NET, ideais para o desenvolvimento de aplicações que requerem alto nível de segurança e integração com a internet das coisas (IoT). Além do sistema de controle, o projeto incluiu a implementação de um banco de dados MS SQL Server, hospedado em um servidor Windows Server, para gerenciar eficientemente os dados sobre produção, fornecedores, e clientes, mantendo conformidade com a Lei Geral de Proteção de Dados (LGPD). Uma análise gráfica e relatórios detalhados permitem monitorar o desempenho da fazenda e identificar oportunidades de melhoria, garantindo a sustentabilidade e a eficácia das operações. Em resumo, este projeto multidisciplinar não apenas desenvolveu um sistema eficaz para a gestão de fazendas urbanas, mas também estabeleceu uma infraestrutura tecnológica avançada e realizou análises profundas para otimizar a produção alimentar em contexto urbano, contribuindo significativamente para a segurança alimentar e sustentabilidade.

Palavras-chave: Fazenda urbana, análise de sistema, levantamento de projeto, segurança alimentar.

**ABSTRACT**

This technical document describes the development of an integrated control system for an urban farm to optimize the production and distribution of food in urban environments. The project involves not only the implementation of innovative technologies but also the creation of a robust IT infrastructure and detailed data analysis to ensure operational efficiency and food security. The system was developed following the agile software engineering model, to facilitate rapid adaptation to the dynamic needs of an urban farm. The C# language was chosen for its robustness and compatibility with .NET-based systems, ideal for developing applications that require high levels of security and integration with the Internet of Things (IoT). In addition to the control system, the project included the implementation of an MS SQL Server database, hosted on a Windows Server, to efficiently manage data on production, suppliers, and customers, while maintaining compliance with the General Data Protection Regulation (GDPR). Graphic analysis and detailed reports allow for monitoring farm performance and identifying opportunities for improvement, ensuring the sustainability and effectiveness of operations. In summary, this multidisciplinary project not only developed an effective system for managing urban farms but also established an advanced technological infrastructure and conducted in-depth analyses to optimize food production in an urban context, significantly contributing to food security and sustainability.

Keywords: Urban farm, system analysis, project survey, food security.

**LISTA DE FIGURAS**

**Figura 1: Fases do Design Thinking...........................................................................53**

**Figura 2: Protótipo de baixa fidelidade de tela de login...........................................57**

**Figura 3: Apresentação visual das informações cadastradas no banco de dados.............................................................................................................................58**

**Figura 4: Formulário para cadastro............................................................................59**

**Figura 5: Tela de login mobile.....................................................................................60**

**Figura 6: Tela de estoque mobile................................................................................61**

**Figura 7: Logomarca Farmtech...................................................................................62**

**Figura 8: Tela Login Mobile em Alta Fidelidade........................................................63**

**Figura 9: Tela de Login de Alta Fidelidade.................................................................63**

**Figura 10: Tela de Estoque de Alta Fidelidade..........................................................64**

**Figura 11: Tela de Cadastro de Produtos em Alta Fidelidade..................................64**

**Figura 12: Tela de Detalhes e Edição de Produtos em Alta Fidelidade...................65**

**Figura 13: Menu Mobile Alta Fidelidade.....................................................................66**

**Figura 14: Tela Estoque Mobile Alta Fidelidade........................................................66**

**Figura 15: Tela Cadastro Mobile Alta Fidelidade.......................................................67**

**Figura 16: Tela Detalhes Mobile Alta Fidelidade.......................................................67**

**Figura 17: Tela de Clientes Alta Fidelidade...............................................................68**

**Figura 18: Tela Cadastro Clientes Alta Fidelidade....................................................68**

**Figura 19: Tela Detalhes e Edição de Produtos Alta Fidelidade..............................69**

**Figura 20: Tela Clientes Mobile...................................................................................70**

**Figura 21: Tela Cadastro Clientes Mobile..................................................................70**

**Figura 22: Tela Detalhes e Edição Clientes Mobile ..................................................71**

**Figura 23: Tela Fornecedores Mobile.........................................................................71**

**Figura 24: Tela Fornecedores Alta Fidelidade...........................................................72**

**Figura 25: Tela Cadastro Fornecedores Alta Fidelidade..........................................72**

**Figura 26: Tela Detalhes e Edição de Fornecedores Alta Fidelidade......................73**

**Figura 27: Tela Financeiro Alta Fidelidade................................................................73**

**Figura 28: Tela Cadastro Fornecedores Mobile........................................................74**

**Figura 29: Tela Detalhes e Edição Fornecedores Mobile.........................................75**

**Figura 30: Tela Financeiro Mobile...............................................................................76**

**Figura 31: Tela de RH em Alta Fidelidade..................................................................77**

**Figura 32: Tela Cadastro de Funcionários em Alta Fidelidade................................77**

**Figura 33: Tela Detalhes e Edição de Funcionários em Alta Fidelidade.................78**

**Figura 34: Tela RH Mobile............................................................................................79**

**Figura 35: Tela Cadastro Funcionário Mobile............................................................79**

**Figura 36: Tela Detalhes e Edição de Funcionários Mobile.....................................80**

**Figura 37: Diagrama de Classe do pacote DAO........................................................83**

**Figura 38: Código fonte para conexão com o banco de dados...............................84**

**Figura 39: Uso do método getConexão......................................................................85**

**Figura 40: Diagrama de Classe de venda...................................................................86**

**Figura 41: Diagrama de Classe de compra**.**...............................................................87**

**Figura 42: Modelo ER conceitual de venda...............................................................88**

**Figura 43: Modelo ER lógico de venda......................................................................89**

**Figura 44: Modelo ER conceitual de compra............................................................90**

**Figura 45: Modelo ER lógico de compra...................................................................91**

**Figura 46: Resultado do script SQL executado........................................................97**

**Figura 47: Dados pré inseridos na tabela.................................................................97**

**Figura 48: Arquitetura de pacotes do sistema.........................................................99**

**Figura 49: Enum de nível de acesso........................................................................100**

**Figura 50: Atributo NivelAcesso em Usuário..........................................................101**

**Figura 51: Métodos Equals e Hashcode em Usuário.............................................102**

**Figura 52: Método cadastraNovoUsuarioNoSistema.............................................103**

**Figura 53: Método Encriptar utilizando SHA256.....................................................104**

**Figura 54: Enum Status para soft delete.................................................................105**

**Figura 55: Tela de login em console........................................................................106**

**Figura 56: Tela console após acesso do administrador........................................107**

**Figura 57: Cadastro de cliente via console.............................................................108**

**Figura 58: Dados do cliente inseridos no banco.....................................................108**

**Figura 59: Funcionalidade de buscar pelo CPF.......................................................109**

**Figura 60: Diagrama de Caso de Uso do sistema FarmTech.................................114**

**Figura 61: Métricas de desempenho de um funcionário exemplo.........................118**

SUMÁRIO

[1. INTRODUÇÃO 10](#_Toc149045742)

[1.1. Tipo de pesquisa: 1](#_Toc149045743)0

[1.2. Contribuição da pesquisa: 11](#_Toc149045744)

[1.3. Objetivo geral: 11](#_Toc149045745)

[2. REFERENCIAL TEÓRICO 12](#_Toc149045746)

[2.1. Engenharia de software II 12](#_Toc149045747)

[2.2. Análise de sistemas orientados a objetos 1](#_Toc149045748)3

[2.3. Programação orientada a objetos I 1](#_Toc149045749)4

[2.4. Projeto de interface com o usuário 15](#_Toc149045750)

[2.5. Banco de dados 16](#_Toc149045751)

[2.6. Economia e mercado 17](#_Toc149045752)

[2.7. Gestão estratégica de recursos humanos 1](#_Toc149045752)8

[3. DESENVOLVIMENTO](#_Toc149045753) 20

[3.1. Segurança alimentar e banco de alimentos](#_Toc149045754) 20

[3.2. Fazendas urbanas](#_Toc149045755) 21

[3.2.1. Tipos de fazendas urbanas](#_Toc149045756) 22

[3.2.2. Historico das fazendas urbanas](#_Toc149045757) 22

[3.3. ESG e ODS 2](#_Toc149045761)3

[3.4. COP30](#_Toc149045771) 24

[3.4.1. Objetivos da cop30 2](#_Toc149045772)4

[3.4.2. O envolvimento do brasil 2](#_Toc149045773)5

[3.5. A empresa](#_Toc149045776) 26

[3.6. Economia e mercado](#_Toc149045776) 26

[3.6.1. Estimativa de viabilidade](#_Toc149045776) 26

[3.6.2. Projeção de demanda](#_Toc149045776) 26

[3.6.3. Análise competitiva](#_Toc149045776) 27

[3.6.4. Análise de custo benefício](#_Toc149045776) 27

[3.6.5. Cenário de retorno sobre investimento (ROI)](#_Toc149045776) 27

[3.6.6. Impacto social e sustentabilidade](#_Toc149045776) 27

[3.6.7. Conclusão da análise](#_Toc149045776) 28

[3.7. Regras de negócio](#_Toc149045776) 28

[3.7.1. Regras de negócio na programação](#_Toc149045776) 29

[3.7.2. Regras de negócio da farmtech](#_Toc149045776) 30

[3.8. Requisitos dos usuários](#_Toc149045776) 50

[3.9. Requisitos de sistema](#_Toc149045776) 51

[3.10. Desenvolvimento da interface de usuário](#_Toc149045776) 52

[3.10.1. Fases do design thinking](#_Toc149045776) 52

[3.10.1.1. Empatia](#_Toc149045776) 53

[3.10.1.2. Definição dos problemas](#_Toc149045776) 54

[3.10.1.3. Idealização](#_Toc149045776) 54

[3.10.1.4. Prototipação](#_Toc149045776) 57

[3.10.1.4.1. Responsividade](#_Toc149045776) 59

[3.10.1.4.2. Identidade visual](#_Toc149045776) 62

[3.10.1.4.3. Prototipo de alta fidelidade](#_Toc149045776) 63

[3.10.1.5. Testes de interface](#_Toc149045776) 81

[3.11. Diagrama de classes do sistema](#_Toc149045776) 82

[3.12. Banco de dados do sistema](#_Toc149045776) 87

[3.13. Sistema orientado a objetos](#_Toc149045776) 98

[3.14. Testes de software](#_Toc149045776) 105

[3.14.1. Testando a funcionalidade de Login](#_Toc149045776) 106

[3.14.2. Testando a funcionalidade de cadastro de clientes](#_Toc149045776) 107

[3.15. Plano de teste](#_Toc149045776) 109

[3.16. Manual de uso do sistema](#_Toc149045776) 113

[3.16.1. Login](#_Toc149045776) 114

[3.16.2. Visualização e cadastro de itens](#_Toc149045776) 115

[3.16.3. Edição/atualização dos dados cadastrados](#_Toc149045776) 115

[3.16.4. Relatórios da aba financeiro](#_Toc149045776) 116

[3.17. Gestão estratégica de recursos humanos](#_Toc149045776) 116

[4. CONCLUSÃO](#_Toc149045778) 119

[REFERÊNCIAS BIBLIOGRÁFICAS](#_Toc149045779) 120

1. INTRODUÇÃO

Neste projeto, exploramos a intersecção entre inovação tecnológica e segurança alimentar, aplicando os conhecimentos adquiridos ao longo do curso de Análise e Desenvolvimento de Sistemas nas disciplinas de Engenharia de Software II, Análise de Sistemas Orientada a Objetos, Programação Orientada a Objetos I, entre outras. Em resposta aos crescentes desafios globais de insegurança alimentar exacerbados pela pandemia, a startup "GreenTech Innovations" propõe a criação de uma fazenda urbana inteligente, empregando tecnologias para otimizar a produção e distribuição de alimentos em ambientes urbanos.

O projeto visa desenvolver um sistema de controle abrangente para a fazenda urbana, que gerenciará a produção, fornecedores, estoques, cadastros de clientes, funcionários, dentre outros. Além disso, o auxiliará para ser uma ferramenta proativa na promoção da segurança alimentar e sustentável.

Crucialmente, o sistema será desenvolvido em conformidade com a Lei Geral de Proteção de Dados (LGPD), garantindo que todas as interações e transações de dados sejam realizadas de forma segura e ética. Para atender a essa exigência, será implementado um módulo específico dentro do sistema para gerenciar consentimentos e garantir que os direitos dos usuários sejam respeitados.

As etapas do projeto incluem a definição detalhada das regras de negócio, a modelagem dos processos através de diagramas UML, o desenvolvimento de uma prova de conceito em console e a preparação de infraestrutura de banco de dados e servidores conforme as especificações técnicas necessárias.

Este empreendimento não apenas demonstra a aplicação prática de avançadas técnicas de programação e análise de sistemas, mas também enfatiza a importância da colaboração multidisciplinar e do trabalho em equipe para resolver problemas complexos do mundo real.

1.1. Tipo de Pesquisa:

No contexto deste projeto inovador, a abordagem de pesquisa adotada é a pesquisa aplicada. Essa abordagem prática visa solucionar problemas específicos e atender às necessidades reais da "GreenTech Innovations", abrangendo questões de gerenciamento de produtos, funcionários fornecedores, segurança da informação e conformidade com as regulamentações legais, principalmene LGPD.

1.2. Contribuição da Pesquisa:

A contribuição deste trabalho está centrada na aplicação concreta dos conhecimentos adquiridos durante o curso de Análise e Desenvolvimento de Sistemas. O projeto de gerenciamento do "GreenTech Innovations" busca solucionar desafios específicos e atender às necessidades práticas de uma fazenda urbana, abordando questões de gerenciamento de produtos, funcionários fornecedores, segurança da informação e conformidade com as regulamentações legais, principalmene LGPD.

1.3. Objetivo Geral:

O objetivo geral deste projeto é desenvolver um sistema de gerenciamento completo para o "GreenTech Innovations" com o propósito de aprimorar o controle da fazendo em geral e promover a segurança alimentar de forma inovadora.

1. REFERENCIAL TEÓRICO

Seção que tem por finalidade uma pesquisa teórica sobre os temas a serem abordados no trabalho.

## **Engenharia de Software II**

Em nosso trabalho, a matéria de Engenharia de Software II foi utilizada para enfatizar a aplicação integrada de práticas essenciais para o desenvolvimento de um sistema de controle para fazendas urbanas. O projeto é guiado por uma estrutura rigorosa de requisitos funcionais e não funcionais, que detalha desde a gestão de inventário até requisitos de segurança e desempenho, seguindo as diretrizes das normas ISO 25000 e ISO 9001 (ISO, 2015; ISO, 2018).

Conforme Sommerville (2015), o sistema deve aderir a regras de negócio claramente definidas que orientam todas as operações e processos, alinhando-se com as estratégias organizacionais e regulamentações de segurança alimentar. A abordagem de desenvolvimento utiliza o Modelo Espiral, que, segundo Boehm (1988), permite iterações incrementais e flexibilidade para ajustar o software às necessidades emergentes, enquanto integra testes e avaliações em cada fase.

Adotamos métodos ágeis para facilitar a rápida resposta às mudanças e otimizar a entrega contínua, conforme descreve Beck et al. (2001), que promovem a colaboração efetiva dentro das equipes de projeto e com os stakeholders. Para garantir a satisfação do usuário e a adequação do sistema, aplicamos práticas de Garantia da Qualidade de Software (GQS), baseando-nos em padrões como a NBR 9126-1 e a ISO 9000 (ABNT, 2003; ISO, 2005), que ajudam a manter um alto padrão de qualidade através de revisões sistemáticas e testes rigorosos.

Além disso, a manutenção do software é planejada de acordo com as práticas recomendadas por Pressman (2014), que enfatiza a importância de uma manutenção eficaz para a longevidade e eficácia do software. O gerenciamento de configuração, crucial para o controle de versões e mudanças no software, segue as orientações da ISO 10007 (ISO, 2003), garantindo que todas as alterações sejam devidamente documentadas e rastreáveis.

Portanto, este projeto multidisciplinar não apenas desenvolve um sistema eficaz para a gestão de fazendas urbanas mas também se baseia em uma fundação teórica robusta e práticas comprovadas para garantir sua sustentabilidade e eficácia operacional a longo prazo.

## **Análise de Sistemas Orientados a Objetos**

A disciplina de Análise de Sistemas Orientada a Objetos é fundamental para a estruturação e desenvolvimento do sistema de controle em nossa fazenda urbana. Essa abordagem facilita a compreensão e a modelagem de sistemas complexos através da encapsulação de dados e comportamentos em objetos, o que é vital para o manejo eficiente de processos agrícolas e de distribuição em ambientes urbanos.

Orientação a objetos é uma metodologia de projeto que organiza software como uma coleção de objetos que incorporam tanto dados quanto os procedimentos necessários para operá-los, como explicado por Booch, Rumbaugh e Jacobson (1999). Este paradigma é especialmente útil em nosso projeto devido à sua habilidade em modelar processos de negócios complexos e inter-relacionados de uma maneira intuitiva e modular.

A linguagem de programação escolhida para o desenvolvimento do sistema é C#, conhecida por sua robustez e suporte extensivo à programação orientada a objetos. Segundo Albahari e Albahari (2017), C# oferece uma combinação poderosa de desempenho, flexibilidade e clareza, tornando-a ideal para implementar as funcionalidades necessárias em nosso sistema de controle.

A análise e o projeto do sistema utilizam técnicas detalhadas de engenharia de requisitos e modelagem de processos de negócio, garantindo que todas as necessidades operacionais sejam cuidadosamente especificadas e que o sistema final esteja alinhado com os objetivos estratégicos da fazenda. De acordo com Sommerville (2016), a engenharia de requisitos não só define o que o sistema deve fazer, mas também como ele deve interagir com outros sistemas e processos empresariais, uma consideração crítica para a integração eficaz das operações da fazenda urbana.

Diagramas UML (Unified Modeling Language) são extensivamente utilizados para a representação visual das especificações do sistema, facilitando a comunicação entre os desenvolvedores e os stakeholders e assegurando uma compreensão clara dos requisitos e da arquitetura do sistema, como apontado por Fowler (2004).

Portanto, a análise de sistemas orientada a objetos não apenas fornece a metodologia para o design e implementação do nosso sistema de fazenda urbana, mas também assegura que ele seja escalável, manutenível e alinhado com as práticas contemporâneas de desenvolvimento de software.

## **Programação Orientada a Objetos I**

A disciplina de Programação Orientada a Objetos I é essencial para o desenvolvimento do nosso projeto de fazenda urbana, oferecendo as ferramentas necessárias para criar um sistema robusto, modular e fácil de manter. Através da utilização de conceitos como classes, construtores, herança e objetos, a programação orientada a objetos permite que complexidades do sistema sejam encapsuladas de maneira eficiente e eficaz.

Classes em C#, conforme explicado por Troelsen e Japikse (2017), são os blocos de construção básicos de uma aplicação em C#. No contexto da nossa fazenda urbana, cada classe representa uma entidade distinta com atributos e métodos específicos; por exemplo, classes para gerenciamento de fornecedores, controle de estoque, e funcionários. Estas classes são fundamentais para organizar o código de maneira lógica e reutilizável.

Construtores são usados para inicializar objetos de uma classe e são um conceito chave na criação de instâncias operacionais dos módulos do sistema. Segundo Albahari e Albahari (2017), construtores em C# garantem que o objeto comece sua vida em um estado válido, configurando padrões essenciais antes de o objeto ser utilizado.

Herança é outro conceito poderoso da programação orientada a objetos, permitindo que novas classes adquiram propriedades de classes existentes. Isso facilita a reutilização de código e a criação de uma hierarquia de classes que pode ser expandida e adaptada sem modificar as classes existentes, minimizando a duplicação de código e os erros, como destacado por Freeman e Freeman (2020).

O uso de objetos, que são instâncias de classes, permite a manipulação dos dados e a execução das funcionalidades e comportamentos definidas nas classes, facilitando a gestão eficaz dos diversos aspectos operacionais da fazenda urbana.

## **Projeto de Interface com o Usuário**

Esta matéria aborda técnicas e ferramentas para o design e avaliação de interfaces que são essenciais para garantir uma experiência do usuário eficiente e agradável. Utilizamos o Figma, uma ferramenta de design interativo, para prototipar e iterar sobre os layouts das interfaces, o que permite testes rápidos e feedback contínuo.

"Fracassar cedo para ter sucesso mais cedo..." (Borba, 2018) esta filosofia é é fundamental no desenvolvimento de interfaces, pois permite identificar problemas de usabilidade e interação antes que se tornem custosos de resolver. Este princípio foi aplicado através de múltiplas sessões de prototipação, utilizando o Figma para ajustar e aprimorar a interface com base nos feedbacks dos usuários.

A Interacção Humano-Computador (IHC) é um componente essencial do projeto, focando-se na criação de interfaces que não só atendam às necessidades operacionais, mas que também proporcionem uma qualidade de uso superior. Seguindo as diretrizes estabelecidas na ISO 9241, parte 11, sobre usabilidade, que define usabilidade como "a medida em que um produto pode ser usado por usuários específicos para alcançar objetivos específicos com eficácia, eficiência e satisfação em um contexto de uso específico" (ISO, 2018).

Além disso, o aspecto de segurança é tratado como uma prioridade no design da interface, assegurando que todas as interações cumpram rigorosos padrões de segurança para proteger os dados dos usuários e do sistema. As interfaces são projetadas para serem intuitivas, facilitando a navegação e minimizando o risco de erros, o que é vital em um ambiente que lida diretamente com a produção e distribuição de alimentos.

Em resumo, o curso de Projeto de Interface com o Usuário proporcionou as competências necessárias para desenvolver interfaces que não só melhoram a interação dos usuários com o sistema, mas também reforçam a operacionalidade, segurança e satisfação do usuário. A aplicação desses conhecimentos foi demonstrada através da criação de protótipos eficazes e seguros, avaliados por meio de testes de usabilidade para garantir sua eficácia no contexto real de uso.

## **Banco de Dados**

A disciplina de Banco de Dados é fundamental para o sucesso de qualquer sistema de informação, neste contexto, o MS SQL Server foi escolhido como sistema de gerenciamento de banco de dados devido à sua robustez, segurança e capacidade de lidar com grandes volumes de dados e transações complexas, essencial para o controle de produção, fornecedores e vendas (Elmasri & Navathe, 2016).

Utilizamos Modelagem de Dados, incluindo Diagrama Entidade-Relacionamento (DER) e Modelo Entidade-Relacionamento (MER), para estruturar e organizar a informação de maneira eficiente e intuitiva. Esses modelos são cruciais para o desenvolvimento de um sistema coeso e para garantir a integridade dos dados. A criação de uma DLL (Data Definition Language) foi fundamental para definir a estrutura do banco de dados, enquanto a DML (Data Manipulation Language) foi utilizada para a inserção, consulta, atualização e remoção de dados dentro do sistema (Date, 2013).

O diagrama ER foi desenvolvido para representar visualmente a estrutura de dados do sistema, facilitando a compreensão das relações entre as entidades do domínio do problema (Chen, 1976). Além disso, os scripts de carga de dados de teste e o dicionário de dados foram meticulosamente preparados para garantir que todos os dados inseridos no sistema fossem precisos e úteis para a realização de testes eficazes.

Para a verificação e validação do sistema, foram elaborados scripts de dados iniciais de testes e homologação do sistema, cobrindo testes de unidade, integração, performance e usabilidade. Esses testes são essenciais para assegurar a funcionalidade do sistema antes de sua implantação final (Myers, Sandler & Badgett, 2011). Ademais, as queries no banco de dados foram cuidadosamente desenhadas para verificar a correção dos dados e garantir que as operações de banco de dados estivessem otimizadas e seguras, prevenindo falhas e perdas de dados (Silberschatz, Korth & Sudarshan, 2011).

Em suma, a aplicação dos conhecimentos adquiridos na disciplina de Banco de Dados permitiu a criação de uma base sólida para o sistema de gerenciamento da fazenda urbana, assegurando a integridade, segurança e eficiência na manipulação dos dados, aspectos cruciais para o sucesso e sustentabilidade do projeto.

## **Economia e mercado**

A matéria de Economia e Mercado é crucial para entender as dinâmicas econômicas que influenciam diretamente a viabilidade e sustentabilidade de projetos como nossa fazenda urbana. Neste curso, exploramos conceitos chave como inflação, crescimento e desenvolvimento econômico, além de aprendermos a realizar estimativas de viabilidade econômica que são essenciais para qualquer empreendimento no setor agrícola urbano.

A inflação é um fator crítico que afeta todos os aspectos econômicos de um projeto, desde os custos de insumos até o preço final dos produtos. Segundo Mankiw (2014), entender as tendências inflacionárias permite às empresas ajustar suas estratégias de preços e custos, garantindo que permaneçam competitivas e rentáveis no mercado. Para o projeto da fazenda urbana, a monitorização constante da inflação é vital para manter a sustentabilidade financeira.

O crescimento e desenvolvimento econômico são indicadores de como uma região ou país está expandindo sua capacidade econômica e melhorando o padrão de vida de sua população. Krugman e Wells (2018) destacam que projetos que contribuem para o crescimento econômico local, como fazendas urbanas, podem beneficiar-se de incentivos governamentais e apoio comunitário, além de contribuir significativamente para a segurança alimentar e emprego local.

Finalmente, a estimativa da viabilidade é o processo de análise de todos os dados econômicos e de mercado para determinar se o projeto é financeiramente viável. Blanchard (2017) explica que essa análise deve considerar não apenas os custos e receitas projetados, mas também avaliar os riscos associados, incluindo variações de mercado e mudanças na política econômica. Para nosso projeto de fazenda urbana, realizamos uma análise de viabilidade detalhada que incluiu projeções de demanda, análise de custo-benefício e cenários de retorno sobre investimento, assegurando que o empreendimento é economicamente sustentável e capaz de adaptar-se a mudanças econômicas.

Em resumo, a disciplina de Economia e Mercado fornece as ferramentas necessárias para compreender e aplicar conceitos econômicos que garantem a viabilidade e eficácia de projetos inovadores como fazendas urbanas, alinhando-os com as exigências econômicas e de mercado atuais.

## **Gestão Estratégica de Recursos Humanos**

A disciplina de Gestão Estratégica de Recursos Humanos desempenha um papel fundamental na concepção e implementação do nosso projeto de fazenda urbana. Neste curso, exploramos diversos aspectos que impactam diretamente a gestão de equipes e o desempenho organizacional, desde a estruturação do organograma até a avaliação do clima e cultura organizacional.

O desafio da gestão de RH reside na capacidade de alinhar os objetivos estratégicos da organização com as necessidades e expectativas dos colaboradores. Robbins e Judge (2018) destacam que isso requer não apenas a implementação de práticas eficazes de recrutamento, seleção e treinamento, mas também uma compreensão profunda das motivações e aspirações individuais.

O organograma é uma representação visual da estrutura hierárquica da organização, definindo as relações de autoridade e responsabilidade entre os diferentes níveis e departamentos. Chiavenato (2014) ressalta que um organograma claro e bem definido facilita a comunicação e coordenação dentro da equipe, promovendo eficiência e eficácia operacional.

A cultura e clima organizacional são aspectos essenciais que influenciam a motivação, satisfação e desempenho dos colaboradores. Schein (2010) destaca que uma cultura organizacional forte e positiva promove a identificação dos funcionários com os valores e objetivos da empresa, enquanto um clima organizacional saudável estimula a criatividade, inovação e colaboração.

Os modelos de organizações desempenham um papel crucial na definição da estrutura e funcionamento das equipes. Mintzberg (1989) identifica diferentes tipos de modelos organizacionais, como burocráticos, matriciais e empreendedores, cada um com suas próprias vantagens e desafios, que devem ser considerados ao projetar a estrutura organizacional de nossa fazenda urbana.

As relações industriais também são um aspecto importante da gestão de RH, envolvendo a negociação e administração de relações entre empregadores e empregados. Dessler e Várkonyi (2017) explicam que uma boa gestão das relações industriais promove um ambiente de trabalho harmonioso e colaborativo, essencial para o sucesso de qualquer empreendimento.

Por fim, os relatórios de desempenho dos funcionários são ferramentas fundamentais para avaliar o desempenho individual e coletivo da equipe. Milkovich e Newman (2016) destacam que relatórios precisos e objetivos permitem identificar pontos fortes e áreas de melhoria, orientando o desenvolvimento profissional e contribuindo para o alcance dos objetivos organizacionais.

Em resumo, a disciplina de Gestão Estratégica de Recursos Humanos fornece as bases teóricas e práticas necessárias para a eficaz gestão de equipes, promovendo um ambiente de trabalho produtivo, colaborativo e alinhado com os objetivos estratégicos da organização.

1. DESENVOLVIMENTO

Nesta seção mostraremos o desenvolvimento do trabalho prático, utilizando os conhecimentos das disciplinas: Engenharia de Software II, Análise de Sistemas Orientada a Objetos, Programação Orientada a Objetos I, Projeto de Interface com o Usuário, Banco de Dados, Economia e Mercado e Gestão Estratégica de Recursos Humanos.

* 1. **Segurança alimentar e banco de alimentos**

A segurança alimentar é uma questão fundamental para o desenvolvimento sustentável das sociedades, especialmente em contextos urbanos onde o acesso a alimentos frescos e nutritivos pode ser limitado. Segundo a Organização das Nações Unidas para a Alimentação e a Agricultura (FAO, 2021), a segurança alimentar é alcançada "quando todas as pessoas, em todos os momentos, têm acesso físico e econômico a alimentos suficientes, seguros e nutritivos para satisfazer suas necessidades alimentares e preferências para uma vida ativa e saudável."

Bancos de alimentos desempenham um papel crítico neste contexto, atuando como entidades que coletam, armazenam e distribuem alimentos para aqueles que estão em risco de insegurança alimentar. Conforme destaca Bazerghi, McKay, e Dunn (2016), bancos de alimentos não apenas mitigam o problema da fome, mas também ajudam a reduzir o desperdício de alimentos, uma preocupação significativa em muitas cidades ao redor do mundo.

O impacto dos bancos de alimentos vai além da mera distribuição de recursos; eles também são importantes para a conscientização sobre questões de pobreza e fome, conforme observado por Tarasuk, Dachner, e Loopstra (2014). Além disso, muitos bancos de alimentos estão agora integrando programas que também focam em aspectos nutricionais, garantindo que os alimentos fornecidos suportem dietas balanceadas e saudáveis.

Em nosso projeto de fazenda urbana, a conexão com bancos de alimentos locais será fundamental. A integração permitirá que parte da produção seja direcionada para famílias e indivíduos em necessidade, garantindo que o impacto social do projeto seja ampliado. Além disso, essa colaboração contribui para a sustentabilidade do projeto, já que produtos que poderiam ser desperdiçados são efetivamente utilizados para aliviar a fome local.

Por fim, a segurança alimentar e os bancos de alimentos são essenciais para o desenvolvimento de comunidades urbanas resilientes e sustentáveis. Segundo Foley et al. (2011), a integração de práticas agrícolas urbanas com redes de segurança alimentar pode transformar espaços urbanos, melhorando a acessibilidade alimentar e fortalecendo a coesão comunitária.

* 1. **Fazendas urbanas**

Nesta seção, apresentamos o conceito de fazendas urbanas como parte da proposta inovadora de nossa startup, que busca integrar a produção de alimentos de forma sustentável e acessível diretamente nos centros urbanos. Este modelo agrícola urbano surge como uma solução estratégica para enfrentar desafios de segurança alimentar, promovendo bem-estar comunitário e práticas ambientalmente responsáveis.

As fazendas urbanas são cruciais para combater a insegurança alimentar, oferecendo acesso direto a alimentos nutritivos em regiões urbanas onde supermercados e opções saudáveis são limitados. Elas também desempenham um papel vital na mitigação dos impactos ambientais da produção alimentar, reduzindo a necessidade de transporte de longa distância e as emissões de gases do efeito estufa.

Do ponto de vista econômico, as fazendas urbanas criam empregos e estimulam a economia local, proporcionando oportunidades de negócio e formação, especialmente em comunidades marginalizadas. Socialmente, promovem uma maior conexão com a alimentação, incentivando hábitos alimentares saudáveis e a conscientização sobre práticas sustentáveis.

As fazendas urbanas, portanto, não são apenas projetos agrícolas; elas são parte integrante de uma estratégia maior para criar sistemas alimentares sustentáveis que possam suportar as pressões de uma população urbana crescente e desafios econômicos globais. Sua expansão é essencial para enfrentar a fome e promover a igualdade no acesso a recursos alimentares, apoiando diretamente os esforços para alcançar os Objetivos de Desenvolvimento Sustentável da ONU, particularmente aqueles relacionados à erradicação da fome e à promoção de comunidades sustentáveis. Em suma, as fazendas urbanas são uma resposta multifacetada aos desafios contemporâneos de urbanização, segurança alimentar e sustentabilidade.

### Tipos de fazendas urbanas

As fazendas urbanas variam em tipo e escopo, adaptando-se às especificidades e necessidades das áreas urbanas onde são implantadas. Algumas das formas mais comuns incluem:

**Jardins Comunitários:** Estes são espaços compartilhados onde membros da comunidade cultivam alimentos em pequenas parcelas. Esses jardins fortalecem laços comunitários e facilitam o acesso a produtos frescos e saudáveis.

**Fazendas Verticais:** Empregam tecnologias avançadas como hidroponia e aquaponia para maximizar a produção em ambientes controlados e verticalizados. Este tipo de fazenda é particularmente eficaz em áreas urbanas densas, onde o espaço é limitado.

**Telhados Verdes:** Transformam os telhados de edifícios urbanos em áreas de cultivo, contribuindo não apenas para a produção de alimentos, mas também para a redução do efeito de ilha de calor, melhorando a qualidade do ar e reduzindo o consumo de energia.

**Fazendas de Periferia:** Localizadas na periferia das cidades, utilizam métodos de cultivo mais tradicionais e sustentáveis para fornecer alimentos frescos para os centros urbanos.

### Histórico das fazendas urbanas

A prática de agricultura urbana não é uma novidade. Durante a Segunda Guerra Mundial, os "jardins da vitória" nos Estados Unidos e no Reino Unido foram criados como uma resposta direta às restrições alimentares da época. Nas últimas décadas, contudo, o conceito de fazendas urbanas evoluiu significativamente, impulsionado pela necessidade de sustentabilidade e resiliência comunitária.

No Brasil, o movimento ganhou impulso como uma maneira de combater a crise econômica e a insegurança alimentar, com iniciativas em grandes metrópoles como São Paulo e Rio de Janeiro. Esses projetos se tornaram centros de fornecimento de alimentos locais e frescos e de educação ambiental.

### ESG e ODS

O conceito de ESG (Environmental, Social, and Governance - Ambiental, Social e Governança) surgiu como uma resposta a uma provocação do ex-secretário-geral da ONU, Kofi Annan, aos CEOs das grandes instituições financeiras sobre como incorporar aspectos sociais, ambientais e de governança nas decisões de mercado de capitais. Esta abordagem é vista como crucial para o futuro sustentável dos negócios, enfatizando que o sucesso corporativo a longo prazo depende de uma estratégia que respeite esses três pilares.

**Ambiental:** No pilar ambiental, as organizações são incentivadas a adotar práticas que minimizem seu impacto negativo no meio ambiente. Isso inclui gerenciar recursos naturais de maneira eficiente, reduzir emissões de poluentes, investir em tecnologias sustentáveis e apoiar a preservação da biodiversidade.

**Social:** O aspecto social do ESG foca no impacto da empresa na sociedade. Isso engloba desde o tratamento justo dos funcionários e a manutenção de um ambiente de trabalho seguro e inclusivo, até o envolvimento e suporte às comunidades locais. Iniciativas como a promoção da diversidade e inclusão, investimentos em educação e programas de saúde para colaboradores são exemplos de práticas que se encaixam neste pilar.

**Governança:** A governança se refere à forma como a empresa é administrada, incluindo práticas de gestão ética, transparência nas operações, responsabilidade corporativa e conformidade com a legislação. Uma boa governança assegura que existam controles internos fortes, auditorias regulares e uma clara divisão de poderes que previnam a corrupção e garantam que as decisões sejam tomadas no melhor interesse da empresa e de seus stakeholders.

O ESG se alinha diretamente aos Objetivos de Desenvolvimento Sustentável (ODS), pois proporciona um framework prático para as empresas contribuírem para essas metas globais. Os ODS são 17 e incluem: Erradicação da Pobreza, Fome Zero e Agricultura Sustentável, Saúde e Bem-Estar, Educação de Qualidade, Igualdade de Gênero, Água Limpa e Saneamento, Energia Acessível e Limpa, Trabalho Decente e Crescimento Econômico, Indústria, Inovação e Infraestrutura, Redução das Desigualdades, Cidades e Comunidades Sustentáveis, Consumo e Produção Responsáveis, Ação Contra a Mudança Global do Clima, Vida na Água, Vida Terrestre, Paz, Justiça e Instituições Eficazes, Parcerias e Meios de Implementação. Por exemplo, ao investir em tecnologias limpas e práticas de trabalho justo, as empresas não apenas melhoram sua própria sustentabilidade, mas também promovem o desenvolvimento sustentável de forma mais ampla..

### COP30

A 30ª Conferência das Partes da Convenção-Quadro das Nações Unidas sobre Mudança do Clima (COP30) é um evento de grande relevância tanto para o Brasil quanto para a comunidade internacional, marcado para ocorrer em Belém, Pará, em 2025. Esta conferência, a primeira a ser realizada na região Amazônica, destaca o papel crítico deste bioma na regulação climática global e na conservação da biodiversidade.

### Objetivos da COP30

**Promoção de Soluções Sustentáveis:** A COP30 tem como meta impulsionar a adoção de soluções sustentáveis e inovadoras frente aos desafios climáticos, com ênfase particular na conservação da Amazônia e na utilização sustentável de seus recursos.

**Redução do Desmatamento:** Um dos focos principais é intensificar as políticas e ações voltadas para a diminuição do desmatamento, uma das principais fontes de emissões de carbono na região.

**Transição Energética:** A conferência procura promover a transição para fontes de energia mais limpas e renováveis, essenciais para minimizar a dependência dos combustíveis fósseis e as consequentes emissões.

**Engajamento e Inclusão Social:** A COP30 valoriza o engajamento dos povos indígenas e das comunidades locais nas políticas e na gestão ambiental, reconhecendo sua importância vital na preservação dos ecossistemas.

**Fortalecimento da Resiliência Climática:** Um objetivo crucial é reforçar a resiliência dos ecossistemas e das comunidades humanas frente aos impactos das mudanças climáticas, especialmente nas áreas vulneráveis como a Amazônia.

### O envolvimento do Brasil

O Brasil, como anfitrião e protagonista ativo nas negociações climáticas, desempenha um papel central na COP30. A escolha do Brasil como sede reflete a importância global da Amazônia e a necessidade de compromisso internacional renovado para a sua proteção e uso sustentável. A liderança brasileira, já evidente na COP28, continuará a enfatizar a sustentabilidade ambiental e a redução das desigualdades sociais.

A significativa presença de autoridades brasileiras e o foco na Amazônia nas discussões anteriores reforçam o papel do Brasil como líder emergente em questões ambientais no cenário global. O sucesso da COP30 dependerá da capacidade do Brasil e de outros países em catalisar ações concretas e compromissos substanciais para enfrentar as mudanças climáticas.

A realização da conferência em Belém é uma oportunidade para o Brasil demonstrar ao mundo que é possível conciliar desenvolvimento econômico com conservação ambiental e justiça social. Espera-se que a COP30 seja um marco na luta global contra as mudanças climáticas, enfatizando o papel crucial da Amazônia e das políticas sustentáveis na mitigação de seus efeitos.

### A empresa

A TechFarm é uma empresa inovadora no setor de agricultura urbana. Com o objetivo de revolucionar a maneira como as fazendas urbanas operam, a TechFarm está na vanguarda da tecnologia agrícola.

A TechFarm reconhece a necessidade de um sistema de gerenciamento eficiente para otimizar suas operações. Por isso, está em processo de desenvolvimento de um software de gerenciamento abrangente. Este software não é apenas um aplicativo, mas um sistema completo que inclui uma interface de usuário intuitiva e um back-end robusto.

A interface do usuário, projetada com cuidado e atenção aos detalhes, apresentará várias telas que facilitam a interação do usuário com o sistema. Cada tela foi projetada para oferecer a melhor experiência ao usuário, permitindo que ele gerencie eficientemente as operações da fazenda.

No back-end, o software será alimentado por uma aplicação escrita em C# e um banco de dados SQL SERVER. Esta combinação permitirá que o software lide com uma grande quantidade de dados e execute várias tarefas simultaneamente, tornando o gerenciamento da fazenda uma tarefa simples e sem complicações.

Em resumo, a TechFarm está se esforçando para criar um software de gerenciamento que possa transformar a indústria da agricultura urbana. Com sua abordagem inovadora e foco na eficiência, a TechFarm está pronta para liderar o caminho na agricultura urbana.

### Economia e mercado

No contexto do desenvolvimento sustentável e da crescente demanda por alimentos produzidos localmente, o projeto de fazenda urbana em São Paulo representa uma inovação significativa no setor agrícola urbano. Este segmento de economia e mercado visa explorar a viabilidade econômica, a projeção de demanda, a competitividade no mercado local, a análise de custo-benefício e o retorno sobre investimento do projeto. Além disso, são discutidos o impacto social e as práticas de sustentabilidade que alinham o projeto com objetivos mais amplos de desenvolvimento sustentável. A análise econômica detalhada que segue oferece uma visão abrangente da estrutura financeira e do potencial de mercado deste empreendimento.

### Estimativa de viabilidade

O projeto de fazenda urbana em São Paulo apresenta uma proposta inovadora para integrar a agricultura sustentável no cenário urbano através de programas de assinatura de cestas de produtos. Considerando a crescente demanda por produtos orgânicos e de origem conhecida na região, a viabilidade do projeto é alta, com custos iniciais projetados em aproximadamente R$ 1.000.000,00. Esses custos cobrem infraestrutura, tecnologia de cultivo, contratação inicial e campanhas de marketing.

### Projeção de demanda

A Grande São Paulo, sendo um grande centro urbano, possui uma população crescente interessada em saúde e sustentabilidade. Espera-se que a demanda inicial atinja cerca de 500 assinaturas, com um crescimento esperado de 15% ao mês. A segmentação de mercado incluirá famílias e jovens profissionais que buscam ingredientes frescos e de alta qualidade.

### Análise competitiva

O mercado de fazendas urbanas em São Paulo é relativamente novo, com poucos competidores diretos, mas está em crescimento. O diferencial do nosso projeto reside no uso de tecnologias sustentáveis e na proximidade com o consumidor final, o que garante frescor inigualável e reduz significativamente a pegada de carbono dos produtos.

### Análise de custo benefício

Os benefícios do projeto ultrapassam os custos iniciais pela capacidade de alcançar um mercado em expansão para produtos sustentáveis e pela oportunidade de estabelecer a marca como líder em inovação agrícola urbana. A operação da fazenda é projetada para ser altamente eficiente, utilizando menos água e energia comparada a métodos tradicionais de agricultura, e maximizando a produção por metro quadrado (EcoInnovate, 2021).

### Cenário de retorno sobre investimento (ROI)

Com base no crescimento da assinatura, os rendimentos aumentam mensalmente. Supondo uma receita média de R$200 por assinatura, as receitas mensais começam em R$100.000 e crescem para aproximadamente R$465,800 ao final do 12º mês. O retorno sobre o investimento é projetado para ocorrer dentro de 12 a 15 meses após o início das operações.

### Impacto social e sustentabilidade

O projeto atende aos Objetivos de Desenvolvimento Sustentável (ODS) e segue rigorosamente os princípios de ESG, promovendo práticas que beneficiam o ambiente, a sociedade e a governança. O impacto social inclui a criação de empregos locais, educação sobre sustentabilidade alimentar e melhoria da qualidade de vida urbana através do acesso a alimentos mais saudáveis (UNDP, 2021).

### Conclusão da análise

A fazenda urbana proposta está bem posicionada para se tornar uma parte vital da infraestrutura alimentar de São Paulo, alinhando interesses econômicos com imperativos ambientais e sociais. A análise detalhada indica que o projeto não apenas é viável, mas tem o potencial de contribuir significativamente para os objetivos sociais e ambientais mais amplos.

### Regras de negócio

Independentes do setor empresarial, as regras de negócio desempenham um papel crucial ao estabelecer os princípios fundamentais que orientam o funcionamento de uma organização. Essas normativas estabelecem os fundamentos e as condições essenciais para alcançar os objetivos específicos da empresa, exercendo influência tanto nas operações diárias quanto nas decisões estratégicas.

Em sua essência, as regras de negócio representam especificações que abordam as características únicas do ambiente em que uma aplicação será utilizada. Em outras palavras, elas refletem as particularidades e as exigências específicas do contexto em que estão inseridas.

Um exemplo prático no setor financeiro ilustra o impacto significativo das regras de negócio. Em instituições financeiras, uma regra pode ser implementada para evitar que clientes realizem transferências que ultrapassem o saldo disponível em suas contas. Essa medida visa assegurar a integridade financeira da instituição, destacando como as regras de negócio desempenham um papel crucial na preservação da estabilidade e no funcionamento eficiente das organizações.

### Regras de Negócio na Programação

Na programação, as regras de negócio se manifestam a partir dos requisitos funcionais, os quais esboçam as funcionalidades específicas que o sistema deve oferecer para atender às demandas necessárias. Essas regras descrevem minuciosamente as operações e a lógica para alcançar os objetivos estabelecidos.

Tomemos como exemplo um sistema de gestão de estoque. Um requisito funcional poderia estipular que o estoque mínimo de um produto nunca deve ser ultrapassado. Essa condição é então descrita de forma detalhada em uma regra de negócios, informando as operações essenciais para alcançar o resultado esperado. A implementação destas regras no software é crucial para garantir a clareza, manutenibilidade e eficácia do sistema.

Essas especificações capturam elementos fundamentais relacionados ao setor, práticas e normas que regem o contexto de uso do software (Pressman, 2014). Essa integração efetiva entre requisitos funcionais e regras de negócio é essencial para o desenvolvimento de sistemas que atendam de maneira.

### Regras de Negócio da FarmTech

As tabelas a seguir apresentam as regras de negócio meticulosamente definidas para nosso sistema. Cada regra é descrita com detalhes para elucidar seu propósito e implementação dentro do contexto do projeto. Essas regras são elementos vitais para garantir que todas as funcionalidades do sistema não apenas operem conforme o esperado, mas também reflitam os objetivos estratégicos e operacionais da organização. A clareza e precisão na definição dessas regras são fundamentais para o sucesso do desenvolvimento e operação do sistema, assegurando que ele responda adequadamente às exigências do ambiente de negócios em que está inserido.

|  |  |
| --- | --- |
| Identificador | RN 0001 |
| Nome | Restrição de sistema |
| Módulo | Controle de Acesso |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração | 02/05/2024 |
| Autor da ultima alteração | FarmTech Innovations |
| Versão | 2.0 |
| Descrição | Esta regra de negócio impõe restrições de acesso no ato do cadastro, sendo eles: Admin, funcionário e cliente. O sistema verificará o tipo de cadastro para determinar a quais áreas e informações ele pode ter acesso, restringindo o acesso a informações sensíveis a funções apropriadas. |
| Justificativa | Essa restrição visa garantir que cada usuário tenha acesso apenas às informações e operações relevantes para suas responsabilidades, melhorando a segurança e a eficiência operacional. |

|  |  |
| --- | --- |
| Identificador | RN 0002 |
| Nome | Segurança de acesso |
| Módulo | Controle de Acesso |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração | 02/05/2024 |
| Autor da ultima alteração | FarmTech Innovations |
| Versão | 2.0 |
| Descrição | Esta regra de negócio estabelece protocolos de segurança para o acesso ao sistema, incluindo a autenticação de usuários através de senhas seguras. O sistema também deve monitorar e registrar tentativas de acesso para detectar e responder a atividades suspeitas de maneira proativa. |
| Justificativa | A segurança de acesso é essencial para proteger o sistema contra acessos não autorizados, garantindo a confidencialidade e a integridade dos dados. |

|  |  |
| --- | --- |
| Identificador | RN 0003 |
| Nome | Backup Regular |
| Módulo | Geral |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio define a política de backup do sistema, estabelecendo a frequência, os métodos e os locais de armazenamento dos backups. Os backups devem ser realizados diariamente de forma automatizada, com cópias de segurança armazenadas em locais físicos seguros e em nuvem para garantir redundância. Além disso, os backups devem ser testados regularmente para garantir sua integridade e a eficácia do processo de recuperação de dados. |
| Justificativa | A realização regular de backups é crucial para minimizar a perda de dados em caso de falhas técnicas, incidentes ou desastres, assegurando a continuidade operacional. |

|  |  |
| --- | --- |
| Identificador | RN 0004 |
| Nome | Auditoria de ações do usuário |
| Módulo | Geral |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio estabelece a obrigatoriedade de um sistema de auditoria que registra todas as ações dos usuários dentro do sistema, incluindo logins, alterações de dados, operações de consulta e exclusões. Os registros de auditoria devem incluir a identificação do usuário, o tipo de ação, a data e hora, e o contexto da ação realizada. Estes registros devem ser armazenados de forma segura e inalterável para fins de conformidade e segurança. |
| Justificativa | A auditoria proporciona transparência nas atividades do sistema, facilitando a identificação de comportamentos inadequados, rastreando alterações e auxiliando em investigações de segurança. |

|  |  |
| --- | --- |
| Identificador | RN 0005 |
| Nome | Expiração de Sessão |
| Módulo | Geral |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio define que as sessões dos usuários no sistema devem automaticamente expirar após um período de inatividade especificado. O sistema deve desconectar o usuário automaticamente após o tempo definido sem interações, requerendo nova autenticação para acessar novamente o sistema. Este tempo de inatividade deve ser configurável conforme as políticas de segurança da organização. |
| Justificativa | A expiração de sessão contribui para a segurança, especialmente em ambientes compartilhados, evitando acessos não autorizados em caso de negligência do usuário em encerrar a sessão. |

|  |  |
| --- | --- |
| Identificador | RN 0006 |
| Nome | Recuperação de Senha |
| Módulo | Segurança de Acesso |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração | 02/05/2024 |
| Autor da ultima alteração | FarmTech Innovations |
| Versão | 2.0 |
| Descrição | Esta regra define o procedimento para a recuperação de senha dos usuários. O sistema deve fornecer uma opção de "Esqueceu a senha?" na tela de login. Ao selecionar esta opção, o usuário deve inserir seu CPF para receber um link de redefinição de senha por e-mail. O link deve expirar em 24 horas para garantir segurança. O usuário será direcionado a uma página segura onde poderá inserir uma nova senha. |
| Justificativa | Garante a conformidade com regulamentações de segurança alimentar, evitando a inclusão de ingredientes inadequados nos produtos. |

|  |  |
| --- | --- |
| Identificador | RN 0007 |
| Nome | Controle de validade |
| Módulo | Produtos |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio estabelece que o sistema deve monitorar e gerenciar a validade de todos os produtos em estoque. O sistema automaticamente alertará os responsáveis quando produtos estiverem próximos à data de expiração, permitindo ações proativas para sua utilização ou descarte adequado. Além disso, o sistema deve priorizar a venda ou uso de produtos com datas de validade mais próximas, para garantir a rotatividade eficiente do estoque. |
| Justificativa | Minimiza o desperdício ao assegurar que produtos próximos ao vencimento sejam identificados e tratados adequadamente. |

|  |  |
| --- | --- |
| Identificador | RN 0008 |
| Nome | Controle de estoque mínimo |
| Módulo | Produtos |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio define que o sistema deve monitorar continuamente o nível de estoque de cada produto e compará-lo com um limiar predefinido de estoque mínimo. Quando o estoque de um produto cai abaixo desse limiar, o sistema deve notificar automaticamente o gerente de compras para iniciar o processo de reabastecimento, garantindo a disponibilidade contínua do produto sem interrupção. |
| Justificativa | Assegura que a empresa mantenha um estoque suficiente para atender à demanda, evitando a indisponibilidade de itens populares. |

|  |  |
| --- | --- |
| Identificador | RN 0009 |
| Nome | Calculo de valores de produtos |
| Módulo | Produtos |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio especifica que o sistema deve calcular o preço de venda de cada produto com base em fatores como custo de produção, margem de lucro desejada e condições de mercado. O sistema deve incluir uma funcionalidade que permite ajustar automaticamente os preços em resposta a variações nos custos de insumos ou mudanças estratégicas de mercado. |
| Justificativa | Proporciona transparência e consistência no estabelecimento de preços, contribuindo para a rentabilidade da empresa. |

|  |  |
| --- | --- |
| Identificador | RN 0010 |
| Nome | Forma de pagamento |
| Módulo | Venda |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio estabelece que o sistema deve suportar diversas formas de pagamento, incluindo PIX, boleto, cartão de crédito, cartão de débito, e pagamentos digitais, como carteiras eletrônicas. O sistema deve assegurar a integração segura com gateways de pagamento e bancos para processar todas as transações de forma segura e eficiente. |
| Justificativa | Oferece flexibilidade aos clientes e garante a segurança nas transações financeiras. |

|  |  |
| --- | --- |
| Identificador | RN 0011 |
| Nome | Calculo de frete e entrega |
| Módulo | Venda |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio determina que o sistema deve calcular automaticamente os custos de frete baseados no endereço de entrega do cliente e nas tarifas dos serviços de entrega disponíveis. O sistema deve ser capaz de apresentar diferentes opções de entrega, incluindo prazos estimados e custos, permitindo aos clientes escolher a opção que melhor atenda suas necessidades. |
| Justificava | Fornece informações transparentes aos clientes sobre os custos associados à entrega e ajuda na gestão logística da empresa. |

|  |  |
| --- | --- |
| Identificador | RN 0012 |
| Nome | Restrição de venda por região |
| Módulo | Venda |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio estabelece que o sistema deve verificar o local de entrega informado pelo cliente e determinar se a venda de certos produtos é permitida naquela região. O sistema deve bloquear automaticamente a finalização da compra de itens que não podem ser comercializados em áreas específicas devido a restrições legais ou políticas de distribuição da empresa. |
| Justificativa | Atende a regulamentações locais e estratégias de mercado, garantindo conformidade e segurança nas transações. |

|  |  |
| --- | --- |
| Identificador | RN 0013 |
| Nome | Tempo de entrega |
| Módulo | Venda |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio especifica que o sistema deve calcular e apresentar o tempo estimado de entrega para o cliente durante o processo de finalização da compra. O cálculo deve considerar a localização de entrega, a disponibilidade dos produtos no estoque e a logística de transporte. O sistema também deve atualizar os clientes sobre quaisquer mudanças ou atrasos no tempo estimado de entrega assim que eles forem identificados. |
| Justificativa | Gerencia as expectativas dos clientes, fornecendo informações claras sobre quando podem esperar a entrega dos produtos. |

|  |  |
| --- | --- |
| Identificador | RN 0014 |
| Nome | Cadastro de clientes |
| Módulo | Venda |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio estabelece que o sistema deve permitir o cadastro de clientes, solicitando informações essenciais como nome, endereço, telefone, e-mail e, se aplicável, preferências de produtos. O cadastro deve ser feito de maneira segura e em conformidade com a LGPD, garantindo a proteção dos dados pessoais dos clientes. |
| Justificativa | Facilita a personalização do atendimento, possibilita a fidelização e contribui para a análise de perfil do cliente. |

|  |  |
| --- | --- |
| Identificador | RN 0015 |
| Nome | LGPD |
| Módulo | Venda |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio assegura que todas as operações de coleta, armazenamento, tratamento e compartilhamento de dados pessoais de clientes, funcionários e parceiros estejam em conformidade com a Lei Geral de Proteção de Dados Pessoais (LGPD). O sistema deve implementar medidas técnicas e administrativas adequadas para proteger os dados contra acessos não autorizados, perdas ou divulgações indevidas. |
| Justificativa | Atende às normas legais de proteção de dados, fortalecendo a confiança dos clientes e evitando possíveis sanções legais. |

|  |  |
| --- | --- |
| Identificador | RN 0016 |
| Nome | Regularização de salário por cargo |
| Módulo | Recursos Humanos |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra estabelece que os salários dos funcionários devem estar alinhados com os padrões de mercado para cada cargo dentro da organização. Os salários são revisados anualmente com base em pesquisas salariais do mercado e ajustados conforme necessário para garantir competitividade e equidade. Ajustes salariais são realizados considerando a performance individual, a inflação anual e as práticas do setor. |
| Justificativa | Manter a competitividade dos salários é crucial para reter talentos e garantir a satisfação dos colaboradores, além de manter a empresa em conformidade com as práticas de mercado e legislação trabalhista. |

|  |  |
| --- | --- |
| Identificador | RN 0017 |
| Nome | Controle de pagamento de fornecedores |
| Módulo | Financeiro |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio garante que a remuneração dos funcionários esteja em conformidade com os padrões competitivos de mercado para cada cargo específico. Os salários são revisados anualmente, baseando-se em pesquisas de mercado atualizadas, para realizar ajustes necessários que refletem a inflação, as mudanças no custo de vida, a performance individual dos funcionários e as práticas correntes do setor. |
| Justificativa | Assegura a pontualidade nos pagamentos, fortalecendo relacionamentos com fornecedores e evitando possíveis penalidades. |

|  |  |
| --- | --- |
| Identificador | RN 0018 |
| Nome | Monitoramento de fluxo de caixa |
| Módulo | Financeiro |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio define a obrigatoriedade de um monitoramento contínuo e detalhado do fluxo de caixa da empresa. O processo inclui o registro sistemático de todas as entradas e saídas financeiras, permitindo análises periódicas da liquidez, investimentos, despesas e receitas. O sistema deve ser capaz de gerar relatórios periódicos que ajudem na visualização do estado atual e projeções futuras do fluxo de caixa. |
| Justificativa | Facilita a tomada de decisões financeiras, garantindo a saúde financeira da empresa e antecipando possíveis desafios. |

|  |  |
| --- | --- |
| Identificador | RN 0019 |
| Nome | Complience fiscal |
| Módulo | Financeiro |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio estabelece a obrigatoriedade de monitoramento contínuo e detalhado do fluxo de caixa da empresa. O processo envolve o registro sistemático de todas as entradas e saídas financeiras, o que facilita análises periódicas importantes para a gestão financeira. O sistema é responsável por gerar relatórios regulares que proporcionam uma visão clara do estado atual do fluxo de caixa e ajudam nas projeções de futuro, permitindo uma melhor visibilidade e controle sobre a saúde financeira da empresa. |
| Justificativa | Evita penalidades legais, mantendo a empresa em conformidade com as leis fiscais e regulamentações governamentais. |

|  |  |
| --- | --- |
| Identificador | RN 0020 |
| Nome | Controle de despesas fixas |
| Módulo | Financeiro |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio determina a implementação de um sistema para o controle rigoroso das despesas fixas da empresa. O sistema deve catalogar todas as despesas recorrentes, como aluguéis, salários, serviços de utilidade pública, e manutenção regular. Deve também fornecer funcionalidades para análise e comparação de custos ao longo do tempo, identificando oportunidades para otimização de gastos e potenciais reduções de custos sem comprometer a operação da empresa. |
| Justificativa | Permite uma gestão eficiente dos custos fixos, contribuindo para a estabilidade financeira e a definição de estratégias de redução de despesas. |

|  |  |
| --- | --- |
| Identificador | RN 0021 |
| Nome | Controle de pagamento jurídico |
| Módulo | Financeiro |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio estabelece o procedimento para a gestão e controle de todos os pagamentos relacionados a questões jurídicas da empresa. O sistema deve permitir o registro e acompanhamento detalhado de todos os pagamentos realizados a advogados, consultores jurídicos, taxas de processos, e qualquer outro custo legal. O sistema também deverá manter um registro histórico para referência futura e preparação para auditorias, garantindo que todos os pagamentos sejam feitos de acordo com os contratos estabelecidos e as leis aplicáveis. |
| Justificativa | Assegura o cumprimento de obrigações legais, proporcionando transparência e organização nas relações jurídicas da empresa. |

|  |  |
| --- | --- |
| Identificador | RN 0022 |
| Nome | Jornada de trabalho |
| Módulo | Financeiro |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio especifica as diretrizes para a jornada de trabalho de todos os empregados da empresa, conforme estabelecido pelas leis trabalhistas locais e nacionais. O sistema deve ser capaz de registrar, monitorar e relatar as horas trabalhadas, horas extras, feriados e quaisquer desvios das normas padrão de jornada de trabalho. A funcionalidade deverá incluir alertas automáticos quando os limites de horas trabalhadas forem excedidos, garantindo que a empresa permaneça em conformidade com a legislação e ajudando a prevenir o excesso de trabalho e o desgaste dos funcionários. |
| Justificativa | Assegura o cumprimento das leis trabalhistas, promove o equilíbrio entre vida profissional e pessoal, além de otimizar a produtividade da equipe. |

|  |  |
| --- | --- |
| Identificador | RN 0023 |
| Nome | Política de férias |
| Módulo | Financeiro |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra de negócio estabelece o processo de planejamento, aprovação e registro das férias de todos os funcionários. O sistema deve permitir aos funcionários solicitar períodos de férias e aos gestores aprová-las com base na disponibilidade, necessidades operacionais e políticas internas. O sistema também deve assegurar que todas as férias sejam registradas e monitoradas adequadamente, permitindo uma visão clara das ausências planejadas para facilitar o planejamento de recursos e evitar a sobrecarga de trabalho durante períodos de ausência significativa de funcionários. |
| Justificativa | Garante a equidade na distribuição das férias, evitando impactos negativos na operação da empresa e proporcionando períodos de descanso adequados. |

|  |  |
| --- | --- |
| Identificador | RN 0024 |
| Nome | Avaliação de desempenho |
| Módulo | Financeiro |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra define a implementação e gerenciamento de um sistema de avaliação de desempenho para todos os funcionários. O processo inclui a definição de critérios claros e mensuráveis, baseados nas responsabilidades de cada cargo e nos objetivos gerais da empresa. O sistema deve permitir avaliações regulares que envolvem autoavaliação, avaliação por superiores, e, quando aplicável, avaliação por pares. O sistema também deve integrar mecanismos para feedback construtivo e estabelecimento de planos de desenvolvimento individual, visando à melhoria contínua do desempenho e ao alinhamento das expectativas entre funcionários e gestão. |
| Justificativa | Contribui para o desenvolvimento profissional, identificação de talentos, promoções e aprimoramento contínuo da equipe. |

|  |  |
| --- | --- |
| Identificador | RN 0025 |
| Nome | Recrutamento de seleção |
| Módulo | Recursos humanos |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração | 02/05/2024 |
| Autor da ultima alteração | FarmTech Innovations |
| Versão | 2.0 |
| Descrição | Esta regra estabelece procedimentos padrão para o recrutamento e seleção de novos funcionários. O processo inclui a divulgação de vagas, triagem inicial de candidatos, entrevistas, avaliações técnicas e comportamentais, e verificações de referências. O sistema deve apoiar a coleta e análise de informações sobre candidatos de forma eficiente e em conformidade com a legislação vigente sobre proteção de dados. Além disso, o processo de seleção deve ser estruturado para garantir a imparcialidade e a objetividade, priorizando competências e alinhamento cultural com a empresa. |
| Justificativa | Assegura a contratação de profissionais alinhados aos valores e necessidades da empresa, otimizando o processo de admissão. |

|  |  |
| --- | --- |
| Identificador | RN 0026 |
| Nome | Treinamento e desenvolvimento |
| Módulo | Recursos humanos |
| Data | 10/03/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração | 02/05/2024 |
| Autor da ultima alteração | FarmTech Innovations |
| Versão | 1.0 |
| Descrição | Esta regra estabelece a obrigação de implementar e manter um programa sistemático de treinamento e desenvolvimento para todos os funcionários. O programa incluirá uma variedade de atividades de capacitação, como workshops, cursos online, treinamentos internos e externos, e programas de mentoria. O sistema deve permitir o agendamento, o registro e a avaliação de todas as atividades de treinamento, bem como a monitorização da progressão e do impacto dessas atividades no desempenho dos funcionários e na consecução dos objetivos da empresa. |
| Justificativa | Promove a atualização constante das habilidades da equipe, impulsionando a inovação, satisfação no trabalho e retenção de talentos. |

|  |  |
| --- | --- |
| Identificador | RN 0027 |
| Nome | Padronização de Login |
| Módulo | Segurança de Acesso |
| Data | 02/05/2024 |
| Autor | FarmTech Innovations |
| Data da última alteração |  |
| Autor da ultima alteração |  |
| Versão | 1.0 |
| Descrição | Esta regra determina que o CPF do usuário será utilizado como identificador único para o processo de login no sistema. Todos os usuários devem usar seu CPF registrado, sem pontos ou traços, como nome de usuário para acessar o sistema. |
| Justificativa | Utilizar o CPF como login padroniza o processo de autenticação e simplifica a gestão de identidades dentro do sistema. Essa abordagem reduz a possibilidade de duplicidade de logins e facilita a integração com outros sistemas administrativos que utilizam o CPF como identificador primário. |

### Requisitos dos usuários

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| **NOME** | **CLASSIFICAÇÃO** | **DESCRIÇÃO** | **ESPECIFICAÇÃO** |
|  |  |  |  |
| Gerenciar Produtos | FUNCIONAL | Os usuários devem ser capazes de adicionar, editar, remover e consultar produtos no sistema. | NÃO SE APLICA |
|  |  |  |  |
| Gerenciar Fornecedores | FUNCIONAL | Os usuários devem ser capazes de adicionar, editar, remover e consultar fornecedores no sistema. | NÃO SE APLICA |
|  |  |  |  |
| Gerar Relatórios | FUNCIONAL | Os usuários devem ser capazes de gerar relatórios detalhados sobre vendas, estoque, fornecedores e produtos. | NÃO SE APLICA |
|  |  |  |  |
| Facilidade de Uso | NÃO FUNCIONAL | O software deve ser intuitivo e fácil de usar, mesmo para usuários com pouca experiência em tecnologia. | USABILIDADE |
|  |  |  |  |
| Suporte Multicanal | NÃO FUNCIONAL | Os usuários devem ser capazes de acessar o sistema através de vários canais, como um aplicativo de desktop, um aplicativo móvel ou um navegador da web. | USABILIDADE |
|  |  |  |  |
| Privacidade | NÃO FUNCIONAL | O software deve garantir a privacidade dos usuários, protegendo suas informações pessoais e garantindo que elas sejam usadas apenas para os fins pretendidos | SEGURANÇA |

### Requisitos de sistema

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| **NOME** | **CLASSIFICAÇÃO** | **DESCRIÇÃO** | **ESPECIFICAÇÃO** |
|  |  |  |  |
| Gerenciar Produtos | FUNCIONAL | Os usuários devem ser capazes de adicionar, editar, remover e consultar produtos no sistema. | NÃO SE APLICA |
|  |  |  |  |
| Gerenciar Fornecedores | FUNCIONAL | Os usuários devem ser capazes de adicionar, editar, remover e consultar fornecedores no sistema. | NÃO SE APLICA |
|  |  |  |  |
| Gerar Relatórios | FUNCIONAL | Os usuários devem ser capazes de gerar relatórios detalhados sobre vendas, estoque, fornecedores e produtos. | NÃO SE APLICA |
|  |  |  |  |
| Facilidade de Uso | NÃO FUNCIONAL | O software deve ser intuitivo e fácil de usar, mesmo para usuários com pouca experiência em tecnologia. | USABILIDADE |
|  |  |  |  |
| Suporte Multicanal | NÃO FUNCIONAL | Os usuários devem ser capazes de acessar o sistema através de vários canais, como um aplicativo de desktop, um aplicativo móvel ou um navegador da web. | USABILIDADE |
|  |  |  |  |
| Privacidade | NÃO FUNCIONAL | O software deve garantir a privacidade dos usuários, protegendo suas informações pessoais e garantindo que elas sejam usadas apenas para os fins pretendidos | SEGURANÇA |

### Desenvolvimento da interface de usuário

O desenvolvimento da interface de usuário, deve, assim como as fases de desenvolvimento do sistema, satisfazer plenamente todas as necessidades do cliente, e para isso, utilizamos a filosofia do Design Thinking. Essa filosofia consiste na abordagem centrada no ser humano para a inovação, utilizando de técnicas do design para melhorar a vida das pessoas, não utilizando de uma fórmula específica para a sua realização, mas sim de ideias que surgem de colaboradores dentro do projeto e que podem ser utilizadas para a realização do objetivo.

Simplificando, é uma disciplina que utiliza a sensibilidade e os métodos do designer para alinhar as necessidades das pessoas com o que é tecnologicamente viável e o que uma estratégia de negócios viável pode transformar em valor para o cliente e em oportunidades de mercado (BROWN, 2008).

### Fases do Design Thinking

As fases do Design Thinking consistem em etapas onde iremos realizar todo o processo de Criação da interface, desde o entendimento do problema do cliente, até a sua fase final de testes de interface.

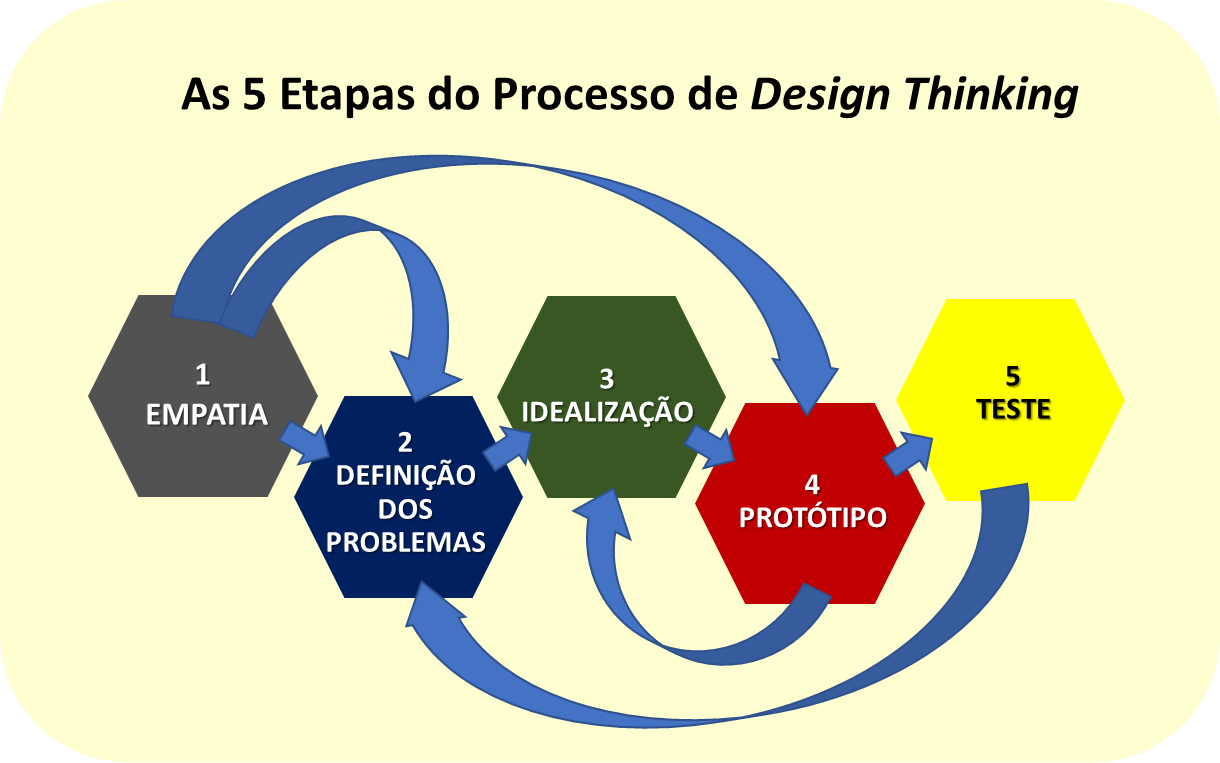


Figura 1 - Fases do Design Thinking

### Empatia

Durante a fase de empatia, nosso objetivo é compreender quais as angústias e necessidades do cliente, para isso, devemos entrar na sua rotina, realizando pesquisas, questionários, observando o dia-a-dia e entendendo em que parte podemos ajudar a melhorar a realização de tarefas dentro da empresa.

Nosso cliente, a empresa Farmtech, uma startup do ramo do agronegócio, possui ideais nobres com relação a boa alimentação da população brasileira, bem como a preservação do ecossistema do País em que vivemos. Por conta disso, conseguimos entender o motivo do pedido para desenvolvimento do sistema de gerenciamento, tendo em vista que não existem muitos softwares similares no mercado, e devido a localização da empresa, a necessidade de um sistema próprio para um melhor gerenciamento de sua fazenda urbana, se faz tão importante quanto imaginamos.

### Definição dos problemas

Após análises minuciosas sobre as necessidades da FarmTech, ficou claro que os problemas estavam na disponibilidade de softwares específicos para realizar o gerenciamento da fazenda. Os softwares disponíveis já estão ultrapassados, principalmente em relação a sua interface, devemos levar em consideração as novas gerações de pessoas que estão entrando e ainda vão entrar no mercado de trabalho, e que possuem a interação humano computador (IHC) de maneira bem mais prática e intuitiva em comparação a gerações que já estão no mercado de trabalho. Entre as pessoas que ainda entrarão no mercado de trabalho, estão os já conhecidos Next Billion Users (Próximo Bilhão de Usuários), pessoas essas que ainda vão conhecer o uso real da internet, e que, em sua maioria, irão utilizar apenas um dispositivo móvel e de baixo custo. Com base neste tema, as interfaces de usuário devem se adaptar a esse futuro próximo, facilitando o uso de sistemas, aplicando conceitos claros de usabilidade e padrões de uso, para melhor adaptação de pessoas que recém chegaram ao mundo digital.

### Idealização

Durante a fase de idealização, surgem as ideias e conceitos para se utilizar nas interfaces. Nessa parte do processo, todos os membros da equipe podem e são incentivados a expressarem suas opiniões e pensamentos sobre como desenvolver o produto, nada deverá ser descartado, tudo poderá ser utilizado, e para isso, devemos analisar as possibilidades.

Abaixo temos algumas das ideias apresentadas pelos membros envolvidos no projeto:

**Funcionalidades:**

* Cadastro de produtos:
  + Nome do produto
  + Descrição
  + Tipo de planta (hortaliça, erva, fruta, etc.)
  + Variedade
  + Quantidade em estoque
  + Fornecedor (opcional)
  + Data de plantio
  + Data de colheita estimada
  + Foto (opcional)
* Exclusão de produtos:
  + Busca por nome, tipo, variedade ou data de plantio
  + Confirmação antes da exclusão
* Atualização de informações de produtos:
  + Qualquer campo do cadastro de produtos
  + Histórico de alterações

**Interface:**

* Design intuitivo e fácil de usar:
  + Layout organizado e limpo
  + Ícones e menus claros
  + Linguagem simples e direta
* Visualização de dados:
  + Gráficos e tabelas para acompanhar o crescimento das plantas
  + Filtros e opções de personalização
* Relatórios:
  + Geração de relatórios sobre estoque, colheitas e vendas
  + Exportação para Excel ou PDF

**Outras funcionalidades:**

* Controle de estoque:
  + Monitoramento do nível de estoque de cada produto
  + Geração de alertas para produtos com baixo estoque
* Gerenciamento de tarefas:
  + Atribuição de tarefas aos funcionários (regar, colher, etc.)
  + Acompanhamento do progresso das tarefas
* Integração com outros sistemas:
  + Ponto de venda (PDV)
  + Sistema de contabilidade

Como já visto anteriormente, várias dessas ideias foram utilizadas ao longo do projeto, mostrando que, apesar da variedade dos membros de uma equipe, pessoas com pensamentos diferentes podem ter visões muito interessantes sobre certo assunto, e que, se utilizados de maneira correta, podem criar grandes projetos, e isso se aplica também na interface de usuário.

### Prototipação

Após a fase de idealização, devemos filtrar algumas ideias que foram propostas, o que é algo natural, afinal, devemos sempre levar em consideração a necessidade do usuário, além claro, de respeitar conceitos de usabilidade e acessibilidade do sistema.

Logo em seguida, começamos a aplicar os conceitos aproveitáveis em telas realizadas de maneira simples e com baixo comprometimento ao resultado final, esse é o conhecido protótipo de baixa fidelidade.

Segundo Tatiane (2021) “Um protótipo de baixa fidelidade é bastante utilizado em fases iniciais e exploratórias de um projeto para validar um conceito e decidir se uma ideia tem ou não valor funcional. Devem ser rápidos, rudimentares e baratos.”

As primeiras telas criadas foram utilizadas para demonstrar as funcionalidades mais básicas do sistema, como o login do usuário, exibição, cadastro e edição de produtos, com a maioria dos campos de uso já definidos.

A partir deste momento, serão apresentados os conceitos iniciais do design, utilizando conceitos já conhecidos na internet.

A primeira tela (Figura 2) demonstra a primeira interação que o usuário terá na utilização do sistema, que é o acesso ao mesmo. De maneira bem simples, o usuário terá que colocar um login e uma senha, e logo em seguida clicar em entrar para ser realizada a validação das informações (o maior detalhamento do processo estará no manual de uso do sistema).

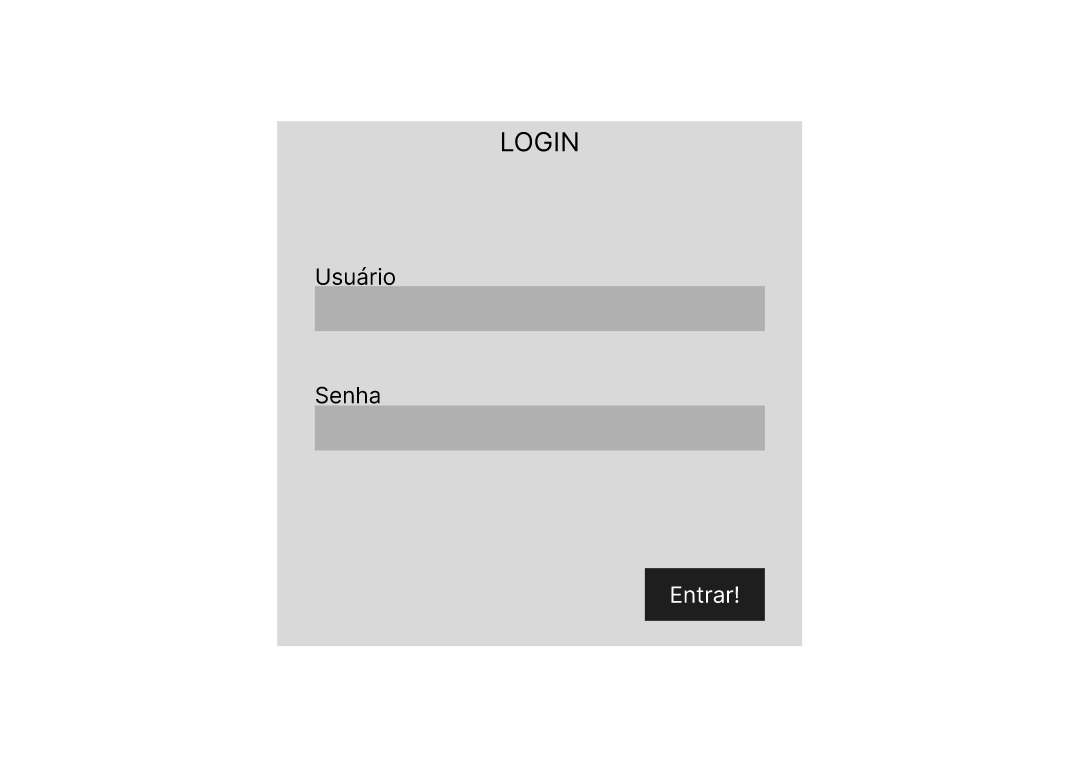
.

Figura 2 - Protótipo de baixa fidelidade de tela de login

Na Figura a seguir (Figura 3), vemos o um menu lateral, contendo as principais abas que o sistema terá, bem como a identidade gráfica da tabela, contendo informações necessárias no entendimento do cliente. Além disso, temos também a opção de cadastro, que nesta tela, é o cadastro de produtos, e uma pequena barra de pesquisa, onde será possível pesquisar informações específicas do item em exibição na tela, para uso mais prático e uma maior facilitação de pesquisa dos itens.

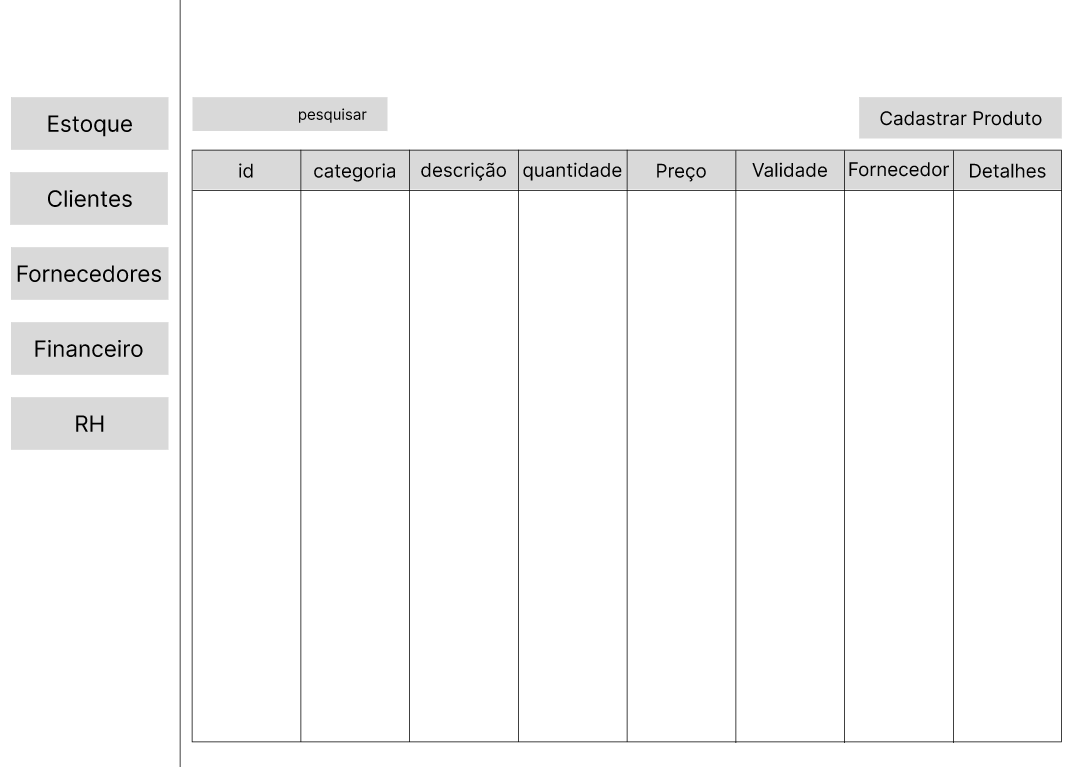


Figura 3 - Apresentação visual das informações cadastradas no banco de dados

Na Figura 4, temos um formulário, que surge como um modal, após acionar a função de cadastro. Essa ideia surgiu a partir de tendências de mercado atuais, priorizando um maior aproveitamento de tela, afinal, levando em conta o número relativamente baixo de informações necessárias para o cadastro, não compensaria dedicar uma tela inteira para funções básicas.

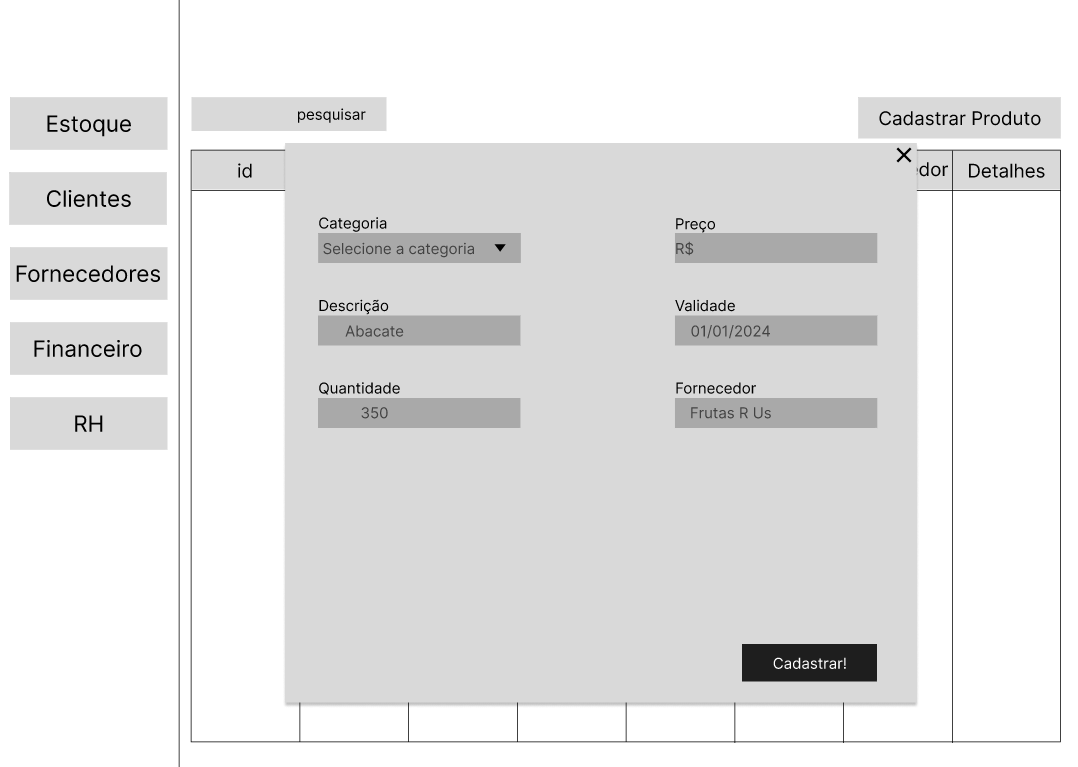


Figura 4 - formulário para cadastro

### Responsividade

Esse conceito surge a partir do aumento do uso de dispositivos móveis, com dimensões de telas diferentes em comparação ao padrão utilizados nos desktops, gerando dificuldades e desconforto ao usuário, que tinha que se adaptar ao uso de uma tela com proporções completamente distintas ao que tinha sido originalmente projetada. A responsividade é a capacidade um site ou sistema de se adaptar a diferentes tamanhos de tela, gerando proporcionalidade e conforto ao usuário, sem perder nenhum tipo de informação na troca de telas.

Com o aumento expressivo de pessoas que utilizam apenas dispositivos móveis para acesso a internet, nosso sistema também teve que se adaptar a essa realidade, e com a exigência de nosso cliente, desenvolvemos uma interface própria para ser utilizada em um aplicativo mobile, contendo todas as funções que o site e o aplicativo desktop terão em sua versão final.

Acompanhe abaixo os protótipos de baixa fidelidade da interface mobile, mostrando as mesmas funcionalidades do protótipo de desktop:



Figura 5 - tela de login mobile



Figura 6 - tela de estoque mobile

Para uma melhor compreensão do usuário, redimensionamos vários aspectos da tela usada nos computadores, incluindo o menu lateral, que virou um “menu hamburguer”, que é muito utilizado em dispositivos móveis para esconder e mostrar funções apenas quando o usuário tiver interesse, poupando espaço de tela para mostrar informações mais importantes de acordo com o uso.

### Identidade visual

A identidade visual de uma empresa é o seu cartão de visitas, aquilo que representa com fidelidade sua história e seus valores, bem como sua influência na sociedade, e possuímos exemplos disso espalhados por todo o mundo, como os arcos dourados do Mcdonald’s, a maçã da Apple, ou até mesmo um simples F branco em um fundo azul, da rede social Facebook.

Nosso cliente também possui uma boa identidade visual, afinal, uma fazenda urbana tem por características a preservação do meio ambiente por meio de métodos melhorados de cultivo de seus frutos. A Farmtech possui uma logomarca muito interessante, se utilizando de sua origem, a fazenda, bem como o extenso uso de azul, para representar o apreço pela água, um dos bens mais preciosos para a humanidade.

Nosso cliente também possui uma boa identidade visual, afinal, uma fazenda urbana tem por características a preservação do meio ambiente por meio de métodos melhorados de cultivo de seus frutos. A Farmtech possui uma logomarca muito interessante, se utilizando de sua origem, a fazenda, bem como o extenso uso de azul, para representar o apreço pela água, um dos bens mais preciosos para a humanidade.



Figura 7 - Logomarca Farmtech

### Protótipo de alta fidelidade

Depois de definidos os conceitos de baixa fidelidade, bem como a identidade visual do sistema, agora é chegada a hora de mostrar a interface de usuário com o seu padrão muito próximo ao produto final, com funções já bem definidas e padrões estruturados e alinhados com o cliente e com a equipe de desenvolvimento do software.

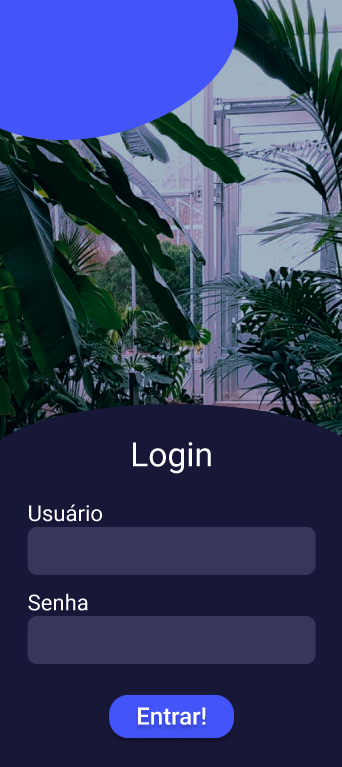
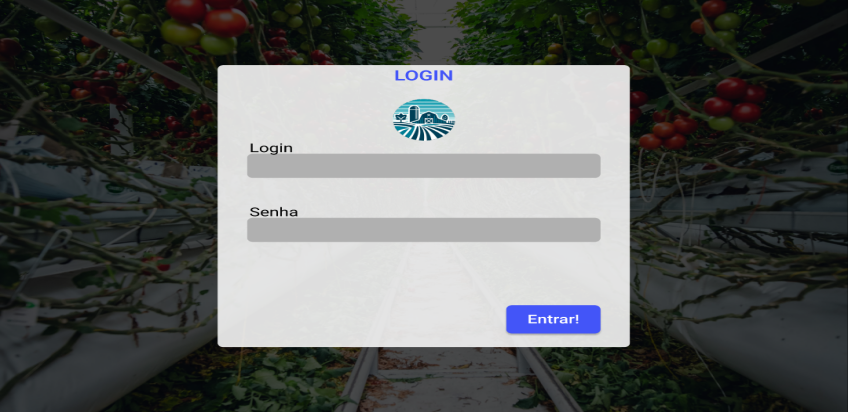


Figura 9 - Tela de Login de Alta Fidelidade

Figura 8 - Tela Login Mobile em Alta Fidelidade

A tela final apresenta as mesmas funções já apresentadas no protótipo de baixa fidelidade, porém com as alterações necessárias para melhor compreensão do usuário, além de uma identidade visual fidedigna aos valores da empresa, com cores representativas e uma Figura de fundo mostrando a riqueza do projeto de uma fazenda urbana.

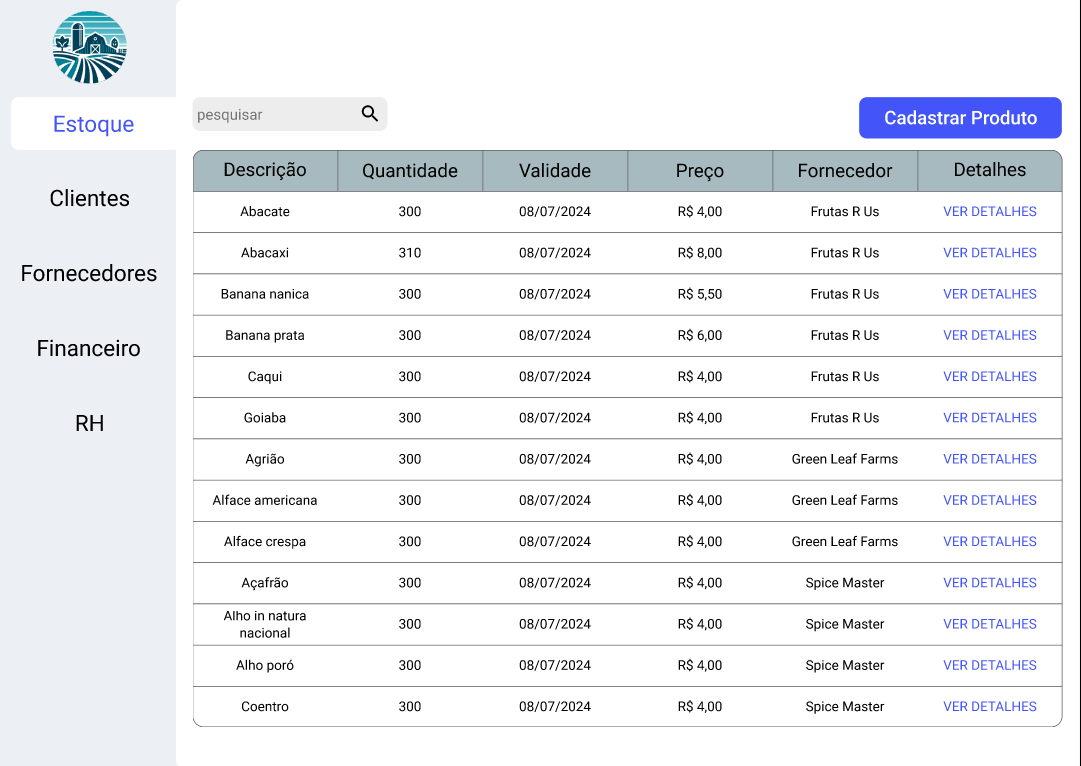


Figura 10 - Tela de Estoque de Alta Fidelidade

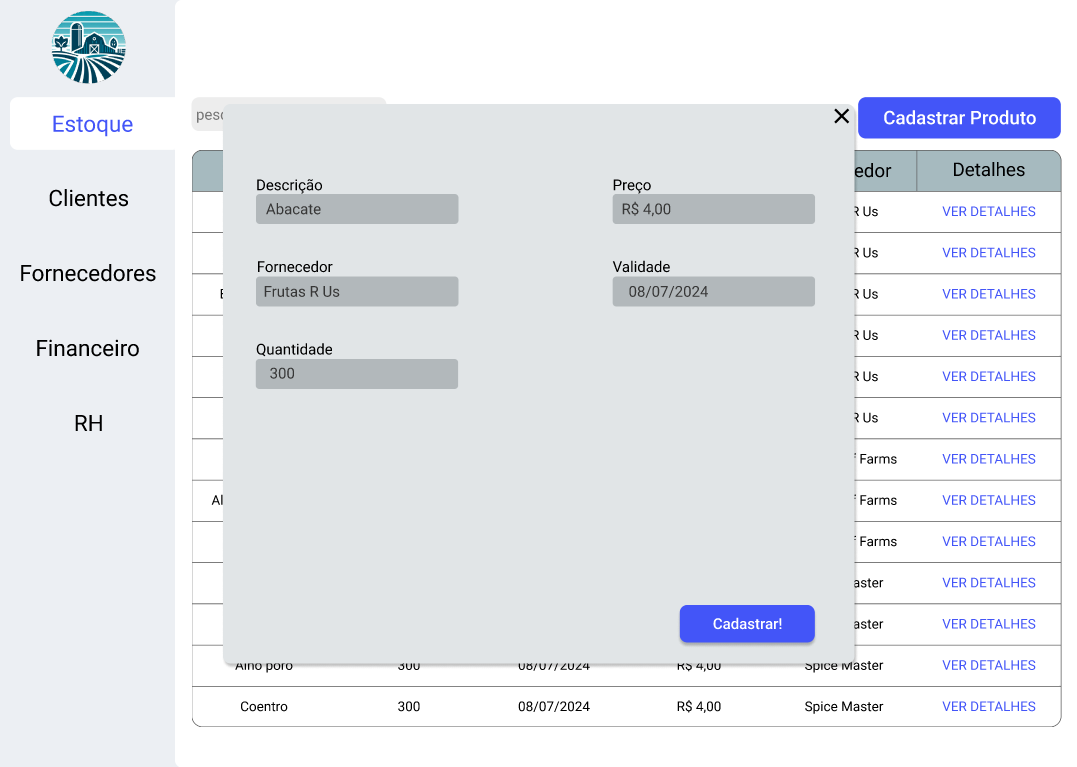


Figura 11 - Tela de Cadastro de Produtos em Alta Fidelidade

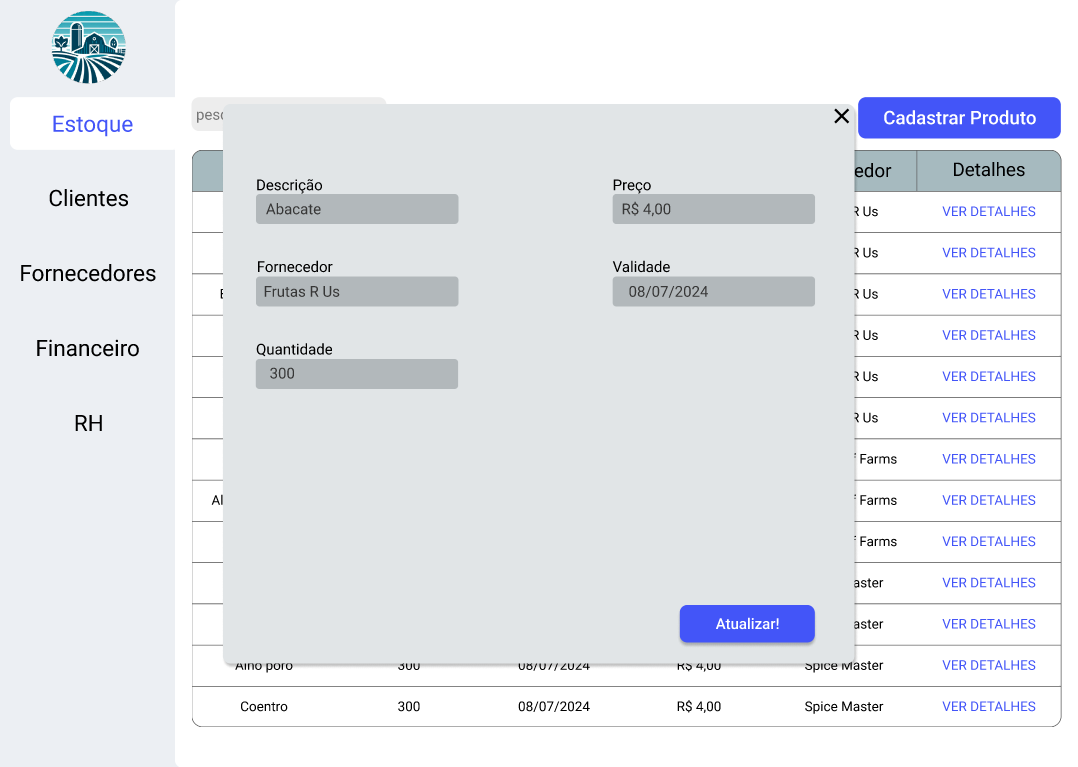


Figura 12 - Tela de Detalhes e Edição de Produtos em Alta Fidelidade

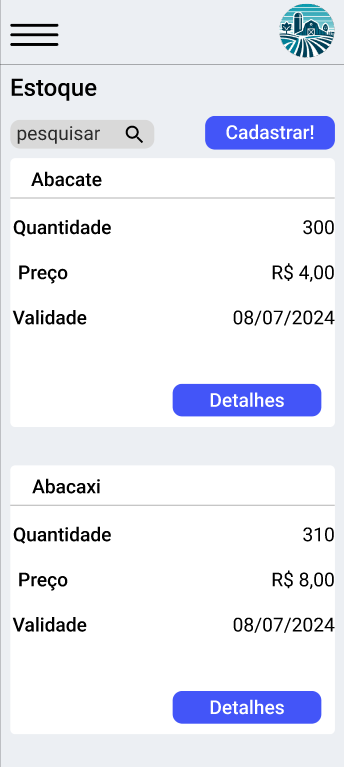


Figura 13 - Menu Mobile Alta Fidelidade Figura 14 - Tela Estoque Mobile Alta Fidelidade

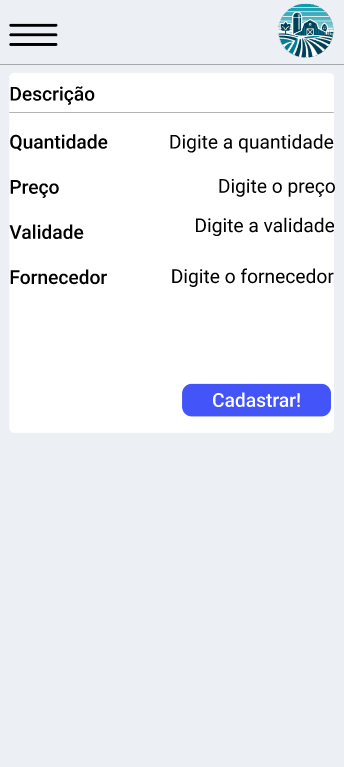
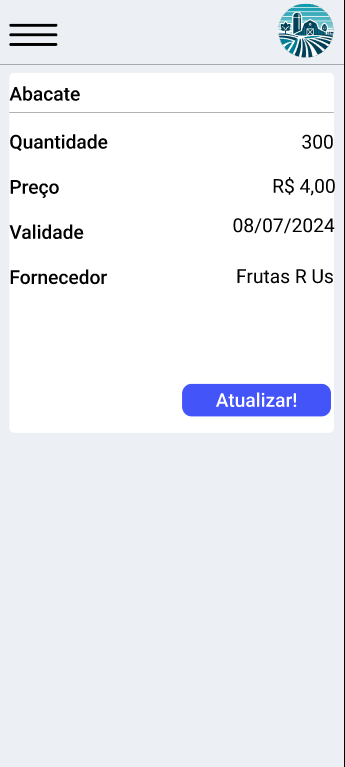


Figura 15 - Tela Cadastro Mobile Alta Fidelidade Figura 16 - Tela Detalhes Mobile Alta Fidelidade

As telas de estoque sofreram alterações, principalmente nos dispositivos móveis, para uma melhor usabilidade do sistema, alterando o clássico modelo de tabela para pequenas janelas contendo todas as informações necessárias, de maneira a melhorar a visualização e operação.

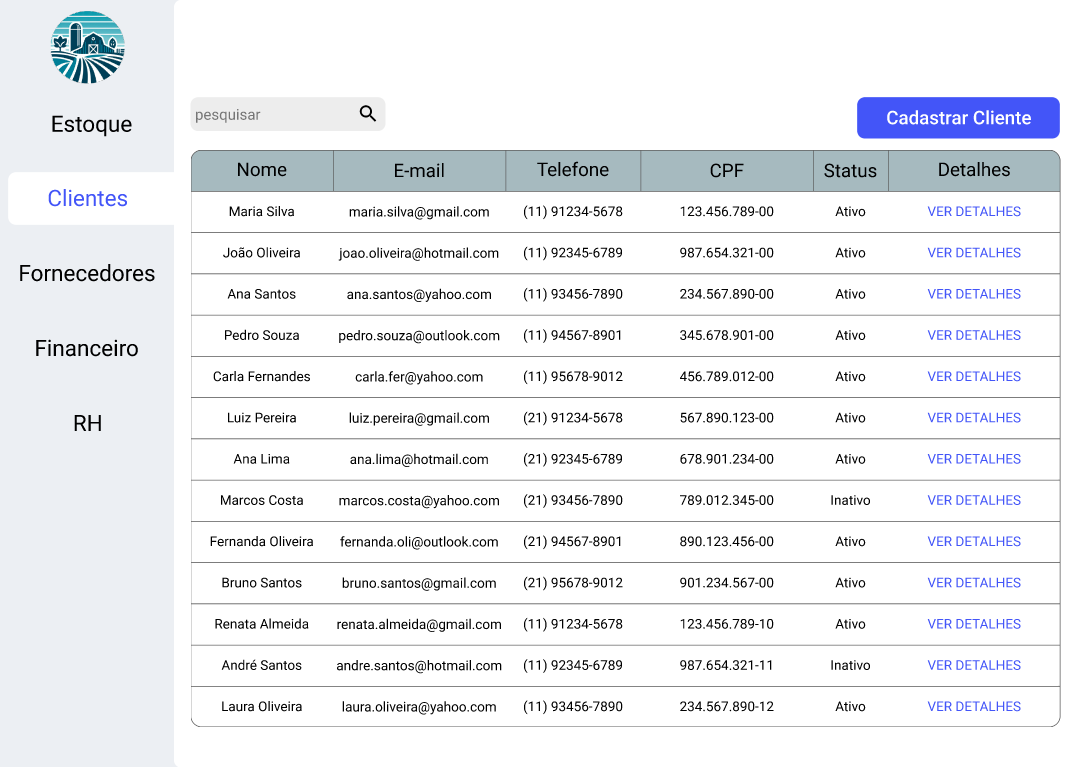


Figura 17 - Tela de Clientes Alta Fidelidade

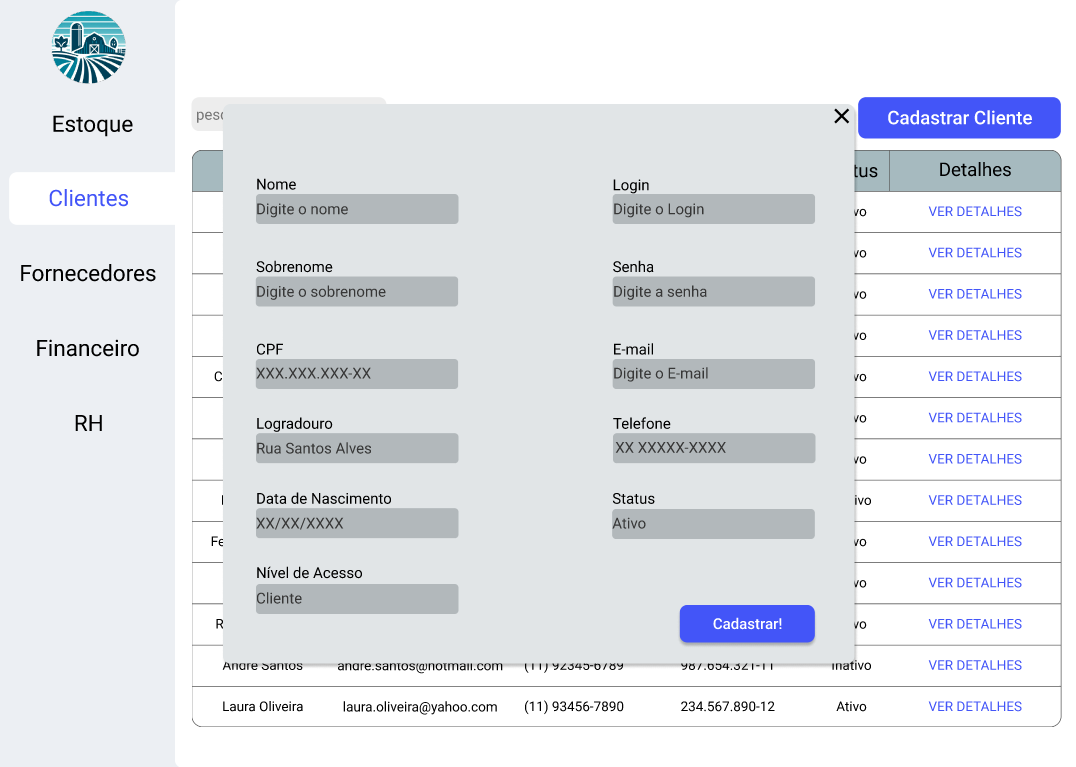


Figura 18 - Tela Cadastro Clientes Alta Fidelidade

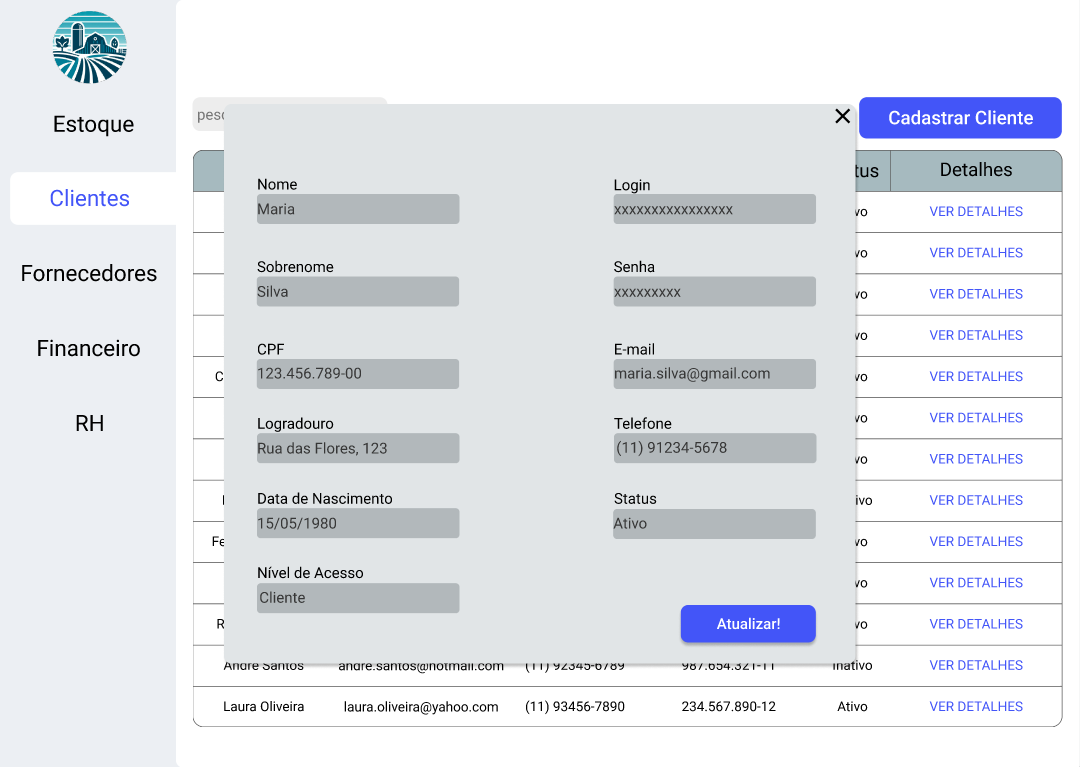


Figura 19 - Tela Detalhes e Edição de Produtos Alta Fidelidade

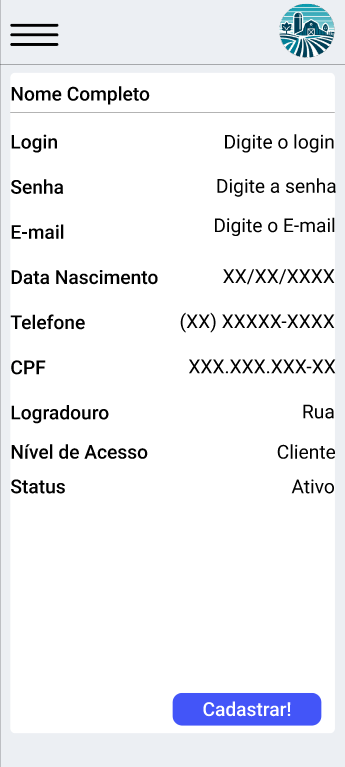
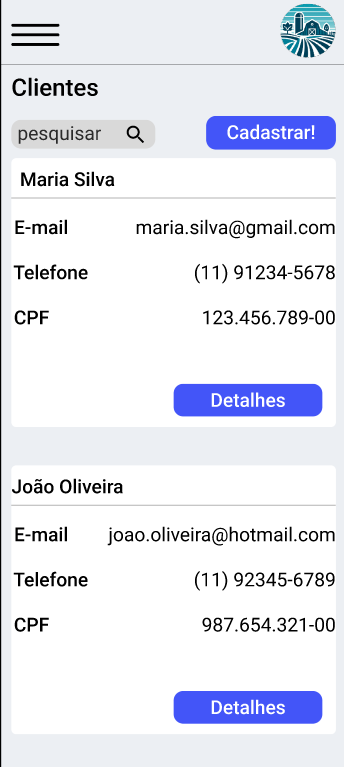


Figura 20 - Tela Clientes Mobile Figura 21 - Tela Cadastro Clientes Mobile

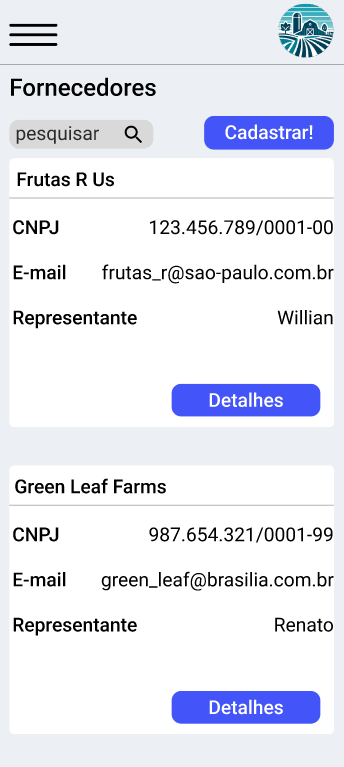


Figura 22 - Tela Detalhes e Edição Clientes Mobile Figura 23 - Tela Fornecedores Mobile



Figura 24 - Tela Fornecedores Alta Fidelidade

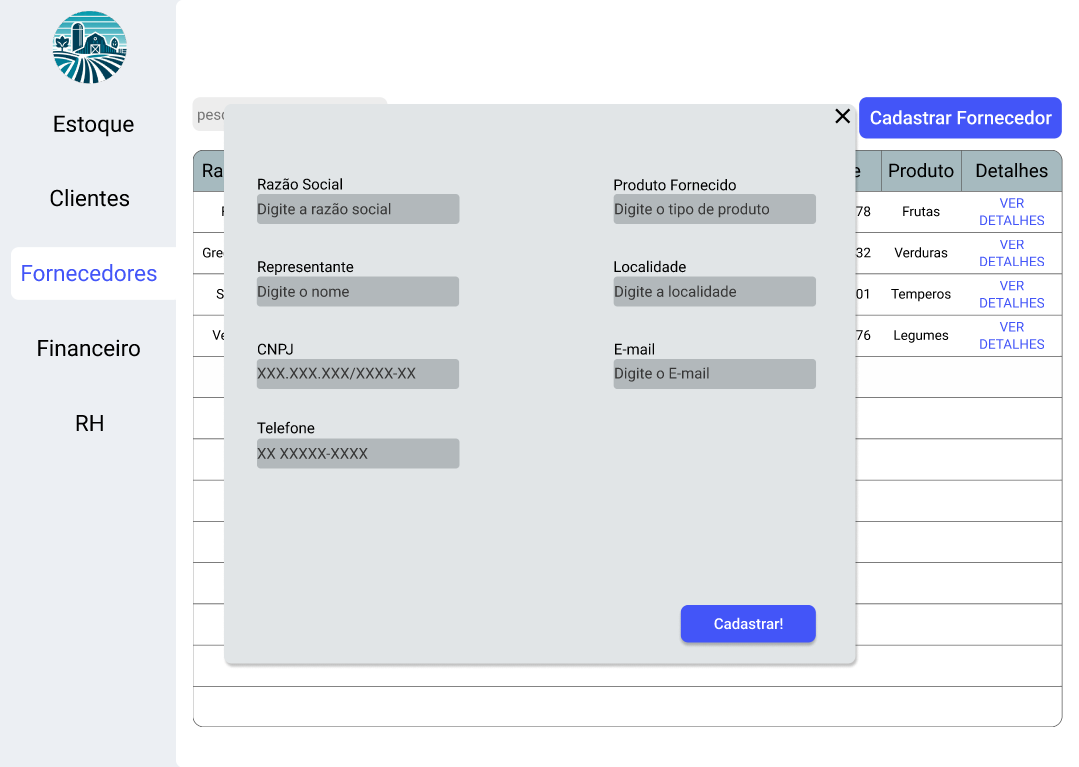


Figura 25 - Tela Cadastro Fornecedores Alta Fidelidade

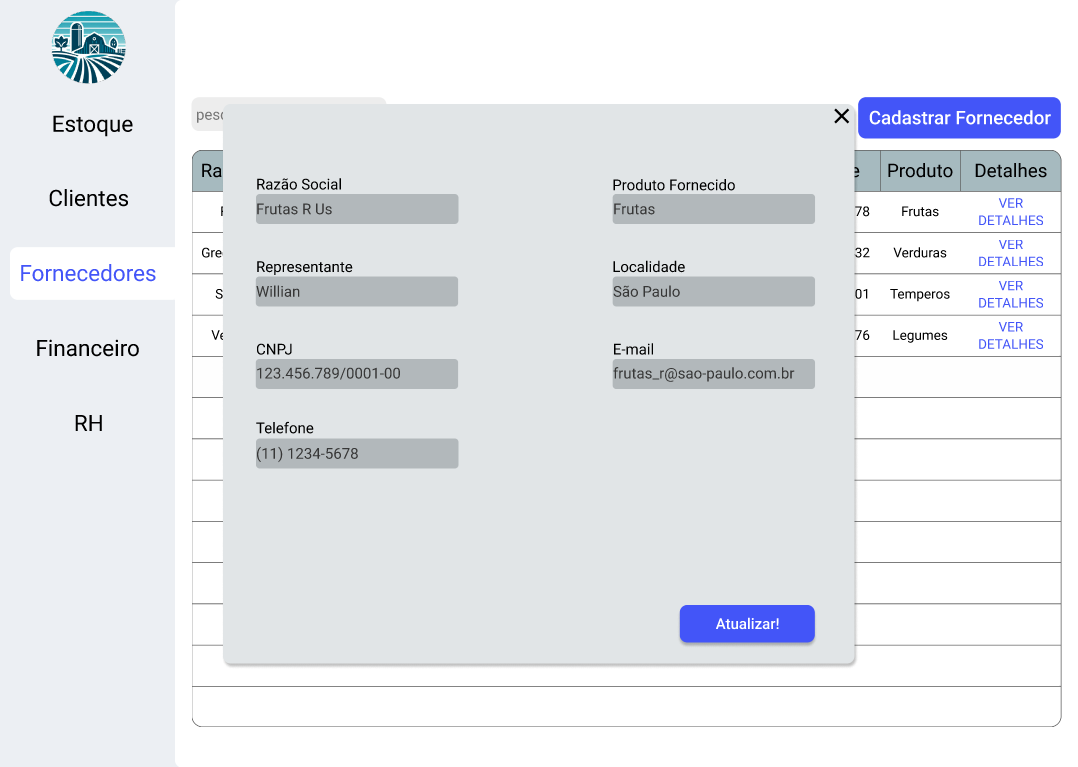


Figura 26 - Tela Detalhes e Edição de Fornecedores Alta Fidelidade

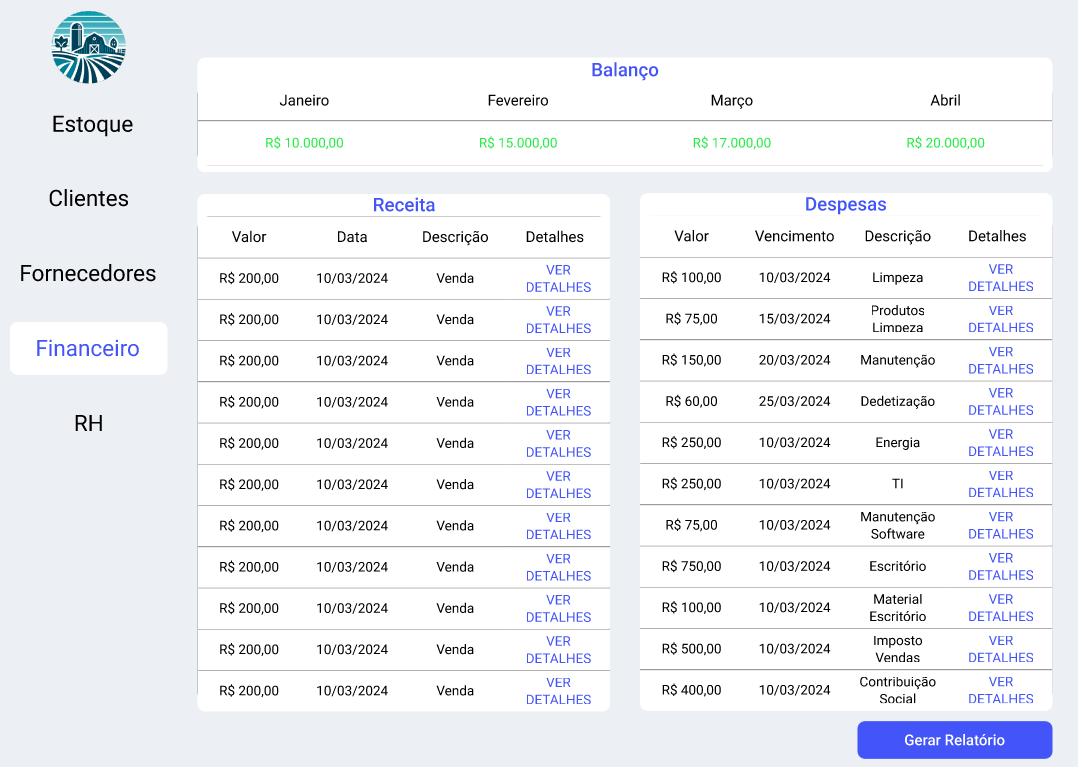


Figura 27 - Tela Financeiro Alta Fidelidade



Figura 28 - Tela Cadastro Fornecedores Mobile



Figura 29 - Tela Detalhes e Edição Fornecedores Mobile



Figura 30 - Tela Financeiro Mobile

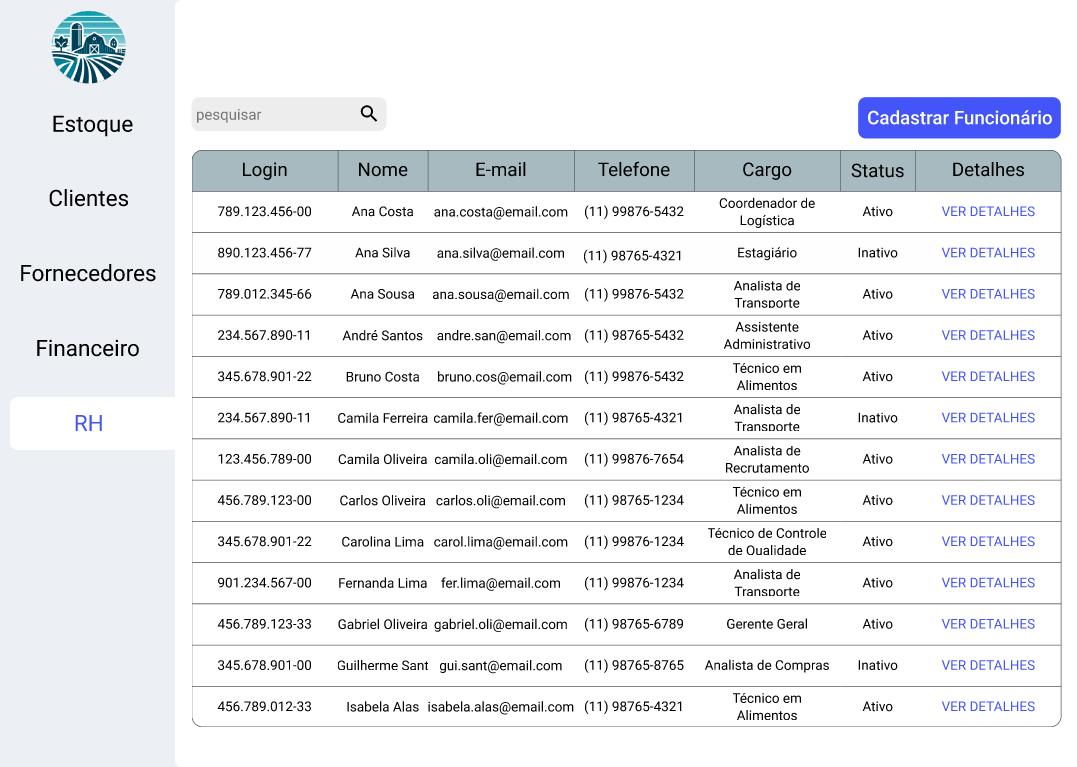


Figura 31 - Tela de RH em Alta Fidelidade

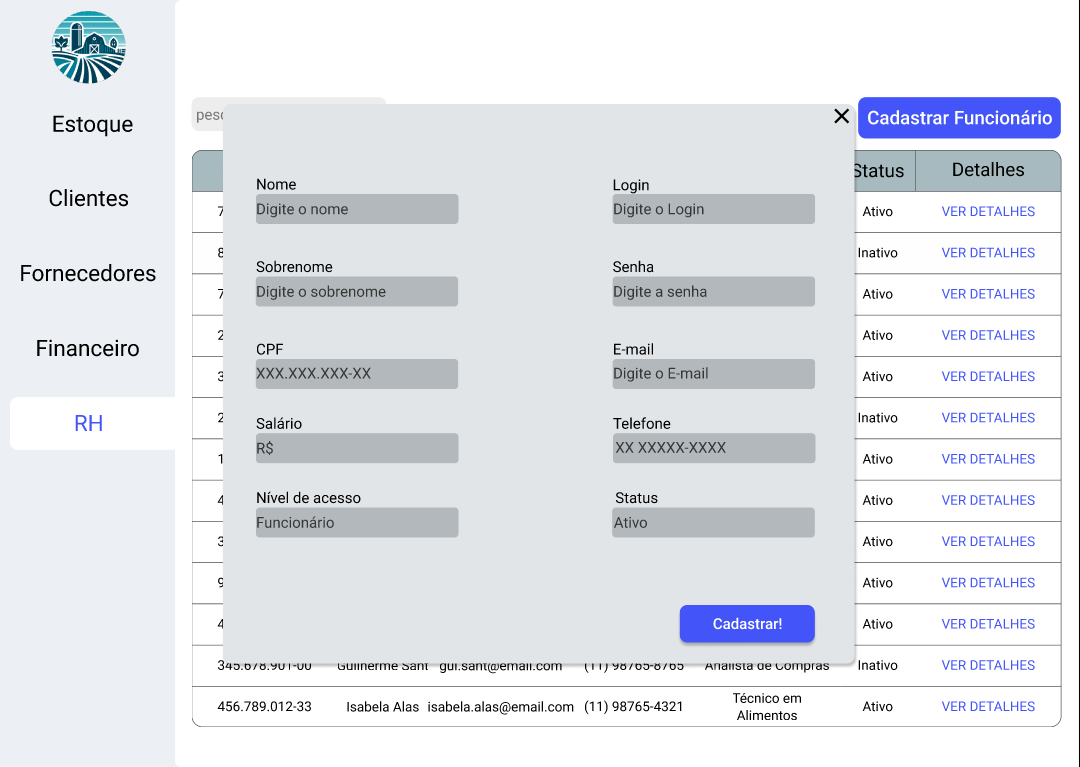


Figura 32 - Tela Cadastro de Funcionários em Alta Fidelidade

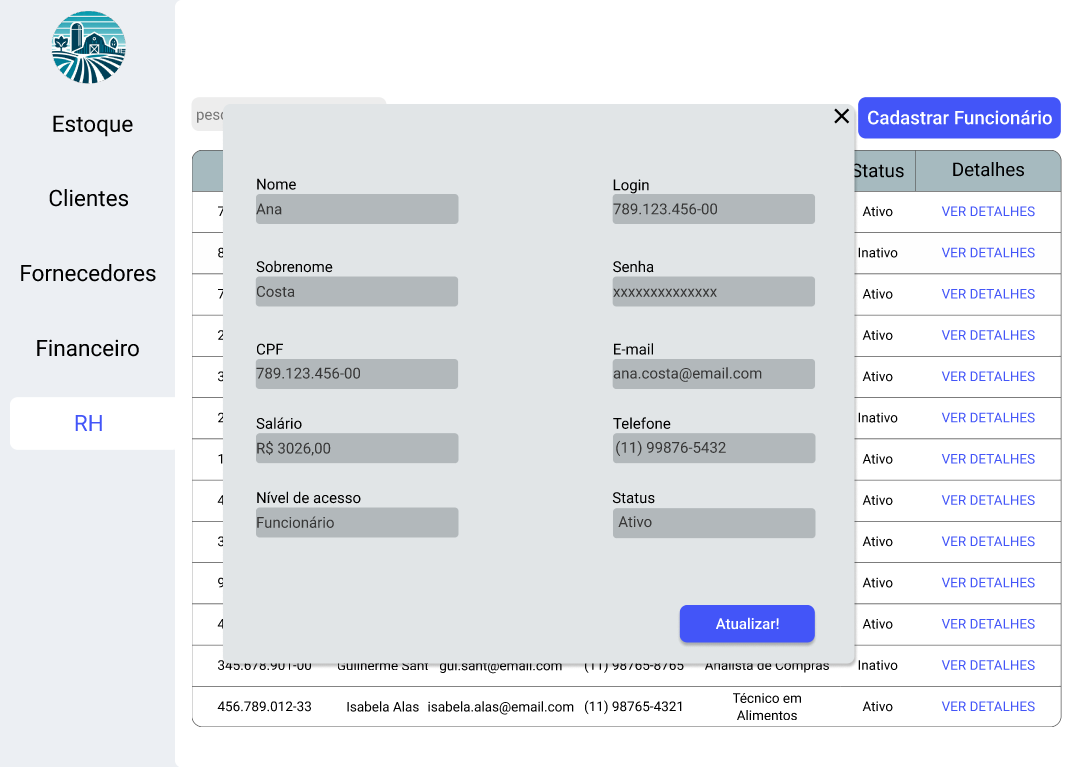


Figura 33 - Tela Detalhes e Edição de Funcionários em Alta Fidelidade

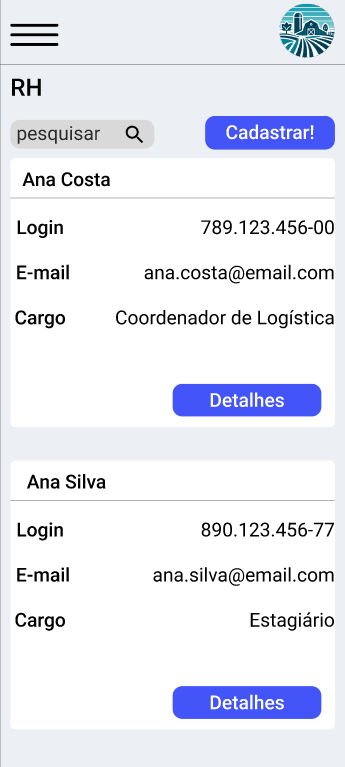
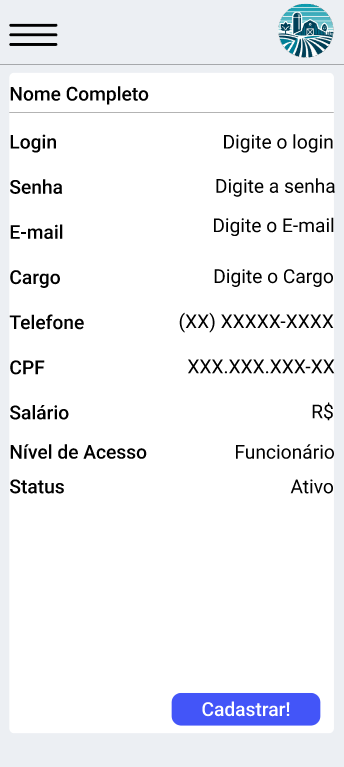


Figura 34 - Tela RH Mobile Figura 35 - Tela Cadastro Funcionário Mobile



Figura 36 - Tela Detalhes e Edição de Funcionários Mobile

### Testes de interface

Com o plano de testes abaixo, realizaremos testes das funções básicas do sistema, como o login, o cadastro, a pesquisa e a edição de informações.

|  |  |  |
| --- | --- | --- |
| Descrição | Resultados esperados | Resultados reais |
| Digitar Login, Digitar Senha, Clicar em Entrar. | Após checagem de dados, acessar a tela inicial do sistema. | O teste foi realizado com sucesso, o fluxo foi seguido de maneira correta e não houve problemas |
| Clicar no botão de cadastro, preencher todos os campos necessários, como nome, cpf, endereço, telefone, e-mail e etc… Clicar em Cadastrar | Realizar o cadastro do item informado, com todas as informações preenchidas e disponíveis para alteração caso necessário. | O teste foi realizado com sucesso, o fluxo foi seguido de maneira correta e não houve problemas |
| Clicar no botão Ver Detalhes, Atualizar quaisquer informações necessárias, Clicar em Atualizar | Visualizar todas as informações cadastradas no item específico, alterar as informações necessárias, e após clicar em atualizar, as informações devem ter sido alteradas e salvas para serem visualizadas novamente caso necessário. | O teste foi realizado com sucesso, o fluxo foi seguido de maneira correta e não houve problemas |
| Clicar na barra de pesquisa e digitar a informação que queira encontrar de acordo com aba selecionada | As informações deverão ser mostradas conforme o escrito na barra de pesquisa. | O teste foi realizado com sucesso, o fluxo foi seguido de maneira correta e não houve problemas |

### Diagrama de classes do sistema

No contexto do nosso sistema, a princípio foram criados três diagramas de classes principais para representar diferentes aspectos do sistema.

O primeiro diagrama de classes é dedicado à camada de persistência DAO (Data Access Object). Esta camada é crucial para a manipulação de dados em nosso sistema. Ela fornece uma abstração entre a lógica de negócios e o acesso aos dados, permitindo que o sistema interaja com o banco de dados. Cada classe que termina com o sufixo DAO representa uma tabela no banco de dados e fornece métodos para recuperar, inserir, atualizar e excluir (desativar) registros nessa tabela.

Segundo Oliveira (2015), o Data Access Object (DAO) encapsula o acesso ao banco de dados, separando as regras de persistência das regras de negócio. Isso significa que a lógica de como os dados são armazenados e recuperados do banco de dados fica isolada do restante da aplicação, o que traz diversos benefícios:

Flexibilidade: O acesso ao banco de dados pode ser alterado sem que a aplicação principal seja afetada. Por exemplo, você pode trocar o tipo de banco de dados que está usando sem precisar modificar o código da aplicação.

Organização: A aplicação pode ter múltiplas formas de acesso aos dados de forma mais organizada, através de diferentes DAOs. Isso facilita a manutenção do código e torna a aplicação mais modular.

Manutenção: O código fica mais fácil de manter porque as regras de persistência e as regras de negócio estão separadas. Isso significa que você pode modificar as regras de persistência sem precisar modificar o código das regras de negócio, e vice-versa.

Abaixo o Diagrama de Classes sobre o pacote DAO do nosso sistema.

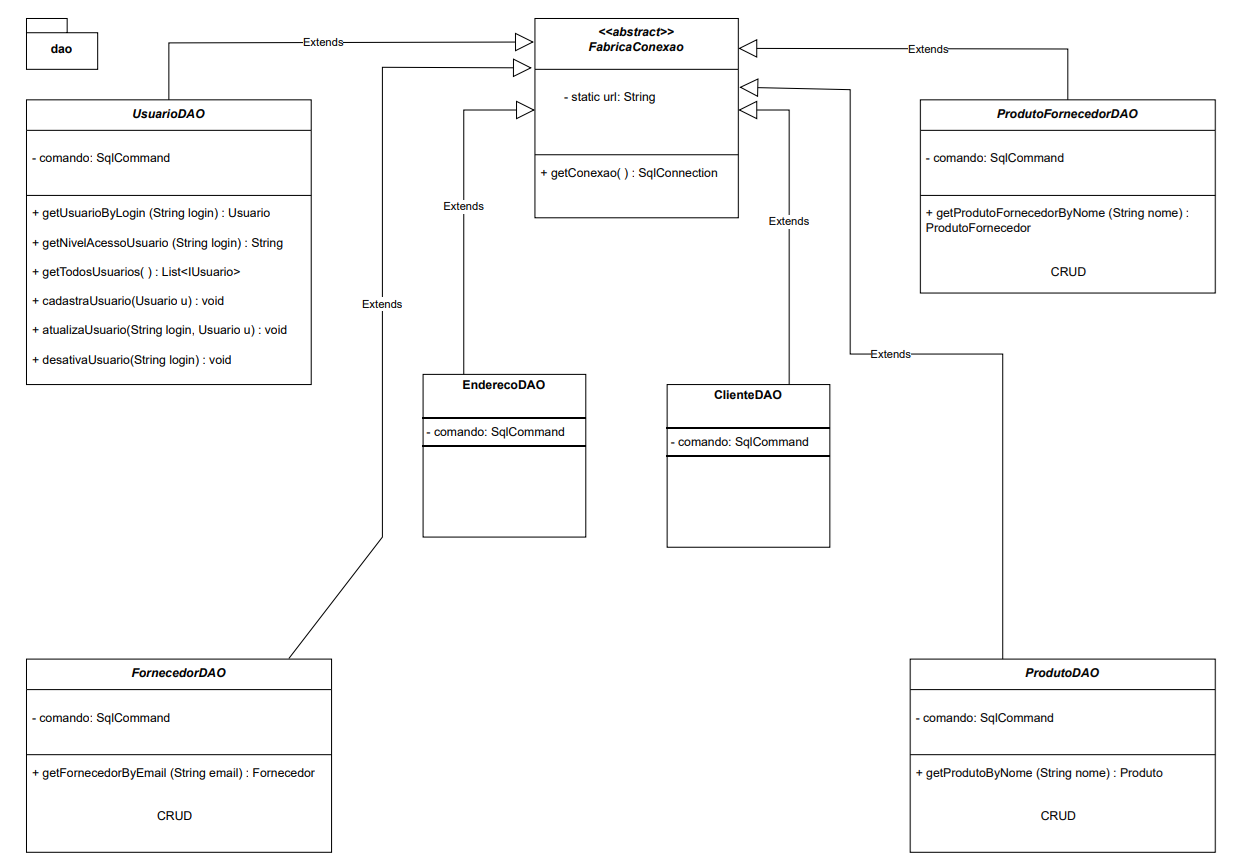


Figura 37 – Diagrama de Classe do pacote DAO.

Em nosso diagrama de classes DAO, observamos que todas as classes são filhas da Classe Abstrata FabricaConexao. Este é um padrão de projeto amplamente reconhecido no desenvolvimento de sistemas orientados a objetos, conhecido como Template Method. Como Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995) afirmam em seu livro Design Patterns: Elements of Reusable Object-Oriented Software (p. 360), o Template Method permite “definir o esqueleto de um algoritmo dentro de uma operação, deixando alguns passos a serem preenchidos pelas subclasses”.

O método público getConexao(), que retorna um objeto do tipo SqlConnection, é responsável por abrir e fechar a conexão com o banco de dados. É importante destacar que a Classe Abstrata possui uma string estática chamada url, que se refere à URL de conexão com o banco de dados, configurada através do arquivo **App.config**.

As classes filhas, também conhecidas como DAOs, herdam o método getConexao() e podem invocá-lo sempre que precisarem estabelecer uma conexão com o banco de dados. Praticamente todos os métodos de gerenciamento (CRUD) necessitarão abrir uma conexão com o banco, realizar as operações necessárias e, em seguida, fechar a conexão.

Vale notar que todas as classes DAO possuem um atributo privado chamado comando do tipo SqlCommand, esse atributo vai ser o responsável por executar os scripts de SQL dentro de cada método.

E por fim vale a ressalva que todas as classes DAO irão realizar o CRUD de suas entidades base. O CRUD é baseado em método de busca personalizada, busca generalizada, atualização de dados, inserção de dados e exclusão de dados.

Logo a seguir uma Figura do código fonte demonstrando o uso do Template Method:

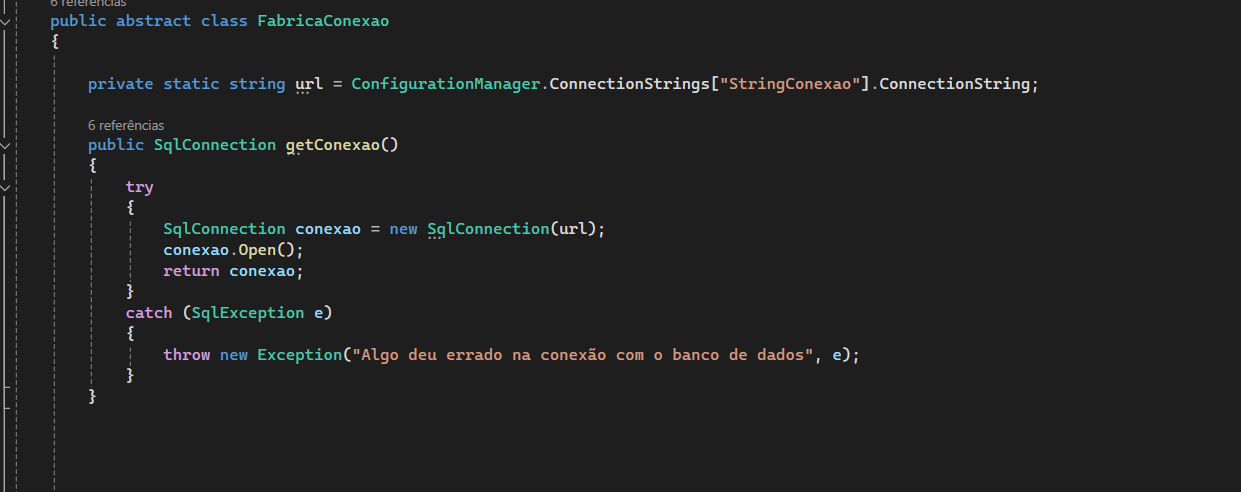


Figura 38 – Código fonte para conexão com o banco de dados.

Vale notar, o bloco try-catch é usado para lidar com exceções que podem ocorrer ao abrir a conexão. Se ocorrer uma SqlException durante a abertura da conexão, o código captura essa exceção e lança uma nova exceção genérica com uma mensagem de erro personalizada e a exceção original como causa interna. Isso permite que o código que chama getConexao() saiba que algo deu errado ao tentar abrir a conexão com o banco de dados.

Agora na figura abaixo, o uso de getConexao em uma classe DAO.

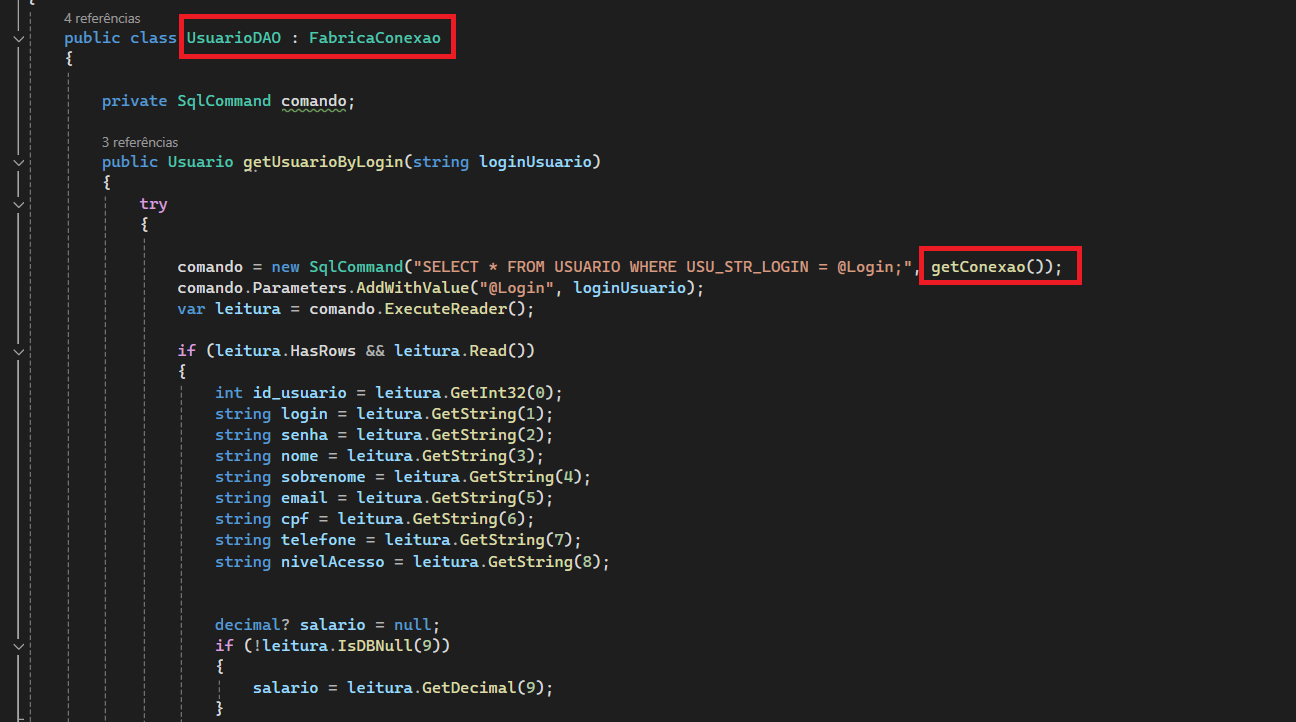


Figura 39 – Uso do método getConexão.

É possível observar que a classe UsuarioDAO herda da classe FabricaConexao. Em um dos seus métodos, getUsuarioByLogin, é necessário utilizar o método getConexao(), que foi definido na superclasse, para abrir uma conexão com o banco de dados e executar uma determinada consulta (query) no banco de dados.

Desta maneira, apenas uma classe é responsável por encapsular o algoritmo de conexão com o banco de dados, tornando o código mais gerenciável e manutenível.

O segundo diagrama de classes representa a venda de produtos aos clientes. Este diagrama ilustra as relações entre os clientes, os produtos e os pedidos. Este diagrama ajuda a entender como os produtos são vendidos aos clientes e como essas vendas são registradas no sistema.

Abaixo a figura ilustra o diagrama de classe que representar uma venda.

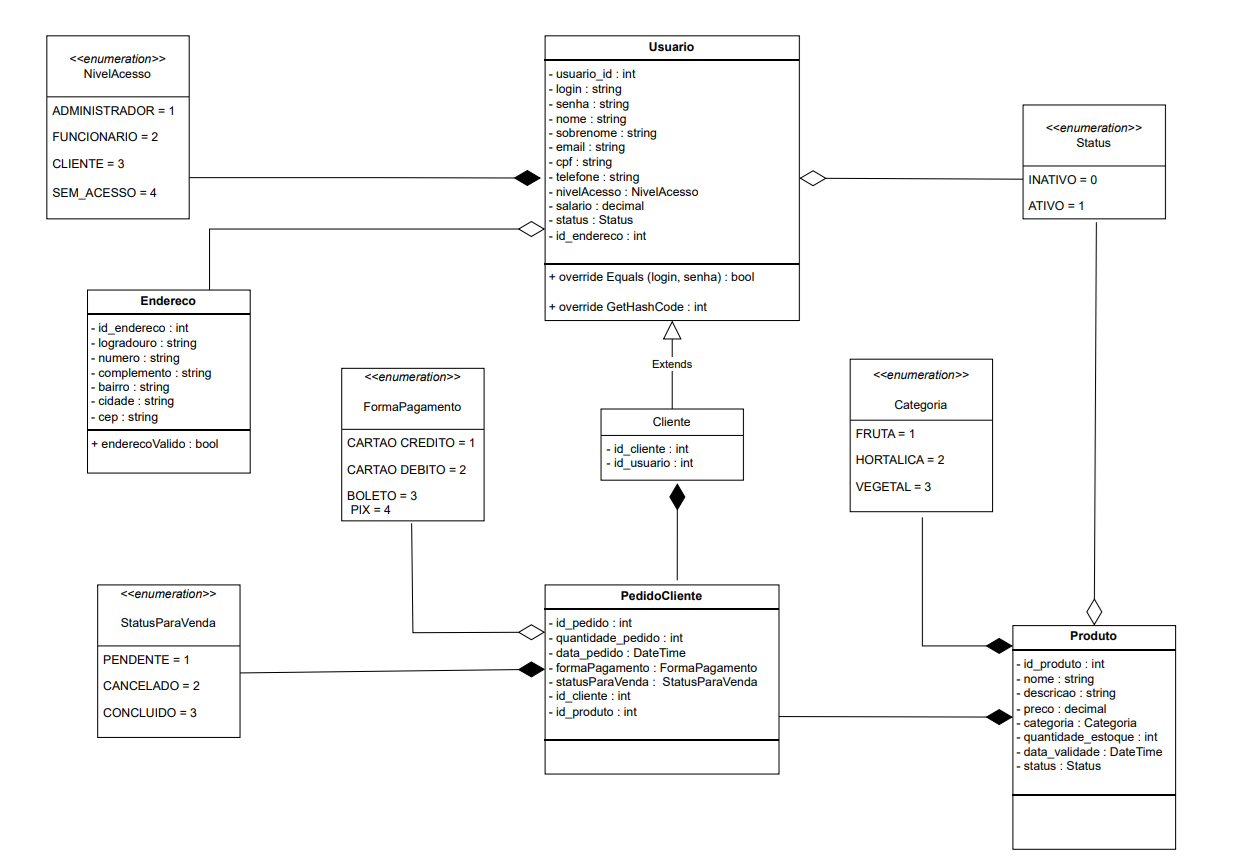


Figura 40 – Diagrama de Classe de venda.

Os elementos-chave deste diagrama são os losangos pintados de preto e os losangos brancos, que representam respectivamente a composição e a agregação. Durante o desenvolvimento do projeto, foi pensado o uso de enumerações (enums) para representar variáveis fixas. Como citado por DEVMEDIA (2024), “Uma enumeração de dados é útil quando precisamos criar estruturas que serão pouco alteradas ao longo do desenvolvimento do projeto”.

Os métodos sobrescritos (override) de Equals e HashCode na Classe Usuário são utilizados durante o projeto para verificar a correspondência entre um objeto do tipo usuário e outro recuperado do banco de dados, com o objetivo de validar uma busca. Se um objeto usuário possui o mesmo login e senha, o sistema o considera como sendo o mesmo objeto.

O terceiro diagrama de classes representa a compra de produtos de fornecedores.

A seguir a ilustração das classes representando uma compra de produtos de fornecedores.

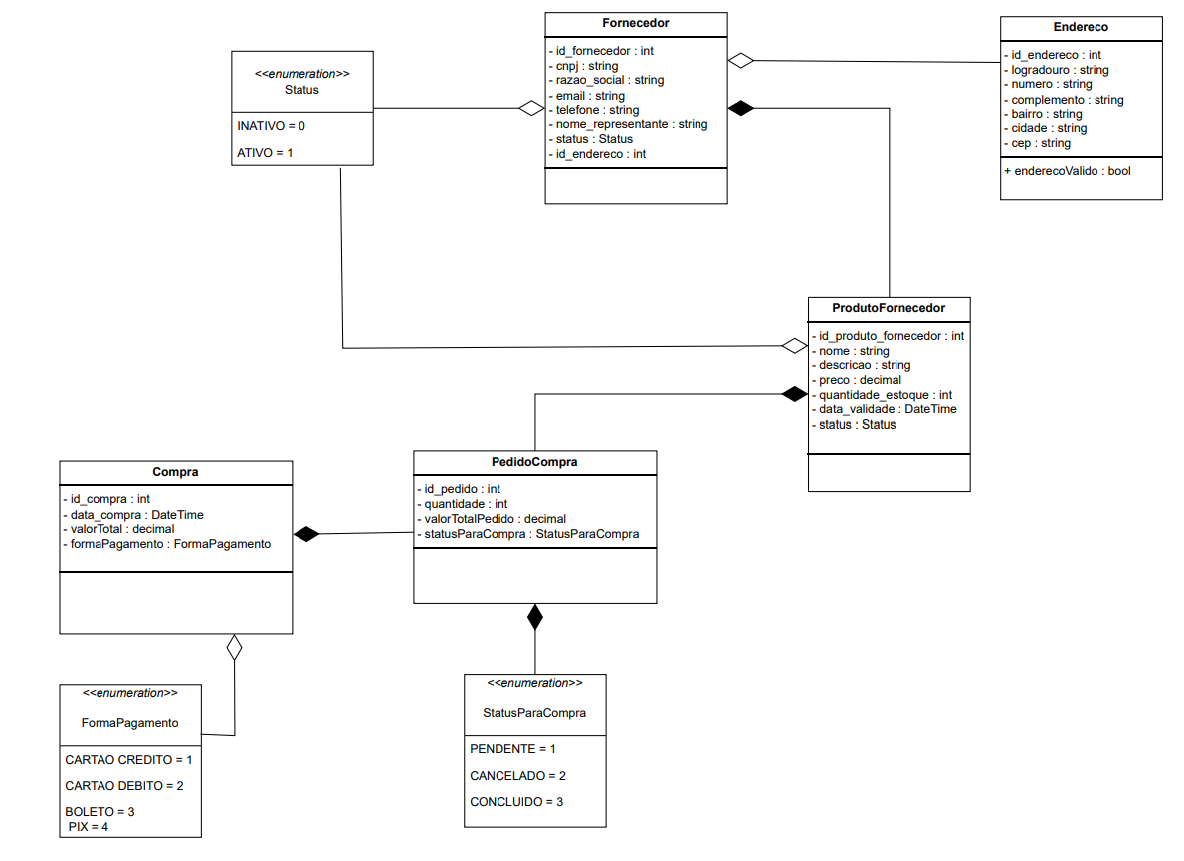


Figura 41 – Diagrama de Classe de compra.

Assim como citado no diagrama que representa vendas, este diagrama de compra também está repleto de associações, agregações e enumerações.

### Banco de dados do sistema

Neste tópico de banco de dados abordaremos a modelagem e implementação de um banco de dados no SQL Server. Iniciaremos com a modelagem conceitual, definindo entidades e suas relações. Em seguida, passaremos para a modelagem lógica, transformando o modelo conceitual em um esquema de banco de dados. Finalmente, implementaremos o esquema usando scripts SQL, criando tabelas e definindo relações entre elas.

Segundo Heuser (1998, p.16), um modelo conceitual é uma representação abstrata e independente de um banco de dados, que não leva em consideração a forma como os dados estão armazenados em um Sistema de Gerenciamento de Banco de Dados (SGBD). Ele destaca a importância do modelo entidade-relacionamento (ER) como uma técnica amplamente utilizada para a modelagem conceitual.

Nosso sistema possui dois modelos para o banco de dados. O primeiro representa a venda de produtos para os clientes e será referido como modelo\_venda para fins didáticos e explicativos. O segundo modelo representa a compra de produtos de fornecedores e será chamado de modelo\_compra. Essa nomenclatura nos ajudará a distinguir claramente entre os dois processos.

A figura abaixo ilustra o modelo conceitual de venda.

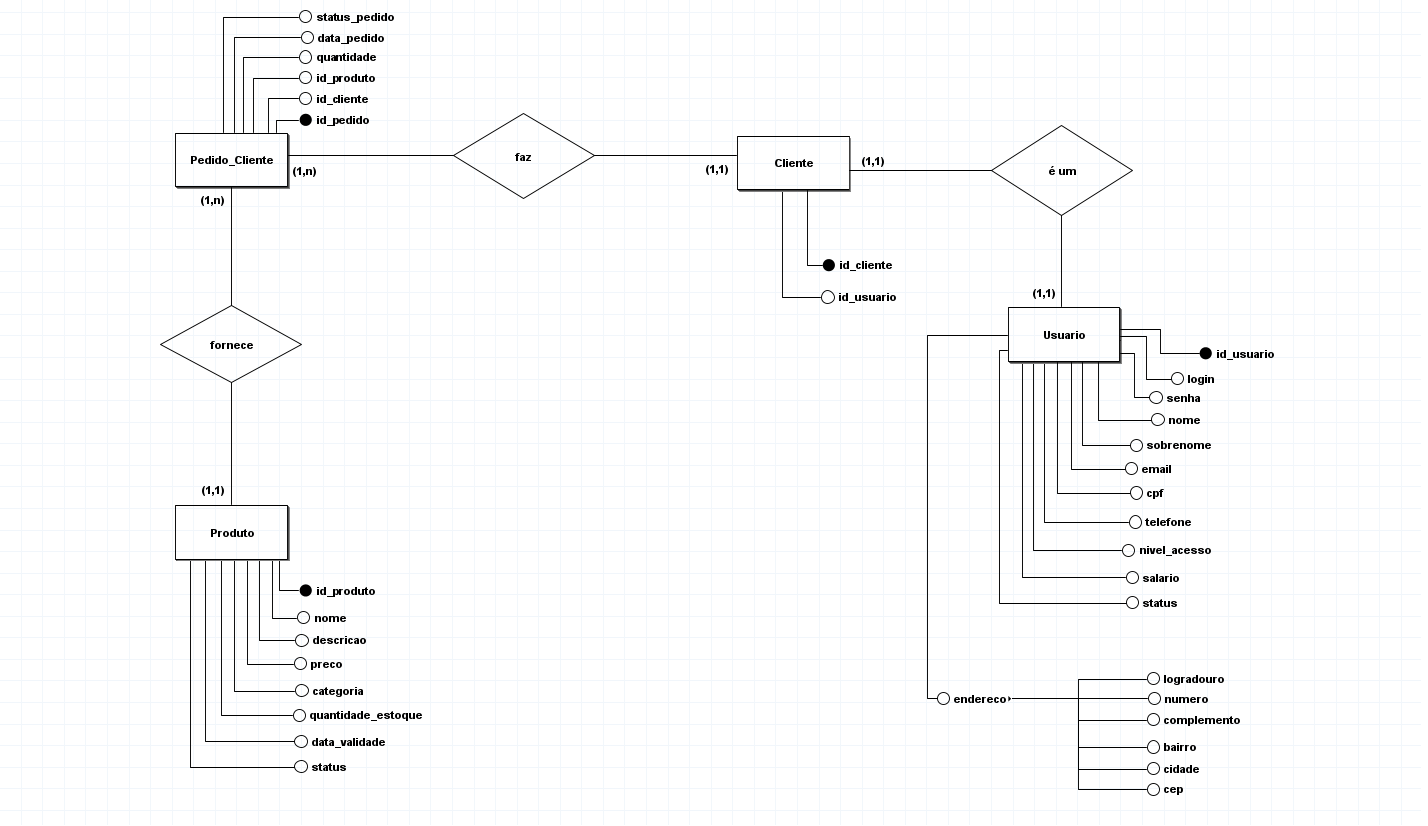


Figura 42 – Modelo ER conceitual de venda.

O modelo\_venda foi concebido inicialmente com o objetivo de mapear as potenciais transações comerciais que a nossa empresa, uma fazenda urbana, realizará com seus clientes. Após uma análise cuidadosa, estabelecemos algumas regras de negócio que se alinham perfeitamente com o nosso modelo de banco de dados. A seguir, apresento essas regras:

Cliente como Usuário: Optamos por considerar o cliente como um usuário, pois, no futuro, ele poderá interagir com o sistema.

Entidade Cliente: Percebemos a necessidade de ter uma entidade Cliente para estabelecer uma relação direta com a venda de produtos. Assim, criamos uma relação 1:1 entre Cliente e Usuário. Nessa relação, a entidade Cliente possui um ID próprio e um ID que referencia a entidade Usuário. É crucial garantir a unicidade deste ID de usuário, uma prática comum em relacionamentos 1:1.

Telefone como Atributo Simples: Em muitos projetos, o telefone é tratado como um atributo multivalorado. No entanto, no nosso projeto, ele será um atributo simples. Cada usuário terá apenas um número de telefone, preferencialmente móvel.

Atributo Composto Endereço: A entidade Usuário terá um atributo composto, o endereço, que será desmembrado em outra tabela contendo logradouro, número, complemento, bairro, cidade e CEP.

Chave Única Nome na Entidade Produto: O atributo nome na entidade Produto será também uma chave única. Acreditamos que isso facilitará a busca do produto, sem a necessidade de passar o ID, como é comum na maioria dos projetos.

Atributo Status: O atributo status, presente tanto na entidade Usuário quanto na entidade Produto, refere-se a uma possível exclusão lógica (softdelete). Normalmente, não é prática comum excluir registros de tabelas em bancos de dados. Portanto, a melhor solução é usar o atributo status para ativar e desativar o referencial.

Após a criação do modelo conceitual, criamos o modelo lógico para modelo\_venda. A seguir uma figura que ilustra o mesmo modelo\_venda só que em uma perspectiva lógica:

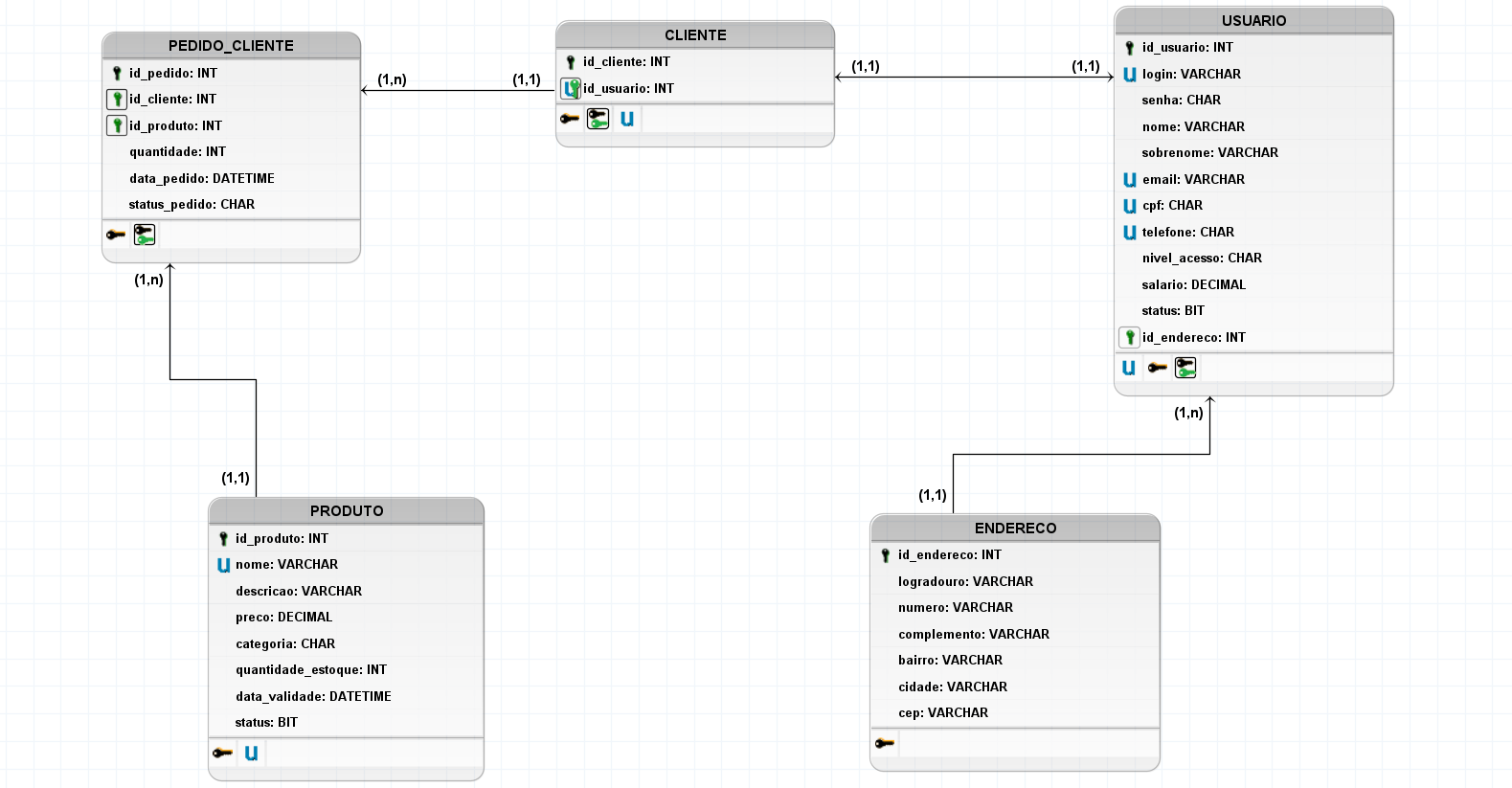


Figura 43 – Modelo ER lógico de venda.

Segundo Heuser (1998, p.18), o modelo lógico é uma representação da estrutura de dados de um banco de dados, conforme percebida pelo usuário do Sistema de Gerenciamento de Banco de Dados (SGBD). Ele serve como uma ponte entre o mundo real e o sistema de banco de dados, facilitando a compreensão e a manipulação dos dados pelo usuário.

Agora iremos mostrar o modelo\_compra conceitual e o modelo\_compra lógico, levando em consideração as mesmas análises citadas anteriormente.

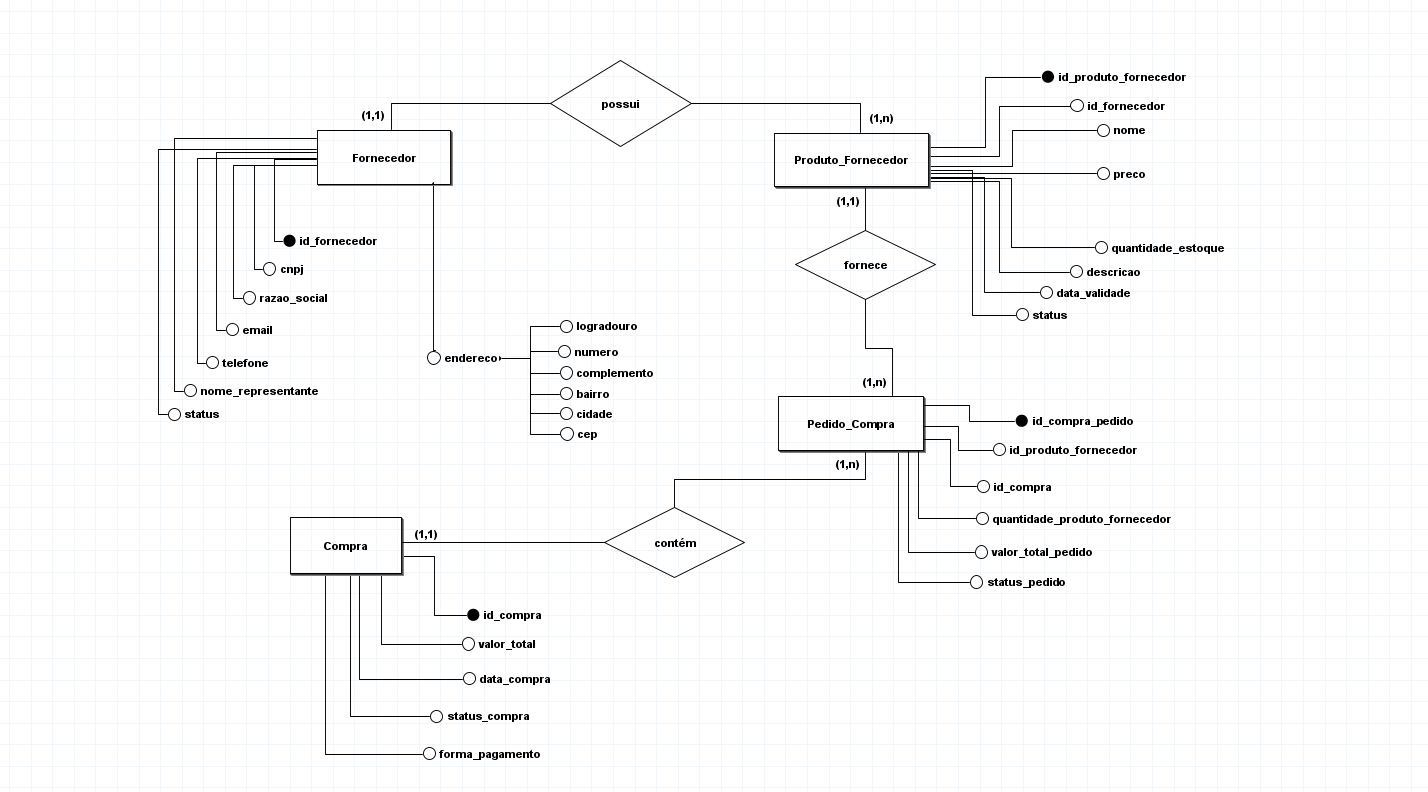


Figura 44 – Modelo ER conceitual de compra.

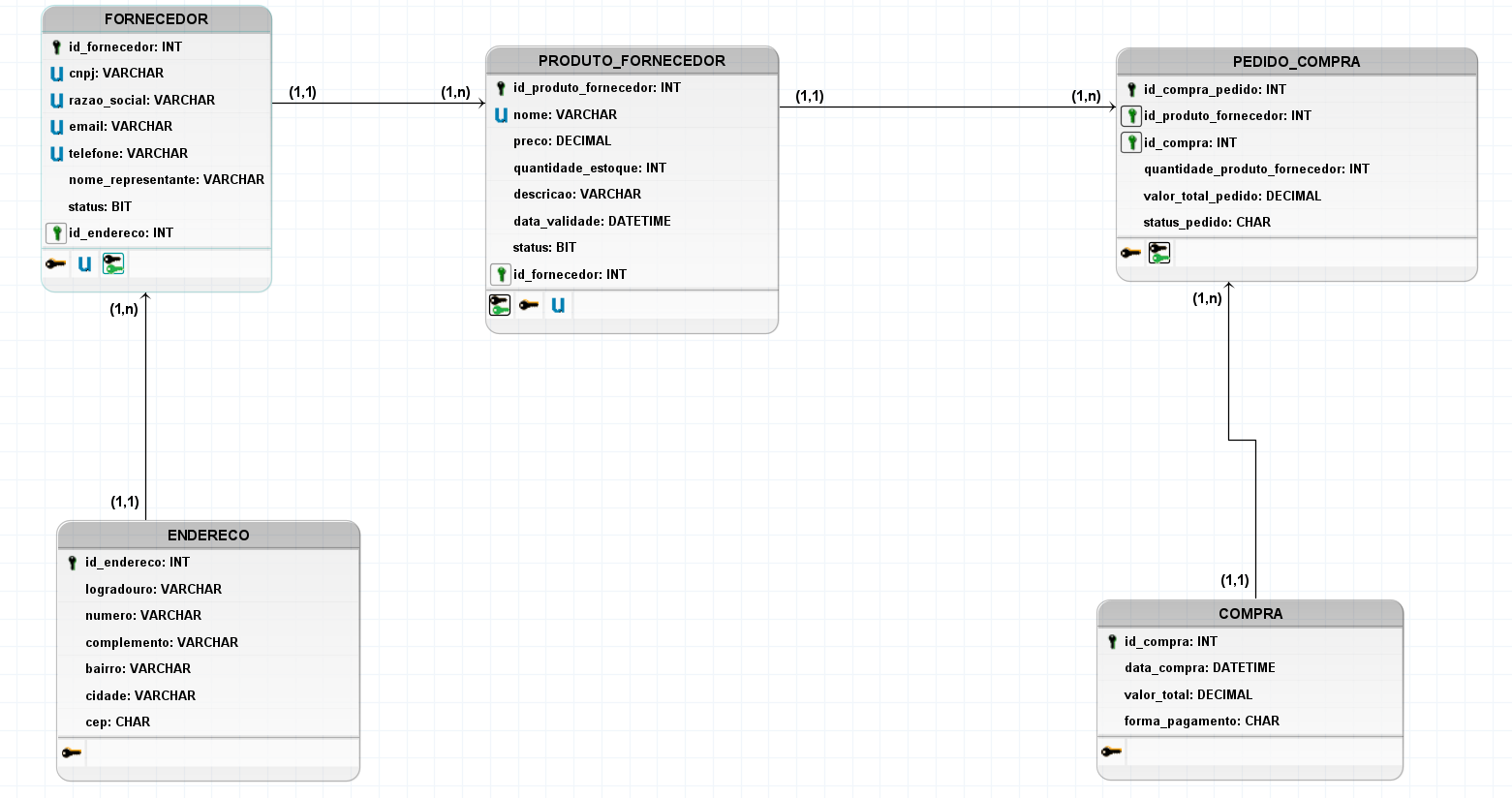


Figura 45 – Modelo ER lógico de compra.

Após todo o processo de modelagem, os scripts de SQL para gerar a criação de tabelas, relacionamentos e restrições é o seguinte:

CREATE DATABASE PIM;

USE PIM;

CREATE TABLE ENDERECO (

END\_INT\_ID INT IDENTITY(1,1),

END\_STR\_LOGRADOURO VARCHAR(100) NOT NULL,

END\_STR\_NUMERO VARCHAR(6) NOT NULL,

END\_STR\_COMPLEMENTO VARCHAR(40),

END\_STR\_BAIRRO VARCHAR(25) NOT NULL,

END\_STR\_CIDADE VARCHAR(25) NOT NULL,

END\_STR\_CEP CHAR(8) NOT NULL,

CONSTRAINT PK\_ENDERECO PRIMARY KEY (END\_INT\_ID)

);

CREATE TABLE USUARIO (

USU\_INT\_ID INT IDENTITY(1,1),

USU\_STR\_LOGIN VARCHAR(50) NOT NULL,

USU\_STR\_SENHA CHAR(64) NOT NULL,

USU\_STR\_NOME VARCHAR(20) NOT NULL,

USU\_STR\_SOBRENOME VARCHAR(40) NOT NULL,

USU\_STR\_EMAIL VARCHAR(50) NOT NULL,

USU\_STR\_CPF CHAR(11) NOT NULL,

USU\_STR\_TELEFONE CHAR(11) NOT NULL,

USU\_STR\_NIVEL\_ACESSO CHAR(1) NOT NULL,

USU\_DEC\_SALARIO DECIMAL(8, 2) DEFAULT NULL,

USU\_BIT\_STATUS BIT NOT NULL DEFAULT 1,

END\_INT\_ID INT NOT NULL,

CONSTRAINT PK\_USUARIO PRIMARY KEY (USU\_INT\_ID),

CONSTRAINT PK\_USUARIO\_LOGIN UNIQUE(USU\_STR\_LOGIN),

CONSTRAINT UK\_USUARIO\_EMAIL UNIQUE (USU\_STR\_EMAIL),

CONSTRAINT UK\_USUARIO\_CPF UNIQUE (USU\_STR\_CPF),

CONSTRAINT UK\_USUARIO\_TELEFONE UNIQUE (USU\_STR\_TELEFONE),

CONSTRAINT FK\_ENDERECO\_USUARIO FOREIGN KEY (END\_INT\_ID) REFERENCES ENDERECO(END\_INT\_ID)

);

CREATE TABLE CLIENTE

(

CLI\_INT\_ID INT IDENTITY (1,1),

USU\_INT\_ID INT,

CONSTRAINT PK\_CLIENTE PRIMARY KEY (CLI\_INT\_ID),

CONSTRAINT FK\_USUARIO FOREIGN KEY (USU\_INT\_ID) REFERENCES USUARIO(USU\_INT\_ID),

CONSTRAINT UK\_USUARIO UNIQUE (USU\_INT\_ID)

);

CREATE TABLE FORNECEDOR (

FOR\_INT\_ID INT IDENTITY(1,1),

FOR\_STR\_CNPJ CHAR(14),

FOR\_STR\_RAZAO\_SOCIAL VARCHAR(80),

FOR\_STR\_EMAIL VARCHAR(50),

FOR\_STR\_TELEFONE CHAR(11),

FOR\_STR\_REPRESENTANTE VARCHAR(50),

FOR\_BIT\_STATUS BIT NOT NULL DEFAULT 1,

END\_INT\_ID INT NOT NULL,

CONSTRAINT PK\_FORNECEDOR PRIMARY KEY (FOR\_INT\_ID),

CONSTRAINT UK\_FORNECEDOR\_CNPJ UNIQUE (FOR\_STR\_CNPJ),

CONSTRAINT UK\_FORNECEDOR\_RAZAOSOCIAL UNIQUE (FOR\_STR\_RAZAO\_SOCIAL),

CONSTRAINT UK\_FORNECEDOR\_EMAIL UNIQUE (FOR\_STR\_EMAIL),

CONSTRAINT UK\_FORNECEDOR\_TELEFONE UNIQUE (FOR\_STR\_TELEFONE),

CONSTRAINT FK\_ENDERECO\_FORNECEDOR FOREIGN KEY (END\_INT\_ID) REFERENCES ENDERECO(END\_INT\_ID)

);

CREATE TABLE PRODUTO\_FORNECEDOR

(

PRO\_FOR\_INT\_ID INT IDENTITY(1,1),

PRO\_FOR\_STR\_NOME VARCHAR(30) NOT NULL,

PRO\_FOR\_DEC\_PRECO DECIMAL(8, 2) NOT NULL,

PRO\_FOR\_INT\_QUANTIDADE\_ESTOQUE INT NOT NULL,

PRO\_FOR\_STR\_DESCRICAO VARCHAR(255),

PRO\_FOR\_DATE\_DATA\_VALIDADE DATETIME NOT NULL,

PRO\_FOR\_BIT\_STATUS BIT NOT NULL DEFAULT 1,

FOR\_INT\_ID INT NOT NULL,

CONSTRAINT PK\_PRODUTO\_FORNECEDOR PRIMARY KEY (PRO\_FOR\_INT\_ID),

CONSTRAINT UK\_PRODUTO\_FORNECEDOR\_NOME UNIQUE (PRO\_FOR\_STR\_NOME),

CONSTRAINT FK\_FORNECEDOR FOREIGN KEY (FOR\_INT\_ID) REFERENCES FORNECEDOR(FOR\_INT\_ID)

);

CREATE TABLE PRODUTO

(

PRO\_INT\_ID INT IDENTITY(1,1),

PRO\_STR\_NOME VARCHAR(30) NOT NULL,

PRO\_DEC\_PRECO DECIMAL(8, 2) NOT NULL,

PRO\_INT\_QUANTIDADE\_ESTOQUE INT NOT NULL,

PRO\_STR\_DESCRICAO VARCHAR(255),

PRO\_DATE\_DATA\_VALIDADE DATETIME NOT NULL,

PRO\_STR\_CATEGORIA CHAR(1) NOT NULL,

PRO\_BIT\_STATUS BIT NOT NULL DEFAULT 1

CONSTRAINT PK\_PRODUTO PRIMARY KEY (PRO\_INT\_ID),

CONSTRAINT UK\_PRODUTO\_NOME UNIQUE (PRO\_STR\_NOME)

);

CREATE TABLE PEDIDO\_CLIENTE

(

PED\_CLI\_INT\_ID INT IDENTITY(1,1),

CLI\_INT\_ID INT NOT NULL,

PRO\_INT\_ID INT NOT NULL,

PED\_CLI\_INT\_QUANTIDADE INT NOT NULL,

PED\_CLI\_DATE\_DATA DATETIME NOT NULL,

PED\_CLI\_STR\_FORMA\_PAGAMENTO CHAR(1) NOT NULL,

PED\_CLI\_STR\_STATUS CHAR(1) NOT NULL,

CONSTRAINT PK\_PEDIDO\_CLIENTE PRIMARY KEY (PED\_CLI\_INT\_ID),

CONSTRAINT FK\_CLIENTE FOREIGN KEY (CLI\_INT\_ID) REFERENCES CLIENTE(CLI\_INT\_ID),

CONSTRAINT FK\_PRODUTO FOREIGN KEY (PRO\_INT\_ID) REFERENCES PRODUTO(PRO\_INT\_ID)

);

CREATE TABLE COMPRA (

COM\_INT\_ID INT IDENTITY(1,1),

COM\_DATE\_DATA DATETIME DEFAULT GETDATE() NOT NULL,

COM\_DEC\_VALOR DECIMAL(8, 2) NOT NULL ,

COM\_STR\_FORMA\_PAGAMENTO CHAR(1) NOT NULL,

CONSTRAINT PK\_COMPRA PRIMARY KEY (COM\_INT\_ID)

);

CREATE TABLE PEDIDO\_COMPRA (

PED\_COM\_INT\_ID INT IDENTITY(1,1),

PRO\_FOR\_INT\_ID INT NOT NULL,

COM\_INT\_ID INT NOT NULL,

PED\_COM\_INT\_QUANTIDADE INT NOT NULL,

PED\_COM\_DEC\_VALOR DECIMAL(8, 2) NOT NULL,

PED\_COM\_STR\_STATUS CHAR(1) NOT NULL,

CONSTRAINT PK\_PEDIDO\_COMPRA PRIMARY KEY (PED\_COM\_INT\_ID),

CONSTRAINT FK\_PRODUTO\_FORNECEDOR FOREIGN KEY (PRO\_FOR\_INT\_ID) REFERENCES PRODUTO\_FORNECEDOR(PRO\_FOR\_INT\_ID),

CONSTRAINT FK\_COMPRA FOREIGN KEY (COM\_INT\_ID) REFERENCES COMPRA(COM\_INT\_ID)

);

INSERT INTO ENDERECO (END\_STR\_LOGRADOURO, END\_STR\_NUMERO, END\_STR\_COMPLEMENTO, END\_STR\_BAIRRO, END\_STR\_CIDADE, END\_STR\_CEP)

VALUES ('Rua do admin', '123', 'Casa', 'Tatuapé', 'São Paulo', '01234567');

INSERT INTO USUARIO (USU\_STR\_LOGIN, USU\_STR\_SENHA, USU\_STR\_NOME, USU\_STR\_SOBRENOME, USU\_STR\_EMAIL, USU\_STR\_CPF, USU\_STR\_TELEFONE, USU\_STR\_NIVEL\_ACESSO, USU\_DEC\_SALARIO, USU\_BIT\_STATUS, END\_INT\_ID)

VALUES ('admin', 'a665a45920422f9d417e4867efdc4fb8a04a1f3fff1fa07e998e86f7f7a27ae3','Rafael', 'Bessa', 'admin@fazendaurbana.com.br', '12345678900', '11999995555', '1', 4560.30, 1, 1);

Existem algumas considerações importantes a serem feitas sobre o script SQL que foi desenvolvido. Primeiramente, todas as entidades, que são as tabelas, possuem uma chave primária denominada “id”. Esta chave tem a propriedade de geração automática, conhecida como IDENTITY.

Adotamos um padrão de nomenclatura para os atributos. Este padrão faz referência às tabelas onde os atributos estão localizados, ao tipo de dado e ao próprio nome do atributo.

Os campos de CPF, CEP, TELEFONE e CNPJ foram anotados como CHAR. Isso permite que eles armazenem exatamente o número correto de caracteres. Optamos por não usar máscaras, pois isso poderia resultar em dados desnecessários sendo adicionados ao banco de dados.

Além disso, um administrador foi inserido junto com as tabelas. Isso permite que ele seja o primeiro a acessar o sistema e iniciar o gerenciamento. É importante notar que a senha do administrador já foi adicionada em forma de hash. Discutiremos mais sobre isso posteriormente.

O relacionamento entre as entidades Usuário e Endereço é de muitos para um (N:1). Isso significa que um endereço pode pertencer a vários usuários, mas um usuário tem apenas um endereço. Portanto, sempre que um usuário for inserido no banco de dados, será necessário fornecer as informações do seu endereço.

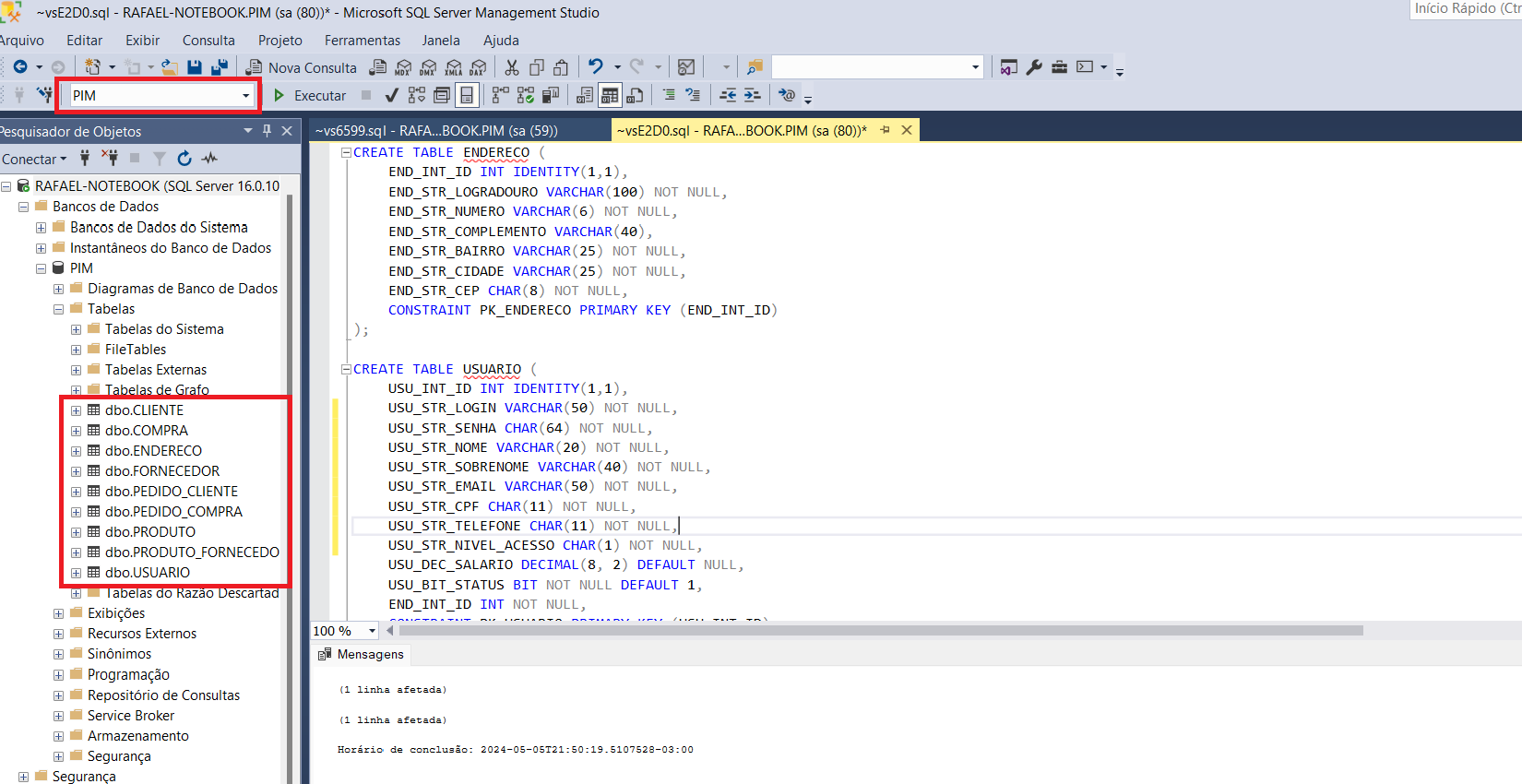


Figura 46 – Resultado do script SQL executado.

Agora vamos ilustrar como fica no SQL Server Management Studio. Mostrando tudo que existe na tabela usuário e endereço, já que o administrador já foi adicionado.

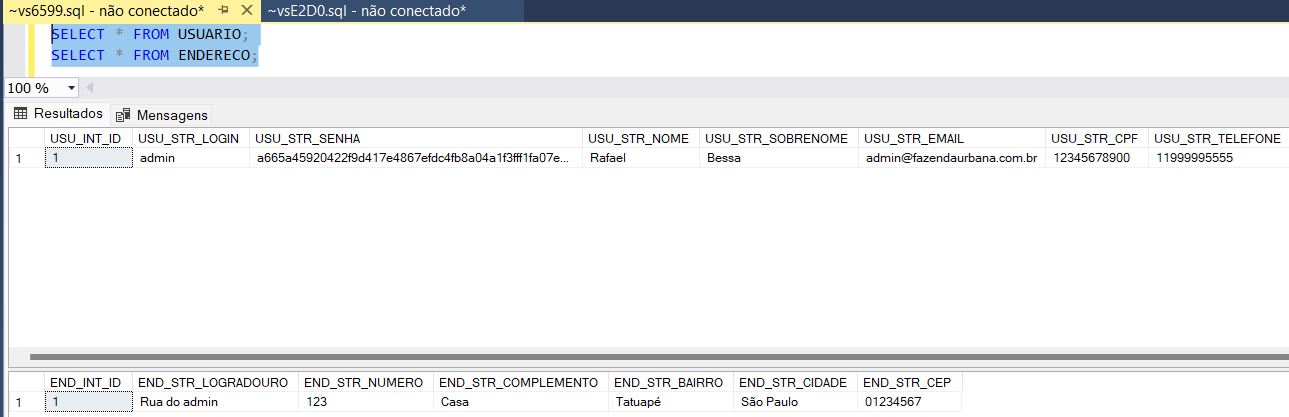


Figura 47 – Dados pré inseridos na tabela.

### Sistema orientado a objetos

Nosso sistema, como já discutido anteriormente, é projetado para facilitar a venda e compra de produtos. A arquitetura do sistema é estruturada em pacotes, cada um contendo classes, interfaces e enums que são agrupados com base em funcionalidades semelhantes.

Pacote DAO: Este pacote contém classes que são responsáveis pelo acesso ao banco de dados e pela execução de consultas SQL.

Pacote MODEL: Este pacote agrupa as classes que atuam como modelos para o sistema. A maioria dessas classes representa as entidades do banco de dados.

Pacote SECURITY: Este pacote é dedicado à segurança do sistema. Ele contém classes e outros componentes relacionados à proteção dos dados e operações do sistema.

Pacote SERVICE: As classes neste pacote são responsáveis por executar as regras de negócio e agrupar algoritmos relevantes para o sistema. Elas desempenham um papel crucial na prevenção do contato direto da aplicação com a camada de persistência (DAO).

Pacote CONSOLE: Este pacote foi criado para fins ilustrativos e contém classes que simulam a utilização do sistema usando o console da aplicação.

Ao longo do desenvolvimento deste projeto, é possível que novos pacotes, classes e funcionalidades sejam implementados. Inicialmente, isso pode ser visto como um exemplo de modelagem, formando o esqueleto da aplicação.

A seguir uma figura ilustrativa de como estão agrupadas as classes.

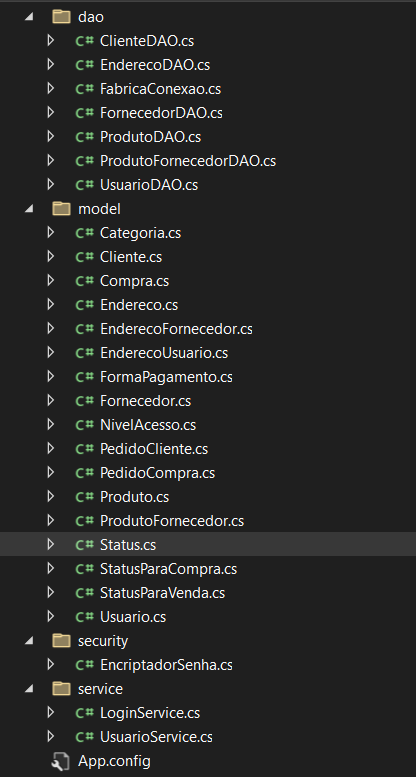


Figura 48 – Arquitetura de pacotes do sistema.

Levando em consideração a segurança do sistema, foi aplicado o RBAC (Role-Based Access Control). O RBAC é um sistema que gerencia o acesso dos usuários com base em suas funções dentro de uma organização. De acordo com o Red Hat, o RBAC é uma alternativa eficiente à necessidade de configurar o acesso individualmente para cada usuário. Ele permite que os administradores de TI configurem o nível de acesso necessário para todos os usuários com um cargo específico. Isso facilita a adição, modificação e remoção de permissões para todos os usuários de um grupo de uma só vez, além de permitir a rápida alteração do nível de acesso de um usuário ao atribuir ou remover uma função (Red Hat, 2024).

O sistema disponibiliza diferentes funcionalidades com base no nível de acesso do usuário que acabou de fazer login.

Se o usuário for um Administrador, terá acesso às opções de gerenciamento de usuários, produtos e fornecedores.

Se o usuário for um Funcionário, poderá gerenciar produtos, clientes e relatórios.

Se o usuário for um Cliente, serão apresentadas opções para possíveis compras.

Na aplicação, foi criado um <<enum>> para representar os níveis de acesso, conforme a figura a seguir.

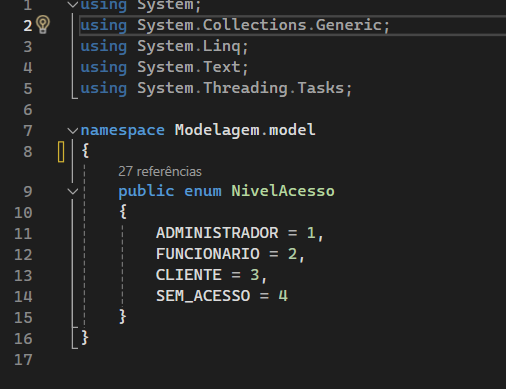


Figura 49 – Enum de nível de acesso.

A Classe Usuário tem um atributo do tipo NivelAcesso, conforme a Figura a seguir:

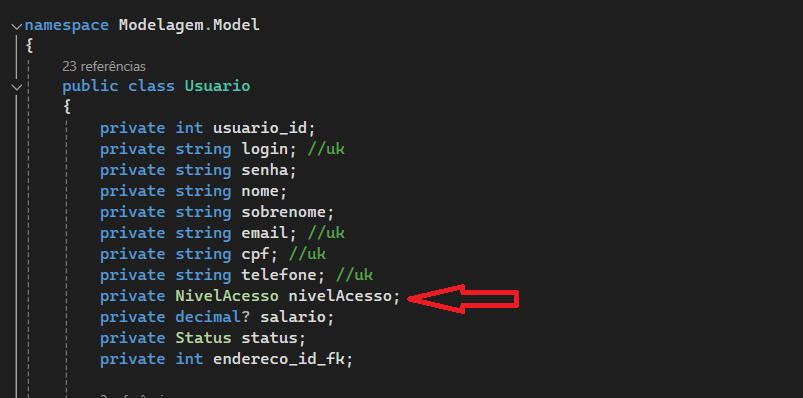


Figura 50 – Atributo NivelAcesso em Usuário.

O atributo nivelAcesso desempenha um papel crucial na determinação das funções acessíveis pelo usuário logado. Conforme ilustrado no enum, o valor 1 é atribuído ao administrador do sistema, enquanto o valor 2 é designado a um funcionário da empresa, neste contexto inicial, sem distinção interna.

Conforme abordado em seções anteriores, o Cliente também é categorizado como um Usuário e, portanto, é atribuído o valor de acesso 3.

Além disso, o nível de acesso é representado no banco de dados como CHAR(1). Essa representação não só é simples, mas também otimiza o uso do espaço de armazenamento.

Ao discutirmos a classe Usuário em nosso sistema, é crucial destacar a importância dos métodos sobrescritos equals e hashCode. Estes foram estrategicamente implementados para considerar dois usuários com o mesmo login e senha como sendo o mesmo objeto. Esta característica é fundamental para o funcionamento eficaz do nosso algoritmo de autenticação. A implementação desses métodos sobrescritos é ilustrada na figura subsequente.

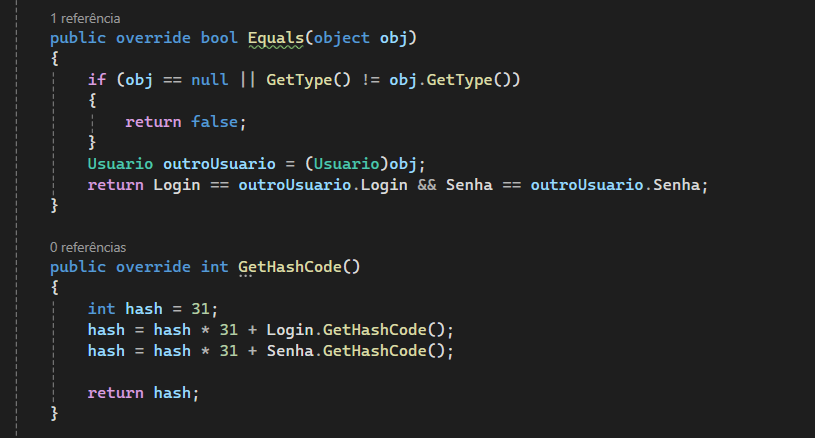


Figura 51 – Métodos Equals e Hashcode em Usuário.

Avançando para o pacote DAO, especificamente na classe UsuarioDAO, implementamos alguns métodos com o objetivo de realizar testes básicos no software. O método getUsuarioByLogin() foi implementado para receber uma string de login e buscar o usuário correspondente no banco de dados. Da mesma forma, o método getUsuarioByCpf() foi criado para buscar um usuário pelo CPF, uma exigência do projeto para melhorar a usabilidade. Além disso, o método cadastraNovoUsuarioNoSistema() foi desenvolvido para receber as informações de um usuário e registrar tanto o usuário quanto seu endereço no banco de dados.

A seguir como o método cadastraNovoUsuarioNoSistema foi implementado.

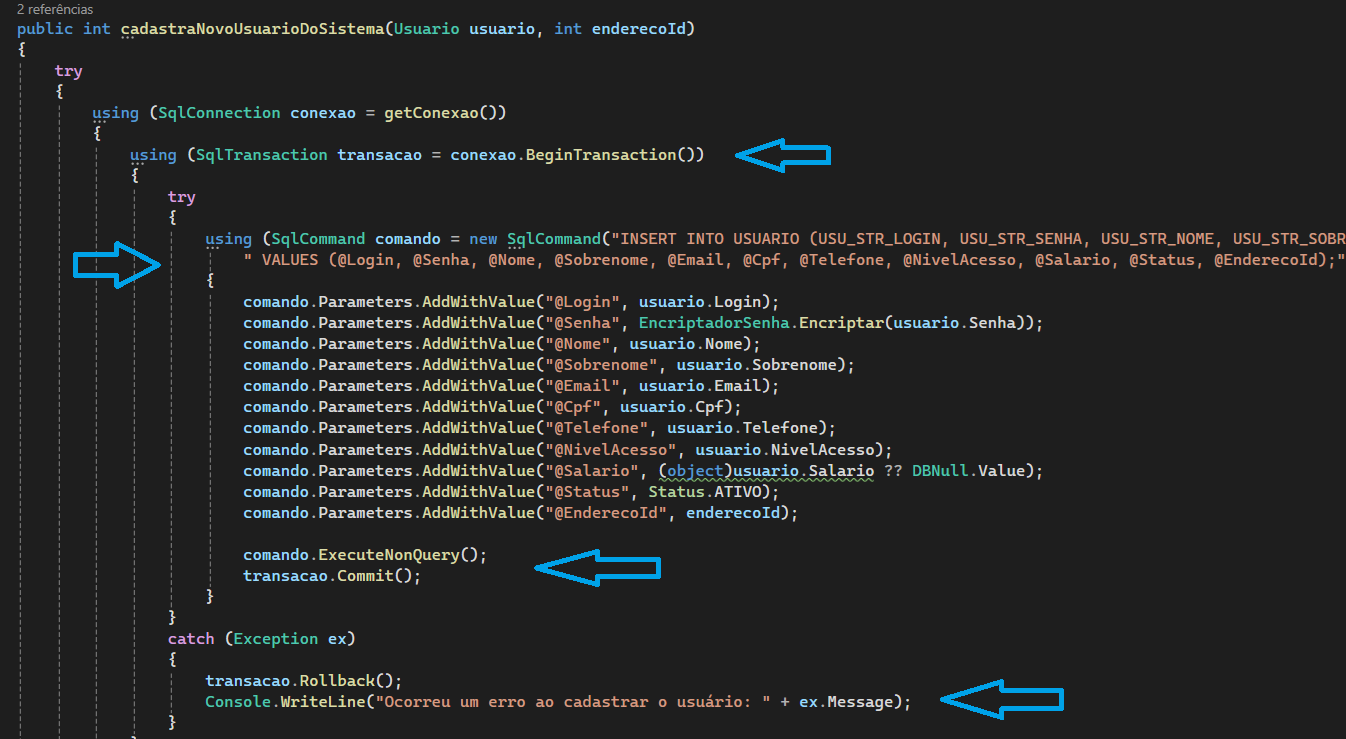


Figura 52 – Método cadastraNovoUsuarioNoSistema.

Para oferecer uma visão concisa do código-fonte, é importante destacar que nos métodos onde ocorrem operações de inserção, atualização ou exclusão no banco de dados - comandos conhecidos como DML (Data Manipulation Language) - é recomendável iniciar transações e tentar efetivar essas transações através de um commit. Caso ocorra algum problema, um rollback é realizado para garantir que nenhuma linha seja indevidamente alterada. Esta premissa foi cuidadosamente considerada ao implementar o método para inserir um novo usuário no sistema.

Além disso, vale ressaltar que a utilização de transações contribui para a integridade dos dados, pois permite que um conjunto de operações seja tratado como uma única unidade de trabalho. Assim, ou todas as operações são bem-sucedidas e os dados permanecem consistentes, ou, em caso de falha, todas as alterações são revertidas, evitando possíveis inconsistências. Esta abordagem é fundamental para manter a confiabilidade do sistema.

Ao explorarmos o pacote SECURITY, gostaríamos de destacar uma funcionalidade essencial do nosso sistema que visa garantir a segurança das informações dos usuários. Uma prática recomendada no desenvolvimento de software é armazenar as senhas no banco de dados em forma de hash.

Armazenar senhas como hashes ajuda a proteger as informações do usuário, mesmo no caso de uma violação de dados. Quando uma senha é transformada em um hash, ela é processada através de uma função que a converte em uma string única. Essa string, ou hash, não pode ser revertida para a senha original, o que significa que mesmo se os hashes de senha forem expostos, a senha original do usuário permanece segura.

Com base no trabalho de Queiroz (2022), a proteção de senhas é de suma importância, especialmente em situações onde o banco de dados possa ser comprometido. A criptografia desempenha um papel crucial nesse aspecto, transformando texto comum em sequências codificadas que parecem aleatórias. Isso impede que informações sensíveis, como senhas, sejam expostas a possíveis invasores. As funções de hash criptográficas são particularmente eficazes nesse contexto, pois convertem um texto simples em um texto cifrado, tornando extremamente difícil a reversão para o texto original.

A figura a seguir mostra a classe com o seu método estático, responsável por criar o hash da senha.

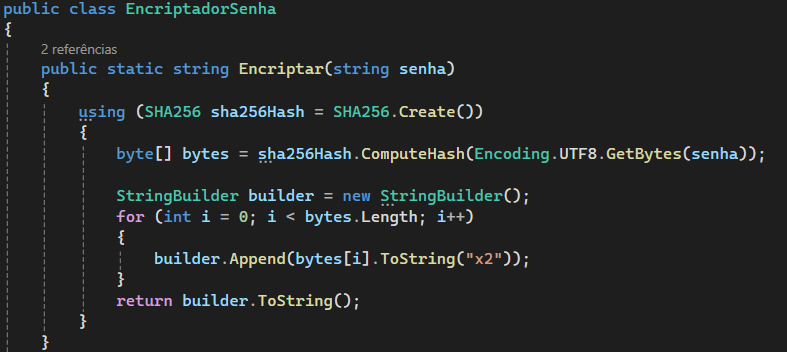


Figura 53 – Método Encriptar utilizando SHA256.

Optamos pelo modelo de hash SHA256 para nosso sistema. Este modelo transforma qualquer string em um hash de 64 caracteres que é armazenado no banco de dados. É importante enfatizar que, uma vez que a senha é criptografada, não há como descriptografá-la - é um processo unidirecional.

Por último, mas não menos importante, gostaríamos de apresentar o processo de ‘soft delete’ em um registro do banco de dados. Implementamos um enum com dois possíveis valores: inativo e ativo. Quando um usuário, fornecedor ou produto precisa ser excluído, seu registro permanece no banco de dados, mas o valor do status é alterado para ‘inativo’, que é representado pelo valor 0. Isso também explica o uso de CHAR(1) durante a criação das tabelas no atributo status.

A figura a seguir mostra o enum para o Status.

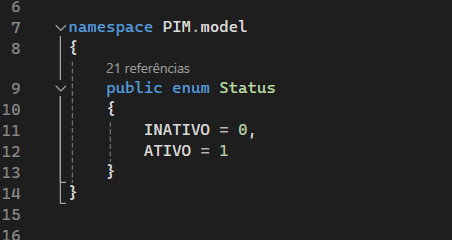


Figura 54 – Enum Status para soft delete.

### Testes de software

A fase inicial deste projeto enfoca primordialmente a modelagem de classes juntamente com o respectivo banco de dados, com o objetivo de identificar relações e funcionalidades relevantes para a gestão de uma fazenda urbana. Para possibilitar a execução de testes preliminares nesta fase, algumas funcionalidades do sistema foram implementadas. Estas incluem o login, o registro de clientes, a pesquisa de um usuário por CPF e o cadastro de novos usuários.

Os testes que serão discutidos nesta seção são todos baseados na técnica de teste de caixa preta. Isto significa que vamos simular a interação de um usuário ou cliente com a aplicação, sem a necessidade de acessar a estrutura interna do software.

De acordo com Eler (s.d.), o teste funcional ou caixa-preta tem o objetivo de verificar se uma funcionalidade, função, método, classe ou programa está de acordo com os requisitos especificados. Os casos de teste são definidos com base na especificação do software (descrições, casos de uso, requisitos, diagramas, etc).

Neste documento, apresentaremos três funcionalidades principais do sistema. A primeira é a funcionalidade de login, a segunda é a funcionalidade de cadastro de clientes e a terceira é a funcionalidade de busca de usuários pelo CPF.

### Testando a funcionalidade de Login

As figuras a seguir ilustram o processo de login no sistema. Quando o banco de dados foi inicialmente criado, um usuário administrador foi automaticamente inserido. Portanto, o primeiro usuário do sistema é este administrador, cujo login é “admin” e a senha foi definida como “123”.

Ao tentar fazer login, o sistema identifica o nível de acesso do usuário e exibe na tela as funcionalidades disponíveis para esse nível de acesso.

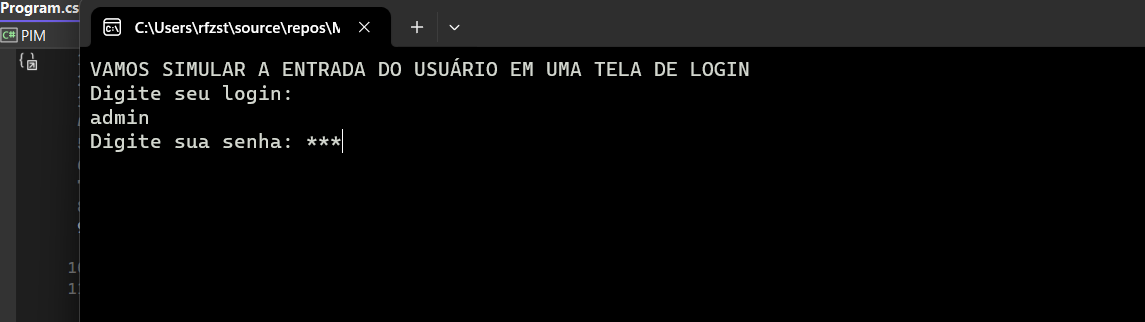


Figura 55 – Tela de login em console.

Após pressionar Enter:

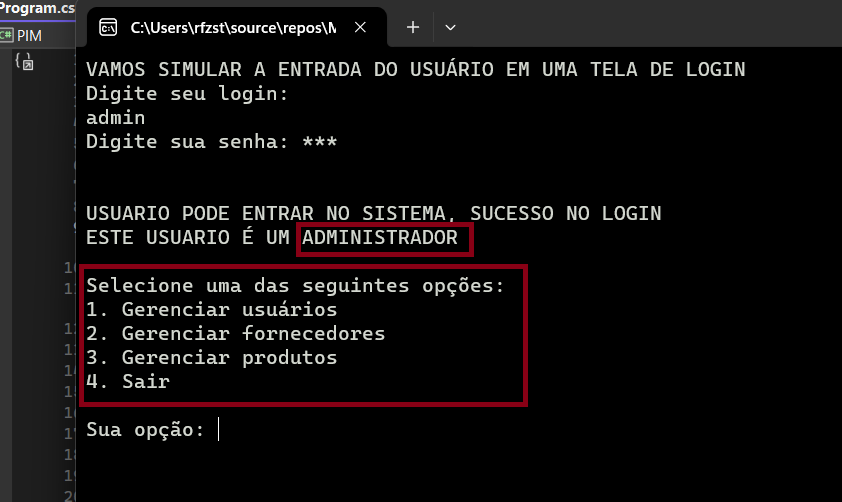


Figura 56 – Tela console após acesso do administrador.

### Testando a funcionalidade de cadastro de clientes

Durante o processo de cadastro, a aplicação em console solicitará algumas informações. Devido à cardinalidade do relacionamento entre as entidades, o endereço do cliente é solicitado antes das suas informações pessoais. Isso ocorre porque é necessário criar primeiro um registro na tabela de endereços, obter o ID desse registro e, em seguida, alocá-lo na chave estrangeira do usuário.

É importante salientar que, por padrão, a coluna ‘salário’ é automaticamente definida como NULL para o cliente, e o status é definido como 1, que representa ‘ATIVO’.

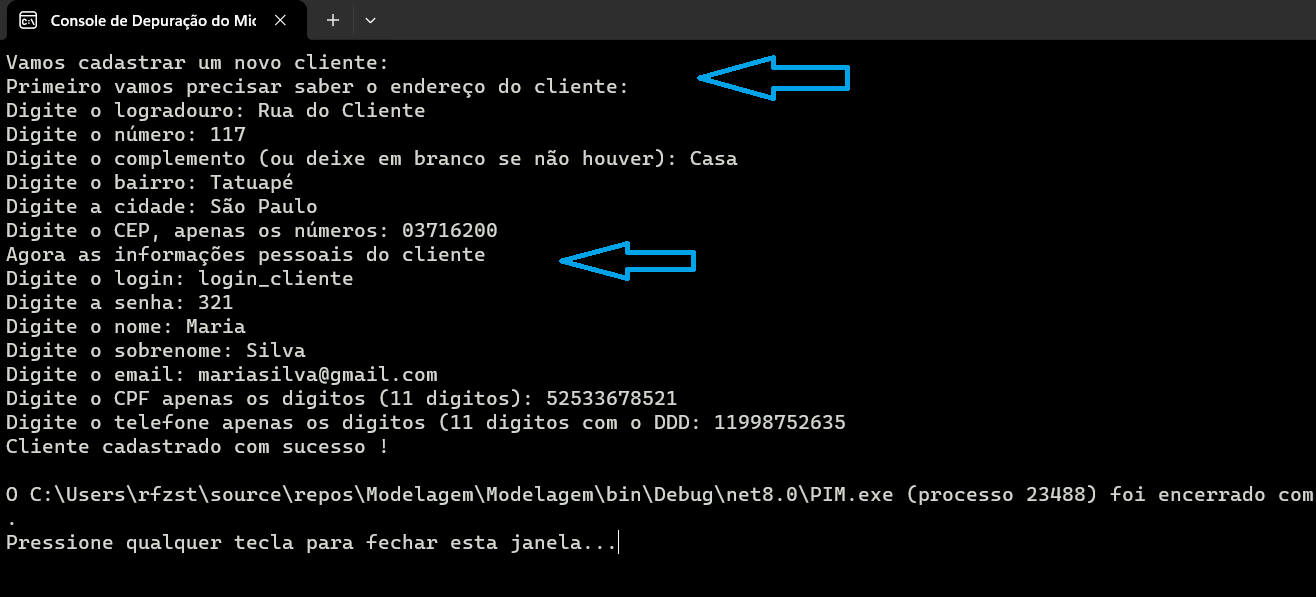


Figura 57 – Cadastro de cliente via console.

A figura a seguir mostra como esses dados foram alocados no banco de dados.

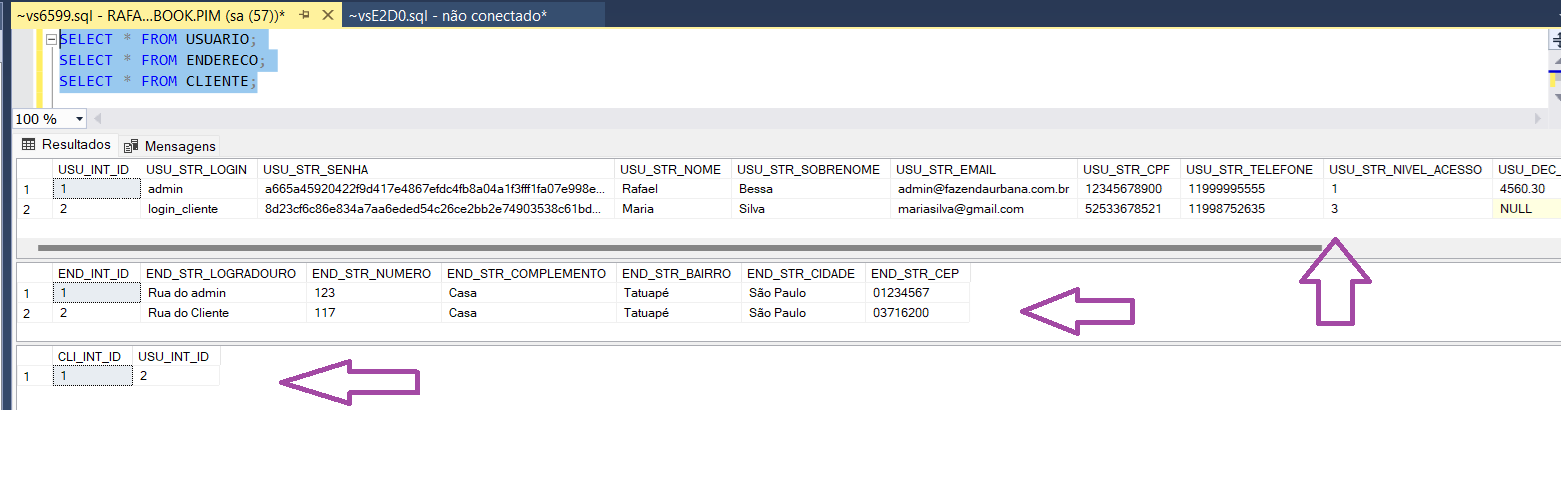


Figura 58 – Dados do cliente inseridos no banco.

A próxima funcionalidade que será testada é a busca de usuários no sistema por meio do CPF. Vale lembrar que o cliente já se cadastrou anteriormente com o CPF 52533678521.

A Figura a seguir ilustra a busca de um usuário específico, utilizando o CPF como critério de busca.

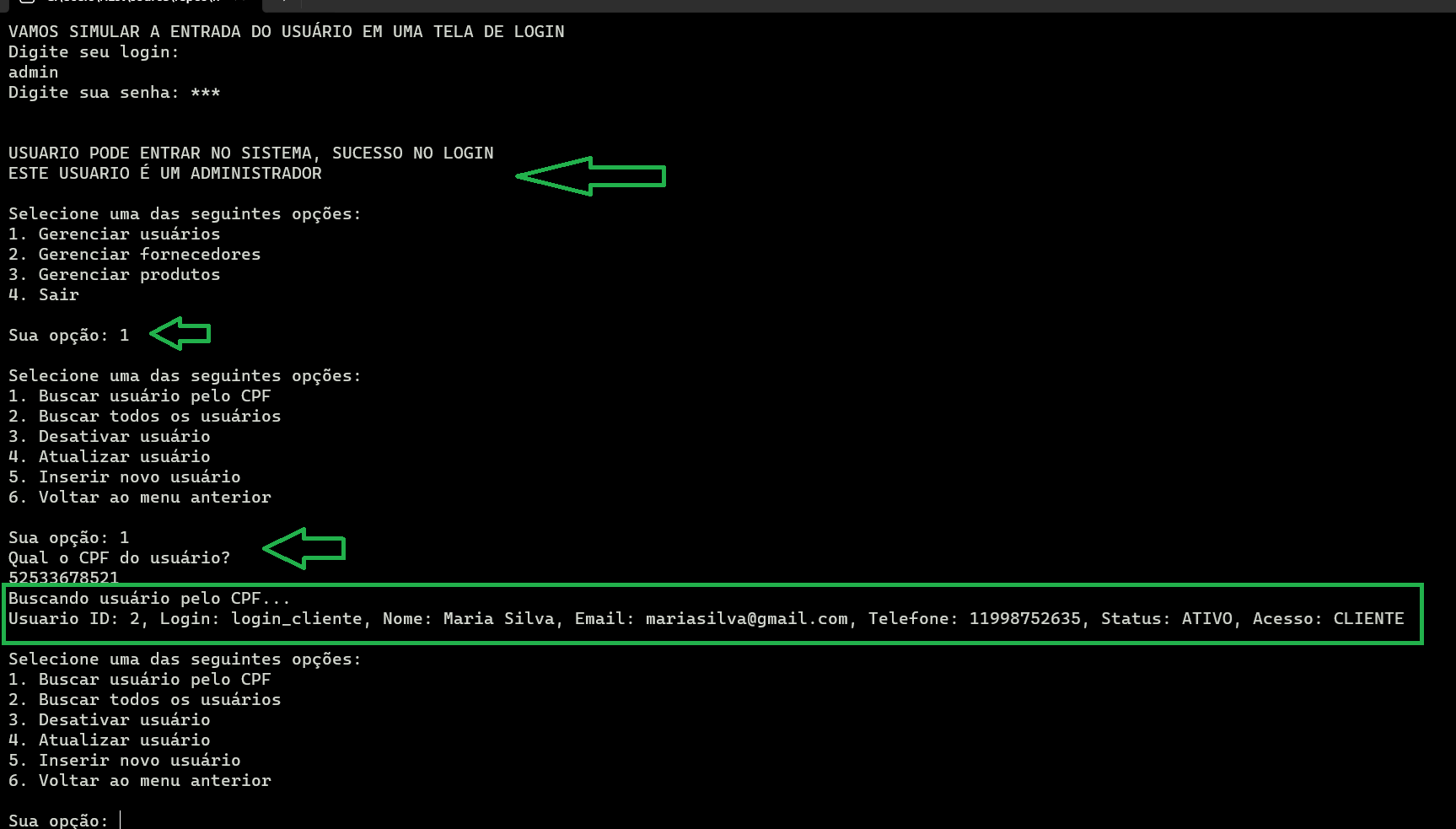


Figura 59 – Funcionalidade de buscar pelo CPF.

Neste trabalho, implementamos um número limitado de funcionalidades, com o principal objetivo de analisar e modelar o sistema em conjunto com seu banco de dados.

Em seguida, apresentaremos uma planilha de testes, semelhante àquelas utilizadas por profissionais de **QA (Quality Assurance)** em seu cotidiano. QA é uma prática que visa garantir a qualidade dos produtos através de processos sistemáticos de revisão e teste. Esta planilha detalhará como testar as funcionalidades, os resultados esperados e os critérios de aceitação do projeto.

### Plano de teste

|  |  |  |  |
| --- | --- | --- | --- |
| **Funcionalidades** | **Comportamento Esperado** | **Verificações** | **Critérios de Aceite** |
| **Cadastro de**  **Clientes** | Ao digitar login, senha, nome, sobrenome, e-mail, cpf, telefone e endereço completo, irá efetuar um cadastro no sistema e o usuário deverá ser redirecionado para a tela de login.  No caso da aplicação console, receberá uma mensagem de “cliente cadastrado com sucesso” e encerrará o sistema. | * Todos os campos devem ser obrigatórios. * Exibir uma mensagem de confirmação em caso positivo. * Redirecionar o usuário para tela de login. * Exibir a mensagem de falha em caso de usuário existente ou algum dado único já estiver cadastrado por outro usuário. * Exibir mensagem de falha ao tentar cadastrar algum campo com string vazia. | * Cliente cadastrados corretamente na tabela de usuário e na tabela de cliente. Com as devida chave estrangeira. * Dados corretamente inseridos no banco de dados, junto com sua senha em forma de Hash. |
| **Login** | Ao digitar seu login e senha corretamente o usuário irá logar no sistema, e receberá uma mensagem dizendo seu nível de acesso.  Ao tentar se logar e falhar receberá uma mensagem.  Se o usuário existir e a senha estiver errada, mensagem de “Senha incorreta”, o login não existir mensagem de “Usuário não encontrado no banco de dados” | * Login no Sistema com sucesso * Login Inválido * Senha Incorreta | * Ter acessibilidade no sistema * Redimensionar a Tela |
| **Cadastro de Usuários via Administrador** | Ao logar com sucesso, receberá a mensagem que é um administrador.  Selecionar a opção número 1, de “gerenciar usuários”, depois escolher a opção 5 “Inserir novo usuário”  Em seguida digitar as informações pendentes na tela de console, é possível cadastrar funcionários, administrador e clientes, porém, o cadastro de clientes via administrador é utilizado apenas em casos especiais.  Deve ser preenchido obrigatoriamente o endereço do usuário ao cadastrar um usuário novo.  Campos como: login, email, telefone e cpf são únicos, se houver outro usuário com valores já cadastrados aparecerá um exception na tela de console. | * Somente administrador possui a funcionalidade de inserir novos usuários do sistema diretamente * Preservação de chaves únicas do banco de dados, garantindo a unicidade de dados já cadastrados. * Sempre será necessário informar o endereço quando a funcionalidade de cadastrar usuário for usada. | * Informações corretamente cadastradas no banco de dados, junto as seus devidas restrições de chaves * Senha cadastrada em forma de Hash no banco de dados |

### Manual de uso do sistema

Para iniciarmos o uso do sistema, devemos primeiro entender os Diferentes Níveis de Acesso, que alteram as funções disponíveis para cada usuário.

**Administrador:** Usuário com maior nível de acesso possível no sistema, consegue utilizar todas as funções, bem como definir o nível de acesso dos demais usuários.

**Funcionário:** Usuário com acesso a funções básicas do sistema, podendo realizar todas as tarefas diárias, porém, não pode interferir no cadastro e edição de outros funcionários e fornecedores.

**Cliente:** Usuário com o menor nível de acesso do sistema, podendo somente realizar login e futuramente poderá visualizar e comprar os itens disponíveis em estoque.

Abaixo será apresentado o diagrama de caso de uso para melhor compreensão dos níveis de acesso e fluxos que cada usuário poderá seguir:

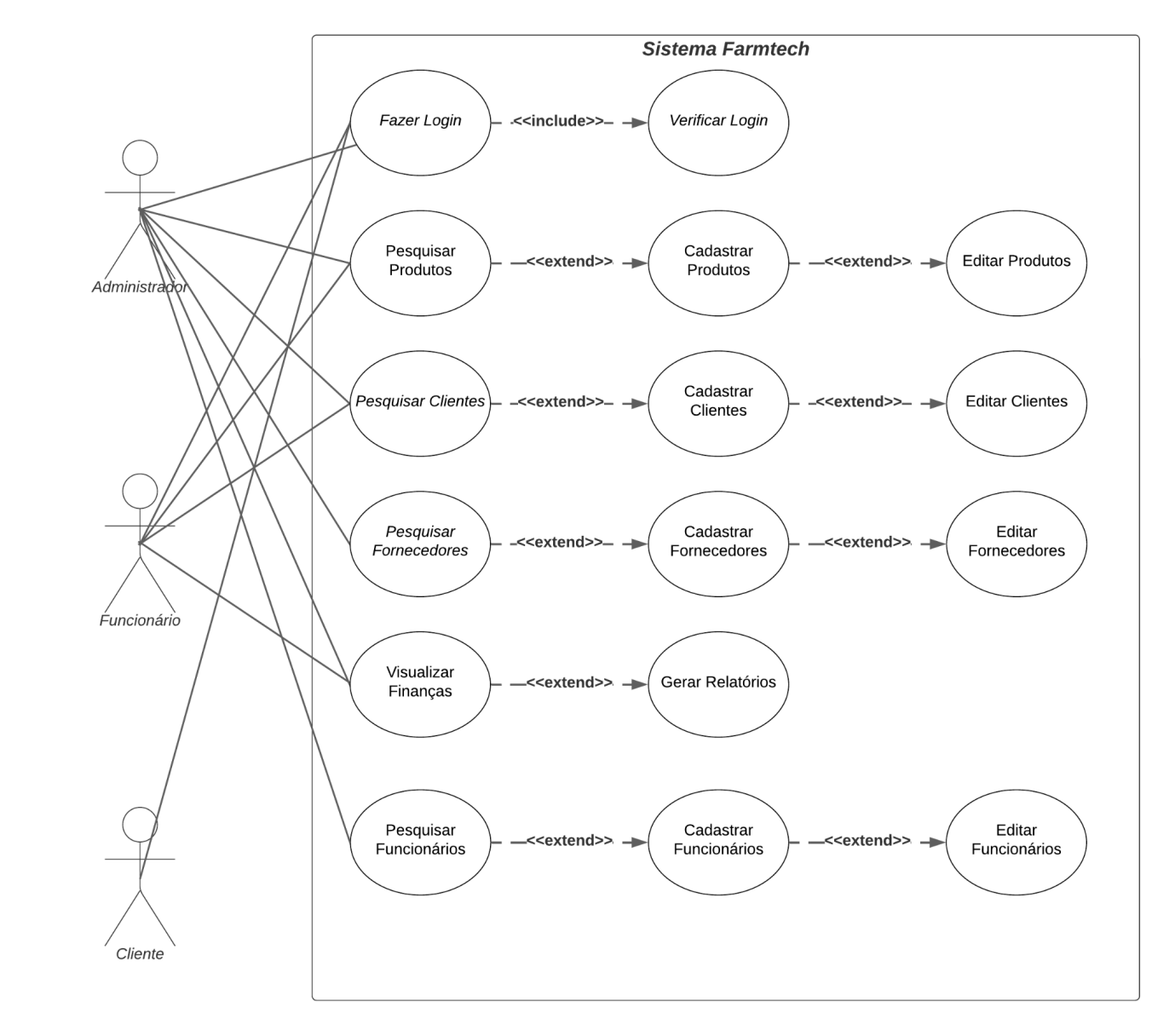


Figura 60 - Diagrama de Caso de Uso do sistema FarmTech

### Login

Para realizar o login, o usuário deverá estar cadastrado no sistema, tendo em vista que é um sistema interno, (somente os administradores têm o direito de cadastrar novos usuários).

O login do usuário é, por padrão, o seu número de cpf, e sua senha, os 4 primeiros dígitos do mesmo, podendo ser alterada após o primeiro acesso. Após preencher os campos de login e senha, o usuário deverá clicar no botão “Entrar!”. Após a realização correta do login, o usuário irá se encontrar na página inicial do sistema.

### Visualização e cadastro de itens

Uma vez logado no sistema, o usuário terá à sua disposição uma série de funcionalidades disponíveis. Na parte esquerda da tela dos desktops (Figura 8), e no canto superior esquerdo dos dispositivos móveis, em forma de menu hambúrguer (Figura 9), o usuário encontrará todas as principais abas do sistema, como o Estoque, Clientes, Fornecedores, Financeiro e RH. Ao entrar em cada uma dessas abas, o usuário perceberá similaridade entre as mesmas, tendo em vista os padrões adotados para melhorar a usabilidade do sistema.

Dentro de cada aba, a função de cadastro estará disponível a depender do nível de acesso mencionado anteriormente, esta função é basicamente igual para todas as abas, com pequenas mudanças nos formulários para a correta adequação ao tipo de informação que será salva. Ao clicar em cadastrar, uma nova tela aparecerá, em formato de modal para o usuários de desktop (Figura 11), e em formato de formulário para os usuários mobile (Figura 15), a partir deste momento, o usuário deverá preencher todos os campos pedidos, utilizando o cadastro de produtos como exemplo, deverão ser preenchidos os campos de descrição, quantidade, fornecedor, preço e data de validade. Após preencher todos os campos, basta clicar em “Cadastrar!” para finalizar o processo e realizar o cadastro.

### Edição/atualização dos dados cadastrados

Após o cadastro do item específico, ele ficará à mostra para ser utilizado conforme o necessário, e, além disso, haverá uma função chamada “Ver Detalhes”, que é utilizada tanto para ver todos os detalhes daquele item, e também para realizar quaisquer edições necessárias nos mesmos. Para realizar a edição, basta clicar em “Ver Detalhes”, então, uma nova tela se abrirá, em formato de modal para os usuários em desktop, e em formato de formulário para usuários mobile, a partir deste momento, basta alterar os campos necessários e clicar na função “Atualizar!”, que estará destacada pelo botão e pelo uso de cores. Após isto, a informação será atualizada e estará disponível para ser visualizada e alterada novamente conforme necessidade.

### Relatórios da aba financeiro

Ao entrar na aba “Financeiro” (Figura 29 e Figura 30), o usuário verá funções diferentes das já mostradas anteriormente, isso se deve ao fato da natureza diferente da aba, que não servirá apenas para a visualização, cadastro e alteração de informações. Nesta aba estão contidas as informações do fluxo financeiro da empresa, contendo informações de despesas, receita e balanço geral dos meses. Na mesma aba, o usuário terá a função “Gerar Relatório” que, quando utilizada, irá gerar um arquivo baixável, contendo todas informações financeiras do mês em que a mesma foi utilizada.

### Gestão estratégica de recursos humanos

A gestão estratégica de recursos humanos procura desenvolver estratégias, que otimizem o desempenho de forma que impulsione os resultados organizacionais da fazenda urbana, sem impactar o meio ambiente. O departamento de recursos humanos é indispensável, alinhando metas e objetivos do estabelecimento. Definindo cargos, salários e recrutando funcionários.

A gestão estratégica envolve implementação e concepção de um conjunto de políticas e práticas, garantindo que as pessoas consigam alcançar seus objetivos. Para isso serão feitas contratações com foco em atrair candidatos qualificados com boa comunicação, e que preferencialmente tenham trabalhado na área.

Conforme a fazenda cresce e o número de clientes aumenta, é necessário expandir o time de funcionários. Os recursos humanos contratarão empregados para as áreas que tiverem maior crescimento e demanda no local. Vagas de emprego poderão ser abertas quando um funcionário for demitido ou transferido para outra empresa.

O departamento de recursos humanos elaborará através de treinamentos em suas funções exercidas para conseguirem aprimorar-se profissionalmente, avaliando o funcionário e trazendo feedbacks. Os treinamentos têm duração de três meses e haverá um sistema de mentoria, na qual o mentor auxiliará o contratado nas suas funções, guiando-os para atingirem seus objetivos na fazenda, auxiliando-os para melhor adaptação ao ambiente de trabalho, e para exercerem seus cargos com maior entendimento.

Métricas de desempenho serão utilizadas á fim de verificar como cada empregado está desempenhando sua função. Gráficos de linha mostrarão o resultado que cada empregado está gerando usando como base o grau de satisfação dos clientes, além de metas variadas para cada setor, podendo variar entre taxa de conversão, e satisfação de clientes.

O departamento de recursos humanos tem como um de seus planejamentos manter relações estáveis entre colegas de trabalho e funcionários, proporcionando um ambiente favorável aos empregados, garantindo valorização e bem estar à eles.

As avaliações de clientes poderão acarretar em mudanças, como melhorias no sistema e correções. O departamento de recursos humanos concederá suporte aos funcionários, além de adaptações sempre que houver mudanças significativas no ambiente. Em ambientes de trabalho, conflitos podem ocorrer entre os funcionários. Caso houver, o departamento de recursos humanos deverá fazer uma mediação imparcial, analisando os argumentos de cada área e o motivo do conflito, procurando uma solução pacífica entre as partes, evitando prejudicar os envolvidos, trazendo uma resolução de problemas favorável a todos.

De acordo com Andrade (2018), a inovação é impulsionada pelo mercado consumidor, que obriga as empresas a aprimorarem seu conhecimento, além de estarem preparadas para constantes mudanças. Consequentemente, serão feitas inovações e adaptações no sistema, juntamente de atualizações. Portanto novas funções serão disponibilizadas conforme a necessidade do mercado, facilitando a usabilidade do sistema para o cliente.

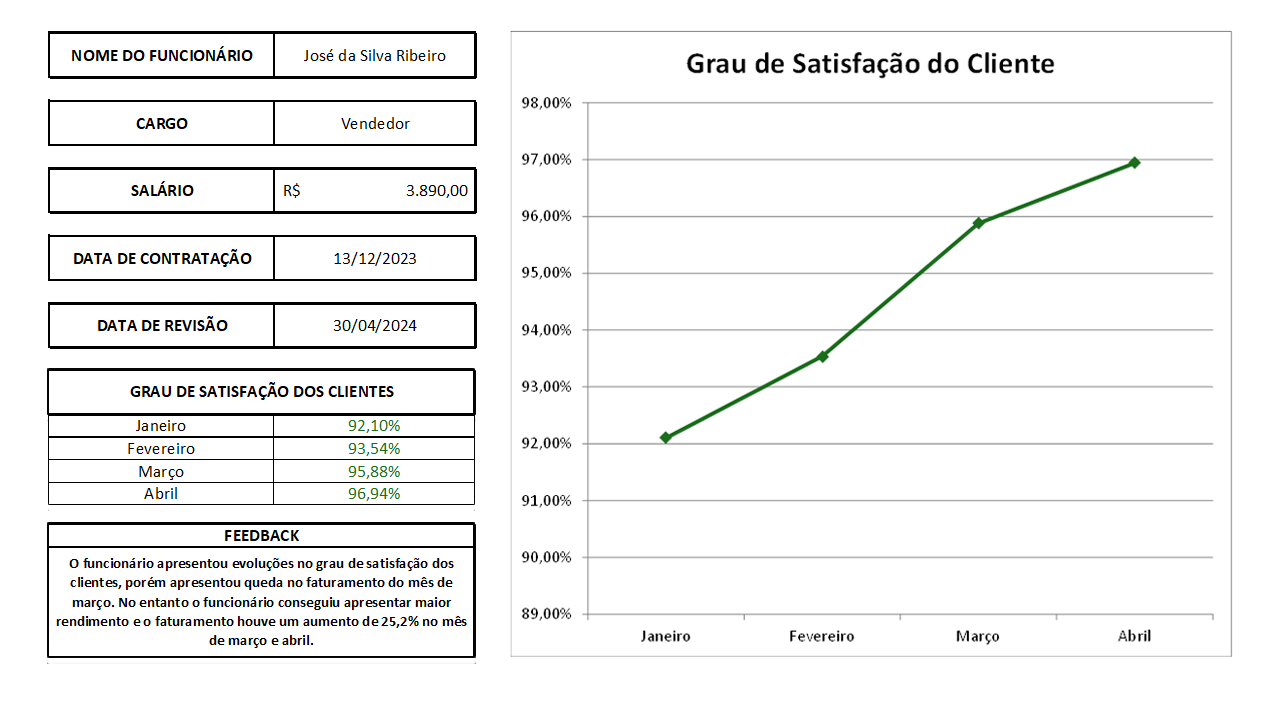


Figura 61 - Métricas de desempenho de um funcionário exemplo

1. CONCLUSÃO

Este trabalho teve como objetivo principal a modelagem de um sistema de gerenciamento inovador para uma fazenda urbana. O sistema foi projetado utilizando a linguagem de programação C# e o banco de dados SQL SERVER, duas ferramentas poderosas e amplamente utilizadas na indústria de tecnologia.

Um aspecto crucial deste projeto foi o desenvolvimento de protótipos de telas para o sistema. Estes protótipos foram cuidadosamente projetados no Figma, uma plataforma líder em design de interface do usuário. O uso do Figma permitiu a criação de um design intuitivo e amigável, que pode facilitar a interação do usuário com o sistema.

Além disso, foram criados diagramas de classes e casos de uso detalhados. Estas ferramentas de modelagem visual foram fundamentais para estruturar o sistema e definir suas funcionalidades. Elas forneceram uma representação clara da estrutura do sistema e dos processos que ele deve realizar.

Embora o sistema não tenha sido completamente implementado, a modelagem detalhada realizada neste trabalho fornece uma base sólida para futuros desenvolvimentos. Acreditamos que, uma vez implementado, este sistema tem o potencial de transformar a maneira como as fazendas urbanas são gerenciadas.

Em conclusão, este trabalho contribuiu significativamente para a agricultura urbana ao modelar um sistema de gerenciamento eficaz e fácil de usar. No futuro, esperamos que este sistema possa ser implementado e adaptado para atender a diversas fazendas urbanas, promovendo a sustentabilidade e a eficiência na agricultura urbana.

REFERÊNCIAS BIBLIOGRÁFICAS

ABNT NBR 9126-1. (2003). Qualidade de software. Associação Brasileira de Normas Técnicas.

Albahari, J., & Albahari, B. (2017). C# 7.0 in a Nutshell: The Definitive Reference. O'Reilly Media.

Ambler, S., & Sadalage, P. J. (2006). Refactoring Databases: Evolutionary Database Design. Addison-Wesley.

Assembleia Geral das Nações Unidas. Transformando nosso mundo: a Agenda 2030 para o Desenvolvimento Sustentável, 2015. Disponível em: http://bit.ly/2030agenda. Acesso em: 20 abr. 2024.

Bazerghi, C., McKay, F. H., & Dunn, M. (2016). The Role of Food Banks in Addressing Food Insecurity: A Systematic Review. Journal of Community Health, 41(4), 732-740.

Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Kern, J. (2001). Manifesto for Agile Software Development.

Blanchard, O. (2017). Macroeconomics. Pearson.

Boehm, B. (1988). A spiral model of software development and enhancement. ACM SIGSOFT Software Engineering Notes.

Booch, G., Rumbaugh, J., & Jacobson, I. (1999). The Unified Modeling Language User Guide. Addison-Wesley Professional.

Chen, P. P. (1976). The entity-relationship model: toward a unified view of data. ACM Transactions on Database Systems (TODS), 1(1), 9-36.

Chiavenato, I. (2014). Gestão de Pessoas: O novo papel dos recursos humanos nas organizações. Elsevier.

CONECTA BRASIL. A importância de ESG nas organizações, o que é e como aplicar, 18 set. 2023. Disponível em: https://conectabrasil.org/#/blogs/details/a-importancia-do-esg. Acesso em: 20 abr. 2024.

Dessler, G., & Várkonyi, A. (2017). Human Resource Management. Pearson.

Date, C. J. (2013). An introduction to database systems. Pearson Education India.

DEVMEDIA. Trabalhando com Enums em C#. Disponível em: <https://www.devmedia.com.br/trabalhando-com-enums-em-csharp/26059>. Acesso em: 5 maio 2024.

EcoInnovate. (2021). Sustainability in Urban Agriculture. Journal of Environmental Technology.

ELER, Marcelo Medeiros. Aula 05 - Teste de Software: Conceitos Básicos e Teste Funcional. Escola de Artes, Ciências e Humanidades - Universidade de São Paulo, s.d. Disponível em: [https://edisciplinas.usp.br/pluginfile.php/8005446/mod\_resource/content/1/05 - Teste de Software\_ Conceitos básicos e Teste Funcional.pdf](https://edisciplinas.usp.br/pluginfile.php/8005446/mod_resource/content/1/05%20-%20Teste%20de%20Software_%20Conceitos%20b%C3%A1sicos%20e%20Teste%20Funcional.pdf). Acesso em: 07 maio 2024.

Elmasri, R., & Navathe, S. (2016). Fundamentals of database systems. Pearson.

Freeman, E., & Freeman, E. (2020). Head First Design Patterns: A Brain-Friendly Guide. O'Reilly Media.

Freeman, E., & Freeman, E. (2020). Head First Design Patterns: Building Extensible and Maintainable Object-Oriented Software. O'Reilly Media

Foley, J. A., Ramankutty, N., Brauman, K. A., Cassidy, E. S., Gerber, J. S., Johnston, M., ... & Zaks, D. P. M. (2011). Solutions for a cultivated planet. Nature, 478(7369), 337-342.

FOLHA DE SÃO PAULO. Fome de quê? Soluções que inspiram. Folha Social+, 27 set. 2023. Disponível em: https://www1.folha.uol.com.br/folha-social-mais/2023/09/boas-praticas-no-combate-a-fome-no-brasil-sao-a-nova-causa-do-ano.shtml. Acesso em: 20 abr. 2024.

Fowler, M. (2004). UML Distilled: A Brief Guide to the Standard Object Modeling Language. Addison-Wesley Professional.

GAMMA, E.; HELM, R.; JOHNSON, R.; VLISSIDES, J. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Professional, 1995.

GROBA, Paula. Brasil é confirmado como sede da COP 30 em 2025. Agência Senado, 12 dez. 2023. Disponível em: https://www12.senado.leg.br/noticias/materias/2023/12/12/brasil-e-confirmado-como-sede-da-cop-30-em-2025#:~:text=O%20Brasil%20foi%20formalmente%20confirmado,21%20de%20novembro%20de%202025. Acesso em: 20 abr. 2024.

HEUSER, Carlos A. Projeto de Banco de Dados. 4ª ed. Porto Alegre: Bookman, 1998.

International Organization for Standardization. (2018). ISO 9241-11:2018 Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts. ISO.

ISO 10007. (2003). Quality management – Guidelines for configuration management. International Organization for Standardization.

ISO 9000. (2015). Quality management systems — Fundamentals and vocabulary. International Organization for Standardization.

ISO 9001. (2018). Quality management systems — Requirements. International Organization for Standardization.

ISO 25000. (2014). Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE. International Organization for Standardization.

Kramer, M. R., & Porter, M. E. (2011). Creating Shared Value. Harvard Business Review. Retrieved from https://hbr.org/2011/01/the-big-idea-creating-shared-value

Krugman, P., & Wells, R. (2018). Economics. Worth Publishers.

LADIES THAT UX BRASIL. Protótipos: Baixa, média ou alta fidelidade? Disponível em: https://medium.com/ladies-that-ux-br/prot%C3%B3tipos-baixa-m%C3%A9dia-ou-alta-fidelidade-71d897559135. Acesso em: 05/05/2024.

Mankiw, N. G. (2014). Principles of economics. Cengage Learning.

Martin, R. C. (2008). Clean Code: A Handbook of Agile Software Craftsmanship. Prentice Hall.

Milkovich, G. T., & Newman, J. M. (2016). Compensation. McGraw-Hill Education.

Mintzberg, H. (1989). Mintzberg on Management: Inside Our Strange World of Organizations. Free Press.

MOTA, Diego. Next Billion Users: Você já ouviu falar? Disponível em: https://brasil.uxdesign.cc/next-billion-users-voce-ja-ouviu-falar-c60ccd3e7490. Acesso em: 05/05/2024.

Myers, G. J., Sandler, C., & Badgett, T. (2011). The art of software testing. John Wiley & Sons.

OLIVEIRA, L. B. R. Data Access Object (DAO). In: SSC 621: Análise e Projeto Orientados a Objetos. 2015. 2 p. Apresentação de PowerPoint. Disponível em: <https://edisciplinas.usp.br/pluginfile.php/385964/mod_resource/content/2/Aula11_PatternsDAO.pdf>. Acesso em: 03 mai. 2024.

OLIVEIRA, Paulo. Design Thinking: A inovação em processos de negócio. Disponível em: https://www.sienge.com.br/blog/design-thinking-paulo-oliveira/. Acesso em: 05/05/2024.

ONU BRASIL. Sobre o nosso trabalho para alcançar os Objetivos de Desenvolvimento Sustentável no Brasil. Disponível em: https://brasil.un.org/pt-br/sdgs. Acesso em: 20 abr. 2024.

POISSON, André. Geografia - Espaço Geográfico. São Paulo: Scipione, 2004. Disponível em: https://www.poisson.com.br/livros/geo/geo1/GEO1.pdf. Acesso em: 07 maio 2024. (Pg. 42, Cap. 3, Inovação).

Pressman, R. S. (2014). Engenharia de Software: Uma Abordagem Profissional.

QUEIROZ, Lucas Grisi Oliveira de. Funções de Hash e Gerenciamento de Senhas. Recife: Federal University of Pernambuco, 2022.

RED HAT. O que é controle de acesso baseado em função? Disponível em: <https://www.redhat.com/pt-br/topics/security/what-is-role-based-access-control>. Acesso em: 6 maio 2024.

ROCK CONTENT. Design Thinking: O que é, como aplicar e cases de sucesso. Disponível em: https://rockcontent.com/br/blog/design-thinking/. Acesso em: 05/05/2024.

Robbins, S. P., & Judge, T. A. (2018). Comportamento organizacional. Pearson.

Schein, E. H. (2010). Organizational Culture and Leadership. John Wiley & Sons.

SERASA EXPERIAN. O que é e qual a importância do RH para a empresa? Disponível em: https://www.serasaexperian.com.br/carreiras/blog-carreiras/rh-departamento-funcionarios-e-importancia-para-a-empresa/. Acesso em: 05/05/2024.

SESI RS. O papel do RH no treinamento e desenvolvimento de profissionais. Disponível em: https://blog-educacao.sesirs.org.br/o-papel-do-rh-no-treinamento-e-desenvolvimento-de-profissionais/. Acesso em: 05/05/2024

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2011). Database system concepts.

Sommerville, I. (2015). Software Engineering. Addison-Wesley.

Sullivan, R. (2009). Kofi Annan and the UN Global Compact: A test of leadership. Global Governance: A Review of Multilateralism and International Organizations, 15(4), 405-421.

Tarasuk, V., Dachner, N., & Loopstra, R. (2014). Food banks, welfare, and food insecurity in Canada. British Food Journal, 116(9), 1405-1417.

Troelsen, A., & Japikse, P. (2017). Pro C# 7: With .NET and .NET Core. Apress.

UNDP. (2021). Sustainable Development Goals and Their Implementation. United Nations Development Programme.

United Nations. (2015). Transforming our world: the 2030 Agenda for Sustainable Development. Retrieved from <https://sustainabledevelopment.un.org/post2015/transformingourworld/publication>