

A TECNOLOGIA DA INFORMAÇÃO APLICADA À GESTÃO DE UMA LOJA DE ARTIGOS AGROPECUÁRIOS

Liniker Gabriel da Silva²⁰

Vitória Terezinha da Silva

Resumo

Este trabalho apresenta o desenvolvimento de uma solução de *software* para otimizar o controle de vendas e gestão de estoque em uma loja de artigos agropecuários, visando modernizar processos manuais. A solução foi motivada pela necessidade de aumentar a eficiência operacional e competitividade da empresa. A metodologia incluiu o levantamento de requisitos com os usuários, análise dos fluxos de vendas e o desenvolvimento iterativo do software. O sistema foi desenvolvido utilizando *Java*, *Spring Framework*, *MySQL*, *HTML5*, *CSS3*, *JavaScript*, *TypeScript* e *React*, permitindo automação de vendas, atualização em tempo real do estoque e geração de relatórios detalhados. Além disso, o sistema contribui para a sustentabilidade ao reduzir o uso de papel. Os resultados indicam melhorias significativas na eficiência operacional e na experiência do cliente, com impacto positivo no crescimento do negócio e no desenvolvimento local.

Palavras-chave: Automação. Desenvolvimento de *software*. Gestão. Vendas.

Abstract

This paper presents the development of a software solution aimed at optimizing sales control and inventory management in an agricultural goods store, with the goal of modernizing manual processes. The solution was driven by the need to increase operational efficiency and the company's competitiveness. The methodology

¹ Graduando em Análise e Desenvolvimento de Sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: linikergabrieldasilva@gmail.com

² Graduando em Análise e Desenvolvimento de Sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: vitoria.c.silva24@gmail.com

included gathering user requirements, analyzing sales workflows, and iteratively developing the software. The system was developed using Java, Spring Framework, MySQL, HTML5, CSS3, JavaScript, TypeScript, and React, enabling sales automation, real-time inventory updates, and detailed report generation. Additionally, the system contributes to sustainability by reducing paper usage. The results show significant improvements in operational efficiency and customer experience, with a positive impact on business growth and local development.

Keywords: Automation. Software development. Management. Sales.

1 Introdução

A tecnologia da informação tem sido fundamental para a modernização e otimização de processos em setores como o agropecuário. Em lojas de artigos agropecuários, a gestão manual de vendas e estoque enfrenta desafios como erros, lentidão e ineficiência. Este projeto busca solucionar a questão: como desenvolver uma solução de software que automatize e otimize o controle de vendas e estoque, aumentando a eficiência operacional de uma loja de artigos agropecuários?

A hipótese levantada é que um sistema automatizado de controle de vendas e estoque pode aumentar a eficiência operacional, modernizar processos manuais e melhorar a competitividade da loja.

Os objetivos deste trabalho são:

Desenvolver uma solução de software que automatize o controle de vendas e a atualização do estoque em tempo real;

Proporcionar relatórios detalhados sobre o desempenho comercial da loja;

Contribuir para a sustentabilidade, reduzindo o uso de papel na gestão de informações;

Melhorar a experiência do cliente e impulsionar o crescimento da empresa.

A justificativa do projeto está na necessidade de modernizar os processos comerciais e administrativos, superando as limitações dos métodos convencionais, como planilhas e registros físicos. A adoção de tecnologia promove

competitividade e pode beneficiar pequenas empresas agropecuárias, contribuindo para o desenvolvimento local.

A relevância do trabalho reside em sua capacidade de transformar a gestão de operações em pequenas e médias empresas do setor, promovendo digitalização, crescimento econômico e sustentabilidade.

O projeto utilizou levantamento de requisitos com usuários, análise de fluxos de vendas e estoque e desenvolvimento iterativo do software. As ferramentas usadas incluíram Java, Spring Framework, MySQL, HTML5, CSS3, JavaScript, TypeScript e React, seguindo uma abordagem ágil, com ciclos contínuos de revisão e aprimoramento.

O trabalho está organizado em capítulos. O Capítulo 2 trata da viabilidade do projeto e ferramentas estratégicas como BMC (Business Model Canvas) e Matriz SWOT. O Capítulo 3 aborda o levantamento de requisitos e o DER (Diagrama Entidade-Relacionamento). O Capítulo 4 descreve as ferramentas e métodos utilizados, enquanto o Capítulo 5 detalha o desenvolvimento do software. Por fim, o Capítulo 6 discute os resultados obtidos, analisando os impactos observados e apresentando propostas para trabalhos futuros.

2 Viabilidade do Projeto

O objetivo de analisar a viabilidade deste projeto é determinar se é possível, de fato, desenvolvê-lo e garantir que realmente seja relevante e útil. Dessa maneira, são evitadas situações frustrantes nas quais os desenvolvedores descobrem, no fim do processo, que seu projeto não era necessário e nem trouxe valor para o negócio.

Tendo isso em vista, após analisar o contexto, entrevistar os *stakeholders* e fazer as validações necessárias, fizemos um BMC (*Business Model Canvas*) para ampliar a visão do negócio, garantindo ainda mais sua assertividade. E no fim, chegamos à conclusão de que o projeto realmente fazia sentido e deveria ser desenvolvido para solucionar as dores dos clientes: lojas do ramo de negócio de artigos agropecuários.

2.1 Canvas de Negócio (*Business Model Canvas* - BMC)

O SEBRAE (Serviço Brasileiro de Apoio às Micro e Pequenas Empresas) sintetizou o conceito de BMC (*Business Model Canvas*):

O Business Model Canvas, mais conhecido como Canvas, é uma ferramenta de planejamento estratégico, que permite desenvolver e esboçar modelos de negócio novos ou existentes.

É um mapa visual pré-formatado contendo nove blocos, são eles:

Quadro 1 – Todos os nove blocos do Business Model Canvas de acordo com o SEBRAE

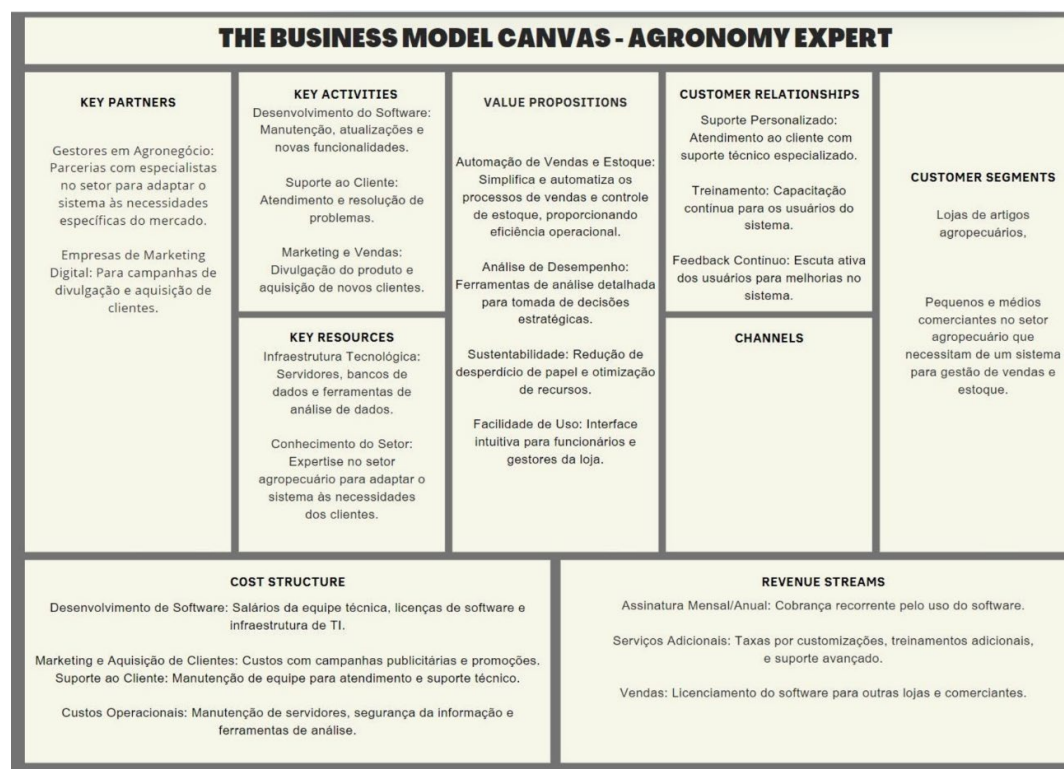
1	É um mapa visual pré-formatado contendo nove blocos, são eles: Proposta de valor: o que sua empresa vai oferecer para o mercado que realmente terá valor para os clientes.
2	Segmento de clientes: quais segmentos de clientes serão foco da sua empresa.
3	Os canais: como o cliente compra e recebe seu produto e serviço.
4	Relacionamento com clientes: como a sua empresa se relacionará com cada segmento de cliente.
5	Atividade-chave: quais são as atividades essenciais para que seja possível entregar a Proposta de Valor.
6	Recursos principais: são os recursos necessários para realizar as atividades-chave.
7	Parcerias principais: são as atividades-chave realizadas de maneira terceirizada e os recursos principais adquiridos fora da empresa;
8	Fontes de receita: são as formas de obter receita por meio de propostas de valor.
9	Estrutura de custos: São os custos relevantes necessários para que a estrutura proposta possa funcionar. As ideias representadas nos nove blocos formam a conceitualização do seu negócio, ou seja, a forma como você irá operar e gerar valor ao mercado, definindo seus principais fluxos e processos, permitindo uma análise e visualização do seu modelo de atuação no mercado. (Canvas: [...], 2019)

Fonte: (Canvas: [...], 2019)

É essa capacidade do *Canvas* de ampliar a visão do desenvolvimento como um todo que agrega tanto valor ao negócio e ao projeto. Ele permite que os objetivos sejam mais claramente definidos e alinhados às expectativas dos *stakeholders*, promovendo uma estratégia mais eficaz e integrada. O *Business Model*

Canvas (BMC) apresentado foi elaborado pelos autores para sintetizar os conceitos discutidos e está ilustrado na Figura 1.

Figura 1 – Business Model Canvas do projeto Agronomy Expert



Fonte: Autoria própria

2.2 Matriz SWOT

A matriz SWOT já foi definida nas palavras da conhecida empresa TOTVS:

A análise SWOT é uma ferramenta de gestão que se baseia no estudo das forças, fraquezas, oportunidades e ameaças a uma situação ou empresa, produto, indústria ou até uma pessoa.

Inclusive, o acrônimo SWOT é uma sigla para Forças (Strengths), Fraquezas (Weaknesses), Oportunidades (Opportunities) e Ameaças (Threats). (Equipe TOTVS, 2023)

Considerando isso, com a matriz SWOT é possível realizar um levantamento eficiente de todos os pontos de uma empresa, de algo ou alguém, isto é, tanto os pontos fortes quanto os fracos. Uma vez que se tem isso levantado, a tomada de decisões é facilitada e guiada por estratégias que considerem as fraquezas, mas que também levem em consideração os pontos fortes almejando a otimização do produto ou tema da análise.

Foi desenvolvida uma matriz SWOT pelos autores para realizar esse levantamento dos principais pontos do projeto, que pode ser visualizada na Figura 2.

Figura 2 – SWOT do projeto Agronomy Expert



Fonte:

Autoria

própria

2.3 Plano de Ação 5W2H do Projeto

Foi elaborado um 5W2H pelos autores, para responder a perguntas importantes sobre o projeto, que pode ser visto no Quadro 2. Quanto ao conceito de 5W2H, de acordo com o SEBRAE (Serviço Brasileiro de Apoio às Micro e Pequenas Empresas):

A sigla 5W2H vem do inglês e é formada pelas letras iniciais das sete perguntas que devem ser feitas ao traçar o plano:

5W

- What – o que será feito?
- Why – por que será feito?
- Where – onde será feito?
- When – quando será feito?
- Who – por quem será feito?

2H

- How – como será feito?
- How much – quanto vai custar?

Ou seja, a função dessa ferramenta é definir o que será feito, por que, onde, por quem, quando, como e quanto isso custará. Dessa forma, o método auxilia na organização e no planejamento de quaisquer processos internos da empresa.

(5w2h: [...], 2023)

Quadro 2 – 5W2H do projeto Agronomy Expert

	WHAT – O QUE SERÁ FEITO (ETAPAS)?	HY – POR QUE SERÁ FEITO (JUSTIFICATIVA)?	HERE – ONDE SERÁ FEITO (LOCAL)?	HEN – QUANDO SERÁ FEITO (TEMPO)?	HO – POR QUEM SERÁ FEITO (RESPONSABILIDADE)?	H HOW – COMO SERÁ FEITO (MÉTODO)?	OW MUCH – QUANTO CUSTARÁ FAZER (CUSTO)?
1	Implementação de um sistema de gestão de vendas através do controle de estoque, ou seja entrada e saída de produtos	Para corrigir a ausência de controle de vendas	Na empresa do setor agropecuário.	Levantamento de Requisitos : 01/04/2024 à 31/05/2024 Desenvolvimento do Sistema: 07/2024 à 10/2024 Testes: 10/2024 Implementação	Liniker Gabriel: Analista de Sistemas (responsável pelo desenvolvimento do sistema, etc) Vitória Silva: Gerente de Negócios (responsável pela comunicação com o	será feita a definição dos requisitos e funcionalidades do software em colaboração com a equipe de uma empresa do Setor Agropecuário. Posteriormente	6 meses



CODITEC

CONGRESSO DE DESENVOLVEDORES E TECNOLOGIA DAS FATECS

Criatividade e Inovação

Franca, SP, 26 de novembro de 2024

				ntação: 11/2024	cliente e pelo levantam ento de requisitos, etc)	será feito o desenvo lvimento interativ o do software , testes em todas as etapas do desenvo lvimento para garantir a qualidad e e funciona lidade do software , para isso serão utilizado s: Junit e Postman . Com o objetivo de gestão de tempo e demand as utilizare	
--	--	--	--	--------------------	---	--	--

						mos o Trello com a função de organiza ção de atividad es e seus respecti vos prazos	
2	desenvolvimento de uma tela de listagem de pedidos para acompanhamento de resultados por vendedor e por período.	para melhorar a gestão de resultados, permitindo que a empresa acompanhe de forma mais eficaz seu desempenho financeiro	na empresa do setor agropecuário	Levantamento de Requisitos : 01/04/2024 à 31/05/2024 Desenvolvimento do Sistema: 07/2024 à 10/2024 Testes: 10/2024	Liniker Gabriel: Analista de Sistemas (responsável pelo desenvolvimento do sistema, etc) ◦ Vitória Silva: Gerente de Negócios (responsável pela comunicação)	serão desenvolvidas rotas na API para exibição dessa listagem, trazendo todos os pedidos feitos na empresa quando o usuário logado tiver acesso	mês



CODITEC

CONGRESSO DE DESENVOLVEDORES E TECNOLOGIA DAS FATECS

Criatividade e Inovação

Franca, SP, 26 de novembro de 2024

		o e operaci onal.		Impleme ntação: 11/2024	ção com o cliente e pelo levantam ento de requisitos, etc)	administ rativo e trazendo somente os pedidos do usuário logado quando este possuir acesso básico.	
3	adronização de cadastros (clientes, cargos, funcionários e produtos)	ara melhora r a gestão das informa ções, permitin do que elas sejam concisa s e padroni zadas.	m empresa s do setor agropecu ário	Levanta mento de Requisit os : 01/04/20 24 à 31/05/20 24 Desenvo lvimento do Sistema: 07/2024 à 10/2024 Testes: 10/2024 Impleme ntação: 11/2024	Liniker Gabriel: Analista de Sistemas (responsá vel pelo desenvolv imento do sistema, etc) ◦ Vitória Silva: Gerente de Negócios (responsá vel pela comunica ção com o cliente e pelo	udo passará a ser feito sistemic amente, visto que se os cadastro s forem feitos sistemati camente , os process os devem ficar padroniz ados, evitando irregulari dades	meses

					levantam ento de requisitos, etc)	cadastra is. s usuários do sistema serão treinado s e instruído s para que utilizem o sistema corretam ente.	
--	--	--	--	--	--	--	--

Fonte: Autoria própria

3 Levantamento de Requisitos

O levantamento de requisitos é o processo de identificar o que o sistema deve fazer (requisitos funcionais) e como ele deve se comportar (requisitos não funcionais). Ele envolve coleta de informações junto aos usuários e *stakeholders* por meio de entrevistas, reuniões e análise de documentos. O objetivo é garantir que todas as necessidades sejam documentadas para que o *software* atenda aos objetivos do projeto com qualidade e eficiência.

3.1 Levantamento de Requisitos Não Funcionais

De acordo com a DevMedia: “um requisito não funcional de software é aquele que descreve não o que o sistema fará, mas como ele fará.” (Antonio, 2008)

Levando esse conceito em consideração, podemos entender que ao levantar os requisitos não funcionais de um sistema, documentaremos seu comportamento de forma que esteja alinhado com as expectativas e necessidades do cliente e dos usuários.

O quadro que lista os requisitos não funcionais do sistema foi elaborado pelos autores, e pode ser visto no Quadro 3.

Quadro 3 – Requisitos Não Funcionais do sistema Agronomy Expert

RNF 001 -Disponibilizar sistema <i>web</i>	O sistema deve atender a plataforma <i>web</i>	O culto	() Desejável (X) Obrigatório	(() Transitório
RNF 002 -Usar banco de dados	Us ar banco de dados relacional <i>MySQL</i>	O culto	() Desejável (X) Obrigatório	(() Transitório
RNF 003 -Garantir segurança na manipulação de senhas	Us ar padrões de segurança e criptografia para a manipulação de senhas, como <i>hash and salt</i> com <i>BCrypt</i>	O culto	() Desejável (X) Obrigatório	(() Transitório
RNF 004 -Utilizar a linguagem de programação <i>Java</i> com o <i>framework Spring</i> para desenvolver o sistema	A linguagem <i>Java</i> deverá ser utilizada para desenvolver o sistema, visto que é uma linguagem que trabalha de maneira bem precisa com sistemas financeiros. Utilizar também o <i>framework Spring</i> .	O culto	() Desejável (X) Obrigatório	(() Transitório
RNF 005 -Buscar aplicar conceitos de SOLID e <i>Clean Code</i> no desenvolvimento	Ap licar conceitos de SOLID e <i>Clean Code</i> no desenvolvimento do código	O culto	() Desejável (X) Obrigatório	(() Transitório
RNF 006 -Buscar aplicar	Ap licar conceitos de	O culto	() Desejável	((X) Permanente

 CODITEC		CONGRESSO DE DESENVOLVEDORES E TECNOLOGIA DAS FATECS <i>Criatividade e Inovação</i> Franca, SP, 26 de novembro de 2024			
os conceitos de	usabilidade no	usabilidade no			
usabilidade no	<i>frontend</i>	como:		X) Obrigatório	() Transitório
<i>frontend</i>	psicologia das	cores e			
	mensagens de	sucesso/erro			

Fonte: Autoria própria

3.2 Diagrama Entidade Relacionamento

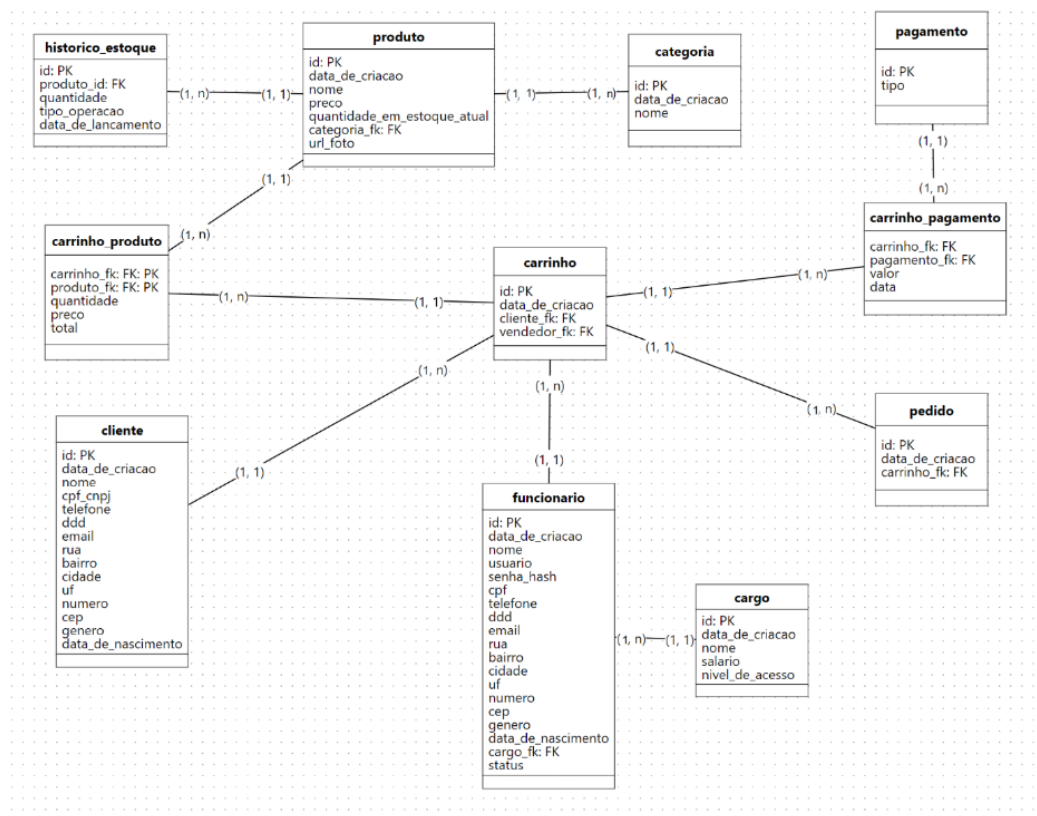
O DER (Diagrama Entidade-Relacionamento) é um diagrama essencial para consolidar o banco de dados usado no projeto. Ele garante uma documentação mapeando e relacionando as entidades, bem como seus atributos e cardinalidades. Serve para ter documentado o estado do banco de dados utilizado, garantindo melhor manutenção, e hoje em dia, algumas ferramentas de SGBD (Sistema de Gerenciamento de Banco de Dados) até mesmo geram automaticamente.

Uma definição pode ser encontrada pela famosa plataforma de design gráfico Canva:

Um diagrama de relacionamento de entidades ou ERD ajuda diferentes indústrias a visualizar e encontrar conexões entre objetos e conceitos para montar e estruturar seus bancos de dados. É vital em engenharia, sistemas de informação, educação e pesquisa, onde a modelagem do relacionamento entre entidades ajuda a identificar falhas lógicas ou de projeto que tornam a produção mais suave e sem problemas. Os projetos de diagramas ER também dependem do nível de detalhe e do escopo de seus dados. (Criador de [...], 2024)

Foi desenvolvido o DER do projeto pelos autores, conforme a Figura 3.

Figura 3 – DER do banco de dados relacional do projeto Agronomy Expert



Fonte: Autoria própria

4 Ferramentas e Métodos

A metodologia deste projeto foi elaborada para garantir que o sistema atendesse às necessidades do negócio. Foram realizadas entrevistas com os *stakeholders* para entender os processos da loja agropecuária e, a partir disso, iniciou-se o levantamento de requisitos funcionais, identificando funcionalidades essenciais como controle de vendas e gestão de estoque. Também foi feito o levantamento de requisitos não funcionais, como segurança, desempenho e usabilidade (ANTONIO, 2008), com destaque para a criptografia de dados e eficiência do sistema.

Além disso, o levantamento de regras de negócio mapeou as normas que regem o funcionamento da loja, enquanto os casos de uso modelaram as interações entre usuários e o sistema. Após essa fase, foram criados diagramas UML, como TAP, BPMN e o diagrama de classes, para estruturar e visualizar o sistema, além da matriz de rastreabilidade e SWOT (EQUIPE TOTVS, 2023), e o *Business Model Canvas* (SEBRAE, 2019).

No aspecto tecnológico, o *backend* está sendo desenvolvido em *Java*, utilizando o *Spring Framework*, enquanto o *MySQL* foi escolhido para gerenciar o banco de dados. O *frontend* será construído com *HTML5*, *CSS3*, *JavaScript*, *TypeScript* e *React*, garantindo uma interface responsiva e dinâmica.

Essas metodologias e ferramentas asseguraram o desenvolvimento de um sistema eficiente, seguro e moderno, contribuindo para a competitividade da loja no mercado agropecuário.

5 Desenvolvimento

O desenvolvimento deste trabalho iniciou com entrevistas com *stakeholders* e o levantamento de requisitos e regras de negócio, seguidos pela modelagem documentada em diagramas UML. O sistema foi construído utilizando *Java* e o *framework Spring*, com *MySQL* como banco de dados, conforme os requisitos não funcionais.

Atualmente, o código está em fase final de desenvolvimento, com o *backend* quase concluído e alguns *endpoints* pendentes. Após essa etapa, será iniciado o desenvolvimento do *frontend*, visando otimizar a experiência do usuário e integrá-lo ao *backend*.

Diversos *endpoints* já estão implementados, incluindo funcionalidades essenciais como a criação de carrinho, que pode ser vista na Figura 4. A criptografia de senha para funcionários, utilizando *BCrypt*, também já foi desenvolvida, conforme ilustrado nas Figuras 5, 6 e 7. A *Clean Architecture* foi adotada no *backend*, sendo visível na estrutura de pastas mostrada na Figura 8.

Figura 4 – Use Case de criação de carrinhos no código de *backend* do projeto

```

16 @Service
17 public class CreateCartUseCaseImpl implements CreateCartUseCase {
18
19     private final CartRepository cartRepository;
20     private final ClientRepository clientRepository;
21     private final EmployeeRepository employeeRepository;
22
23     public CreateCartUseCaseImpl(CartRepository cartRepository,
24                                 ClientRepository clientRepository,
25                                 EmployeeRepository employeeRepository) {
26         this.cartRepository = cartRepository;
27         this.clientRepository = clientRepository;
28         this.employeeRepository = employeeRepository;
29     }
30
31     @Override
32     public Cart createCart(String requesterUsername, CreateCartDto createCartDto) {
33         Employee requester = employeeRepository.findByUsername(requesterUsername);
34         .orElseThrow(() -> new EmployeeNotFoundException("Não foi possível identificar o responsável pela request"));
35
36         if (AccessLevelEnum.BASIC.equals(requester.getRole().getAccessLevel()) && createCartDto.sellerId() != null) {
37             throw new IllegalArgumentException("Você não tem permissão para criar carrinhos para outros funcionários");
38         }
39
40         Employee seller = requester;
41         if (createCartDto.sellerId() != null) {
42             seller = employeeRepository.findById(createCartDto.sellerId());
43             .orElseThrow(() -> new EmployeeNotFoundException("Não foi encontrado nenhum funcionário com o id informado"));
44         }
45
46         Client client = clientRepository.findById(createCartDto.clientId());
47         .orElseThrow(() -> new ClientNotFoundException("Não foi encontrado nenhum cliente com o id informado"));
48
49         Cart cart = Cart.create(client, seller);
50
51         return cartRepository.save(cart);
52     }
53 }
    
```

Fonte: Autoria própria

Figura 5 – Use Case de criação de funcionários fazendo encriptação de senhas

```

@Service
public class CreateEmployeeUseCaseImpl implements CreateEmployeeUseCase {

    private final EmployeeRepository employeeRepository;
    private final RoleRepository roleRepository;
    private final WebSecurityConfig securityConfiguration;

    public CreateEmployeeUseCaseImpl(EmployeeRepository employeeRepository,
                                     RoleRepository roleRepository,
                                     WebSecurityConfig securityConfiguration) {
        this.employeeRepository = employeeRepository;
        this.roleRepository = roleRepository;
        this.securityConfiguration = securityConfiguration;
    }

    @Override
    public Employee createEmployee(CreateEmployeeDto createEmployeeDto) {
        Role role = roleRepository.findById(createEmployeeDto.roleId());
        .orElseThrow(() -> new RoleNotFoundException("Não foi encontrado nenhum cargo com o id informado"));

        String hashedPassword = encryptEmployeePassword(createEmployeeDto.password());

        Employee employee = Employee.create(createEmployeeDto, role, hashedPassword);
        return employeeRepository.save(employee);
    }

    private String encryptEmployeePassword(String password) {
        return securityConfiguration.passwordEncoder().encode(password);
    }
}
    
```

Fonte:

Autoria

própria

Figura 6 – Classe *WebSecurityConfig* com a instância do *BCrypt*

```

@Configuration
@EnableWebSecurity
public class WebSecurityConfig {

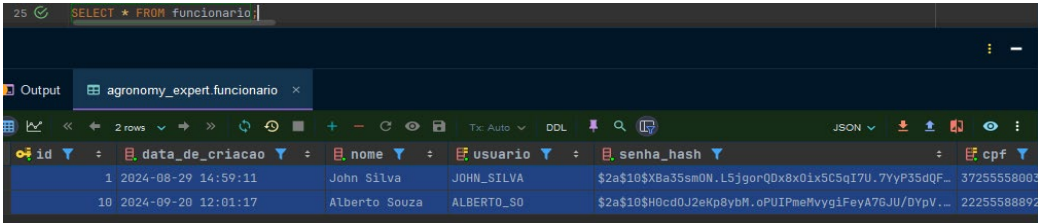
    private final EmployeeAuthenticationFilter employeeAuthenticationFilter;

    @Autowired
    public WebSecurityConfig(EmployeeAuthenticationFilter employeeAuthenticationFilter) {
        this.employeeAuthenticationFilter = employeeAuthenticationFilter;
    }

    @Bean
    public PasswordEncoder passwordEncoder() {
        return new BCryptPasswordEncoder();
    }
}
    
```

Fonte: Autoria própria

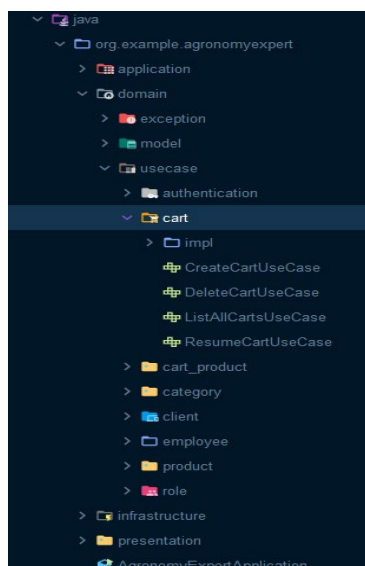
Figura 7 – Resultado da encriptação de senhas dos funcionários no banco de dados



id	data_de_criacao	nome	usuario	senha_hash	cpf
1	2024-08-29 14:59:11	John Silva	JOHN_SILVA	\$2a\$10\$X8a35sm0N.L5jgorQDx8x01x5C5qI7U.7YyP35dQF...	37255558888
10	2024-09-20 12:01:17	Alberto Souza	ALBERTO_S0	\$2a\$10\$H0cdQJ2eKp8ybM.oPUIPmeMvyg1FeyA76JU/DypV...	22255588889

Fonte: Autoria própria

Figura 8 – Estrutura de pastas usando *Clean Architecture*



Fonte: Autoria própria

6 Resultados e discussões

O projeto ainda está em fase de finalização e será implementado definitivamente numa loja de artigos agropecuários. Somente após essa implementação será possível realizar uma avaliação técnica completa deste estudo de caso. No entanto, os testes realizados até o momento no *backend* já demonstram melhorias significativas em relação ao desempenho. Os *endpoints* apresentaram tempos de resposta consideravelmente mais rápidos quando comparados aos processos manuais que atualmente são utilizados pela empresa, que são lentos e sujeitos a erros. A expectativa é que, com a integração do *backend* ao *frontend*, a solução se tornará ainda mais eficiente e eficaz, proporcionando uma automação completa dos processos e resolvendo as principais dificuldades identificadas pelos *stakeholders*.

Considerações finais

Este trabalho destacou a relevância da tecnologia da informação na modernização da gestão de lojas agropecuárias, um setor que exige agilidade e eficiência para manter a competitividade. A digitalização dos processos de vendas e controle de estoque otimizou operações, reduziu o uso de papel e aumentou a precisão das informações.

Os objetivos do projeto, como a automação da gestão de vendas e estoque, foram atingidos parcialmente, com a implementação de *endpoints*, criptografia de senhas e adoção da *Clean Architecture*, garantindo robustez e sustentabilidade. Embora ainda em desenvolvimento, os resultados já indicam avanços em eficiência operacional e segurança de dados.

O projeto permitiu aprendizado em tecnologias como *Java*, Spring Framework e *MySQL*, além de reforçar a importância do levantamento de requisitos, modelagem de sistemas e interação com *stakeholders* para alinhamento às necessidades reais da empresa.

Os principais desafios incluíram adaptação a novas tecnologias, implementação de arquitetura limpa, gestão do tempo e desenvolvimento de uma interface *frontend* simples e acessível.

O futuro do sistema prevê funcionalidades avançadas, como módulos de relatórios detalhados, personalização para outras lojas agropecuárias, integração com *e-commerce* e adoção de metodologias ágeis. Espera-se que o projeto evolua para um ERP completo e contribua para a modernização de práticas no setor agropecuário, incentivando pequenas e médias empresas a adotarem soluções digitais, promovendo eficiência, sustentabilidade e competitividade.

Referências bibliográficas

5W2H: o que é, para que serve e por que usar na sua empresa. Sebrae, 20.nov.2023. Disponível em: <https://www.sebrae-sc.com.br/blog/5w2h-o-que-e-para-que-serve-e-por-que-usar-na-sua-empresa>. Acesso em: 09.out.2024.

ANTONIO. Artigo Engenharia de Software 3 - Requisitos Não Funcionais. DevMedia, 2008. Disponível em: <https://www.devmedia.com.br/artigo-engenharia-de-software-3-requisitos-nao-funcionais/9525>. Acesso em: 14.ago.2024.

Canvas: como estruturar seu modelo de negócio. Sebrae, 30.out.2019. Disponível em: <https://sebrae.com.br/sites/PortalSebrae/ufs/pr/artigos/canvas-como-estruturar-seu-modelo-de-negocio,6dab288acc58d510VgnVCM1000004c00210aRCRD>. Acesso em: 07.ago.2024.

Criador de Diagrama Entidade Relacionamento. Canva, 2024. Disponível em: https://www.canva.com/pt_br/graficos/diagrama-entidade-relacionamento/. Acesso em: 21.ago.2024.

DANIELLA. Elicitação de Requisitos: Levantamento de requisitos e técnicas de Elicitação. DevMedia, 2014. Disponível em: <https://www.devmedia.com.br/elicitacao-de-requisitos-levantamento-de-requisitos-e-tecnicas-de-elicitacao/31872>. Acesso em: 09.ago.2024.

EQUIPE TOTVS. Análise SWOT: o que é, para que serve, benefícios e como fazer. Totvs, 08.nov.2023. Disponível em: <https://www.totvs.com/blog/negocios/analise-swot/>. Acesso em: 22.jul.2024.