

UNIVERSIDADE PAULISTA – UNIP EaD
PROJETO INTEGRADO MULTIDISCIPLINAR VI

Curso Superior de Tecnologia em
Análise e Desenvolvimento de sistemas

FELIPE ÍTALO DA PAIXÃO PEREIRA – 0621436

Sistema de venda de jogos eletrônicos, acessórios e produtos geek

FELIPE ÍTALO DA PAIXÃO PEREIRA – 0621436

Sistema de venda de jogos eletrônicos, acessórios e produtos geek

Projeto Integrado Multidisciplinar em

Análise e Desenvolvimento de Sistemas

Projeto Integrado Multidisciplinar para obtenção do título de tecnólogo em

Análise e Desenvolvimento de Sistemas, apresentado à Universidade Paulista –

UNIP EaD.

Orientador (a): Professor Robson Batista Alves.

RESUMO

Este trabalho teve como objetivo o desenvolvimento de um site de vendas de produtos geek, abrangendo jogos eletrônicos, acessórios e itens relacionados. Foram realizadas análises, modelagem e implementação para atender aos requisitos funcionais e não funcionais do sistema. Por meio de diagramas de casos de uso e de classes, as funcionalidades e as entidades do sistema foram representadas de forma clara. Com base nesses diagramas, foi criado um modelo entidade-relacionamento para o banco de dados, permitindo a persistência das informações relacionadas aos produtos, vendas e clientes. O resultado é um site intuitivo, seguro e eficiente, oferecendo aos usuários uma experiência de compra online completa no universo geek. Em resumo, este trabalho envolveu o desenvolvimento de um site de vendas de produtos geek, que foi estruturado por meio de análises, modelagem e implementação. Utilizando diagramas de casos de uso e de classes, foi possível representar as funcionalidades e entidades do sistema de forma clara. O modelo entidade-relacionamento foi criado para a persistência dos dados, possibilitando o armazenamento das informações dos produtos, vendas e clientes. O resultado é um site que proporciona uma experiência de compra online agradável, segura e completa para os amantes do universo geek.

Palavras-chave: Sistema; site; modelagem; diagrama de casos; banco de dados; vendas; geek; produtos.

ABSTRACT

This work aimed at developing a website for selling geek products, including video games, accessories, and related items. Analyses, modeling, and implementation were carried out to meet the functional and non-functional requirements of the system. Through use case and class diagrams, the system's functionalities and entities were represented clearly. Based on these diagrams, an entity-relationship model was created for the database, allowing the persistence of information related to products, sales, and customers. The result is an intuitive, secure, and efficient website that provides users with a complete online shopping experience in the geek universe. In summary, this work involved the development of a website for selling geek products, which was structured through analyses, modeling, and implementation. Using use case and class diagrams, the system's functionalities and entities were clearly represented. The entity-relationship model was created for data persistence, enabling the storage of information regarding products, sales, and customers. The result is a website that offers a pleasant, secure, and comprehensive online shopping experience for enthusiasts of the geek universe.

Keywords: System; site; modeling; case diagram; database; sales; geek; products.

SUMÁRIO

1 INTRODUÇÃO	6
2 ANÁLISE DE REQUISITOS	7
2.2 ATORES ENVOLVIDOS NO SISTEMA	8
2.3 OBJETIVOS DO SISTEMA	9
2.4 DEFININDO OS REQUISITOS DO SISTEMA	10
4 ESPECIFICAÇÕES DE CASOS DE USO	14
5 DESCRIÇÃO DOS CASOS DE USO.....	17
6 DIAGRAMA DE CLASSES	22
7 MER.....	24
7.1 RELAÇÕES NO MODELO	25
8 CONCLUSÃO	27
9 REFERÊNCIAS	28

1 INTRODUÇÃO

A evolução tecnológica tem impulsionado o desenvolvimento de soluções eficientes e inovadoras para atender às demandas de empresas dos mais diversos setores. Nesse contexto, a indústria de vendas de jogos eletrônicos, acessórios e produtos geek não fica alheia a essa realidade. Diante da crescente necessidade de otimizar a gestão de estoque e as operações de vendas, uma instituição desse ramo optou por contratar uma empresa especializada para desenvolver um sistema customizado e alinhado às suas necessidades específicas.

O objetivo principal desse sistema é estabelecer uma estrutura robusta e integrada que permita realizar, de forma ágil e eficiente, os processos de cadastro, alteração, consulta e exclusão relacionados aos produtos comercializados pela empresa, bem como os registros dos clientes. Além disso, é imprescindível a implementação de um controle de acesso com níveis de login, visando garantir a segurança das informações e restringir o acesso apenas aos colaboradores autorizados, como atendentes, estoquistas e o supervisor da loja.

Um dos aspectos centrais a serem considerados é a exigência de que todo o acesso ao sistema seja realizado exclusivamente nas dependências da loja, mediante a utilização de login e senha individuais. Essa medida assegura a confidencialidade dos dados e a proteção contra possíveis ameaças externas. Cada colaborador terá suas próprias credenciais de acesso, permitindo que desempenhem suas funções com eficácia e precisão.

O cadastro dos produtos que compõem o catálogo da loja é de responsabilidade dos estoquistas, profissionais incumbidos de organizar e atualizar o estoque. Para uma melhor categorização e localização dos itens, é necessário dividir os produtos em três principais categorias: jogos, acessórios e produtos geek. Essa estruturação facilita a gestão e a visualização dos produtos disponíveis, além de contribuir para a experiência de compra do cliente. No registro de cada produto, informações como código de barras, nome, categoria, fabricante, quantidade em estoque e valor devem ser devidamente especificadas. Adicionalmente, para jogos e acessórios, é importante informar a plataforma em que serão utilizados e o prazo de garantia estabelecido.

O cadastro dos clientes é um elemento essencial para a personalização do atendimento e a fidelização dos consumidores. Portanto, é imprescindível que o sistema colete e armazene informações relevantes, como código, RG, CPF, nome,

data do cadastro, endereço, telefone e e-mail do cliente. Esses dados permitem um relacionamento mais próximo e individualizado, contribuindo para um serviço de qualidade e a criação de estratégias de fidelização.

No processo de vendas, o sistema deve possibilitar o registro dos dados do cliente e dos produtos adquiridos. Cada transação de venda será identificada por um código único, contendo informações como data, valor, opções de pagamento (dinheiro ou cartão), status de pagamento e status da venda em si. O atendente terá a autorização para excluir produtos da venda caso o cliente decida não os adquirir, enquanto a exclusão da venda como um todo será restrita ao supervisor da loja, que deverá autenticar.

2 ANÁLISE DE REQUISITOS

A análise e o levantamento de requisitos desempenham um papel fundamental no processo de desenvolvimento de sistemas, especialmente quando se trata da criação de um sistema destinado à venda de jogos eletrônicos, acessórios e produtos geek. Essas etapas iniciais têm como objetivo primordial compreender a natureza e as necessidades do negócio, assim como as expectativas dos usuários, a fim de estabelecer uma base sólida para o desenvolvimento do sistema.

No contexto da análise de requisitos, busca-se identificar, compreender e documentar as funcionalidades e características que o sistema deve possuir. Isso implica em compreender o fluxo de trabalho da empresa, as interações entre os diversos atores envolvidos, como atendentes, estoquistas e o supervisor da loja, bem como as necessidades dos clientes em relação aos produtos e serviços oferecidos.

Para isso, é essencial realizar um levantamento detalhado das informações relevantes, como os cadastros de produtos e clientes. O estoquista desempenha um papel central nesse processo, sendo responsável por cadastrar os produtos a serem vendidos na loja, os quais devem ser categorizados de acordo com sua natureza, como jogos, acessórios e produtos geek. Os cadastros de clientes devem conter informações essenciais, tais como código, RG, CPF, nome, data do cadastro, endereço, telefone e e-mail.

Além disso, é fundamental considerar os requisitos específicos dos produtos. Cada item deve ser identificado por um código de barras e possuir informações como nome, categoria, fabricante, quantidade e valor. No caso de jogos e acessórios, é

importante registrar a plataforma na qual serão utilizados, assim como o prazo de garantia oferecido. Esses detalhes permitem um controle eficiente do estoque e uma gestão adequada dos produtos disponíveis para venda.

No que diz respeito às vendas, o sistema deve ser capaz de registrar os dados do cliente e dos produtos adquiridos. Um código único deve ser gerado para cada venda, contendo informações como a data da transação, o valor, as opções de pagamento (dinheiro ou cartão), o status de pagamento e o status da venda em si. É importante destacar que, embora o atendente possua a permissão para excluir produtos da venda, essa ação requer a autenticação do supervisor da loja, que deve fornecer um usuário e senha válidos.

Adicionalmente, o sistema deve oferecer a funcionalidade de consulta de preços, permitindo que o atendente atenda prontamente às solicitações dos clientes. Por fim, o cancelamento de uma venda é uma ação exclusiva do supervisor da loja, exigindo autenticação por meio de usuário e senha válidos. Nesse momento, o código da venda deve ser enviado para o sistema financeiro, a fim de que sejam realizados os devidos registros e ajustes.

Em resumo, o processo de análise e levantamento de requisitos para o sistema de vendas de jogos eletrônicos, acessórios e produtos geek envolve uma compreensão profunda das necessidades e expectativas dos usuários, bem como dos aspectos específicos relacionados aos produtos, clientes e transações. Essa etapa é essencial para a construção de um sistema eficiente, que atenda às demandas do negócio e proporcione uma experiência satisfatória tanto para os atendentes quanto para os clientes.

2.2 ATORES ENVOLVIDOS NO SISTEMA

No contexto do sistema proposto para a empresa de vendas de jogos eletrônicos, acessórios e produtos geek, é possível identificar os principais atores que desempenham papéis fundamentais no funcionamento do sistema.

Primeiramente, temos os atendentes, que são os funcionários responsáveis por atender os clientes e realizar as transações comerciais. Eles interagem diretamente com os clientes, registrando as vendas, fornecendo informações sobre os produtos disponíveis, auxiliando na escolha dos itens e garantindo uma experiência satisfatória

de compra. Os atendentes são os principais usuários do sistema, utilizando-o para realizar os cadastros, consultas e alterações relacionados aos produtos e clientes.

Em seguida, temos os estoquistas, que desempenham um papel crucial na gestão do estoque da loja. Eles são responsáveis por cadastrar os produtos que serão vendidos, dividindo-os por categorias, como jogos, acessórios e produtos geek. Os estoquistas também atualizam as quantidades disponíveis de cada item, registram as entradas e saídas de mercadorias, garantindo que o estoque esteja sempre atualizado e que os produtos estejam prontos para serem vendidos.

Além disso, o supervisor da loja desempenha um papel de supervisão e controle no sistema. Ele tem acesso privilegiado ao sistema e é responsável por garantir o bom funcionamento das operações da loja. O supervisor pode realizar ações como exclusão de produtos da venda, desde que forneça um usuário e senha válidos, além de ter a capacidade de cancelar uma venda e enviar o código correspondente para o sistema financeiro. Sua função é fundamental para assegurar que todas as atividades estejam de acordo com as políticas e procedimentos estabelecidos.

Por fim, temos os clientes, que são os usuários finais do sistema. Eles interagem com os atendentes, buscando informações sobre os produtos, solicitando preços, efetuando compras e realizando cadastros para facilitar futuras transações. Os clientes são atores essenciais no sistema, pois suas necessidades e expectativas devem ser atendidas de forma eficiente, proporcionando uma experiência satisfatória de compra.

A identificação e compreensão desses atores envolvidos no sistema são de extrema importância para o desenvolvimento de um sistema adequado, que atenda às necessidades de cada um deles. O envolvimento de todos os atores desde as fases iniciais de análise e levantamento de requisitos é fundamental para garantir a eficácia e eficiência do sistema, bem como a satisfação dos usuários finais.

2.3 OBJETIVOS DO SISTEMA

O sistema proposto para a empresa de vendas de jogos eletrônicos, acessórios e produtos geek possui uma série de objetivos principais que visam aprimorar o controle e a eficiência das operações da loja.

Em primeiro lugar, o sistema tem como objetivo centralizar e otimizar o controle do estoque dos produtos. Isso inclui o cadastro detalhado dos produtos, a divisão por categorias (jogos, acessórios e produtos geek), a atualização das quantidades

disponíveis, o registro de entradas e saídas de mercadorias, bem como a definição de informações relevantes, como o estoque atual, estoque mínimo e histórico de movimentação dos produtos. Dessa forma, o sistema permite um melhor gerenciamento do estoque, evitando a falta de produtos e proporcionando uma visão ampla e atualizada do status dos itens disponíveis para venda.

Outro objetivo importante do sistema é facilitar o processo de vendas e atendimento aos clientes. Os atendentes utilizam o sistema para registrar as vendas, incluindo os dados dos clientes e os produtos adquiridos. Além disso, o sistema gera um código único para cada venda, com informações como a data da venda, o valor total, as opções de pagamento (dinheiro/cartão) e o status de pagamento e venda. Isso agiliza o processo de registro, facilita o acompanhamento das transações realizadas e fornece informações importantes para análises futuras.

Além disso, o sistema também visa garantir a segurança e a integridade das transações. A exclusão de produtos da venda é permitida apenas pelo supervisor da loja, mediante a autenticação com usuário e senha válidos. Isso evita erros ou manipulações indevidas nas vendas. Da mesma forma, o cancelamento de uma venda também é restrito ao supervisor, que deve fornecer as credenciais corretas. Nesse momento, o sistema envia o código da venda para o sistema financeiro, garantindo uma sincronização adequada entre as operações.

Em resumo, os principais objetivos do sistema são: gerenciar o estoque de forma centralizada e eficiente, agilizar o processo de vendas e atendimento aos clientes, e garantir a segurança e integridade das transações realizadas. Esses objetivos contribuem para o bom funcionamento da loja, a satisfação dos clientes e a maximização dos resultados comerciais.

2.4 DEFININDO OS REQUISITOS DO SISTEMA

A tabela abaixo mostra a relação entre os atores e os casos de uso no sistema proposto para a empresa de vendas de jogos eletrônicos, acessórios e produtos geek:

Tabela 01: Relação Atores e Casos de Uso

Ator	Caso de Uso
Sistema (pagamento)	Realizar venda

	Consultar preços dos Produtos
	Excluir Produtos da Venda
Estoquista	Cadastrar Produtos
	Atualizar Estoque
Admin do Site	Excluir Produto da Venda
	Cancelar Venda
	Gerar Relatório de Vendas
Cliente (usuário)	Realizar Compra
	Consultar Produtos Disponíveis

Fonte: Autor (2023)

Essa tabela apresenta os principais atores envolvidos no sistema, como o Atendente, Estoquista, Supervisor da Loja e Cliente. Em seguida, são listados os casos de uso relacionados a cada ator. O sistema (pagamento), por exemplo, está relacionado aos casos de uso "Realizar Venda", "Consultar Preços dos Produtos" e "Excluir Produtos da Venda". O Estoquista tem como principais casos de uso o "Cadastrar Produtos" e "Atualizar Estoque". Já o Admin do site está envolvido nos casos de uso "Excluir Produto da Venda", "Cancelar Venda" e "Gerar Relatório de Vendas". Por fim, o Cliente (usuário) pode realizar ações como "Realizar Compra" e "Consultar Produtos Disponíveis". Essa tabela ajuda a visualizar de forma clara a relação entre os atores do sistema e os casos de uso que cada um deles está envolvido.

3 REQUISITOS FUNCIONAIS E NÃO FUNCIONAIS

Os requisitos funcionais são aqueles que descrevem as funcionalidades e comportamentos específicos que o sistema deve apresentar. No contexto do sistema de vendas de produtos geek, alguns requisitos funcionais são identificados. Em primeiro lugar, é necessário ter um cadastro de produtos, no qual seja possível registrar informações como código de barras, nome, categoria, fabricante, quantidade em estoque, valor e informações específicas de cada categoria, como plataforma de jogos e prazo de garantia para jogos e acessórios. Além disso, é fundamental ter um cadastro de clientes, com campos para armazenar dados como código, RG, CPF, nome, data de cadastro, endereço, telefone e e-mail. Também é necessário implementar um

controle de estoque, permitindo a atualização dos níveis de estoque com base nas vendas realizadas.

Os requisitos não funcionais definem as características e restrições do sistema, que não estão diretamente relacionados às funcionalidades específicas, mas afetam a sua qualidade e desempenho. Para o sistema de vendas de produtos geek, alguns requisitos não funcionais são relevantes. Em primeiro lugar, é importante considerar a segurança do sistema, garantindo que o acesso seja restrito a usuários autorizados por meio de login e senha. Além disso, é necessário garantir a eficiência e o desempenho do sistema, assegurando que as operações de cadastro, consulta e exclusão sejam realizadas de maneira rápida e responsiva. A interface do sistema também deve ser intuitiva e amigável, facilitando o uso pelos atendentes, estoquistas e supervisor da loja. Por fim, é importante considerar a escalabilidade do sistema, de modo que ele possa lidar com um aumento no número de produtos, clientes e transações sem comprometer a sua performance.

Esses requisitos funcionais e não funcionais são essenciais para o desenvolvimento de um sistema eficiente e eficaz para a empresa de vendas de jogos eletrônicos e produtos geek. A análise e o atendimento desses requisitos garantirão que o sistema atenda às necessidades dos usuários, oferecendo um controle adequado do estoque, gerenciamento das vendas e proporcionando uma experiência satisfatória tanto para os funcionários quanto para os clientes da loja. Segue abaixo tabela especificando os requisitos não funcionais.

Tabela 02: Requisitos não funcionais

Identificador	Categoria	Nome	Descrição
RNF1	Desempenho	Tempo de Resposta	O sistema deve apresentar um tempo de resposta rápido, permitindo aos usuários navegar e interagir de forma ágil, sem atrasos significativos.
RNF2	Usabilidade	Interface Intuitiva	A interface do sistema deve ser intuitiva e de fácil utilização, permitindo que os usuários encontrem facilmente os produtos desejados, adicionem ao carrinho e realizem o pagamento de forma rápida e sem complicações.

RNF3	Segurança	Proteção de Dados	Todos os dados dos usuários, incluindo informações pessoais e dados de pagamento, devem ser protegidos por medidas de segurança adequadas, como criptografia, para garantir a confidencialidade e integridade das informações.
RNF4	Disponibilidade	Alta Disponibilidade	O sistema deve estar disponível e acessível aos usuários de forma contínua, com tempo de inatividade mínimo e uma infraestrutura robusta capaz de lidar com picos de tráfego e garantir a continuidade das operações.
RNF5	Escalabilidade	Escalabilidade Vertical	O sistema deve ser capaz de lidar com o aumento no número de usuários, transações e produtos sem comprometer o desempenho e a qualidade do serviço, permitindo que o sistema seja escalável verticalmente conforme necessário.
RNF6	Confiabilidade	Backup e Recuperação	Deve ser implementado um sistema de backup regular para garantir a recuperação de dados em caso de falhas ou perdas, minimizando a possibilidade de perda de informações críticas dos usuários e transações realizadas.

Fonte: Autor (2023)

Esses requisitos não funcionais abordam aspectos importantes do sistema que vão além das funcionalidades principais. Eles abrangem a performance, segurança, usabilidade, confiabilidade, escalabilidade e portabilidade, garantindo uma experiência satisfatória para os usuários e um ambiente robusto e confiável para o funcionamento do sistema.

Durante a análise dos casos de uso do sistema de vendas de produtos geek, é relevante verificar a presença de cenários do tipo <<extends>> e <<include>>, que trazem maior flexibilidade e reutilização de funcionalidades. Os cenários do tipo <<extends>> são identificados quando um caso de uso pode ser estendido com

funcionalidades adicionais em situações específicas, sem alterar seu fluxo principal. Um exemplo de cenário de <<extends>> no sistema de vendas de produtos geek pode ser a opção de aplicar um desconto promocional no valor total da compra, caso o cliente atinja um determinado valor mínimo de compra. Essa extensão ocorre apenas quando a condição é satisfeita, proporcionando uma personalização no cálculo do preço final.

Por outro lado, os cenários do tipo <<include>> são observados quando há a inclusão de funcionalidades obrigatórias em um ou mais casos de uso. Essa inclusão evita a repetição de código e promove a reutilização de funcionalidades comuns. No contexto do sistema de vendas de produtos geek, um exemplo de cenário de <<include>> pode ser a inclusão do registro de pagamento em todos os casos de uso relacionados à finalização da venda. Independentemente do fluxo principal do caso de uso, o registro do pagamento é uma etapa essencial que deve ser executada em todas as transações, garantindo a integridade das informações financeiras.

Ao identificar e analisar esses cenários de extensão e inclusão, é possível modularizar o sistema, tornando-o mais flexível e adaptável a diferentes necessidades. Além disso, a utilização de cenários do tipo <<extends>> e <<include>> promove a reutilização de funcionalidades, reduzindo a redundância de código e melhorando a coesão do sistema. Essa abordagem também facilita a manutenção e evolução do software, pois alterações ou adições em funcionalidades específicas podem ser realizadas nos cenários correspondentes, sem impactar todo o sistema.

Portanto, a verificação de cenários do tipo <<extends>> e <<include>> no sistema de vendas de produtos geek é essencial para identificar oportunidades de personalização e reutilização de funcionalidades. Por meio da extensão, é possível adicionar recursos extras quando necessários, sem comprometer o fluxo principal do caso de uso. Já a inclusão garante a execução de funcionalidades obrigatórias de forma consistente e padronizada em diversos casos de uso. Esses cenários contribuem para a eficiência, flexibilidade e manutenibilidade do sistema, atendendo às demandas do negócio de forma adequada.

4 ESPECIFICAÇÕES DE CASOS DE USO

A modelagem de casos de uso é uma prática essencial na engenharia de software para capturar os requisitos funcionais do sistema e entender como os

usuários interagem com ele. Para o sistema de jogos eletrônicos, produtos e acessórios geek, podemos identificar diversos casos de uso relevantes que descrevem as funcionalidades principais do sistema e as interações dos usuários.

Um dos casos de uso importantes é o "Pesquisar Produtos", no qual os usuários podem realizar buscas por jogos eletrônicos, produtos e acessórios geek específicos. Esse caso de uso permite que os usuários encontrem os produtos desejados com base em critérios como nome, categoria, plataforma de jogo ou preço.

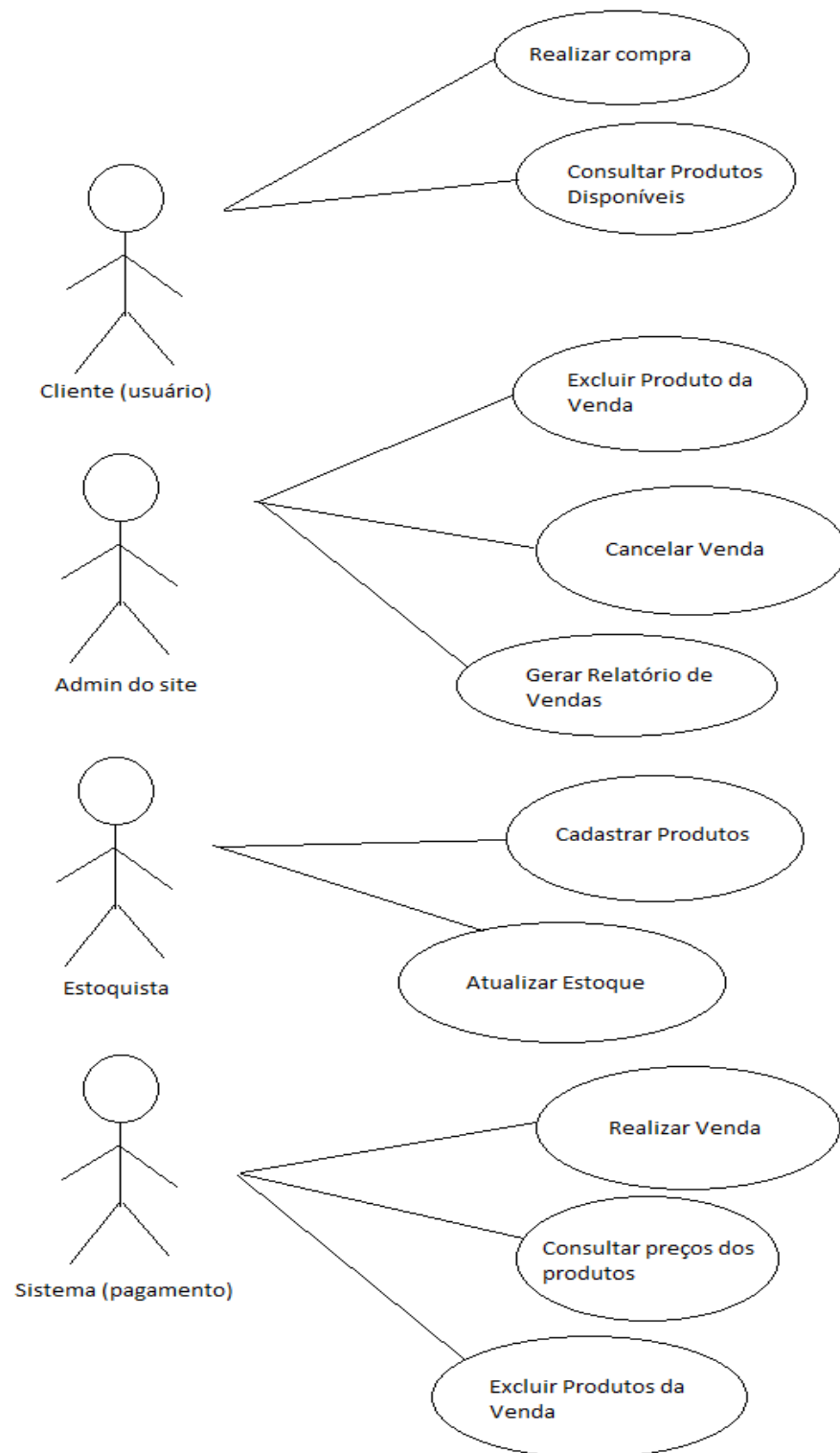
Outro caso de uso relevante é o "Adicionar Produto ao Carrinho". Nesse caso de uso, os usuários podem selecionar os produtos de seu interesse e adicioná-los ao carrinho de compras virtual. Esse processo inclui a validação da disponibilidade do produto, a atualização do valor total da compra e a possibilidade de ajustar a quantidade de itens no carrinho.

Um caso de uso fundamental é o "Realizar Compra". Esse caso de uso permite que os usuários concluam suas compras, fornecendo as informações necessárias, como endereço de entrega e opções de pagamento. Durante esse processo, o sistema deve garantir a segurança das transações e fornecer um feedback adequado ao usuário sobre o status da compra.

Além disso, é importante considerar o caso de uso "Gerenciar Perfil do Usuário". Esse caso de uso permite que os usuários atualizem suas informações pessoais, como nome, endereço de entrega e preferências de comunicação. Também pode incluir a possibilidade de acompanhar o histórico de compras, avaliar produtos e visualizar recomendações personalizadas.

O diagrama de casos de uso abaixo, ilustra as principais interações entre os usuários e o sistema, representando os casos de uso mencionados anteriormente. Cada caso de uso pode ser detalhado com fluxos de eventos, pré-condições, pós-condições e exceções, a fim de fornecer uma compreensão mais completa das funcionalidades do sistema e das interações dos usuários.

Figura 01 – Diagrama de casos de uso



Fonte: Autor (2023)

5 DESCRIÇÃO DOS CASOS DE USO

Tabela 03: Caso de Uso: Realizar Venda

Identificação	Realizar Venda
Escopo	Vendas
Descrição	Registrar a venda de produtos
Ator Primário	Atendente
Interessados	Atendente, Sistema de Vendas
Pré-condições	O atendente deve estar autenticado no sistema
Pós-condições	A venda é registrada no sistema
Fluxo Normal	

Fonte: Autor (2023)

O atendente seleciona a opção de realizar venda.

O sistema exibe a tela de registro de venda.

O atendente seleciona os produtos desejados e suas quantidades.

O sistema calcula o valor total da venda.

O atendente registra os dados do cliente, se necessário.

O sistema gera o comprovante de venda.

Fluxo Alternativo: 1a. No passo 3, se o produto não estiver disponível em estoque, o sistema exibe uma mensagem de aviso ao atendente. O atendente pode substituir o produto ou cancelar a venda. 3a. No passo 5, se o cliente não desejar se identificar, o atendente pode registrar a venda como venda anônima.

Requisitos Relacionados: RNF01, RNF02, RNF03

Tabela 04: Caso de Uso: Consultar Preços dos Produtos

Identificação	Consultar Preços dos Produtos
Escopo	Vendas
Descrição	Consultar os preços dos produtos disponíveis
Ator Primário	Atendente
Interessados	Atendente, Sistema de Vendas
Pré-condições	O atendente deve estar autenticado no sistema
Pós-condições	-
Fluxo Normal	

Fonte: Autor (2023)

O atendente seleciona a opção de consultar preços dos produtos.

O sistema exibe a lista de produtos disponíveis com seus respectivos preços.

Fluxo Alternativo -

Requisitos Relacionados: RNF01

Tabela 05: Caso de Uso: Excluir Produtos da Venda

Identificação	Excluir Produtos da Venda
Escopo	Vendas
Descrição	Remover produtos da venda em andamento
Ator Primário	Atendente
Interessados	Atendente, Sistema de Vendas
Pré-condições	O atendente deve estar autenticado no sistema e ter uma venda em andamento
Pós-condições	-
Fluxo Normal	

Fonte: Autor (2023)

O atendente seleciona a opção de excluir produtos da venda.

O sistema exibe a lista de produtos registrados na venda.

O atendente seleciona o produto a ser excluído.

O sistema remove o produto da venda.

Fluxo Alternativo -

Requisitos Relacionados: RNF01

Tabela 06: Caso de Uso: Cadastrar Produtos

Identificação	Cadastrar Produtos
Escopo	Estoquista
Descrição	Cadastrar novos produtos no sistema
Ator Primário	Estoquista
Interessados	Estoquista, Sistema de Vendas
Pré-condições	O estoquista deve estar autenticado no sistema
Pós-condições	O produto é cadastrado no sistema
Fluxo Normal	

Fonte: Autor (2023)

O estoquista seleciona a opção de cadastrar produtos.

O sistema exibe o formulário de cadastro de produtos.

O estoquista preenche as informações do produto (nome, preço, quantidade em estoque, etc.).

O sistema valida as informações e cadastra o produto.

Fluxo Alternativo -

Requisitos Relacionados: RNF01

Tabela 07: Caso de Uso: Atualizar Estoque

Identificação	Atualizar Estoque
Escopo	Estoquista
Descrição	Atualizar a quantidade em estoque de um produto
Ator Primário	Estoquista
Interessados	Estoquista, Sistema de Vendas
Pré-condições	O estoquista deve estar autenticado no sistema
Pós-condições	A quantidade em estoque do produto é atualizada
Fluxo Normal	

Fonte: Autor (2023)

O estoquista seleciona a opção de atualizar estoque.

O sistema exibe a lista de produtos cadastrados.

O estoquista seleciona o produto desejado.

O sistema exibe as informações atuais do produto (nome, preço, quantidade em estoque).

O estoquista atualiza a quantidade em estoque do produto.

O sistema registra a atualização do estoque.

Fluxo Alternativo –

Requisitos Relacionados: RNF01

Tabela 08: Caso de Uso: Excluir Produto da Venda

Identificação	Excluir Produto da Venda
Escopo	Supervisor da Loja
Descrição	Remover produtos da venda em andamento
Ator Primário	Supervisor da Loja
Interessados	Supervisor da Loja, Sistema de Vendas
Pré-condições	O supervisor da loja deve estar autenticado no sistema e ter uma venda em andamento
Pós-condições	-
Fluxo Normal	

Fonte: Autor (2023)

O supervisor da loja seleciona a opção de excluir produtos da venda.

O sistema exibe a lista de produtos registrados na venda.

O supervisor da loja seleciona o produto a ser excluído.

O sistema remove o produto da venda.

Fluxo Alternativo –

Requisitos Relacionados: RNF01

Tabela 09: Caso de Uso: Cancelar Venda

Identificação	Cancelar Venda
Escopo	Supervisor da Loja
Descrição	Cancelar uma venda em andamento
Ator Primário	Supervisor da Loja
Interessados	Supervisor da Loja, Sistema de Vendas
Pré-condições	O supervisor da loja deve estar autenticado no sistema e ter uma venda em andamento
Pós-condições	A venda é cancelada
Fluxo Normal	

Fonte: Autor (2023)

O supervisor da loja seleciona a opção de cancelar venda.

O sistema confirma a ação e exibe uma mensagem de aviso.

O supervisor da loja confirma o cancelamento.

O sistema cancela a venda em andamento.

Fluxo Alternativo –

Requisitos Relacionados: RNF01

Tabela 10: Caso de Uso: Gerar Relatório de Vendas

Identificação	Gerar Relatório de Vendas
Escopo	Supervisor da Loja
Descrição	Gerar um relatório com as vendas realizadas
Ator Primário	Supervisor da Loja
Interessados	Supervisor da Loja, Sistema de Vendas
Pré-condições	O supervisor da loja deve estar autenticado no sistema
Pós-condições	O relatório de vendas é gerado
Fluxo Normal	

Fonte: Autor (2023)

O supervisor da loja seleciona a opção de gerar relatório de vendas.

O sistema exibe as opções de filtro para o relatório (por período, por categoria, etc.).

O supervisor da loja seleciona os filtros desejados.

O sistema processa as informações e gera o relatório de vendas.

Fluxo Alternativo –

Requisitos Relacionados: RNF01

Tabela 11: Caso de Uso: Realizar Compra

Identificação	Realizar Compra
Escopo	Cliente
Descrição	Realizar a compra de produtos
Ator Primário	Cliente
Interessados	Cliente, Sistema de Vendas
Pré-condições	-
Pós-condições	A compra é registrada no sistema
Fluxo Normal	

Fonte: Autor (2023)

O cliente acessa o sistema e seleciona os produtos desejados.

O sistema exibe os detalhes dos produtos selecionados (nome, preço, etc.).

O cliente confirma a compra.

O sistema registra a compra e gera o comprovante de pagamento.

Fluxo Alternativo –

Requisitos Relacionados: RNF01, RNF02, RNF03

Tabela 12: Caso de Uso: Consultar Produtos Disponíveis

Identificação	Consultar Produtos Disponíveis
Escopo	Cliente
Descrição	Consultar os produtos disponíveis para compra
Ator Primário	Cliente
Interessados	Cliente, Sistema de Vendas
Pré-condições	-
Pós-condições	-
Fluxo Normal	

Fonte: Autor (2023)

O cliente acessa o sistema e seleciona a opção de consultar produtos disponíveis.

O sistema exibe a lista de produtos disponíveis com seus respectivos detalhes (nome, preço, etc.).

Fluxo Alternativo –

Requisitos Relacionados: RNF01

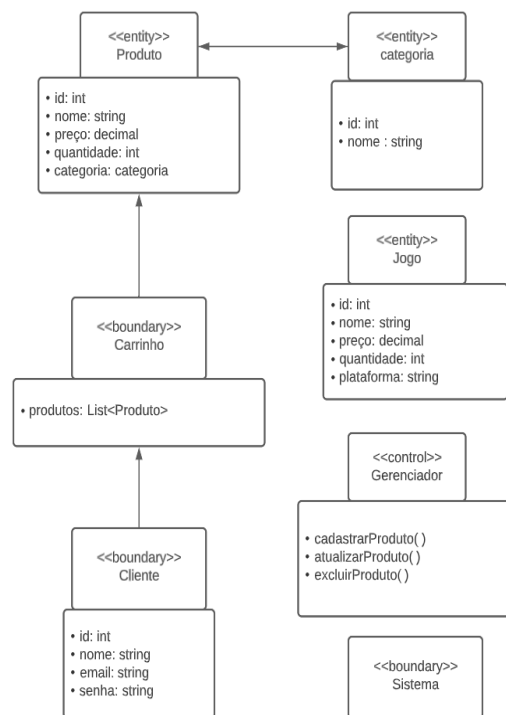
6 DIAGRAMA DE CLASSES

O diagrama de classe é uma representação visual da estrutura estática de um sistema, enfatizando as classes e os relacionamentos entre elas. Ele é amplamente utilizado na modelagem de software orientado a objetos para descrever as entidades do sistema, suas propriedades (atributos) e comportamentos (métodos), bem como as associações, heranças e dependências entre as classes.

O principal objetivo do diagrama de classe é capturar a estrutura estática do sistema, fornecendo uma visão clara e organizada das entidades e suas interações. Ele é uma ferramenta poderosa para comunicar e documentar o design do sistema, facilitando a compreensão entre os membros da equipe de desenvolvimento e servindo como base para a implementação do software. (Martin Fowler, "UML Distilled: A Brief Guide to the Standard Object Modeling Language")

No contexto do sistema de vedas de jogos eletrônicos, acessórios e produtos geek, podemos criar um diagrama de classes que representa as principais entidades envolvidas no sistema. Aqui está o diagrama de classes:

Figura 02: Diagrama de Classes



Fonte: Autor (2023)

O diagrama de classes apresentado representa a estrutura do sistema de vendas de jogos eletrônicos, acessórios e produtos geek. Ele é uma representação visual das classes e suas relações, permitindo uma compreensão mais clara e organizada da arquitetura do sistema.

O diagrama é composto por diversas classes, identificadas por retângulos, que representam entidades do sistema. As classes relevantes incluem "Produto", "Categoria", "Jogo", "Carrinho", "Cliente" e "Gerenciador". Cada classe possui seus atributos e métodos específicos, que desempenham papéis importantes no funcionamento do sistema.

A classe "Produto" representa um item disponível para venda e contém atributos como "id", "nome", "preço" e "quantidade". Além disso, possui uma relação de associação com a classe "Categoria", indicando que um produto pertence a uma determinada categoria.

A classe "Categoria" representa as categorias às quais os produtos podem ser atribuídos. Ela possui o atributo "id" e "nome" para identificar e descrever cada

categoria. A classe "Jogo" é uma especialização da classe "Produto" e representa jogos eletrônicos disponíveis para venda. Além dos atributos herdados da classe "Produto", possui o atributo adicional "plataforma" para indicar a plataforma na qual o jogo é executado.

A classe "Carrinho" é responsável por armazenar os produtos selecionados pelo cliente para compra. Ela possui uma associação de composição com a classe "Produto", permitindo que um carrinho contenha vários produtos. Seus atributos incluem "produtos", representado como uma lista de produtos no carrinho.

A classe "Cliente" representa os usuários do sistema que realizam compras. Ela possui atributos como "id", "nome", "email" e "senha" para identificar e autenticar cada cliente. Por fim, a classe "Gerenciador" atua como um controlador do sistema, contendo métodos como "cadastrarProduto()", "atualizarProduto()" e "excluirProduto()", que são responsáveis por gerenciar as operações relacionadas aos produtos.

No diagrama, foram identificadas as associações entre as classes, indicando a natureza dos relacionamentos. Além disso, as multiplicidades são especificadas para indicar quantos objetos de uma classe estão relacionados a objetos de outra classe. Essas informações ajudam a compreender como as diferentes classes interagem entre si no sistema.

Em resumo, o diagrama de classes fornece uma visão abrangente da estrutura do sistema de vendas de jogos eletrônicos, acessórios e produtos geek. Ele descreve as classes, seus atributos e métodos, além das associações entre elas. Essa representação visual é útil para entender a arquitetura do sistema, facilitando o desenvolvimento, manutenção e compreensão.

7 MER

O Modelo Entidade-Relacionamento (MER) é uma técnica de modelagem de dados amplamente utilizada na área de banco de dados. Ele fornece uma representação visual das entidades relevantes de um sistema, seus atributos e os relacionamentos entre elas. Essa abordagem permite descrever a estrutura e o comportamento dos dados de forma clara e concisa.

No MER, as entidades são representadas por retângulos, os atributos são representados por elipses e os relacionamentos entre as entidades são representados

por linhas. Essa notação gráfica facilita a compreensão e comunicação das informações relacionadas aos dados de um sistema.

O objetivo principal do MER é identificar e definir as entidades que fazem parte do sistema, seus atributos e os relacionamentos entre elas. Ele serve como uma ferramenta fundamental na modelagem conceitual de um banco de dados, ajudando a capturar a estrutura lógica dos dados e as restrições do sistema.

De acordo com Elmasri e Navathe (2010), o MER é uma técnica de modelagem de dados baseada no conceito de entidade, que é uma representação abstrata de um objeto ou conceito do mundo real com características próprias e identificável. O MER permite especificar as entidades relevantes para o sistema, identificar os atributos que descrevem essas entidades e estabelecer os relacionamentos entre elas.

Uma entidade pode ser uma pessoa, um objeto, um evento ou qualquer outro elemento do domínio do problema. Os atributos são as características ou propriedades das entidades, como nome, idade, endereço, etc. Os relacionamentos representam as associações e dependências entre as entidades, como "pertence a", "tem" ou "ocorre entre".

O MER é uma etapa essencial no processo de desenvolvimento de sistemas de banco de dados, pois ajuda a entender e documentar a estrutura dos dados necessários para o funcionamento do sistema. Ele serve como base para a criação do modelo lógico do banco de dados, que define as tabelas, colunas e restrições específicas para armazenar os dados.

Em resumo, o Modelo Entidade-Relacionamento (MER) é uma técnica de modelagem de dados que permite representar as entidades, atributos e relacionamentos de um sistema. É uma ferramenta importante na etapa de análise e projeto de sistemas de banco de dados, auxiliando na compreensão e documentação dos requisitos de dados.

7.1 RELAÇÕES NO MODELO

Com base no diagrama de classes apresentado anteriormente, identifiquei as classes que precisam ser persistidas no banco de dados e criei tabelas correspondentes para cada uma delas. A seguir, apresento as tabelas resultantes, com suas respectivas colunas e chaves primárias:

Tabela 13: Produto

Coluna	Tipo de Dado	Chave Primária
ID	INT	Sim
Nome	VARCHAR(100)	Não
Descrição	TEXT	Não
Preço	DECIMAL(10,2)	Não
Estoque	INT	Não

Fonte: Autor (2023)

A tabela "Produto" armazena informações sobre os produtos disponíveis no sistema. Cada produto possui um ID único para identificação, nome, descrição, preço e quantidade em estoque.

Tabela 14: Categoria

Coluna	Tipo de Dado	Chave Primária
ID	INT	Sim
Nome	VARCHAR(100)	Não

Fonte: Autor (2023)

A tabela "Categoria" representa as categorias às quais os produtos podem ser associados. Ela possui um ID único para identificação e um nome descritivo.

Tabela 15: Pedido

Coluna	Tipo de Dado	Chave Primária
ID	INT	Sim
Data	DATETIME	Não
Total	DECIMAL(10,2)	Não
Status	VARCHAR(50)	Não

Fonte: Autor (2023)

A tabela "Pedido" registra informações sobre os pedidos realizados pelos clientes. Cada pedido é identificado por um ID único e contém a data do pedido, o total do valor da compra e o status atual do pedido.

Tabela 16: Pedido_Item (Relacionamento 1..n entre Pedido e Produto)

Coluna	Tipo de Dado	Chave Primária
PedidoID	INT	Sim
ProdutoID	INT	Sim

Quantidade	INT	Não
PreçoUnitário	DECIMAL(10,2)	Não

Fonte: Autor (2023)

A tabela "Pedido_Item" é usada para representar a relação 1..n entre os pedidos e os produtos que foram incluídos nesses pedidos. Ela possui chaves estrangeiras para os IDs de Pedido e Produto, além de armazenar a quantidade de itens do produto no pedido e o preço unitário.

Tabela 17: Produto_Categoria (Relacionamento n..n entre Produto e Categoria)

Coluna	Tipo de Dado	Chave Primária
ProdutoID	INT	Sim
CategoriaID	INT	Sim

Fonte: Autor (2023)

A tabela "Produto_Categoria" é utilizada para representar o relacionamento n..n entre produtos e categorias. Ela contém chaves estrangeiras para os IDs de Produto e Categoria, permitindo a associação de múltiplos produtos a múltiplas categorias.

Essas tabelas representam a estrutura do banco de dados para o sistema de vendas de produtos geek. Ao modelar o banco de dados, é fundamental garantir a integridade dos dados e a consistência das relações entre as entidades. Além disso, é importante considerar os requisitos e regras de negócio específicos do sistema em questão.

8 CONCLUSÃO

Ao longo deste trabalho, foi realizada a análise e modelagem de um sistema de vendas de produtos geek, abrangendo desde a identificação dos requisitos de negócio até a construção de diagramas e modelos essenciais para o desenvolvimento do sistema.

O trabalho foi iniciado com a definição dos requisitos funcionais e não funcionais, que nortearam todo o processo de modelagem. Com base nesses requisitos, foi feito o diagrama de casos de uso, que descreve as interações entre os atores e o sistema, evidenciando as principais funcionalidades a serem implementadas.

Em seguida, foi desenvolvido o diagrama de classes, que representa a estrutura do sistema, identificando as classes, atributos, relacionamentos e métodos. Esse diagrama foi essencial para a compreensão das entidades envolvidas no sistema e para o estabelecimento das relações entre elas.

Com base no diagrama de classes, foi realizada a modelagem do banco de dados utilizando o Modelo Entidade-Relacionamento (MER). Nesse processo, foram identificadas as classes que precisavam ser persistidas e então, foram criadas as tabelas correspondentes, definindo as chaves primárias, os atributos e as relações entre as entidades. Essa modelagem garante a organização e a integridade dos dados, possibilitando o armazenamento e recuperação eficiente das informações do sistema.

Por fim, concluiu-se o trabalho com a descrição detalhada das tabelas resultantes da modelagem do banco de dados, evidenciando as colunas, as chaves primárias e as relações estabelecidas entre as entidades. Além disso, destacando a importância de refinar e aprimorar a modelagem de acordo com as necessidades do projeto e as especificidades do sistema.

Neste trabalho, foi cumprido o objetivo de realizar a análise e modelagem de um sistema de vendas de produtos geek, aplicando conceitos acadêmicos e metodologias de engenharia de software. O processo de modelagem permitiu uma compreensão mais clara e estruturada do sistema, facilitando o desenvolvimento e garantindo a consistência das informações.

Em suma, a modelagem realizada neste trabalho proporcionou uma visão abrangente e precisa do sistema de vendas de produtos geek, fornecendo as bases necessárias para o desenvolvimento futuro do sistema. O cumprimento dos requisitos e a aplicação dos conceitos estudados demonstram a aplicabilidade dos conhecimentos acadêmicos estudados em cada disciplina.

9 REFERÊNCIAS

Cockburn, A. (2000). **Escrevendo Casos de Uso Eficazes**.

Elmasri, R., & Navathe, S. B. (2010). **Fundamentals of Database Systems**. Pearson Education.

FOWLER, M. **UML Essencial: um breve guia para linguagem padrão**. 3 ed. Bookman, 2011.

Larman, C. (2004). **Aplicando UML e Padrões: Uma Introdução à Análise e Design Orientados a Objetos e ao Desenvolvimento Iterativo.**

Martin Fowler, "UML Distilled: **A Brief Guide to the Standard Object Modeling Language**"

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2010). **Sistema de Banco de Dados.** Editora LTC.

Pressman, R. S. (2015). **Engenharia de Software: Uma Abordagem Profissional.**