

UNIP EaD  
Projeto Integrado Multidisciplinar  
Curso Superior de Tecnologia

LEVANTAMENTO E ANÁLISE DE REQUISITOS DE UM SISTEMA DE CONTROLE  
DE VENDAS DE UMA LOJA DE JOGOS, ACESSÓRIOS E PRODUTOS GEEK

São Paulo  
2022

UNIP EaD  
Projeto Integrado Multidisciplinar  
Curso Superior de Tecnologia

LEVANTAMENTO E ANÁLISE DE REQUISITOS DE UM SISTEMA DE CONTROLE  
DE VENDAS DE UMA LOJA DE JOGOS, ACESSÓRIOS E PRODUTOS GEEK

**Alunos:**

Antônia Daniele Araújo Alves RA:2114449  
Flavio Grandine RA:0439236  
Henrique Cesar Toledo Ferraro RA:0420525  
Marcos Vinicius de Almeida Martins RA:0427923  
Rodrigo Amorin Evangelista RA:0426989

Curso: Análise e Desenvolvimento de  
Sistemas

Semestre: 2º Semestre

São Paulo

2022

## **RESUMO**

O objetivo principal desse projeto, é desenvolver o levantamento e análise de requisitos de um sistema de controle de vendas de uma loja de jogos, acessórios e produtos geek.

Devido a alta demanda e a falta de eficiência e eficaz para realizar uma venda, que hoje são processadas tudo em planilhas em Excel, surgiu a necessidade de desenvolver um software que possa atender as necessidades da loja, com a intenção de ganhar mais agilidade, eficiência, maior controle nas vendas, melhoria no atendimento e até mesmo agilidade ao realizar uma análise para uma tomada de decisão.

Para isso o cliente contratou a empresa BJ Geek Soluções, especialista em desenvolvimento e software, para desenvolver um sistema que atenda todas as suas necessidades. Sendo assim será necessário fazer um levantamento dos requisitos e as devidas funcionalidades, como cadastro de cliente, cadastro de produtos, cadastro de funcionario, template de acesso aos sistema, controle de vendas e estoque, claro que levando em consideração a usabilidade, confiabilidade, acessibilidade entre outros.

Sendo assim, vamos utilizar os conhecimentos adquiridos nas disciplinas análise de sistema orientada a objetos, banco de dados e gestão estratégica de recursos humanos para realizar este projeto.

Palavras-chave: Engenharia de Software, Orientação a Objetos, Banco de Dados e Gestão de Recursos Humanos.

## **ABSTRACT**

The main objective of this project is to develop the survey and analysis of requirements for a sales control system for a game store, accessories and geek products.

Due to the high demand and the lack of efficiency and effectiveness to make a sale, which today are all processed in Excel spreadsheets, the need arose to develop a software that can meet the needs of the store, with the intention of gaining more agility, efficiency, greater control in sales, improvement in service and even agility when performing an analysis for decision making.

For this, the client hired the company BJ Geek Soluções, a specialist in development and software, to develop a system that meets all its needs. Therefore, it will be necessary to survey the requirements and the appropriate functionalities, such as customer registration, product registration, employee registration, system access template, sales and stock control, of course taking into account usability, reliability, accessibility between others.

Therefore, we will use the knowledge acquired in the disciplines object-oriented system analysis, database and strategic management of human resources to carry out this project.

Keywords: Software Engineering, Object Orientation, Database and Human Resources Management.

## SUMÁRIO

1. Introdução.....	6
2. Processo atual do cliente.....	7
3. Softwares disponíveis no mercado.....	7
4. Desenvolvimento do Software.....	8
5. Casos de uso.....	8
6. Metodologia.....	16
7. Análise e levantamento dos requisitos.....	16
7.1. Requisitos.....	16
7.1.1. Requisitos funcionais.....	16
7.1.2. Requisitos não funcionais.....	19
8. Modelagem de processos de negócios.....	21
9. Regras de negócio.....	22
10. Diagrama da classes.....	24
10.1. Descrição das classes.....	25
11. Modelo de entidade relacional.....	26
12. Conclusão.....	29
13. Referências.....	30

## **INTRODUÇÃO**

O projeto a ser desenvolvido trata-se do levantamento e análise de requisitos de um controle de vendas de uma loja de jogos, acessórios e produtos geek com a intenção de substituir o sistema que é utilizado atualmente para um sistema autoamtizado facilitando a usabilidade e desempenho para o cliente.

Além dos conceitos de recursos humanos, que vai contextualizar o processo de recrutamento e seleção e alguns funcionários, outros conceitos serão utilizados, como o conceito de banco de dados querepresenta o comportamento dos dados no SGBD e definindo um modelo de entidade relacional e o conceito da análise de sistema orientada a objetos que tem como principio fazer um levantamento e análise dos requisitos para um desenvolvimento mais sólido e acertivo proporcionando todos os requisitos solicitados pelo cliente.

## **2. PROCESSO ATUAL DO CLIENTE**

O cliente, atualmente, possui seus controles através de arquivos em Excel, desde cadastro de clientes, vendas até o controle de estoque, deixando de ser eficiente e eficaz, contribuindo para falhas nos dados cadastrados acarretando muita perda de tempo e alta demanda de mão de obra na alimentação das planilhas e na análise dos dados.

Sem ter um controle de acesso eficaz, hoje todos os colaboradores possuem acessos a todas as informações da empresa, principalmente da parte financeira onde possuem informações confidenciais.

A fim de garantir maior agilidade, controle nas vendas e uma ter uma análise de dados mais assertiva na hora de tomar uma decisão, a empresa de produtos geek decidiu automatizar os processos para cadastro de cliente, vendas, controle de estoque e financeiro. Para isso decidiu contratar a empresa BJ Geek Soluções, que é especialista de desenvolvimento de software para ajudá-la nesta solução.

## **3. SOFTWARES DISPONÍVEIS NO MERCADO**

Nos dias de hoje, independente do segmento e do tamanho da empresa, é extremamente importante dispor de um software que auxilie no gerenciamento do seu negócio. Eles possuem um papel fundamental pois facilitam a entrada e saída de mercadorias, maior controle de estoque, maior controle no fluxo de caixa, entre outros.

Com o aumento da tecnologia, existem diversas empresas que trabalham no desenvolvimento de sistemas de gerenciamento de vendas, os famosos CRM e ERP. Possuem os softwares gratuito, que possuem algumas restrições como não emitem NF-E, não possui controle de estoque, entre outros e os pagos, sendo que alguns já prontos, geralmente já são softwares já específicos para o ramo de atuação e outros que são desenvolvidos desde o início, atendendo todas as necessidades do cliente.

Analisando em todas as necessidades do cliente a empresa BJ Geek Soluções irá desenvolver um software para a empresa Produtos Geek atendendo todas as suas necessidades e aos requisitos do software, requisitos funcionais e não funcionais, tornando uma ferramenta viável para o cliente e ao mesmo tempo disponibilizando para o mercado.

A vantagem de um software ser desenvolvido com as peculiaridades da empresa são os conhecimentos do trabalho diário, sentindo as “dores” que a empresa tem, para proporcionarmos maior produtividade e assertividade nesse desenvolvimento com a intenção de contribuir para a facilidade no treinamento de funcionários, facilidade na implementação, custo-benefício e flexibilidade.

Objetivos do sistema a ser desenvolvido pela BJ Geek Soluções, será criar um software com os templates para cadastro de clientes, cadastro de produtos, vendas, controle de estoque e financeiro, com a possibilidade de gerar relatórios, restringindo

acessos para os funcionários, controle no fluxo de caixa, cada produto terá código de barras, e muito mais.

#### 4. DESENVOLVIMENTO DO SOFTWARE

Por se tratar de um projeto novo, não somente para o cliente, mas para disponibilizar para o mercado, a BJ Geek Soluções terá que contratar mais três colaboradores. Para tais contratações, a área de Recursos Humanos foi acionada para desempenhar este papel de recrutamento. O recrutador técnico é o responsável por identificar os candidatos mais adequados para a vaga.

Após as avaliações e entrevistas com os candidatos, o recrutador técnico irá selecionar os candidatos mais adequados para a vaga e que possuem o mesmo perfil da empresa para fazerem parte desse time.

Realizado as contratações, o time para desenvolver este software são: 1 coordenador de projetos, 1 Scrum Master e 4 desenvolvedores (sendo um deles que atuará como PO).

#### 5. CASOS DE USO

Segundo o Wikipédia, “**caso de uso** (do [inglês](#) *use case*) é um tipo de classificador representando uma unidade funcional coerente provida pelo sistema, subsistema, ou classe manifestada por sequências de mensagens intercambiáveis entre os sistemas e um ou mais atores.”

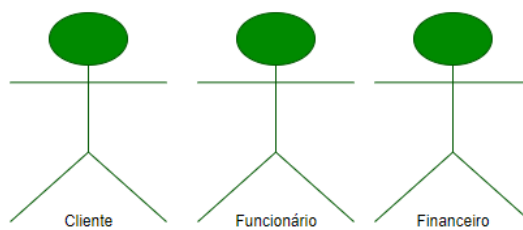
Para o desenvolvimento do novo software, será utilizado o Diagrama de caso de uso, que tem como objetivo descrever de como será o uso de uma funcionalidade no sistema. Este diagrama não tem a intenção de descrever como será construído o software, mas sim de como ele deve se comportar quando estiver concluído. É uma excelente ferramenta para fazer o levantamento de requisitos funcionais.

Para auxiliar na modelagem de estrutura que vai compor a aplicação e suportar nos conceitos de Orientação ao objeto, vamos utilizar a Linguagem de Modelagem Unificada (UML). Traz para o contexto computacional a padronização de uma linguagem utilizada para representar informações referentes a um sistema.

Para a construção dos casos de uso, pode-se destacar alguns elementos fundamentais como: ator e o cenário. Ator são pessoas, organizações que interagem diretamente com o sistema. E Cenário é a sequência específica de ações e interações entra atores e o sistema. Conforme mostra na figura 1, os atores são:

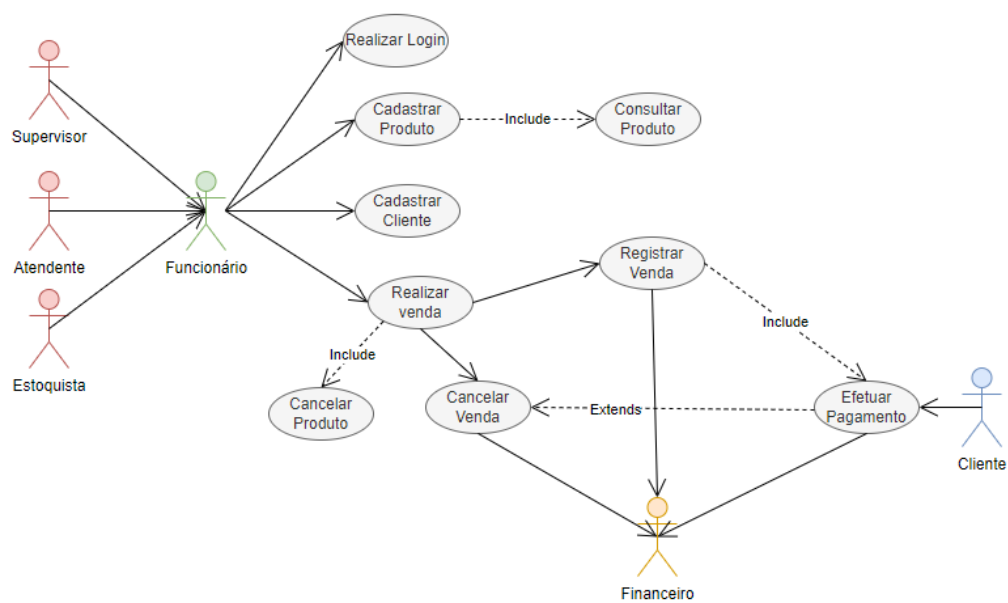


Figura. 1 – Atores do Sistema



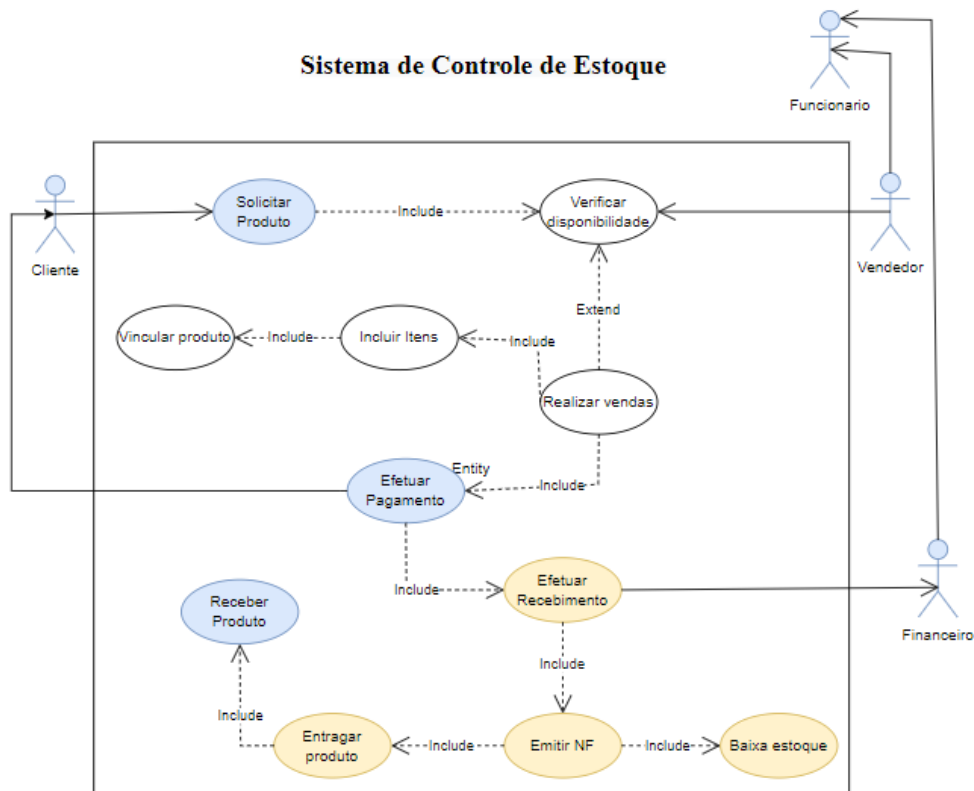
Fonte: o autor (2022)

Figura. 2 – Diagrama de caso de uso



Fonte: o autor (2022)

Figura. 3 - Diagrama UML de Estoque



Fonte: o autor (2022)

Seguem os modelos de casos de uso:

### Template 1 — US001 | Login e Senha

<b>Identificação:</b>	Acessar Sistema
<b>Escopo:</b>	Sistema configurado no computador.
<b>Descrição:</b>	Este caso de uso permite ao funcionário realizar acesso ao sistema.
<b>Ator Primário:</b>	Funcionário / Atendente / Supervisor / Estoquista
<b>Interessados:</b>	Funcionário / Atendente / Supervisor / Estoquista
<b>Pré-Condições:</b>	O usuário deve possuir login e senha.
<b>Pós-Condições:</b>	Autenticação do usuário válida, usuário entra no sistema.
<b>Fluxo Principal</b>	<ol style="list-style-type: none"> <li>1. O Usuário digita o Login e Senha.</li> <li>2. O Sistema vai checar a autenticação.</li> </ol>

	3 . Autenticação válida. 4 . Acesso ao sistema.
<b>Fluxo Alternativo</b>	2.1. ERROR: Em caso de autenticação inválida,o usuário deve solicitar acesso ao administrador.
<b>Requisitos Relacionados:</b>	RF001 - acessar sistema RNF005 - Autenticação de acesso

Fonte: O autor (2022)

## Template 2 — US002 | Cadastro Cliente

<b>Identificação:</b>	Cadastrar Cliente
<b>Escopo:</b>	O atendente deve cadastrar o cliente no sistema.
<b>Descrição:</b>	Este caso de uso permite ao funcionário/ atendente realizar o cadastro do cliente no sistema.
<b>Ator Primário:</b>	Atendente
<b>Interessados:</b>	Funcionário / Atendente / Supervisor / Cliente
<b>Pré-Condições:</b>	O atendente deve estar logado no sistema.
<b>Pós-Condições:</b>	Após o cliente ser cadastrado, permite que ele realize compras futuras sem a necessidade de realizar um novo cadastro
<b>Fluxo Principal:</b>	1. O Cliente deseja realizar a compra do produto/serviço. 2. Atendente verifica cadastro do cliente. 3. Cliente não cadastrado, atendente realiza cadastro. 4. Cliente cadastrado, atendente prossegue com a venda.
<b>Fluxo Alternativo:</b>	* Em caso de não haver cadastro 3.1. Atendente entra no template do cadastro. 3.2. Atendente cadastra dados do cliente. 3.3. Atendente salva dados do cliente
<b>Requisitos Relacionados:</b>	RF002 - Cadastrar Cliente

Fonte: O autor (2022)

### Template 3 — US003 | Cadastro Produtos

<b>Identificação:</b>	Cadastrar Produtos
<b>Escopo:</b>	O Estoquista deve cadastrar os produtos no sistema.
<b>Descrição:</b>	Este caso de uso permite ao funcionário / Estoquista realizar o cadastro dos produtos o sistema.
<b>Ator Primário:</b>	Estoquista
<b>Interessados:</b>	Funcionário / Atendente / Estoquista
<b>Pré-Condições:</b>	O estoquista deve estar logado no sistema.
<b>Pós-Condições:</b>	Após o produto cadastrado, permite que a atendente realize as vendas.
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"><li>1. O estoquista acessa o sistema.</li><li>2. O estoquista preenche os dados do produto.</li><li>3. O estoquista seleciona a categoria do produto.</li><li>4. O estoquista salva o produto no sistema.</li><li>5. O supervisor altera e exclui os dados do produto no sistema.</li></ol>
<b>Fluxo Alternativo:</b>	<ol style="list-style-type: none"><li>4.1. Em caso de falha de inserção de produto, item 2, verificar se os dados foram preenchidos corretamente.</li></ol>
<b>Requisitos Relacionados:</b>	RF003 - Cadastrar produtos

Fonte: O autor (2022)

### Template 4 — US004 | Consulta Produtos

<b>Identificação:</b>	Consultar Produtos
<b>Escopo:</b>	Possibilita a consulta de produtos no sistema.
<b>Descrição:</b>	Este caso de uso permite o atendente realizar a consulta de produtos no sistema caso o cliente solicite alguma informação sobre o produto. Para o estoquista permite verificar o controle de estoque.
<b>Ator Primário:</b>	Funcionário / Atendente / Estoquista
<b>Interessados:</b>	Funcionário / Atendente / Estoquista
<b>Pré-Condições:</b>	O estoquista e o atendente deve estar logado no sistema e o produto deve estar cadastrado.
<b>Pós-Condições:</b>	Possibilita maior agilidade na consulta nas descrições do produto.
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"><li>1. O Cliente solicita informação sobre o produto/serviço.</li></ol>

	2. Atendente entra no template de Produtos. 3. Atendente consulta produto pelo código de barra ou código. 4. Atendente informa cliente. Por exemplo: "valor do produto"
<b>Fluxo Alternativo:</b>	3.1. Em caso de produto não cadastrado, atendente entra em contato com estoquista.
<b>Requisitos Relacionados:</b>	RF004 - Consultar produtos

Fonte: O autor (2022)

### Template 5 — US005 | Registrar Venda

<b>Identificação:</b>	Registrar Venda
<b>Escopo:</b>	Registro das vendas realizadas.
<b>Descrição:</b>	Este caso de uso permite registrar todas as vendas realizadas, assim como análise de faturamento e controle de estoque
<b>Ator Primário:</b>	Atendente
<b>Interessados:</b>	Atendente / Cliente / Sistema Financeiro
<b>Pré-Condições:</b>	O atendente deve estar logado no sistema e deve ter produto disponível em estoque
<b>Pós-Condições:</b>	Atendente verifica produto em estoque, caso tenha, cliente realiza a compra.
<b>Fluxo Principal:</b>	1. O cliente realiza a compra de um produto/serviço. 2. Atendente verifica disponibilidade de produto. 3. A atendente registra venda no sistema. 4. Atendente solicita a forma de pagamento. 5. Cliente realiza pagamento
<b>Fluxo Alternativo:</b>	3.1. Em caso de cancelamento da venda solicitada pelo cliente, o supervisor deve ser, pois somente ele pode cancelar uma venda, devendo digitar o login e senha.
<b>Requisitos Relacionados:</b>	RF005 - Registrar venda

Fonte: O autor (2022)

## Template 6 — US006 | Cancelar Venda

<b>Identificação:</b>	Cancelar Venda
<b>Escopo:</b>	Cancelamento da venda solicitado pelo cliente.
<b>Descrição:</b>	Este caso de uso permite cancelar a venda, caso o cliente deseja desistir da compra.
<b>Ator Primário:</b>	Supervisor
<b>Interessados:</b>	Atendente / Supervisor / Cliente / Sistema Financeiro
<b>Pré-Condições:</b>	O atendente deve estar logado no sistema e supervisor deve informar login e senha para realizar o cancelamento.
<b>Pós-Condições:</b>	Após o cancelamento da venda, é gerado um código de cancelamento e encaminhado para o sistema financeiro
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"><li>1. O cliente solicita o cancelamento da venda.</li><li>2. A atendente solicita ao supervisor o cancelamento.</li><li>3. O supervisor informa seu login e senha.</li><li>4. O supervisor cancela a venda.</li><li>5. Sistema realiza cancelamento da venda.</li><li>6. Sistema envia um código de cancelamento para o financeiro.</li></ol>
<b>Fluxo Alternativo:</b>	5.1. No item 5, em caso de falha no cancelamento, sair da função e repetir os passos a partir do passo 4.
<b>Requisitos Relacionados:</b>	RF008 - Cancelar vendas

Fonte: O autor (2022)

## Template 7 — US007 | Cancelar Produto

<b>Identificação:</b>	Cancelar Produtos
<b>Escopo:</b>	Cancelamento do produto solicitado pelo cliente.
<b>Descrição:</b>	Este caso de uso permite cancelar o produto, caso o cliente deseja desistir de um determinado produto.
<b>Ator Primário:</b>	Supervisor
<b>Interessados:</b>	Atendente / Supervisor / Cliente / Sistema Financeiro

<b>Pré-Condições:</b>	O atendente deve estar logado no sistema e supervisor deve informar login e senha para realizar o cancelamento.
<b>Pós-Condições:</b>	Supervisor cancela produto.
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"> <li>1. O cliente solicita o cancelamento do produto.</li> <li>2. A atendente solicita ao supervisor o cancelamento.</li> <li>3. O supervisor informa seu login e senha.</li> <li>4. O supervisor informa qual produto será removido.</li> <li>5. Supervisor realiza cancelamento do produto.</li> <li>6. Produto volta para o estoque.</li> </ol>
<b>Fluxo Alternativo:</b>	5.1. No item 5, em caso de falha no cancelamento, sair da função e repetir os passos novamente.
<b>Requisitos Relacionados:</b>	RF007 - Cancelar Produtos

Fonte: O autor (2022)

## Template 8 — US008 | Efetuar Pagamento

<b>Identificação:</b>	Efetuar Pagamento
<b>Escopo:</b>	Cliente realiza o pagamento.
<b>Descrição:</b>	Este caso de uso permite que o cliente realiza o pagamento da sua venda, podendo ser realizado em dinheiro ou cartão.
<b>Ator Primário:</b>	Cliente
<b>Interessados:</b>	Atendente / Supervisor / Cliente / Sistema Financeiro
<b>Pré-Condições:</b>	O atendente deve estar logado no sistema.
<b>Pós-Condições:</b>	Cliente realiza o pagamento e retira o produto adquirido.
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"> <li>1. O cliente escolhe a forma de pagamento.</li> <li>2. O atendente informa ao sistema a forma de pagamento.</li> <li>3. Sistema valida a forma de pagamento.</li> <li>4. Pagamento realizado.</li> </ol>
<b>Fluxo Alternativo:</b>	1.1. Caso o cliente deseja pagar em cartão e aparece alguma mensagem de erro, a atendente deve retirar o cartão e inserir novamente na máquina.
<b>Requisitos Relacionados:</b>	RF006 - Efetuar pagamento

Fonte: O autor (2022)

## **6. METODOLOGIA**

Com o objetivo de ter uma maior interação entre cliente e equipe de desenvolvimento e afim de entregar um produto de qualidade, este projeto será apoiado pela metodologia Scrum. Este desenvolvimento é composto por mini entregas ao longo de um ciclo que pode durar entre 1 a 4 semanas. Já que nesse período o projeto passa por feedbacks, acreditamos que possa auxiliar em um desenvolvimento mais ágil e com menos manutenção.

## **7. ANÁLISE E LEVANTAMENTO DOS REQUISITOS**

### **7.1. REQUISITOS**

Tudo que deve ser desenvolvido em um sistema, precisa conter alguns requisitos como, necessidades do usuário, regras de negócio, ideias e desejos da empresa. A descrição das características e funcionalidades que um sistema deve realizar, são chamadas de requisitos.

Os requisitos funcionais e requisitos não funcionais são as duas categorias que se destacam no quesito de requisitos. Os requisitos funcionais especificam as ações que o sistema deve executar, ou seja, eles descrevem o que o software deve realizar. Já os requisitos não funcionais especificam as ações de como o todo do sistema funcionará. Os requisitos de usabilidade, desempenho e atributos de qualidade são alguns exemplos de requisitos importantes para o desenvolvimento do software, eles quem definiram a arquitetura do projeto, como será implementado, quais tecnologias serão utilizadas, e muito mais.

Uma das principais atividades, no momento do planejamento de um software, é a captação dos requisitos, são eles que darão a direção nesse processo de desenvolvimento. É neste momento que será coletado informações sobre como é o sistema atualmente, quais são os processos que os colaboradores realizam hoje para realizar a venda de um produto, o que o cliente deseja etc.

Existem diversas formas que se realizar esta coleta, são elas: Entrevistas, cenários, casos de uso, entre outros.

#### **7.1.1 REQUISITOS FUNCIONAIS**

Segundo Pfleeger (2004):

“requisitos funcionais são como o detalhamento da interação entre o sistema de software e o seu ambiente. Podemos considerar que um componente importante do ambiente do software é o seu usuário. Logo, podemos considerar que o requisito funcional é uma descrição detalhada da interação entre o software e o usuário.”



SOMMERVILLE (2010) descreve requisitos funcionais como:

“o comportamento esperado de um sistema de software, explicita o que o sistema deve fazer e idealmente o que o sistema não deve fazer.”

Segue a descrição dos requisitos funcionais (RF) do sistema:

<b>Acesso</b>	
<b>RF 001</b>	
<b>Identificação:</b>	Acessar sistema
<b>Classificação:</b>	Importante
<b>Ator:</b>	Funcionário / Atendente/ Estoquista / Supervisor
<b>Objetivo:</b>	Este requisito especifica um caso de uso o qual o usuário só terá permissão de acesso no sistema através de um Login e senha.

<b>Cliente</b>	
<b>RF002</b>	
<b>Identificação:</b>	Cadastrar Cliente
<b>Classificação:</b>	Essencial
<b>Ator:</b>	Atendente
<b>Objetivo:</b>	Demonstrar um caso de uso o qual os clientes deverão ser cadastrados: Nome, RG, CPF, data de nascimento, data de cadastro (Será gerado automaticamente pelo sistema), endereço, telefone, e-mail e um código do cliente (Será gerado automaticamente pelo sistema).

<b>Produtos</b>	
<b>RF003</b>	
<b>Identificação:</b>	Cadastrar Produtos
<b>Classificação:</b>	Essencial
<b>Ator:</b>	Funcionário / Estoquista
<b>Objetivo:</b>	Mostrar um caso de uso em que os produtos que serão vendidos deverão ser cadastrados no sistema e deverão ser divididos por categorias: jogos, acessórios e produtos geek.

<b>Produtos</b>	
<b>RF004</b>	
<b>Identificação:</b>	Consultar Produtos
<b>Classificação:</b>	Essencial
<b>Ator:</b>	Funcionário / Estoquista / Atendente

<b>Objetivo:</b>	Demonstrar o caso de uso em que os produtos cadastrados no sistema devem possuir as seguintes informações de identificação tais como; Código de barras, nomes do produto, categoria, fabricante, quantidade e valor do produto. Para os jogos e os acessórios, deverão ser informados em qual plataforma serão utilizados e qual é o prazo de garantia que o produto possui pelo fabricante. Dessa forma, tais informações serão aproveitadas e utilizadas para gerenciamento no estoque e apresentação na nota fiscal do consumidor.
------------------	---

Venda	
<b>RF005</b>	
<b>Identificação:</b>	Registrar Venda
<b>Classificação:</b>	Essencial
<b>Ator:</b>	Funcionário / Atendente
<b>Objetivo:</b>	Demonstrar um caso de uso o qual a venda irá conter os dados do cliente e dos respectivos produtos comprados. O sistema irá gerar um código de venda, com a data da venda (Será gerado automaticamente pelo sistema), o valor da venda, opções para pagamento, status de pagamento e status de venda.

Venda	
<b>RF006</b>	
<b>Identificação:</b>	Efetuar pagamento
<b>Classificação:</b>	Importante
<b>Ator:</b>	Cliente
<b>Objetivo:</b>	Demonstrar um caso de uso em que o realiza o pagamento da compra.

Venda	
<b>RF007</b>	
<b>Identificação:</b>	Cancelar Produtos
<b>Classificação:</b>	Essencial
<b>Ator:</b>	Funcionário / Supervisor
<b>Objetivo:</b>	Este requisito especifica o caso de uso em que o atendente poderá excluir produtos das vendas caso o cliente solicite o cancelamento. Porém, apenas o supervisor da loja terá autorização para excluir um produto da venda, devendo informar um usuário e senha.

<b>Venda</b>	
<b>RF008</b>	
<b>Identificação:</b>	Cancelar Venda
<b>Classificação:</b>	Essencial
<b>Ator:</b>	Funcionário / Supervisor
<b>Objetivo:</b>	Demonstrar um caso de uso em que o atendente poderá cancelar uma venda caso o cliente solicite o cancelamento. Logo, apenas o supervisor da loja terá permissão para excluir o produto da venda, devendo informar um usuário e senha válidos. No momento do cancelamento, será gerado um código de cancelamento da venda e será enviado para o financeiro.

### 7.1.2. REQUISITOS NÃO FUNCIONAIS

Para SOMMERVILLE (2010):

“Requisitos não funcionais (RNF) descrevem restrições sobre os serviços oferecidos pelo sistema de software.”

Alguns termos podem substituir o requisito não funcional, como atributo de qualidade, restrições e requisitos não comportamentais, requisitos de qualidade de serviço, objetivos de qualidade, estabilidade e portabilidade.

As duas principais categorias de qualidades são: Qualidade de execução e qualidade de evolução. Qualidade de execução é segurança e usabilidade. Qualidade de evolução é testabilidade, capacidade de manutenção, extensibilidade e escalabilidade.

Segue a descrição dos requisitos não funcionais (RNF) do sistema:

<b>Eficiência</b>	
<b>RNF 001</b>	
<b>Identificação:</b>	Requisitos de Configuração
<b>Classificação:</b>	Importante
<b>Objetivo:</b>	São os requisitos mínimos de configuração que um computador deve ter para rodar o sistema. Requisitos de Configuração do sistema computacional: 20 GB de espaço no disco rígido 4 GB de Memória RAM, Processador Dual Core 2.8 GHz, Sistema Operacional Windows 10 / Linux 15.10

<b>Usabilidade</b>	
<b>RNF002</b>	
<b>Identificação:</b>	Interface do Usuário
<b>Classificação:</b>	Essencial
<b>Objetivo:</b>	Este requisito tem como objetivo proporcionar uma interface do usuário amigável e de fácil entendimento. Os requisitos de usabilidade devem atender as necessidades do cliente como a operacionalidade.

<b>Confiabilidade</b>	
<b>RNF003</b>	
<b>Identificação:</b>	Rotina de Backups
<b>Classificação:</b>	Importante
<b>Objetivo:</b>	O requisito de confiabilidade tem o objetivo de realizar backup diariamente dos dados, a fim de garantir que os dados sejam recuperados em caso de algum problema.

<b>Funcionalidade</b>	
<b>RNF004</b>	
<b>Identificação:</b>	Autenticação de Acessos
<b>Classificação:</b>	Importante
<b>Objetivo:</b>	O requisito de segurança tem como objetivo limitar e garantir a segurança em relação a acesso ao sistema. Somente pessoas autorizadas a cadastradas poderão ter acesso ao sistema utilizando usuário e senha.

<b>Portabilidade</b>	
<b>RNF005</b>	
<b>Identificação:</b>	Compatibilidade em sistemas Operacionais
<b>Classificação:</b>	Importante
<b>Objetivo:</b>	O requisito de compatibilidade tem como objetivo desenvolver um software compatível com diversos sistema operacionais como Windows, Linux e MAC.

Interoperabilidade	
<b>RNF006</b>	
<b>Identificação:</b>	Integração com API de emissão de notas fiscais
<b>Classificação:</b>	Importante
<b>Objetivo:</b>	Este requisito tem como objetivo que o sistema seja capaz de comunicar com API outros sistemas, como cep e até sistemas para emissão de notas fiscais.

## 8. MODELAGEM DE PROCESSOS DE NEGÓCIOS

O conjunto de umas atividades previamente estabelecidas, que visa compreender, comunicar e otimizar processos dentro de uma organização, é chamado de modelagem de processos de negócios.

Esta atividade exige um nível alto de detalhamento como as atividades que devem ser desenvolvidas, relação da comunicação entre as pessoas no quesito de negócio, quem serão as pessoas responsáveis pelas atividades entre outros.

A modelagem de processos auxiliará no entendimento do negócio que será construído pelos profissionais da tecnologia da informação. Alguns princípios da modelagem de processo são: a interação entre o cliente e a empresa, agregar valor para o cliente, diminuir os atrasos no fluxo durante a troca de atividades, evitar automação em excesso, padronização dos processos de negócios, prever a regra de negócios, aplicação padrão de conformidade, validação da modelagem e a simplicidade no desenho dos processos.

As seis perspectivas de um processo de negócio são: Por quê/Motivo, Onde, Quando, Quem, O que e como. Neste projeto as perspectivas são:

- Por quê/Motivo: A empresa de produtos geek quer automatizar os processos de vendas de seus produtos, a fim de ganhar agilidade e eficiência no processo.
- Onde: Este processo será executado, a empresa BJ Geek Soluções e o software será desenvolvido para operar em desktop.
- Quando: Após a realização das pesquisas e do entendimento a previsão é que o software seja realizado dentro de 2 ou no máximo 4 sprints, conforme metodologia ágil Scrum.
- Quem: As pessoas responsáveis por este processo serão os colaboradores da BJ Geek Soluções (Coordenador, Scrum Master, PO, desenvolvedores) e o cliente.

- O que: Será desenvolvido um software com o objetivo de cadastrar clientes, realizar vendas e ter um controle de estoque e financeiro.
- Como: Após a empresa de produtos geek contratar a empresa BJ Geek Soluções, serão realizadas reuniões para o entendimento do projeto e qual é o objetivo que o software deve alcançar. Tudo isso com a participação de todos os envolvidos e procurando atender as expectativas dentro do prazo.

## 9. REGRAS DE NEGÓCIO

A definição do modelo ao qual a organização realiza as atividades e entrega os seus produtos ou serviços, são conhecidos como regra de negócios. A regra de negócios, em um processo de desenvolvimento de software, é um dos elementos mais importantes durante a fase de levantamento e modelagem de requisitos.

Segundo o site da Wikipédia:

**“Regras de Negócio** são declarações sobre a forma da empresa fazer negócio. Elas refletem políticas do negócio. As organizações com isto têm políticas para satisfazer os objetivos do negócio, satisfazer clientes, fazer bom uso dos recursos, e obedecer às leis ou convenções gerais do negócio.”

<b>Identificador:</b>	RN001
<b>Nome:</b>	Acessar sistema
<b>Descrição:</b>	Os acessos deverão ser realizados por login e senha
<b>Autor:</b>	Funcionário / Atendente / Supervisor / Estoquista
<b>Versão:</b>	2.0

<b>Identificador:</b>	RN002
<b>Nome:</b>	Cadastrar Cliente
<b>Descrição:</b>	O cliente deve ser cadastrado no sistema e todos os campos devem ser preenchidos. Os atributos desse módulo são: Nome, RG, CPF, endereço, telefone e e-mail. Os atributos Código do cliente e data do cadastro, serão preenchidos automaticamente pelo sistema.
<b>Autor:</b>	Funcionário / Atendente
<b>Versão:</b>	2.0

<b>Identificador:</b>	RN003
<b>Nome:</b>	Cadastrar Produto
<b>Descrição:</b>	O estoquista deverá cadastrar todos os produtos que serão vendidos. Todos os campos deverão ser devidamente preenchidos. Os atributos do cadastro de produtos são: Nome do produto, categoria, fabricante, quantidade, valor do produto e prazo de garantia. O código do produto e o código de barras serão gerados automaticamente pelo sistema. Ao cadastrar Jogos e acessórios, será liberado um atributo para informar em qual plataforma o produto pode ser utilizado.
<b>Autor:</b>	Funcionário / Estoque
<b>Versão:</b>	2.0

<b>Identificador:</b>	RN004
<b>Nome:</b>	Consultar Produtos
<b>Descrição:</b>	O Atendente poderá realizar a consulta das descrições dos produtos, tais como valor do produto, fabricante, plataforma que pode ser utilizado entre outros.
<b>Autor:</b>	Funcionário / Estoque / Atendente / Supervisor
<b>Versão:</b>	2.0

<b>Identificador:</b>	RN005
<b>Nome:</b>	Registrar Venda
<b>Descrição:</b>	Todas as vendas vão conter os dados do cliente e dos produtos adquiridos. Todos os atributos desse módulo deverão ser preenchidos. Os atributos são: Opções para pagamento (Dinheiro / Cartão), status de pagamento e status de vendas. Os atributos código de venda e a data da venda serão preenchidos automaticamente pelo sistema. O valor da venda será informado de acordo com cada venda.
<b>Autor:</b>	Funcionário / Atendente
<b>Versão:</b>	2.0

<b>Identificador:</b>	RN006
<b>Nome:</b>	Cancelar Venda
<b>Descrição:</b>	Em caso de cancelamento de venda, o atendente deverá informar o supervisor sobre o ocorrido. Somente o supervisor poderá realizar o cancelamento da venda. O supervisor deverá inserir o seu login e senha para concluir o cancelamento. Todo cancelamento, será gerado um código de cancelamento e enviado para o sistema financeiro.
<b>Autor:</b>	Funcionário / Supervisor
<b>Versão:</b>	2.0

<b>Identificador:</b>	RN007
<b>Nome:</b>	Cancelar Produto
<b>Descrição:</b>	Em caso de cancelamento de produto, o atendente deverá informar o supervisor sobre o ocorrido. Somente o supervisor poderá realizar o cancelamento do produto. O supervisor deverá inserir o seu login e senha para concluir o cancelamento.
<b>Autor:</b>	Funcionário / Supervisor
<b>Versão:</b>	2.0

<b>Identificador:</b>	RN008
<b>Nome:</b>	Efetuar Pagamento
<b>Descrição:</b>	O Atendente deve perguntar para o cliente qual é a forma de pagamento (dinheiro / Cartão). Cliente realiza o pagamento e finaliza a compra.
<b>Autor:</b>	Funcionário / Atendente/ Cliente
<b>Versão:</b>	2.0

## 10. DIAGRAMA DE CLASSES

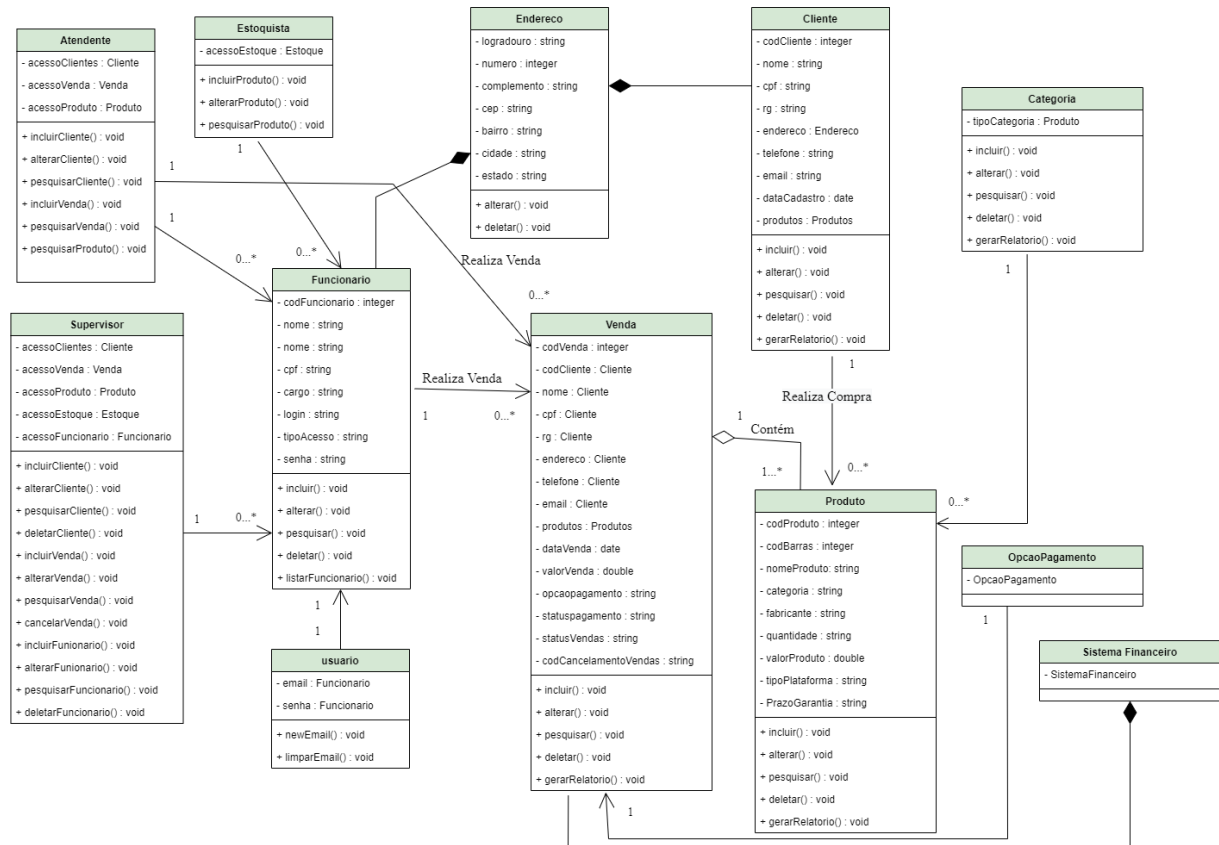
Para modelar a estrutura do software vamos utilizar o diagrama de classes. Este diagrama é um dos mais populares na Linguagem de modelagem unificada (UML) e tem como objetivo de apresentar uma documentação de programação orientada a objetos, mostrando os relacionamentos existentes entre as classes e como esses relacionamentos colaboram com a execução de um projeto. Nesta documentação, apresentará suas classes, seus atributos, operações e as relações entre objetos.

Alguns benefícios para utilizarmos o Diagrama de classe são expressar melhor visão as necessidades específicas de um sistema, fornece uma descrição de implementação de tipos utilizados em um sistema, ajuda na avaliação de desempenho, segurança, rastreamento e fornece diretrizes importantes para desenvolvimento em operação.

Na figura a seguir, mostra a estrutura do sistema representado pelo diagrama de classes.



Figura. 4 – Diagrama de Classe



Fonte: O autor (2022)

## 10.1. DESCRIÇÃO DAS CLASSES

**Cliente:** Esta classe é onde será realizado o cadastro do cliente. Nesta entidade irá conter os seguintes atributos: CodCliente, nome, cpf, rg, endereço, telefone, email, dataCadastro e produtos. Os métodos dessa classe são incluir, alterar, pesquisar, deletar e gerarRelatorio.

**Endereço:** Nesta classe será cadastrado o endereço do cliente e possuem os seguintes atributos: Logradouro, número, complemento, cep, bairro, cidade, estado. Esta classe possui uma relação de composição com a entidade Cliente.

**Funcionário:** Esta classe define os atributos relacionados com os funcionários. Nesta entidade serão cadastrados os seguintes atributos: codFuncionario, nome, cpf, cargo, login, senha e tipoAcesso. Esta entidade possui uma relação de dependência as entidades Supervisor, Atendente e Estoquista e seus métodos são incluir, alterar, pesquisar, deletar e gerarRelatorio. Possui uma relação de agregação com o atributo Endereço.

**Supervisor:** Esta classe vai permitir acesso as classes Cliente, Vendas, Produtos, Funcionários, Estoque e Financeiro. Os métodos para esta entidade

permitirão incluir, alterar, pesquisar e deletar os dados das classes Cliente, Vendas, Produtos Funcionários e Estoque. Esta entidade possui uma relação de herança da classe pai Funcionário.

Atendente: Esta classe vai permitir acesso as classes Cliente, Vendas e Produtos. Os métodos para esta entidade permitirão incluir, alterar, pesquisar os dados das classes Cliente, Vendas e na entidade Produto terá acesso apenas ao método pesquisar. Esta entidade possui uma relação de herança da classe pai Funcionário

Estoquista: Esta classe vai permitir acesso as classes Produtos e Estoque. Os métodos para esta entidade permitirão incluir, alterar, pesquisar os dados das classes Produtos e Estoque. Esta entidade possui uma relação de herança da classe pai Funcionário

Produto: Esta classe será responsável por realizar o cadastro dos atributos do produto, que são: codProduto, codBarras, nomeProduto, categoria, fabricante, quantidade, valorProduto, tipoPlataforma e prazoGarantia. Os métodos utilizados para esta função são incluir, alterar, pesquisar, deletar e gerarRelatorio. Esta entidade possui uma relação de dependência com a entidade cliente e uma relação de agregação com a entidade categoria.

Categoria: Os atributos dessa classe são definidos pela classe Produto. O relacionamento dessa classe com a classe Produto é de herança.

Venda: É nesta classe que serão computados as vendas. Os atributos desta classe são codVenda, codCliente, nome, cpf, rg, produtos, dataVenda, statusVendas, Oopcaopagamento, statusVendas, codCancelamentoVendas. Possui um relação de dependência com a entidade formapagamento

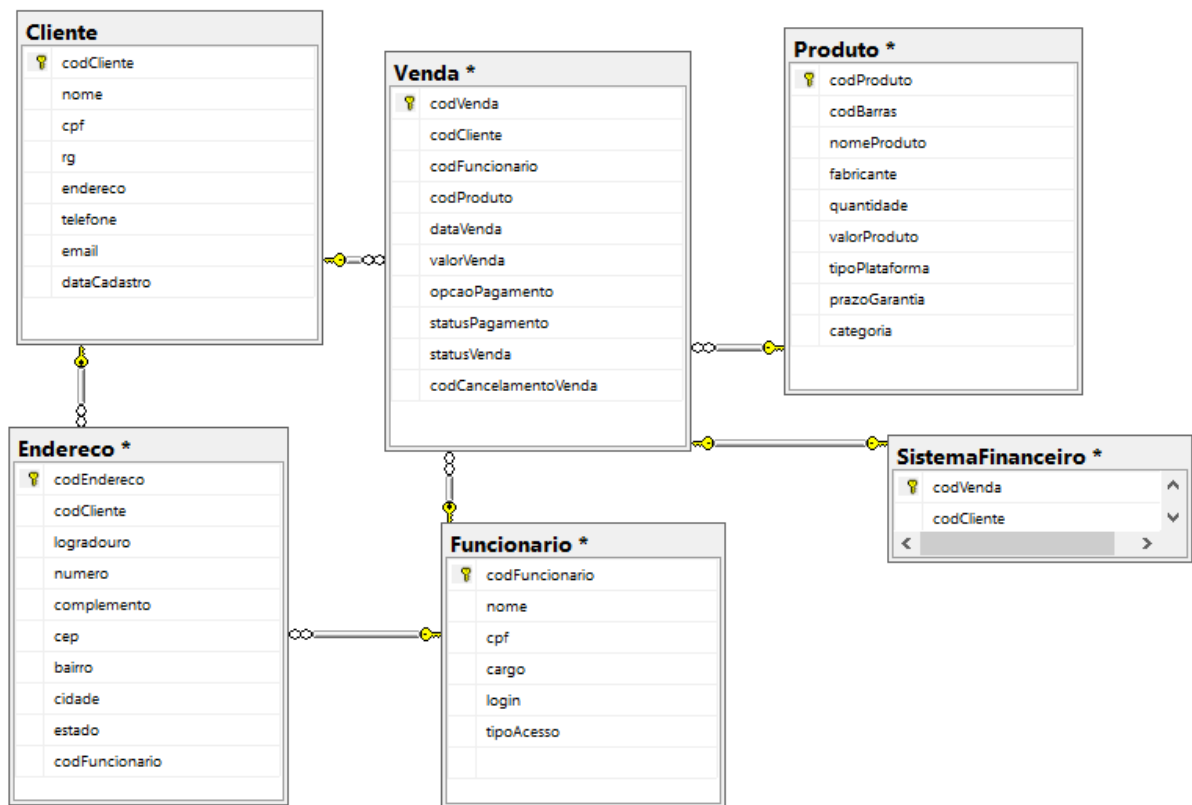
OpcaoPagamento: Os atributos dessa classe são definidos pela classe Venda. O relacionamento dessa classe com a classe Venda é de herança.

Financeiro: Esta classe que vai controlar e gerir os dados financeiros.

## **11. MODELO DE ENTIDADE DE RELACIONAMENTO**

Para este projeto, foi utilizado o SQL Server 15.0.18 como o banco de dados. O SQL Server é um sistema de gerenciamento de banco de dados relacional. Esse SGBD é um dos softwares mais usados no mercado, por ser um banco de dados inteligente que aumenta a velocidade de execução das operações e otimiza o processamento de consultas.

Figura. 5 – Modelo de Entidade de Relacionamento




Fonte: O autor (2022)


Cliente *			
Nome da Coluna	Tipo de Dados	Permitir Nulos	
codCliente	int	<input type="checkbox"/>	
nome	varchar(100)	<input type="checkbox"/>	
cpf	varchar(11)	<input type="checkbox"/>	
rg	varchar(9)	<input type="checkbox"/>	
endereco	varchar(100)	<input type="checkbox"/>	
telefone	varchar(20)	<input type="checkbox"/>	
email	varchar(100)	<input type="checkbox"/>	
dataCadastro	datetime	<input type="checkbox"/>	


Endereco *			
Nome da Coluna	Tipo de Dados	Permitir Nulos	
codEndereco	int	<input type="checkbox"/>	
codCliente	int	<input type="checkbox"/>	
logradouro	varchar(100)	<input type="checkbox"/>	
numero	int	<input type="checkbox"/>	
complemento	varchar(20)	<input type="checkbox"/>	
cep	varchar(9)	<input type="checkbox"/>	
bairro	varchar(50)	<input type="checkbox"/>	
cidade	varchar(100)	<input type="checkbox"/>	
estado	nchar(10)	<input type="checkbox"/>	
codFuncionario	int	<input type="checkbox"/>	

Fonte: O autor (2022)

<b>Venda *</b>			
	Nome da Coluna	Tipo de Dados	Permitir Nulos
	codVenda	int	<input type="checkbox"/>
	codCliente	int	<input type="checkbox"/>
	codFuncionario	int	<input type="checkbox"/>
	codProduto	int	<input type="checkbox"/>
	dataVenda	date	<input type="checkbox"/>
	valorVenda	float	<input type="checkbox"/>
	opcaoPagamento	varchar(15)	<input type="checkbox"/>
	statusPagamento	varchar(50)	<input type="checkbox"/>
	statusVenda	varchar(100)	<input type="checkbox"/>
	codCancelamentoVenda	int	<input type="checkbox"/>
			<input type="checkbox"/>

<b>Produto *</b>			
	Nome da Coluna	Tipo de Dados	Permitir Nulos
	codProduto	int	<input type="checkbox"/>
	codBarras	int	<input type="checkbox"/>
	nomeProduto	varchar(50)	<input type="checkbox"/>
	fabricante	varchar(50)	<input type="checkbox"/>
	quantidade	float	<input type="checkbox"/>
	valorProduto	varchar(100)	<input type="checkbox"/>
	tipoPlataforma	varchar(50)	<input type="checkbox"/>
	prazoGarantia	varchar(20)	<input type="checkbox"/>
	categoria	varchar(30)	<input type="checkbox"/>
			<input type="checkbox"/>

<b>SistemaFinanceiro *</b>			
	Nome da Coluna	Tipo de Dados	Permitir Nulos
	codVenda	int	<input type="checkbox"/>
	codCliente	int	<input type="checkbox"/>
			<input type="checkbox"/>

<b>Funcionario *</b>			
	Nome da Coluna	Tipo de Dados	Permitir Nulos
	codFuncionario	int	<input type="checkbox"/>
	nome	int	<input type="checkbox"/>
	cpf	varchar(20)	<input type="checkbox"/>
	cargo	varchar(9)	<input type="checkbox"/>
	login	varchar(50)	<input type="checkbox"/>
	tipoAcesso	varchar(100)	<input type="checkbox"/>

Fonte: O autor (2022)

## **CONCLUSÃO**

Este projeto foi elaborado com a finalidade de praticar os conceitos passados nas disciplinas de análise de sistema orientada a objetos, banco de dados e gestão estratégica de recursos humanos, no qual realizamos um levantamento e análise de requisitos de um sistema de controle de vendas de uma loja de jogos, acessórios e produtos geek.

A gestão estratégica de recursos humanos foi contextualizada no quesito de realizar as contratações de funcionários para o projeto. A definição dos requisitos, casos de uso, regras de negócios utilizamos a disciplina de análise de sistemas orientada a objetos e os diagramas de classes e modelos de entidade de relacionamento foi utilizado o conceito de banco de dados.

Foi realizado também, uma pesquisa de mercado para apresentar ao cliente as vantagens e desvantagens de se desenvolver um sistema de gerenciamento.

## REFERÊNCIAS

UNIP -Universidade Paulista. Ref.: Análise de Sistemas Orientada a Objetos. São Paulo 2022.

UNIP -Universidade Paulista. Ref.: Banco de dados. São Paulo 2022.

UNIP -Universidade Paulista. Ref.: Gestão Estratégica de Recursos humanos. São Paulo 2022.

VENTURA, Plínio. O que é um Requisito Não-Funcional. Disponível em: <https://www.ateomomento.com.br/o-que-e-um-requisito-nao-funcional/>. Acesso em: 27 maio 2022.

VENTURA, Plínio. O que é Requisito Funcional. Disponível em: <https://www.ateomomento.com.br/o-que-e-requisito-funcional/>. Acesso em: 27 maio 2022.

[https://ava.ead.unip.br/bbcswebdav/pid-2695401-dt-content-rid-2603779\\_1/institution/Conteudos\\_AVA/DISCIPLINAS\\_GERAIS/6858-60%20-%20An%C3%A1lise%20de%20Sistemas%20Orientada%20a%20Objetos/Livro-Texto%20%E2%80%93%20Unidade%20II.pdf](https://ava.ead.unip.br/bbcswebdav/pid-2695401-dt-content-rid-2603779_1/institution/Conteudos_AVA/DISCIPLINAS_GERAIS/6858-60%20-%20An%C3%A1lise%20de%20Sistemas%20Orientada%20a%20Objetos/Livro-Texto%20%E2%80%93%20Unidade%20II.pdf)

<https://www.infoescola.com/engenharia-de-software/uml/>

[https://dtic.tjpr.jus.br/wiki/-/wiki/Governan%C3%A7a-TIC/Modelo+de+Caso+de+Uso/pop\\_up](https://dtic.tjpr.jus.br/wiki/-/wiki/Governan%C3%A7a-TIC/Modelo+de+Caso+de+Uso/pop_up)

[https://pt.wikipedia.org/wiki/Processo\\_de\\_neg%C3%B3cio](https://pt.wikipedia.org/wiki/Processo_de_neg%C3%B3cio)

<https://i.stack.imgur.com/hhTO9.jpg>

<https://www.guj.com.br/t/modelo-entidade-relacionamento-de-vendas/346446>