**Documento de Visão de Projeto**

****

**E-commerce Esboço Comix**

**Histórico de Versões**

| **Data** | **Versão** | **Descrição** | **Autor** | **Revisor** |
| --- | --- | --- | --- | --- |
| 15/04/2025 | 1.0 | Modelagem e desenvolvimento | Nattan Silva de Souza | - |

| **Cliente** | FATEC - Interno |
| --- | --- |
| **Documento** | Documento de Visão de Projeto: *Esboço Comix* |
| **Data** | 15 de abril de 2025 |
| **Autor** | **Nattan Silva de Souza**  nattan.souza@fatec.sp.gov.br |

**Página de Assinaturas**

| Revisado e Aprovado por: |  |  |
| --- | --- | --- |
|  |  | 15/11/25 |

**Índice**

[**1. Objetivo 4**](#_heading=h.nnznzqyjn6ex)

[1.1. Escopo 4](#_heading=h.juzht52u3uel)

[1.2. Referências 4](#_heading=h.psj9br1x8d32)

[**2. Necessidades de Negócio 4**](#_heading=h.l7vl2tzavpri)

[**3. Objetivo do Projeto 4**](#_heading=h.t1028kh9szza)

[**4. Declaração Preliminar de Escopo 5**](#_heading=h.uhr4p8o2ahkb)

[4.1. Descrição 5](#_heading=h.zbiwhfi2pmn2)

[4.2. Produtos a serem entregues 5](#_heading=h.reaflndc7hm)

[4.3. Requisitos 5](#_heading=h.rfn2ol7l2b3n)

[4.3.1. Requisitos Funcionais 5](#_heading=h.jg04uq92vaj4)

[4.3.2. Requisitos Não Funcionais 5](#_heading=h.467h6u2e28up)

[4.3.3. Regras de Negócio 6](#_heading=h.r1qi7zer53wl)

[**5. Premissas 6**](#_heading=h.heptrzabzthf)

[**6. Influência das Partes Interessadas 6**](#_heading=h.9biux6qhr3b4)

[**7. Representação Arquitetural 7**](#_heading=h.f7rwv12xfnk6)

[7.1. Restrições Arquiteturais 8](#_heading=h.hu03w39sko85)

[7.2. Objetivos e Restrições Arquiteturais 8](#_heading=h.vmue58dy7q9u)

[**8. Visão de Use Case 8**](#_heading=h.pdzmxw83mpwy)

[8.1. Diagrama de Casos de Uso Realizar Pedido 9](#_heading=h.gmah0h1cqb5)

[8.2. Diagrama de Caso de Uso Gerenciar Pedidos 10](#_heading=h.mw8mhz022il)

[8.3. Diagrama de Caso de Uso Gerenciar Estoque 10](#_heading=h.wocn6p6asks1)

[8.4. Descrição dos Casos de Uso Arquiteturalmente Significativos 11](#_heading=h.oueoaw35qrq)

[**9. Visão de Lógica 12**](#_heading=h.lv7oa8h1r9zu)

[9.1. Camada de Apresentação 14](#_heading=h.97obiim2ky7a)

[9.2. Camada de Negócio 15](#_heading=h.yfy47jusuizg)

[9.3. Camada de Persistência 18](#_heading=h.useeprcbjkg5)

[9.4. Realização dos Casos de Uso Significativos 19](#_heading=h.lzl7zuefgccd)

[9.4.1. Realizar Pedido 19](#_heading=h.o8jmjmrxzbvr)

[9.4.2. Gerenciar Pedidos 20](#_heading=h.8pa4a5u6ry6r)

[9.4.3. Gerenciar Estoque 20](#_heading=h.kbj17gfveiv2)

[**10. Visão de Implantação 21**](#_heading=h.bte1navug8v)

[**11. Visão de Implementação 22**](#_heading=h.m14c8un5tfo6)

[**12. Visão de Dados 23**](#_heading=h.z3fx2ggfum5l)

[**13. Tamanho e Performance 25**](#_heading=h.lrsj90hdrltg)

[**14. Qualidade 25**](#_heading=h.b9aiam6fukv2)

[**15. Cronograma Macro 25**](#_heading=h.5hcv93cd3p29)

# Objetivo

Este documento tem como objetivo apresentar o projeto do E-commerce Esboço Comix, apresentando uma visão arquitetural desse sistema, como também documentar as necessidades de negócio e o produto que busca satisfazer esses requisitos.

## Escopo

Trata-se do desenvolvimento de um sistema Web E-commerce que atenda as necessidades da livraria de revistas em quadrinhos, Esboço Comix.

Aqui será descrito pontos significativos quanto à arquitetura e design do sistema, sua divisão em subsistemas e pacotes e sua divisão em classes e utilitários de classes.

## Referências

Para a construção deste documento foram utilizadas as seguintes referências:

* Reuniões informais realizadas com o grupo Esboço Comix.

Este documento influencia os seguintes documentos:

* Documento de Requisitos

# Necessidades de Negócio

Um sistema para a venda de revistas em quadrinhos, que permite funcionalidades como:

* Controle de estoque dos produtos
* Realização de pedidos de compra dos clientes
  + Possibilidade de uso de cartões de crédito, cupons promocionais e cupons de troca como forma de pagamento
* Possibilidade de troca e devolução dos produtos
* Gerenciamento de pedidos
  + Ter controle dos status de pedidos, pedidos aprovados, pedidos para entrega
* Análise do histórico de vendas da loja
* Recomendações personalizadas de produtos para cada cliente

# Objetivo do Projeto

Desenvolver um sistema web E-commerce capaz de:

* Armazenar informações em uma base de dados
* Controlar estoque dos pedidos
* Controlar pedidos dos clientes
* Ser executado em diversos navegadores
* Integrar com serviço de IA generativa para recomendação personalizada

# Declaração Preliminar de Escopo

Esta seção descreve, em alto nível, o escopo do projeto. Os requisitos serão melhor detalhados nos documentos de Requisitos e Dicionário WBS.

## Descrição

O produto do projeto se trata de um sistema web de E-commerce de revistas em quadrinhos. A partir dele, os clientes poderão consultar os produtos e realizar pedidos de compra. Com esse sistema, um usuário com perfil de admnistrador poderá fazer o controle dos pedidos e a análise do histórico de vendas. O objetivo é prover uma solução digital para a venda de revistas em quadrinhos da loja Esboço Comix.

## Produtos a serem entregues

Os seguintes itens são considerados produtos do projeto:

* Sistema Web E-commerce, implementado de acordo com a especificação feita na fase de análise, incluindo código fonte.
* Documentos de especificação do sistema, concebido na fase de elaboração
* Hospedagem do sistema em ambiente 24 x 7.

## Requisitos

Os requisitos estão descritos em alto nível. Eles serão detalhados num possível Documento de Requisitos.

### Requisitos Funcionais

O sistema deve ser capaz de efetuar o cadastro, consulta, alteração e exclusão de clientes.

O sistema deve permitir ao cliente realizar uma compra a partir de um repositório temporário (carrinho)

O sistema deve permitir que o cliente efetue um pedido de troca de um produto

O sistema deve permitir ao administrador gerenciar os pedidos dos clientes, alterando status dos pedidos

### Requisitos Não Funcionais

Utilizar linguagem Java.

Utilizar o banco de dados PostgreSQL.

A arquitetura da solução obedecer o design pattern MVC.

O sistema deve rodar nos seguintes browsers:

* IE
* Google Chrome
* Firefox
* Cronograma de Marcos Sumariado

### Regras de Negócio

Ao cadastrar um cliente, devem ser obrigatórios dados como nome e CPF.

Só deve ser permitida a compra do produto se houver quantidade disponível no estoque.

A forma de pagamento do pedido deve ser validada antes do pedido ser feito.

Cada entrada em estoque deve ser validada.

# Premissas

* O projeto será orientado pelo professor Rodrigo Rocha.
* O projeto será entregue até o final do primeiro semestre de 2025.
* Não haverão mudanças drásticas no escopo do projeto.
* Há recursos suficientes para entrega do projeto.

# Influência das Partes Interessadas

* **Nattan Silva de Souza:** gerente de projetos, responsável por liderar e supervisionar todo o planejamento, execução e entrega do sistema. Garante que os objetivos sejam atingidos dentro dos prazos e limites estabelecidos.
* **Rodrigo Rocha Silva:** atua como mentor técnico e acadêmico, proporcionando suporte estratégico e feedback contínuo para garantir o alinhamento do projeto aos objetivos propostos.
* **Loja Esboço Comix:** Representada pelos proprietários ou gestores, define os requisitos e valida as funcionalidades implementadas para assegurar que o sistema atende às suas necessidades comerciais e operacionais.
* **Clientes da Loja:** Usuários finais do sistema de e-commerce. Impactados diretamente pela qualidade e usabilidade da plataforma, seus feedbacks serão cruciais para o aperfeiçoamento contínuo.

# Representação Arquitetural

O sistema será desenvolvido tendo como base a arquitetura representada na figura 1.

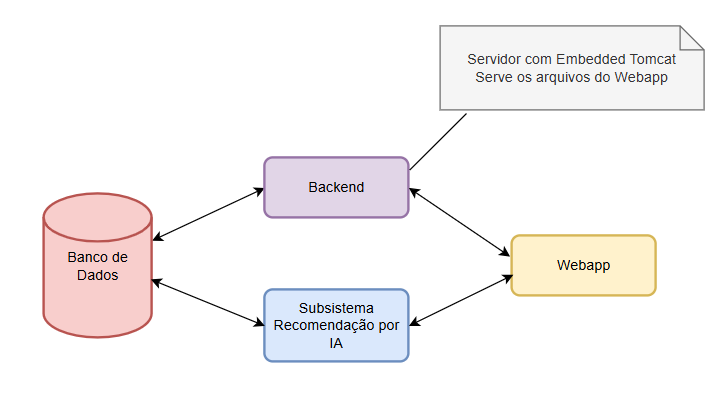


Figura 1 - Modelo Arquitetural

O **Webapp** faz o papel da camada de apresentação. É ele que fará a interação com usuário, por meio de um navegador da Web.

Interage com o **Backend** por meio de requisições por API, que o Backend responde, e interage com o **Subsistema de Recomendação por IA** para ter acesso à funcionalidade de recomendações personalizadas.

Já o **Backend** é o principal responsável por aplicar as regras de negócio do sistema. Aqui são encontradas as principais regras, validações e medidas de segurança do sistema.

Ele se comunica com o **Webapp** respondendo suas requisições API. Vale comentar que o Backend também é responsável por disponibilizar os arquivos de Webapp, por meio do servidor Tomcat.

Ele interage com o **Banco de Dados** quando é necessário lidar com operações de persistência dos dados do sistema.

O **Subsistema de Recomendação por IA** é responsável por fazer uso da IA generativa para gerar recomendações personalizadas para os clientes, com base no histórico de venda e suas preferências. Depende do **Banco de Dados** para treinar o modelo e dar as recomendações para o cliente usando a base de dados do sistema.

O **Banco de Dados** é responsável por fazer a persistência dos dados (como cadastro de clientes, informações dos produtos, pedidos).

## Restrições Arquiteturais

As tecnologias usadas para o sistema serão:

* Java 21 para o Backend.
* JBDC para realizar a conexão entre o Backend e o Banco de Dados.
* HTML, CSS, JS e Bootstrap para a criação do Webapp.
* PostgreSQL como o Banco de dados da aplicação.
* Servidor Tomcat.
* Python para a criação do Subsistema de Recomendação por IA.
* Gemini como modelo de IA generativa.

## Objetivos e Restrições Arquiteturais

Alguns requisitos registrados que impactam diretamente a arquitetura do sistema são:

* O uso de recomendações personalizadas por IA, que implica no uso e integração com um modelo de IA generativa.

Considerando premissas definidas para os Sistemas XXXX pode-se citar as seguintes restrições:

* Utilização da Linguagem Java
* Considerar a utilização de software Livre, quando possível
* O Sistema de Gerenciamento de Banco de Dados a ser considerado em implementações de âmbito corporativo será o SGBD PostgreSQL.

# Visão de Use Case

Esta seção apresenta os Casos de Uso arquiteturalmente significativos, que foram selecionados considerando-se o pacote do Modelo de Casos de Uso que representa o E-commerce.

A classificação dos casos de uso, em termos de significância, foi realizada com base na observação de pelo menos um dos seguintes critérios:

* Casos de uso que estendem outros Casos de Uso
* Casos de Uso que são incluídos em outros Casos de Uso e
* Casos de uso que acessam sistemas externos

## Diagrama de Casos de Uso Realizar Pedido

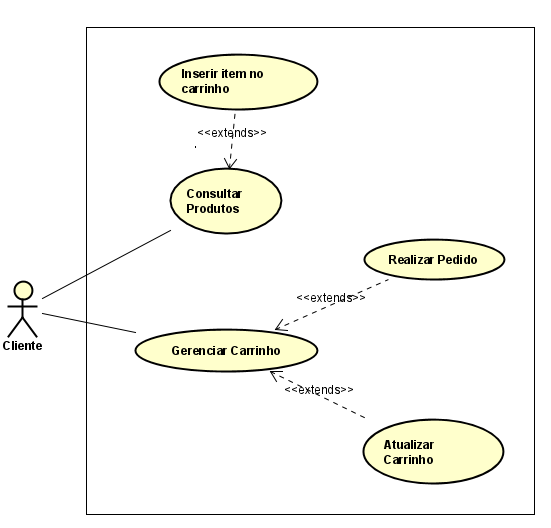


Figura 2 - Diagrama de Caso de Uso de Realizar Pedido

## Diagrama de Caso de Uso Gerenciar Pedidos

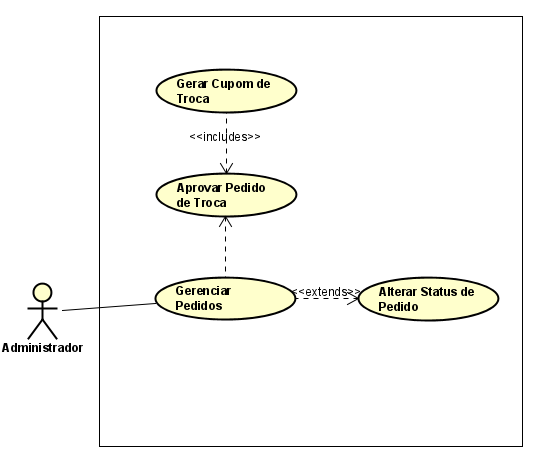


Figura 3 - Diagrama de Caso de Uso de Gerenciar Pedidos

## Diagrama de Caso de Uso Gerenciar Estoque

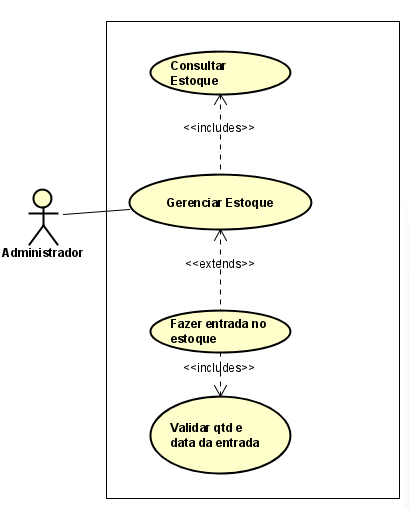


Figura 4 - Diagrama de Caso de Uso de Gerenciar Estoque

## Descrição dos Casos de Uso Arquiteturalmente Significativos

**Consultar Produtos e Gerenciar Carrinho**

Esses dois casos de uso descrevem como o cliente pode consultar produtos e inserir a quantidade de um produto no carrinho. Do carrinho, ele pode alterar a quantidade dos itens ou excluir o item do carrinho. A partir disso, tem a opção de realizar o pedido do carrinho.

**Gerenciar Pedidos**

Esse caso de uso descreve como o administrador pode gerenciar os pedidos. Pode alterar status de pedidos e pode aprovar pedidos de troca, onde o sistema automaticamente gerará um cupom de troca.

**Gerenciar Estoque**

Aqui o administrador pode consultar o estoque dos produtos da loja. Ele tem a opção de fazer uma entrada no estoque de algum produto específico. Após isso, a quantidade inserida no estoque e a data da entrada deverão ser validadas.

# Visão de Lógica

Esta visão apresenta elementos de design significativos do ponto de vista da arquitetura, descrevendo a organização do E-commerce Esboço Comix em pacotes, bem como a organização desses pacotes em camadas.

O Diagrama com as camadas do sistema é ilustrado na figura 5.

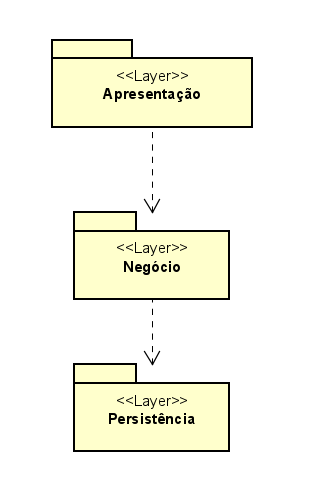


Figura 5 – Diagrama de camadas

**Apresentação**: Contém os arquivos responsáveis pelas interfaces gráficas com os usuários (GUI), desenvolvidas utilizando HTML, CSS e JavaScript. A partir daqui dessas interfaces, os usuários conseguem interagir com o sistema.

**Negócio**: Contém classes que controlam a execução das funcionalidades do E-commerce. Também inclui o Subsistema de Recomendação por IA.

**Persistência**: Contém classes responsáveis por persistir as entidades de modelo. Inclui, por exemplo, classes que lêem e gravam dados no Banco de Dados.

A Figura 6 ilustra o diagrama de camadas com as tecnologias utilizadas no desenvolvimento, já descritas na figura 5.

Vale ressaltar que nenhuma página é renderizada pela camada de negócio. Toda a página é renderizada na camada de apresentação por meio dos arquivos estáticos de HTML, CSS e JS e por requisições AJAX pelo Javascript.

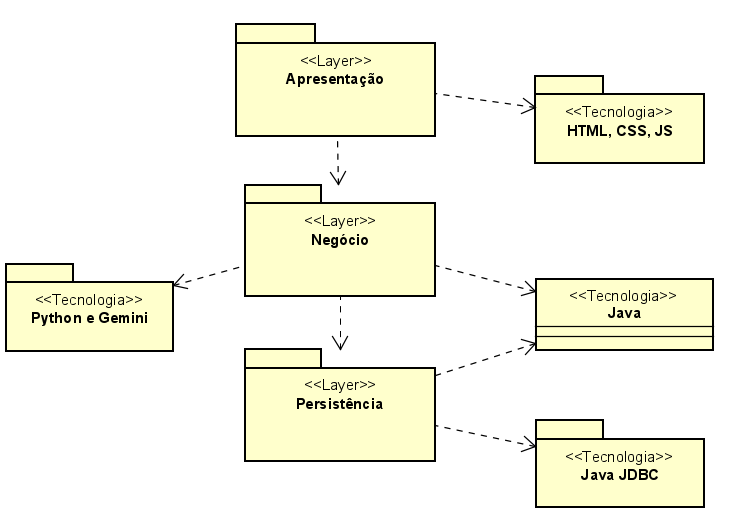


Figura 6 - Camadas com as dependências de tecnologia

## 

## Camada de Apresentação

Nesta camada, temos os pacotes CSS e JS, que são recursos que podem ser usados em qualquer página do site. Dentro do pacote JS, há o pacote API, que armazena os scripts de chamadas AJAX para a camada de negócio. Fora isso, há o pacote “Páginas”, que contém as páginas do E-commerce.

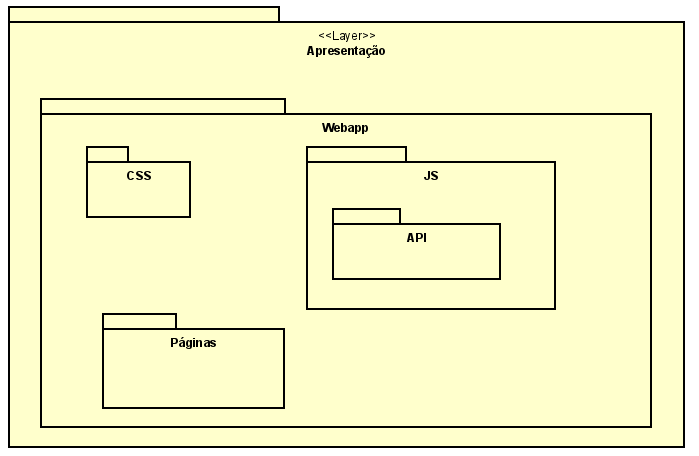


Figura 7 - Camada de Apresentação

## 

## Camada de Negócio

Nesta camada, temos o Backend e o Subsistema de Recomendação por IA.

Dentro do Backend, temos como principais pacotes:

* Controller: responsável por lidar com as chamadas de API do sistema. Qualquer regra ou lógica adicional, repassa para as classes do pacote Service.
* Model: contém as classes que representam o modelo de dados do E-commerce.
  + Ex.: Cliente, cartão de crédito, pedido.
  + Também abriga classes que objetos do negócio. Ex.: Carrinho.
* Service: lida diretamente com os serviços prestados pelo Backend. Aplica regras, validações, medidas de segurança. Conversa com a camada de persistência caso necessário

A imagem 8 ilustra os pacotes descritos.

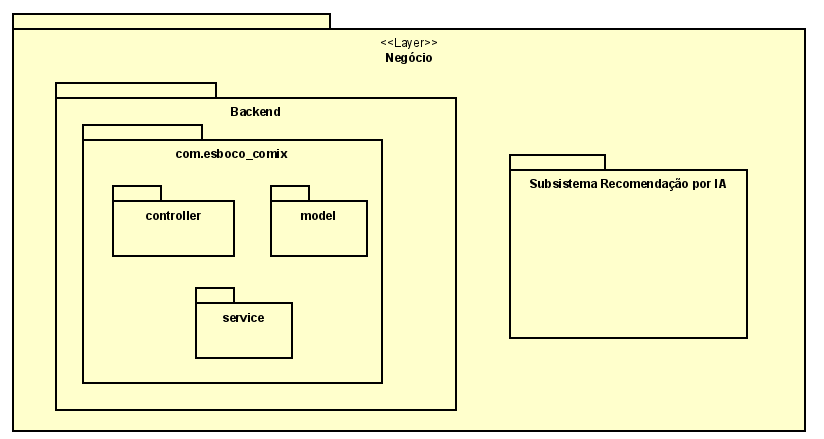


Figura 8 - Camada de Negócios

* + 1. **Pacote Controller e Pacote Service**

A figura 9 ilustra as principais classes de controle e de Service.

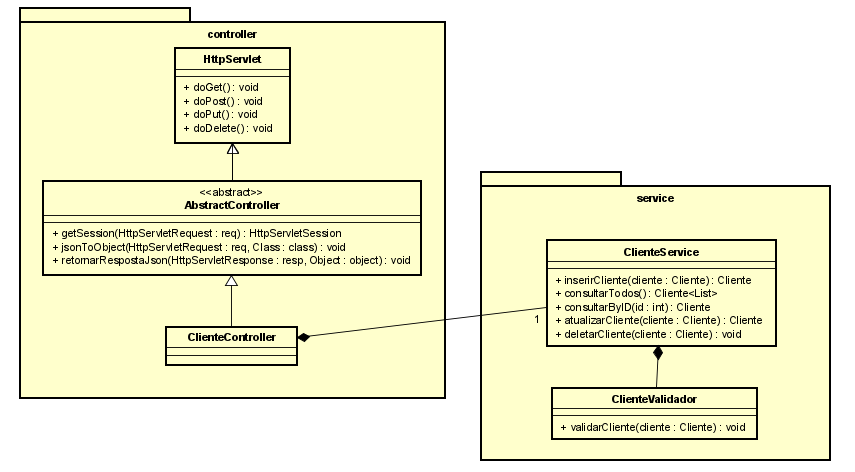
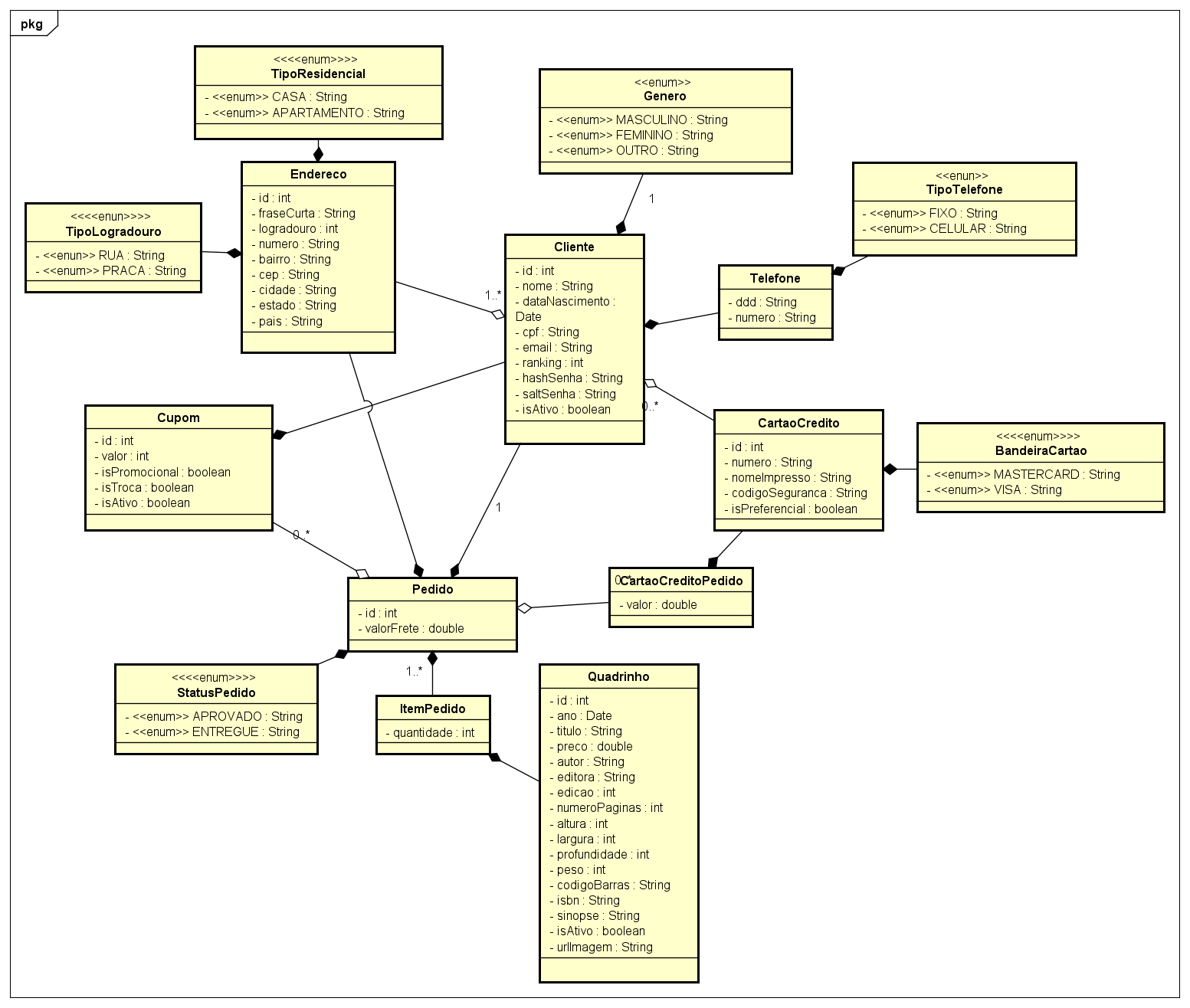


Figura 9 - Classes de controle e service

* + 1. **Pacote Model**

A figura 10 ilustra as principais classes do modelo.

Figura 10 - Classes do Modelo

## Camada de Persistência

Nesta camada temos o pacote dao que contém as classes e interfaces responsáveis por persistir as informações do E-commerce no Banco de Dados. É utilizada a tecnologia Java JDBC para fazer a conexão com o banco.

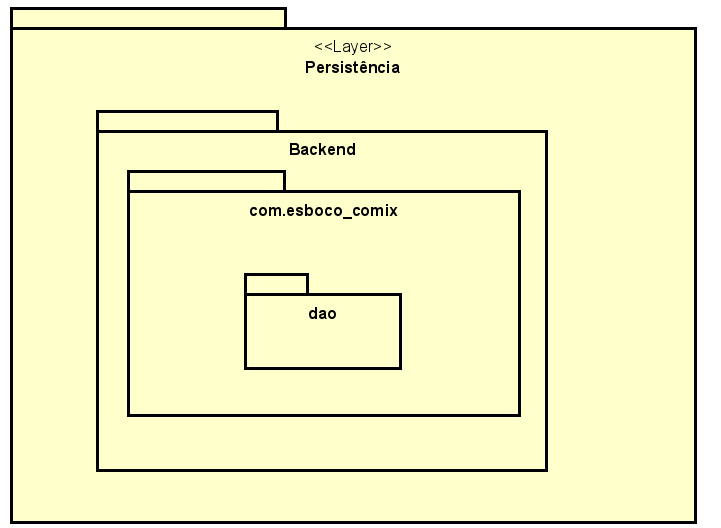


Figura 11 - Camada de Persistência

## Realização dos Casos de Uso Significativos

### Realizar Pedido

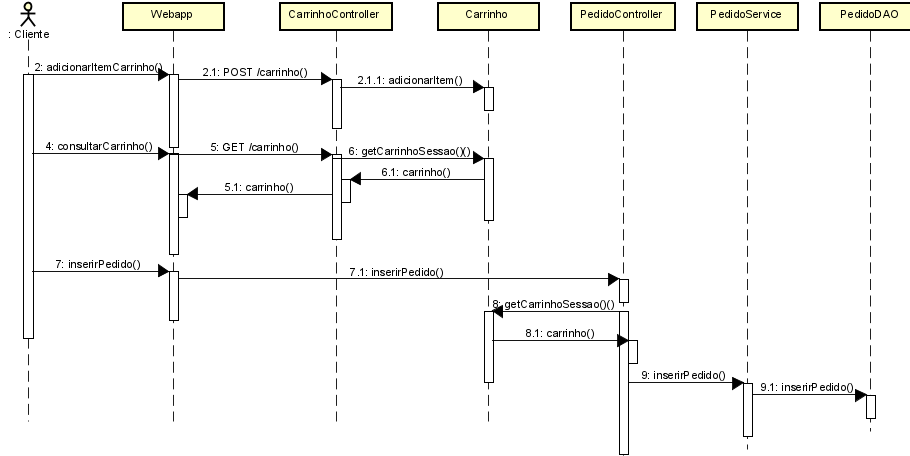


Figura 12 - Diagrama de Atividade de Realizar Pedido

### Gerenciar Pedidos

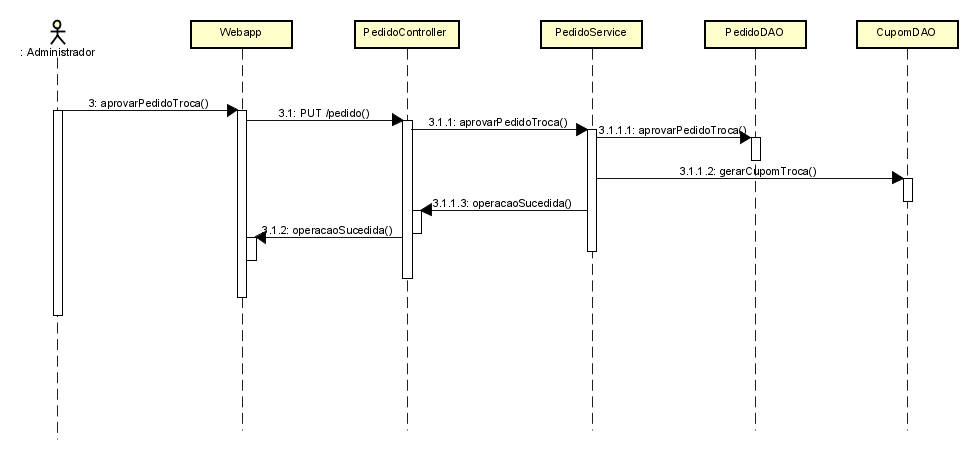


Figura 13 - Diagrama de Atividade de Gerenciar Pedidos

### Gerenciar Estoque

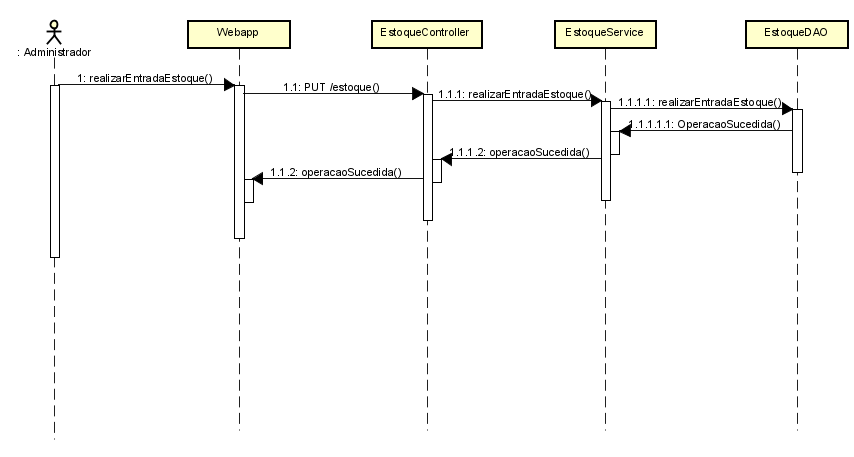


Figura 14 - Diagrama de Atividade de Gerenciar Estoque

# Visão de Implantação

Esta seção descreve as configurações da rede física (hardware) na qual o sistema de e-commerce será implantado e executado. Trata-se de uma visão do Modelo de Implantação que, para a configuração em questão, indica os nós físicos (computadores, servidores, CPUs), que executarão o Backend, o Banco de Dados PostgreSQL e o Subsistema de Recomendação por IA, além das respectivas interconexões (barramento, LAN, etc).

A Figura 15 ilustra o modelo de implantação para o sistema de e-commerce.

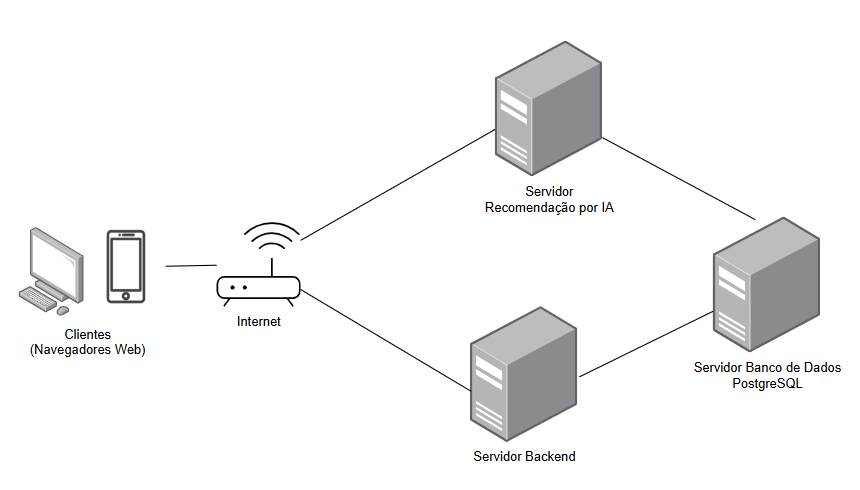


Figura 15 - Visão de Implantação do Esboço Comix

Na Figura 15, observa-se os seguintes nós físicos:

1. **Navegadores Web (Web Client Application - WCA)**: Representam os dispositivos de acesso dos usuários (computadores, tablets, smartphones) que executam aplicativos de interface gráfica via navegador, desenvolvidos em HTML, CSS e JavaScript.
2. **Backend**: Componente responsável por processar a lógica de negócios e gerenciar as requisições enviadas pelos navegadores web. Implementado em Java, utilizando servlets e comunicação via HTTP.
3. **Servidor DB**: Nó que contém o Banco de Dados Central do sistema, executado em PostgreSQL, armazenando informações como dados de usuários, produtos e pedidos.
4. **Subsistema de Recomendação por IA**: Responsável por executar os algoritmos de inteligência artificial que fornecem recomendações personalizadas. Este subsistema pode ser executado em um servidor dedicado ou compartilhado com o Backend.

# Visão de Implementação

Esta visão descreve a estrutura geral de implementação do sistema, apresentando a decomposição do software em camadas de implementação.

A estrutura geral de implementação do **Esboço Comix** segue diretamente a organização definida na **Visão Lógica**, mas com adaptações para atender às necessidades práticas da implementação. Essas adaptações incluem a criação de pacotes adicionais no Backend da camada de negócio, como:

* **DTO (Data Transfer Object):** Para gerenciar objetos usados na transferência de dados entre diferentes camadas do sistema.
* **Config:** Para centralizar configurações específicas do sistema, como configuração de banco de dados ou propriedades do servidor.
* **Utils:** Para agrupar funções e classes auxiliares que são reutilizadas em diferentes partes do sistema.

Esses pacotes adicionam modularidade e facilitam a manutenção e a escalabilidade do sistema, indo além da abstração apresentada na Visão Lógica. Apesar dessas mudanças, a estrutura geral mantém forte alinhamento com os diagramas de camadas e pacotes definidos anteriormente.

# Visão de Dados

Para realizar a persistência dos dados nesse sistema, é feito o uso do SGBD Relacional PostgreSQL. O Backend interage com o banco por meio da conexão JDBC, portanto, ORM não é utilizado nesse sistema. As figuras a seguir apresentam o modelo físico e modelo lógico da aplicação:

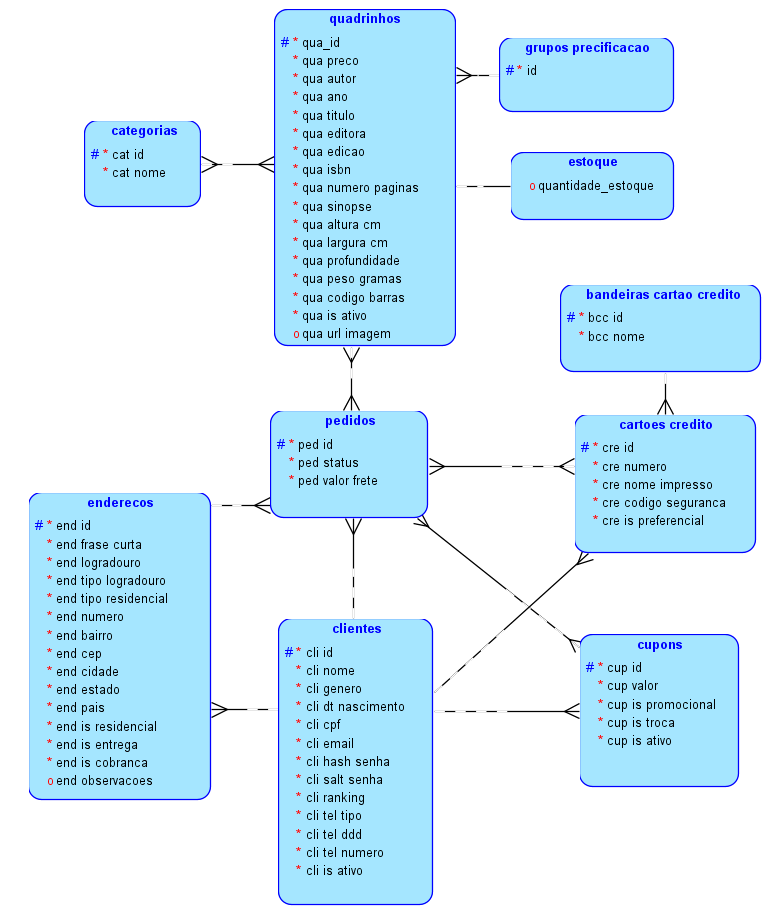


Figura 16 - Modelo Lógico

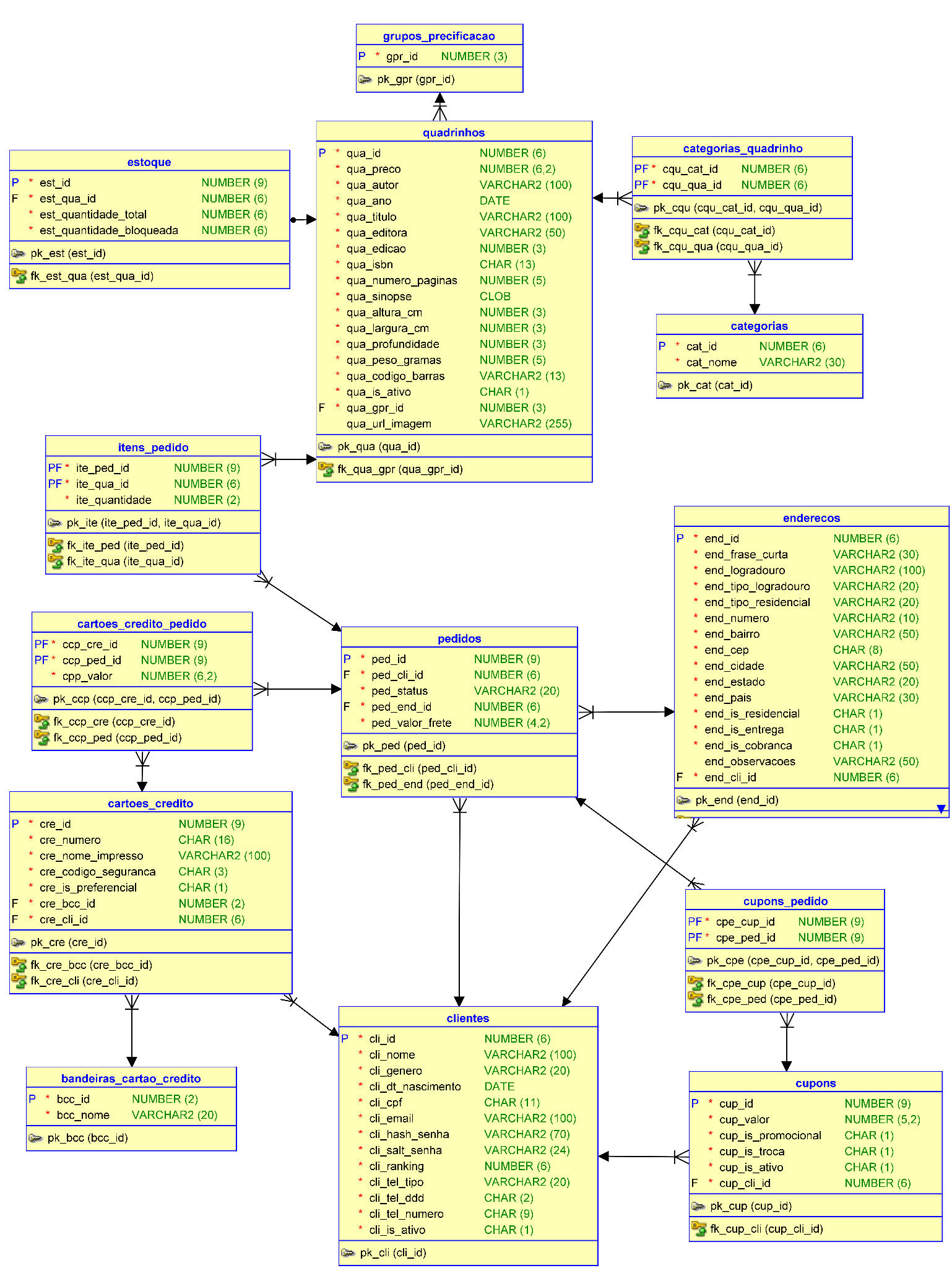


Figura 17 - Modelo Relacional

# Tamanho e Performance

O sistema da loja **Esboço Comix** será usado para gerenciar o e-commerce de vendas de revistas em quadrinhos, lidando com uma base crescente de dados e operações. Seus servidores poderão enfrentar **picos de utilização** em períodos específicos, como durante promoções, lançamentos de edições especiais ou eventos de grande demanda, quando o número de acessos simultâneos ao site pode aumentar consideravelmente.

Estima-se que o número de usuários e a carga de utilização em períodos de pico serão elevados, exigindo que o sistema seja projetado para garantir desempenho eficiente mesmo sob essas condições. Informações mais detalhadas sobre os requisitos relacionados ao tamanho e desempenho do sistema da loja **Esboço Comix** estarão disponíveis no documento de requisitos não funcionais.

# Qualidade

O sistema **Esboço Comix** será responsável pelas vendas de revistas em quadrinhos, lidando com transações financeiras, dados de clientes e histórico de vendas. Pela presença de dados sensíveis no sistema, Falhas no sistema podem gerar prejuízos financeiros e comprometer a credibilidade da marca, portanto a **confiabilidade** e a **robustez** devem ser prioridades no design.

Além disso, a interface do sistema disponível na Internet aumenta o risco de ataques maliciosos, como roubo ou corrupção de dados. Por isso, é essencial implementar uma **infraestrutura de segurança robusta**, incluindo proteção contra acessos não autorizados e criptografia de informações sensíveis.

Mais detalhes sobre os requisitos de qualidade estarão no documento de requisitos não funcionais.

# Cronograma Macro

| **Resultado** |  |
| --- | --- |
| Plano Preliminar | Semana 2 |
| Plano Fase 1 | Semana 3 |
| Especificação Fase 1 | Semana 5 |
| Piloto Fase 1 | Semana 11 |
| Solução Testada Fase 1 | Semana 12 |
| Plano Fase 2 | Semana 13 |
| Especificação Fase 2 | Semana 14 |
| Piloto Fase 2 | Semana 20 |
| Solução Testada Fase 2 | Semana 21 |

Obs: Os prazos apresentados são uma estimativa inicial considerando as informações disponíveis nesta etapa do projeto. Um cronograma detalhado será elaborado na fase de planejamento e, eventualmente, estes prazos podem ser modificados.

## Referências

**Unified Modeling Language (UML):** <http://www.omg.org/technology/documents/formal/uml.htm>

**RUP. Rational Unified Process:** <https://www.ibm.com/docs/en/rup>

**Oracle SQL Developer Data Modeler:** <https://www.oracle.com/database/sqldeveloper/technologies/sql-data-modeler/>

**Astah:** <https://astah.net/>

**Draw.io / Diagrams.net**: <https://app.diagrams.net/>