

# DOCUMENTO DE VISÃO DE PROJETO



## MIRROR FASHION – e-commerce de Roupas Femininas



Faculdade de Tecnologia de Mogi das Cruzes

Tecnologia em Análise e Desenvolvimento de Sistemas | Tecnologia em Agronegócio | Tecnologia em Recursos Humanos



## Histórico de Versões

Data	Versão	Descrição	Autor	Revisor
24/04/2022	1.0	Modelagem e desenvolvimento	Lorena Oliveira	-
24/04/2022	1.1	Primeira versão do documento	Lorena Oliveira	-
17/05/2022	1.2	Revisão do documento com informações inseridas	Lorena Oliveira	-
12/06/2022	1.3	Segunda versão do documento	Lorena Oliveira	-

**Cliente** Mirror Comércio de Roupas LTDA  
**Documento** Documento de Visão de Projeto: *Mirror Fashion*  
**Data** 12 de Junho de 2022  
**Autor** Lorena Oliveira  
lorena.oliveira01@fatec.sp.gov.br

## Página de Assinaturas

Revisado e  
Aprovado por:

12/06/22



## Índice

1. OBJETIVO .....	4
1.1. ESCOPO .....	4
1.2. REFERÊNCIAS.....	4
2. NECESSIDADES DE NEGÓCIO .....	4
3. OBJETIVO DO PROJETO .....	5
4. DECLARAÇÃO PRELIMINAR DE ESCOPO.....	5
4.1. DESCRIÇÃO .....	5
4.2. PRODUTOS A SEREM ENTREGUES.....	5
4.3. REQUISITOS .....	5
4.3.1. REQUISITOS FUNCIONAIS .....	5
4.3.2. REQUISITOS NÃO FUNCIONAIS .....	8
4.3.3. REGRAS DE NEGÓCIO.....	9
4.4. PREMISSAS .....	11
5. INFLUÊNCIA DAS PARTES INTERESSADAS.....	11
6. REPRESENTAÇÃO ARQUITETURAL .....	12
6.1. RESTRIÇÕES ARQUITETURAIS.....	13
6.1.1. OBJETIVOS E RESTRIÇÕES ARQUITETURAIS .....	13
7. VISÃO DE USE CASE .....	14
7.1. DIAGRAMA DE CASO DE USO GERAL DO SISTEMA .....	15
7.2. DESCRIÇÃO DOS CASOS DE USO ARQUITETURALMENTE SIGNIFICATIVOS .....	16
7.3. VISÃO DE LÓGICA.....	17
7.3.1. CAMADA DE APRESENTAÇÃO .....	18
7.3.2. CAMADA DE NEGÓCIO .....	19
7.3.3. PACOTE CONTROLLER.....	19
7.3.4. PACOTE MODEL.....	20
7.3.5. CAMADA DE PERSISTÊNCIA .....	21
7.3.6. REALIZAÇÃO DOS CASOS DE USO SIGNIFICATIVOS .....	23
7.4. VISÃO DE IMPLANTAÇÃO .....	26
7.5. VISÃO DE IMPLEMENTAÇÃO .....	26
7.6. VISÃO DE DADOS.....	26
7.7. TAMANHO E PERFORMANCE .....	31
7.8. QUALIDADE .....	31
7.9. CRONOGRAMA MACRO.....	31
7.10. REFERÊNCIAS .....	32



## 1. OBJETIVO

Este documento tem o objetivo de alinhar as expectativas dos interessados para formalizar o início do projeto e apresentar uma visão arquitetural do sistema da linha do e-commerce de Roupas Femininas, Mirror Fashion. O intuito é salientar diferentes aspectos deste produto, obtidos a partir de decisões arquiteturais realizadas no âmbito do sistema a ser desenvolvido.

Este documento trata principalmente da documentação das necessidades de negócio, da justificativa do projeto, do entendimento atual das necessidades do cliente e descreve resumidamente o novo produto, serviço ou resultado que deve satisfazer esses requisitos.

### 1.1. ESCOPO

O escopo deste documento trata do desenvolvimento de um módulo que atenda todas as necessidades de uma loja de roupas femininas abrangendo qualquer estilo e gênero, sendo que este módulo é uma parte de um sistema maior.

O escopo deste documento é documentar as partes significativas do ponto de vista da arquitetura do modelo de design, como sua divisão em subsistemas e pacotes. Além disso, mostra sua divisão em classes e utilitários de classe.

### 1.2. REFERÊNCIAS

Para a construção deste documento foram utilizadas as seguintes referências:

- Reuniões informais entre o grupo LSO TECH e MIRROR LTDA.
- Documentos elaborados pela MIRROR LTDA intitulado como:
  - DOCUMENTO DE REQUISITOS.doc

Este documento influencia os seguintes documentos:

- Documento de Requisitos
- Estimativa
- Proposta Comercial

## 2. NECESSIDADES DE NEGÓCIO

Um sistema informatizado para vendas de roupas possibilita ter total controle de todos os produtos que ela possui. Com o sistema, também é possível controlar todas as vendas e trocas que são efetuadas. Além de tais funcionalidades, o sistema também poderá auxiliar no controle de devolução de produtos que clientes da Mirror Fashion venham realizar, para que não ocorram perda de lucro e entre outros problemas.

Além disto, deve ser informatizado que a partir dos dados armazenados referentes as operações realizadas pelos clientes gere relatórios que serão utilizados para o controle interno, para fornecer informações aos administradores da empresa na qual está localizada.



### 3. OBJETIVO DO PROJETO

Desenvolver uma plataforma para soluções web capaz de:

- armazenar informações em uma base de dados;
- utilizar o protocolo HTTP;
- ser executado em qualquer navegador.

Desenvolver uma solução para um e-commerce, que possibilite:

- Controlar o estoque de produtos presentes na loja virtual e entrada de novos produtos.
- Gerenciamento das operações realizadas na loja: vendas, devolução de produtos, geração de cupons.
- Maximizando os lucros da empresa e facilitando o intermédio de pagamento online, permitir através de registros obtidos das transações o controle das operações realizadas e uma projeção estatísticas através de gráficos do lucro obtido por um determinado período.

### 4. DECLARAÇÃO PRELIMINAR DE ESCOPO

Esta seção descreve, em alto nível, o escopo do projeto.

Os requisitos serão detalhados melhor nos documentos de Requisitos e Proposta Comercial.

#### 4.1. DESCRIÇÃO

A **Mirror Fashion** é uma aplicação online que funciona como um e-commerce de vendas de roupas femininas. Na aplicação os clientes poderão ver os produtos à venda e fazer pedidos, além de contar com a possibilidade de troca e devolução caso necessitarem. O administrador terá controle sobre o fluxo do negócio, ele irá autorizar vendas e trocas, gerenciar estoque, gerenciar clientes, emitir relatórios das transações, consultar análises através de gráficos.

#### 4.2. PRODUTOS A SEREM ENTREGUES

Os seguintes itens são considerados partes fundamentais do projeto, na sua etapa 1.

- Sistema do módulo de cliente, etapa 1, implementado de acordo com a especificação feita na fase de análise. (código objeto e código fonte).
- Documentos de especificação do sistema, concebido na fase de elaboração.
- Hospedagem do sistema em ambiente 24 x 7.

#### 4.3. REQUISITOS

##### 4.3.1. REQUISITOS FUNCIONAIS

Grupo: Cadastro de Roupas		
RF0011	Cadastrar roupa	O sistema deve manter um cadastro único para cada peça de roupa.



RF0012	Inativar cadastro de roupa	O sistema deve possibilitar que as peças de roupa sejam inativadas.
RF0013	Inativar roupa de forma automática	O sistema deve inativar as peças de roupa sem estoque e que não possuem venda com valor inferior a parâmetro predefinido no sistema.
RF0014	Alterar cadastro de roupa	O sistema deve possibilitar a alteração de dados cadastrais para as peças de roupas.
RF0015	Consulta de roupa	O sistema deve possibilitar que uma peça de roupa seja consultada com base em um filtro definido pelo usuário. Todos os campos utilizados para identificação da peça de roupa podem ser utilizados como filtro, tanto de forma combinada como de forma isolada.
RF0016	Ativar cadastro de roupa	Deve ser possível ativar o cadastro de uma peça de roupa.
<b>Grupo: Cadastro de Clientes</b>		
RF0021	Cadastrar cliente	O sistema deve possibilitar o cadastro de clientes.
RF0022	Alterar cliente	O sistema deve possibilitar a alteração de dados cadastrais de clientes.
RF0023	Inativar cadastro de cliente	O sistema deve possibilitar que clientes sejam inativados.
RF0024	Consulta de clientes	O sistema deve possibilitar que um cliente seja consultado com base em um filtro definido pelo usuário. Todos os campos utilizados para identificação do cliente podem ser utilizados como filtro, tanto de forma combinada como de forma isolada.
RF0025	Consulta de transações	O sistema deve disponibilizar no cadastro de clientes a consulta de todas as transações já realizadas por ele.
RF0026	Cadastro de endereços de entrega	Deve ser possível associar diversos endereços de entrega ao cadastro de um cliente. Cada cadastro de endereço deve ser identificado com um nome composto de uma frase curta.
RF0027	Cadastro de cartões de crédito	Deve ser possível associar diversos cartões de crédito ao cadastro de um cliente. Deve haver um cartão de crédito configurado como preferencial.
RF0028	Alteração apenas de senha	O sistema deve possibilitar que a senha do usuário seja alterada sem que seja necessária a alteração de todos os dados cadastrais.
<b>Grupo: Gerenciar Vendas Eletrônicas</b>		
RF0031	Gerenciar carrinho de compra	O sistema deve permitir que produtos sejam colocados em um repositório temporário para futura compra (carrinho de compra). Deve ser possível adicionar, alterar e excluir itens de compra no carrinho. Também deve ser possível visualizar os itens no carrinho.
RF0032	Definir quantidade de itens no para o carrinho	Deve ser possível editar a quantidade de cada item ao adicionar um produto no carrinho. Também deve ser possível editar a quantidade de itens de um carrinho na visualização dos itens já adicionados.
RF0033	Realizar compra	Deve ser possível a partir de um carrinho de compra realizar uma compra.



RF0034	Calcular frete	O sistema deve calcular o frete da compra com base nos itens selecionados e o endereço apontado pelo cliente.
RF0035	Selecionar endereço de entrega	O cliente pode selecionar qualquer endereço de entrega previamente cadastrado em seu perfil ou um novo endereço de entrega pode ser cadastrado. Caso um novo endereço de entrega seja inserido, deve-se dar a possibilidade que ele seja incorporado ao perfil do cliente.
RF0036	Selecionar forma de pagamento	<p>O cliente pode selecionar qualquer cartão de crédito previamente cadastrado em seu perfil ou um novo cartão de crédito pode ser cadastrado. Caso um novo cartão de crédito seja cadastrado, deve-se dar a possibilidade que ele seja incorporado ao perfil do cliente.</p> <p>O cliente também poderá utilizar um cupom de troca ou um cupom promocional válido.</p> <p>Deve-se possibilitar que o pagamento seja feito utilizando tanto cupons de troca, promocionais e cartão de crédito.</p>
RF0037	Finalizar Compra	Uma compra deve ser finalizada após a seleção da forma de pagamento e endereço de entrega. Após a finalização o status da compra deve ser EM PROCESSAMENTO.
RF0038	Despachar produtos para entrega	O sistema deve possibilitar que um usuário com perfil de administrador selecione vendas já aprovadas para serem entregues. Assim o status deve ficar EM TRÂNSITO.
RF0039	Produtos entregues	O sistema deve possibilitar que um usuário com perfil de administrador confirme entrega de uma compra. Assim o status deve ficar ENTREGUE.
RF0040	Solicitar troca	O sistema deve possibilitar que um item de uma compra seja trocado por um cliente através da visualização de pedidos dele.
RF0041	Autorizar trocas	O sistema deverá possibilitar que o administrador autorize pedidos ou compra com status EM TROCA. Assim o pedido passa ficar com status TROCA AUTORIZADA.
RF0042	Visualização de trocas	O sistema deverá possibilitar que o administrador visualize todos pedidos de troca ou compra com status EM TROCA.
RF0043	Confirmar recebimento de itens para troca	<p>O sistema deverá possibilitar que o administrador confirme o recebimento de pedidos de troca ou compra com status EM TROCA.</p> <p>Nesta confirmação o administrador deverá informar se os itens trocados deverão retornar ao estoque. Em caso positivo deve-se dar entrada no estoque dos respectivos itens.</p>
RF0044	Gerar cupom de troca após recebimento de itens	O sistema deverá gerar um cupom de troca quando o administrador informar que os itens a serem trocados chegaram. Este cupom deverá ser disponibilizado para o cliente para ser utilizado em futuras compras.



Grupo: Controle de estoque		
RF0051	Realizar entrada em estoque	O sistema deve permitir que seja possível realizar entrada de itens de peça de roupa em estoque.  No registro de cada item, deve ser indicado a peça de roupa já previamente cadastrada e a quantidade de itens da peça de roupa.
RF0052	Calcular valor de venda	O sistema deve calcular o valor de venda com base no valor de custo e o grupo de precificação. Sendo que o valor de venda será o valor de compra mais o percentual definido no grupo de precificação relacionado a peça de roupa.
RF0053	Dar baixa em estoque	Para cada venda realizada deve-se dar baixa no estoque do total de itens vendidos.
RF0054	Realizar reentrada em estoque	O sistema deve realizar a reentrada de um item em estoque a partir da troca de um produto.

#### 4.3.2. REQUISITOS NÃO FUNCIONAIS

Grupo: Geral		
RNF0011	Tempo de resposta para consultas	Toda consulta de usuário deve ter resposta em no máximo 1 segundo.
RNF0012	Log de transação	Para toda operação de escrita (Inserção ou Alteração) deve ser registrado data, hora, usuário responsável além de manter os dados alterados.
Grupo: Cadastro de Roupas		
RNF0021	Código de roupa	Toda peça de roupa cadastrada deve receber um código único no sistema.
RNF0013	Cadastro de domínios	Deve haver um script de implantação do sistema que insere todos os registros de tabelas de domínio necessárias por ex: grupo de precificação, fornecedor etc.
Grupo: Cadastro de Clientes		
RNF0031	Senha forte	A senha cadastrada pelo usuário deve ser composta de pelo menos 8 caracteres, ter letras maiúsculas e minúsculas além de conter caracteres especiais.
RNF0032	Confirmação de senha	O usuário obrigatoriamente deve digitar duas vezes a mesma senha no momento do registro dela.
RNF0033	Senha criptografada	A senha deve ser criptografada
RNF0034	Alteração apenas de endereços	O sistema deve possibilitar que endereços de entrega ou cobrança possam ser alterados ou adicionados de forma simples sem a necessidade da edição dos demais dados cadastrais.
RNF0035	Código de cliente	Todo cliente cadastrado deve receber um código único no sistema.





### 4.3.3. REGRAS DE NEGÓCIO

Grupo: Cadastro de Roupas		
RN0011	Dados obrigatórios para o cadastro de uma roupa	Para toda peça de roupa cadastrada é obrigatório o cadastro dos seguintes dados: nome, descrição, categoria, valor de venda, valor de compra, quantidade, grupo de precificação e observação.
RN0012	Associação com categorias	Uma peça de roupa pode estar associada com mais de uma categoria.
RN0013	Definindo valor de venda	Toda peça de roupa após cadastrada deverá ser associada a um grupo de precificação onde o valor deverá ter como base a margem de lucro parametrizado para o grupo definido no cadastro da peça de roupa.
RN0014	Validar margem de lucro	Uma peça de roupa somente pode ter seu valor alterado se estiver dentro da margem de lucro definida pelo critério de grupo de precificação. Para uma peça de roupa ter seu valor alterado para baixo da margem de lucro definida pelo grupo de precificação é necessária uma autorização de um gerente de vendas.
RN0015	Associar motivo de inativação	Toda peça de roupa que for inativado manualmente deve ter uma justificativa e uma categoria de inativação associada.
RN0017	Associar motivo de ativação	Toda peça de roupa que for ativado deve ter uma justificativa e uma categoria de ativação associada.
Grupo: Cadastro de Clientes		
RN0021	Cadastro de endereço de cobrança	Para todo cliente cadastrado é obrigatório o registro de ao menos um endereço de cobrança.
RN0022	Cadastro de endereço de entrega	Para todo cliente cadastrado é obrigatório o registro de ao menos um endereço de entrega.
RN0023	Composição do registro de endereços	Todo cadastro de endereços associados a clientes deve ser composto dos seguintes dados: Tipo de residência (Casa, Apartamento etc.), Tipo Logradouro, Logradouro, Número, Bairro, CEP, Cidade, Estado e País. Todos os campos anteriores são de preenchimento obrigatório. Opcionalmente pode ser preenchido um campo observações.
RN0024	Composição do registro de cartões de crédito	Todo cartão de crédito associado a um cliente deverá ser composto pelos seguintes campos: N° do Cartão, Nome impresso no Cartão, Bandeira do Cartão e Código de Segurança.
RN0025	Bandeiras permitidas para registro de cartões de crédito	Todo cartão de crédito associado a um cliente deverá ser de alguma bandeira registrada no sistema.
RN0026	Dados obrigatórios para o cadastro de um cliente	Para todo cliente cadastrado é obrigatório o cadastro dos seguintes dados: Gênero, Nome, Data de Nascimento, CPF, Telefone (deve ser composto pelo tipo, DDD e número), e-mail, senha, endereço residencial.
RN0027	Ranking de cliente	O cliente deve receber um ranking numérico com base no seu perfil de compra
Grupo: Gerenciar Vendas Eletrônicas		



RN0032	Validar estoque para compra	Ao solicitar a compra de itens que estejam em um carrinho deve-se garantir que tais itens ainda permanecem disponíveis em estoque.
RN0033	Uso de cupom promocional para pagamento	Apenas um cupom promocional pode ser utilizado por compra.
RN0034	Uso de diversos cartões de crédito	Uma compra pode ser paga utilizando mais de um cartão de crédito, porém o valor mínimo para ser pago com cada cartão deve ser R\$ 10,00.
RN0035	Uso de cupons junto a cartão de crédito	<p>Ao realizar pagamento utilizando cupons e cartões em conjunto, deve-se sempre considerar o valor máximo dos cupons.</p> <p>Somente neste caso é permitido que seja realizado um pagamento de um valor menor que R\$ 10,00 no cartão. Exemplo: Uma compra de R\$ 35,00 o cliente pode pagar R\$ 30,00 utilizando cupons de troca ou cupons promocionais e pagar R\$ 5,00 com cartão de crédito.</p>
RN0036	Gerar cupom de troca	Um cupom de troca deve ser gerado quando uma compra for paga com outros cupons em que o valor supere o valor da compra. Obs: O sistema não deve possibilitar o uso de cupons que supere a compra desnecessariamente, ex: a venda tem valor total de R\$ 50,00 e o cliente possui três cupons, um com valor de R\$ 20,00, outro com valor de R\$ 40,00 e um terceiro com valor de R\$ 35,00 o sistema não deve possibilitar o uso dos três cupons nesta compra, deve ser aceito apenas dois cupons e consequentemente gerar um cupom com a diferença de R\$ 5,00, ou R\$ 10,00 ou R\$ 25,00.
RN0038	Alterar status da compra conforme processo de aprovação de forma de pagamento	<p>Caso as formas de pagamento tenham sido validadas com sucesso, a compra deve passar ter o status APROVADA.</p> <p>Caso contrário deve passar a ter o status REPROVADA.</p>
RN0039	Alterar status da compra para transporte	Toda compra selecionada para ser entregue por um administrador deve ter seu status alterado para EM TRANSPORTE.
RN0040	Alterar status da compra após entrega	Toda compra selecionada como entregue por um administrador deve ter seu status alterado para ENTREGUE.
RN0041	Gerar pedido de troca	<p>Todo item selecionado para troca deve gerar um pedido de troca. Este pedido deverá ter o status EM TROCA.</p> <p>Caso o cliente solicite a troca de toda a compra o status do pedido deverá ser EM TROCA.</p>
RN0042	Alterar status do pedido após recebimento de troca	Ao confirmar que os itens de um pedido de troca ou uma compra com status EM TROCA foi recebido o status do pedido ou compra deverá ser TROCADO.
Rn0043	Validação para solicitar troca	Somente itens de pedidos com status ENTREGUE poderão receber solicitação de troca.



RNF0045	Retirar item do carrinho	Toda vez que um item for desbloqueado todos os itens do mesmo produto deverão ser retirados do carrinho de compra que gerou o prazo de bloqueio.
RNF0046	Gerar notificação de autorização de troca	Quando o administrador autorizar uma troca o sistema deverá gerar uma notificação sobre tal ao cliente.
Grupo: Controle de estoque		
RN0051	Validar dados de estoque	Para cada entrada em estoque, deve ser obrigatoriamente informado o produto, a quantidade, o valor de custo, fornecedor, e a data de entrada dos itens de produto.
RN005x	Definir valor de item com diferentes custos	Quando itens de uma determinada peça de roupa forem registrados com valores de custo diferentes deverá ser calculado o valor de venda com base no grupo de precificação, porém o valor de todos os itens deverá ser igual, considerando então o maior valor de custo.
RN0061	Quantidade de itens	Não deve ser permitido que seja realizado a entrada de itens de peça de roupa com quantidade igual a zero.
RN0062	Valor de custo	Para todo item deve haver um valor de custo.
RNF064	Data de entrada	Não deve ser permitido que itens sejam registrados sem que uma data de entrada seja registrada.

#### 4.4. PREMISSAS

- A aplicação deve ser intuitiva ao usuário, que permitirá a realização de vendas, trocas e devoluções sendo cliente e controle de estoque de produtos entre outras funcionalidades como administrador sem maiores problemas.
- O desenvolvimento utilizando boas práticas e padrões do mercado permitirá que a aplicação não tenha preocupações ao seu uso.
- O objetivo da aplicação é oferecer facilidade de gerenciar vendas e estoque que possam contribuir no lucro desejado, mas não abrange a divulgação de vínculos publicitários da marca ou exposição nas demais plataformas.
- Utilização da linguagem de desenvolvimento Java;
- A equipe de desenvolvimento será composta de um desenvolvedor(a).
- O projeto será orientado pelo professor Rodrigo Rocha.

#### 5. INFLUÊNCIA DAS PARTES INTERESSADAS

- João Medeiros: diretor da MIRROR LTDA e patrocinador do projeto. Interessado em minimizar o custo operacional exorbitante da atual loja virtual da Mirror, assim como, acompanhar a tendência dos seus concorrentes, facilitando a gestão de vendas e produtos online afim de analisar lucros obtidos por este canal.
- Lorena Oliveira: gerente e responsável pelo desenvolvimento da Mirror Fashion, interessada que a solução do e-commerce seja capaz de auxiliar toda da MIRROR na gestão de vendas e produtos online, onde o desenvolvimento deve estar atrelado a boas práticas e padrões atuais que possibilitem o funcionamento por muito tempo.



## 6. REPRESENTAÇÃO ARQUITETURAL

Os sistemas serão desenvolvidos tendo como base a arquitetura ilustrada na Figura 1. Toda a arquitetura será baseada nos padrões de projetos tradicionais e nos padrões J2EE sendo executados dentro de um Servidor de Aplicações.

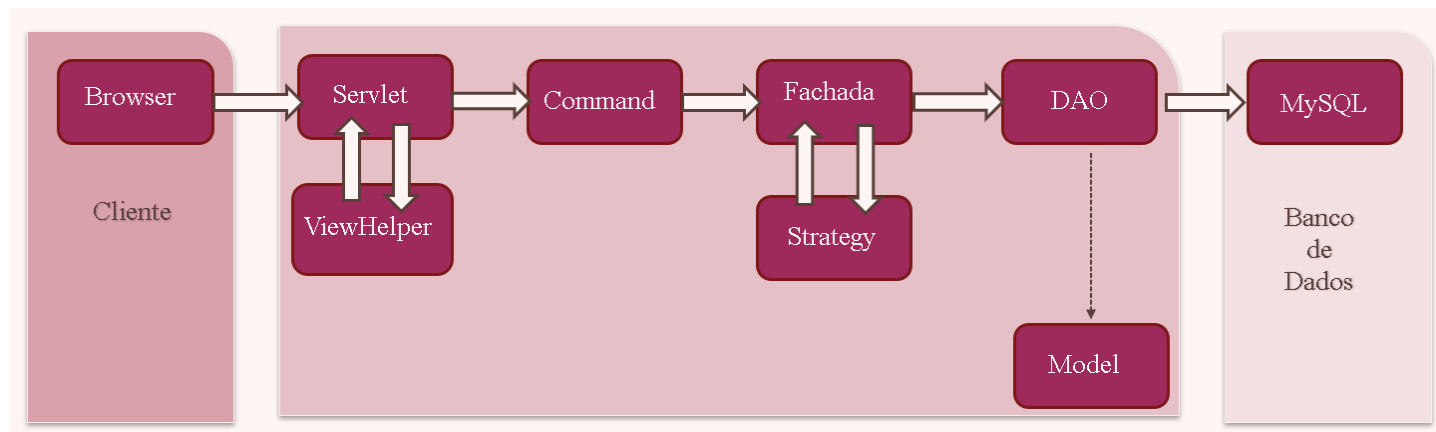


Figura 1 - Modelo Arquitetural

O Container WEB abrigará os componentes da arquitetura responsáveis pela camada de apresentação. Será utilizado para esta camada o JSP – Java Server Pages.

Classes de Domínio são as classes que representam os *Value Object*, contendo somente os atributos e os métodos *getters/setters*.

As Classes de Negócio representam as classes responsáveis por aplicar as regras de negócio do sistema como, por exemplo, Cadastrar Cliente. Constarão também nas classes de negócio o relacionamento com os Repositorys responsáveis por persistir e recuperar os objetos no banco de dados representadas pela DAO.

O diagrama a seguir, Figura 2, representa a organização das classes dentro dos pacotes da aplicação que fará uso da arquitetura. É importante atentar para a padronização de nomes de pacotes.

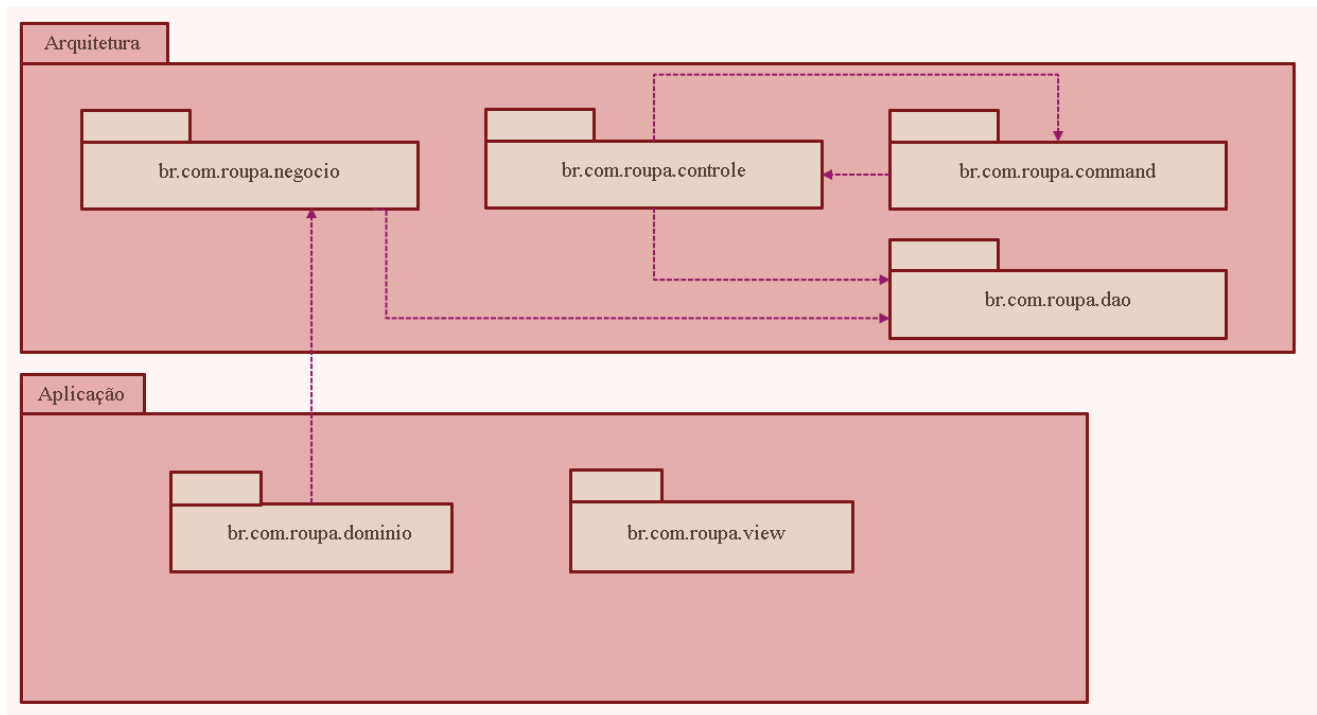


Figura 2 - Diagrama em pacotes

## 6.1. RESTRIÇÕES ARQUITETURAIS

Foram identificadas algumas orientações / restrições pertinentes ao desenvolvimento deste subsistema:

- Utilização do JDK 14.0.1 do Java;
- Utilização do servidor Apache Tomcat/9.0.41;
- Utilização do JSP – *Java Server Pages* para desenvolvimento da camada de apresentação;
- Utilização do SGBD MySQL.

### 6.1.1. OBJETIVOS E RESTRIÇÕES ARQUITETURAIS

Alguns requisitos registrados que impactam diretamente a arquitetura do e-commerce Mirror Fashion, considerando premissas definidas para o sistema 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 MySQL.



## 7. 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 sistema de e-commerce Mirror Fashion.

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



## 7.1. DIAGRAMA DE CASO DE USO GERAL DO SISTEMA



powered by Astah

Figura 3 - Diagrama de Caso de Uso do Sistema - Geral



## **7.2. DESCRIÇÃO DOS CASOS DE USO ARQUITETURALMENTE SIGNIFICATIVOS**

### **– Cadastrar Dados Pessoais**

Este caso de uso se inicia quando um possível cliente registra os seus dados pessoais através do formulário de cadastro.

### **– Realizar Login**

Este caso de uso se inicia quando um cliente ou um administrador do sistema se autentica para realizar suas respectivas operações, é executado em todas as operações de usuários, exceto no cadastro de dados pessoais do cliente.

### **– Realizar pedidos**

Este caso de uso se inicia quando um cliente escolhe as peças de roupa desejadas para compra e define suas características, como a quantidade do item a ser inserido no carrinho de compras. Sua execução permanece durante todo o processo pertinente à compra da(s) roupa(s), passando pela escolha, verificação do carrinho de compras, escolha da forma de pagamento, cálculo de frete até a finalização da transação.

### **– Manter Dados Pessoais, Manter Cartões, Manter Endereços**

Este caso de uso se inicia quando um Cliente autenticado realiza operações sob os seus dados, como alterações de dados pessoais, cadastro de endereços e cadastro de cartões de crédito.

### **– Consultar Pedidos (Cliente)**

Este caso de uso se inicia quando um cliente já autenticado no sistema, consulta suas transações realizadas e efetua operações sobre elas, como a solicitação de troca ou devolução de peças de roupa, além de visualizar o status de pagamentos.

### **– Consultar Pedidos Realizados (Administrador)**

Este caso de uso se inicia quando um administrador autenticado no sistema gerencia os pedidos realizados pelos clientes, tais atividades estão relacionadas ao despacho de roupas para entrega, confirmação de recebimento de pedidos de troca, trocar roupa solicitadas, alterar status sobre pedidos e a consulta de pedidos de compra e troca.

### **– Gerenciar Estoque**

Este caso de uso se inicia quando um administrador autenticado realiza o controle de estoque, tais atividades estão relacionadas à inclusão e baixa de roupas vendidas, além da consulta deles, também é possível cadastrar fornecedores de roupas.





### – Gerenciar Produtos

Este caso de uso se inicia quando um administrador do sistema devidamente autenticado realiza operações específicas sobre as roupas, como a consulta e a inativação dele.

### – Gerenciar Clientes

Este caso de uso se inicia quando um administrador do sistema devidamente autenticado realiza operações específicas sobre os clientes, como a consulta e a inativação dele.

### – Criar Relatórios (Admin)

Este caso de uso se inicia quando um administrador autenticado no sistema gerencia todos os dados gerados por compras, trocas, devoluções e produtos vendidos. O relatório é feito um gráfico de linhas onde consta, filtrado pelos 3 produtos mais vendidos, a quantidade vendida por datas selecionadas.

## 7.3. VISÃO DE LÓGICA

Esta visão apresenta elementos de design significativos do ponto de vista da arquitetura, descrevendo a organização do Sistema **Mirror Fashion** em pacotes, bem como a organização desses pacotes em camadas.

**Apresentação:** Contém classes para as interfaces gráficas com os usuários. Através destas interfaces os Administradores conseguem interagir com o Mirror Fashion, com o intuito de incluir, alterar e excluir produtos.

**Negócio:** Contém classes que controlam a execução das funcionalidades do Mirror Fashion.

**Persistência:** Contém classes responsáveis por persistir as entidades de modelo. Por exemplo, contém as classes que permitem ler e gravar os objetos no banco de dados relacional.

O Diagrama com as camadas do sistema Mirror Fashion é ilustrado na figura 4.

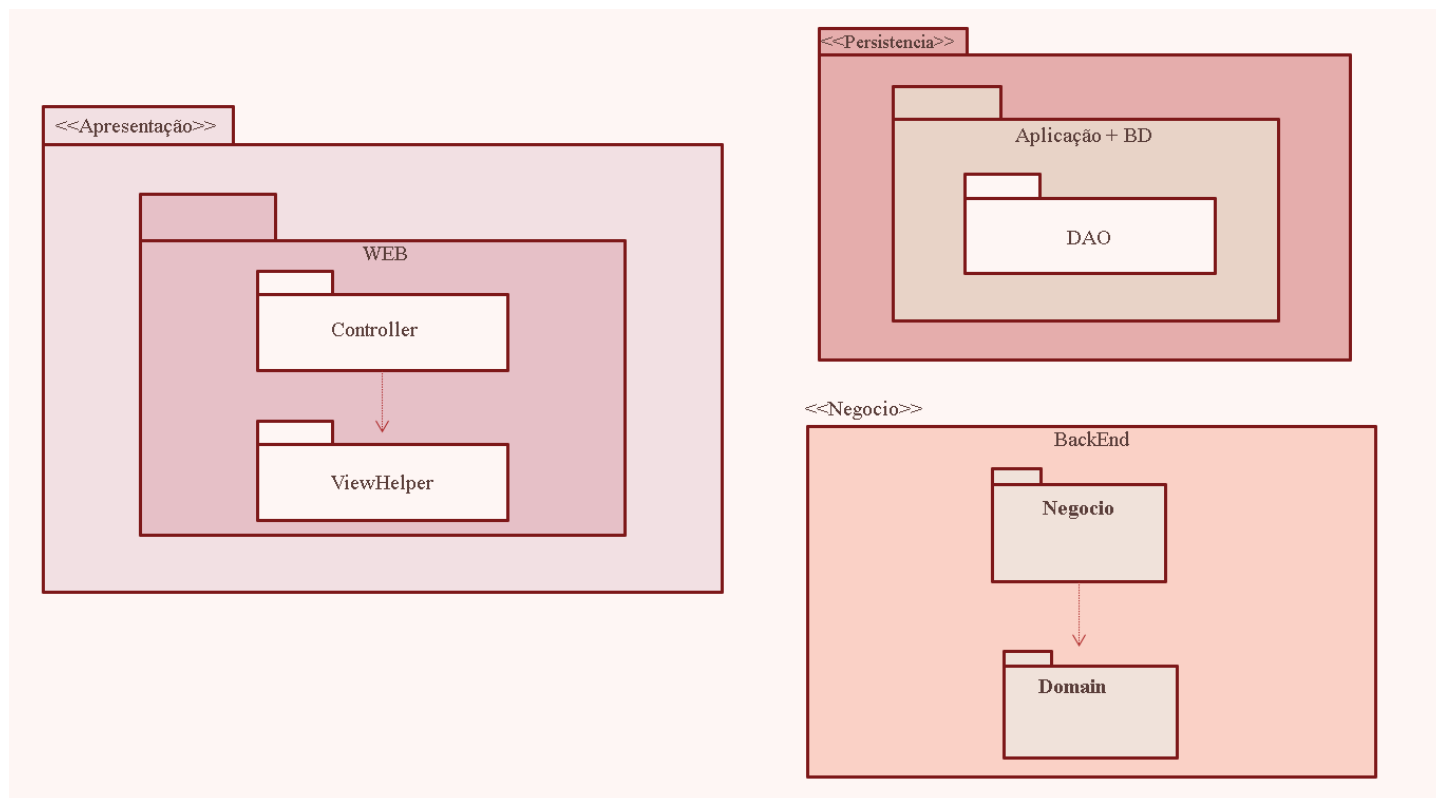


Figura 4 - Diagrama de Camadas

Neste momento, é importante ressaltar que a camada de apresentação envolve componentes que são executados na porção servidora e na porção cliente. Na porção servidora, são executados os componentes que montam as páginas HTML/JSP de resposta e controlam o fluxo de interação com o usuário. Assim, envolve interpretação de páginas JSP, utilização de *viewhelpers* para montagem das páginas de resposta, delegação de solicitação para a camada de negócio, obtenção de respostas da camada de negócio, dentre outros. Já na porção cliente, estão os códigos Javascript que são executados nos navegadores dos usuários com o intuito de facilitar a interação. Por exemplo, podem fazer uma pré-verificação dos dados digitados (como o preenchimento de campos obrigatórios), solicitação de páginas, de modo a tornar mais dinâmica a interação com o usuário, dentre outros.

### 7.3.1. CAMADA DE APRESENTAÇÃO

Nesta camada, temos na Figura 5 o pacote View que contém todos os arquivos relacionados à exibição de informações para usuário, o que engloba páginas JSP e HTML, imagens, Javascript, dentre outros.

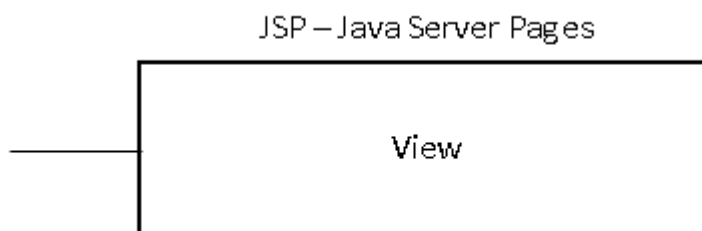


Figura 5: Camada de Apresentação

### 7.3.2. CAMADA DE NEGÓCIO

Nesta camada, na Figura 6 temos o pacote controller que contém as classes responsáveis por controlar as regras de negócio da aplicação. O pacote model, contém as classes que representam o modelo, ou seja, aquelas que contêm as informações sobre o Sistema do Mirror Fashion. O pacote contém as classes de controle do negócio.

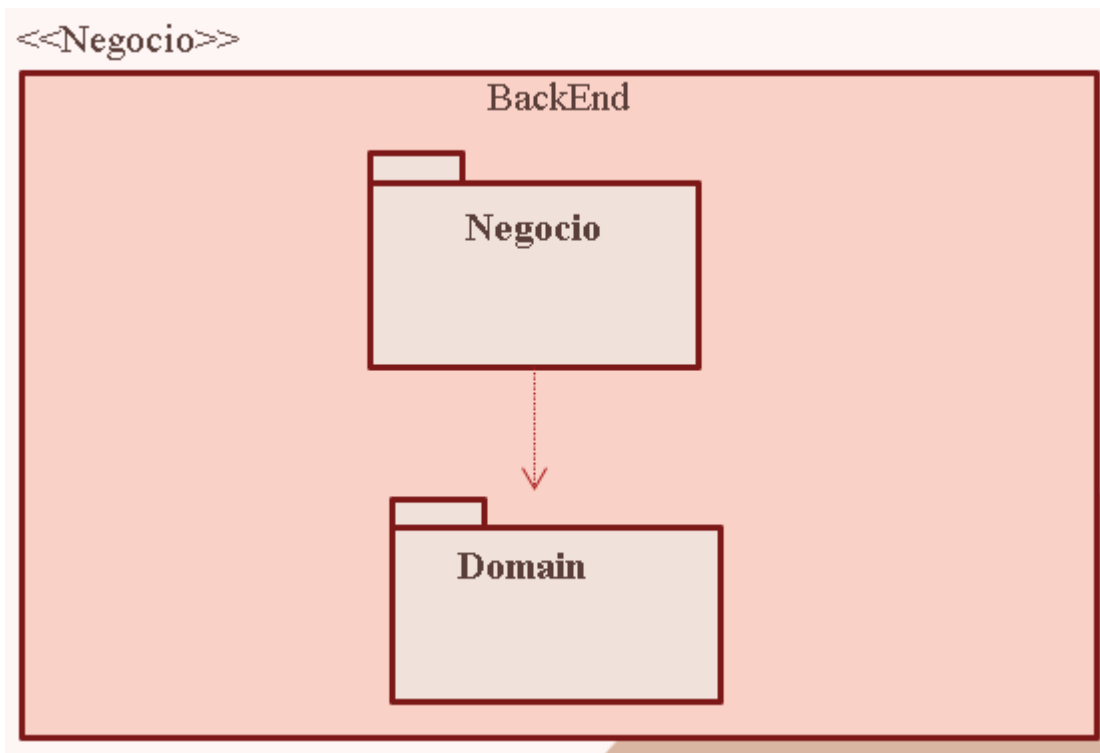


Figura 6: Camada de Negócios

### 7.3.3. PACOTE CONTROLLER

A figura 7 ilustra as principais classes de controle.

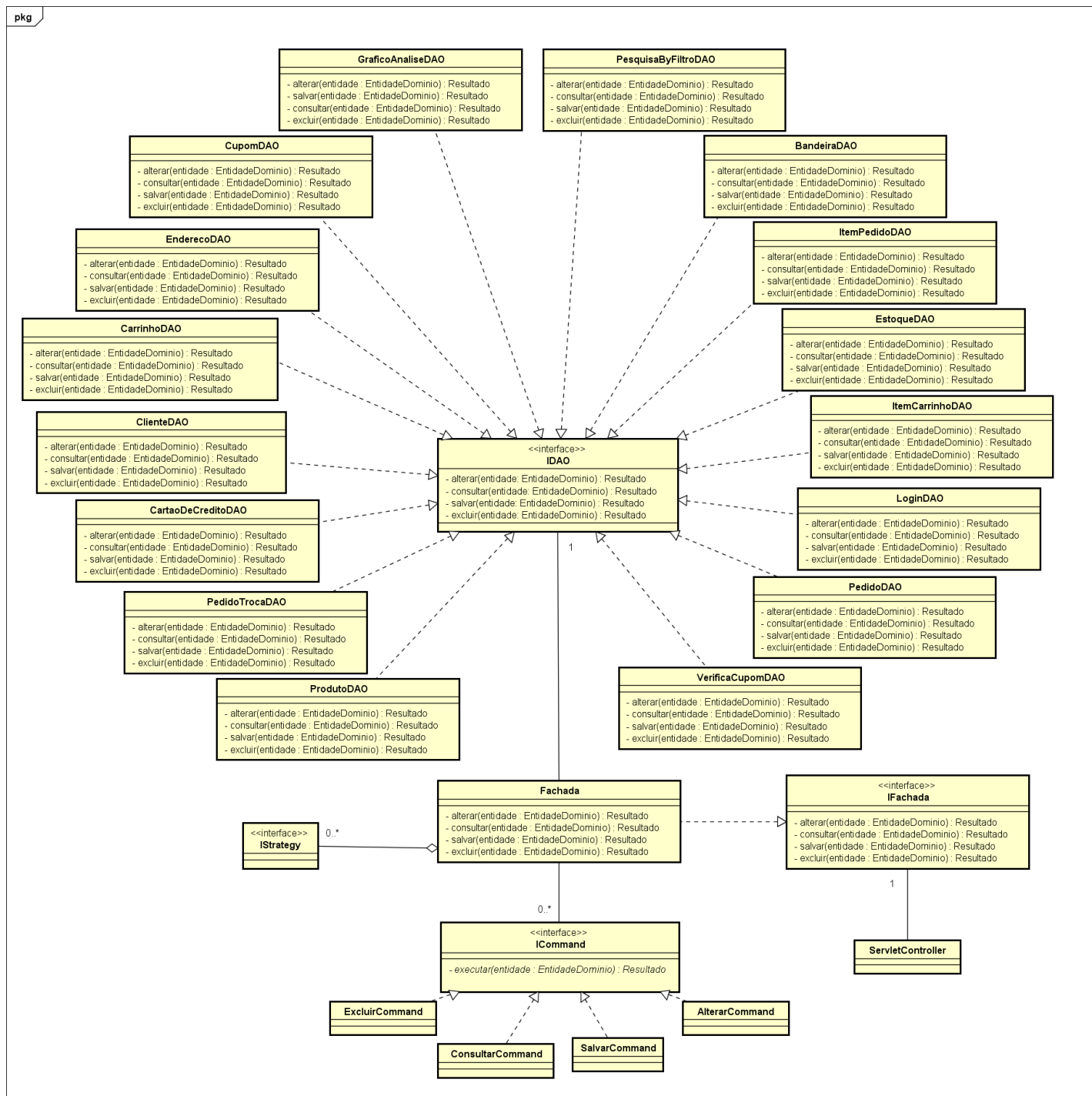
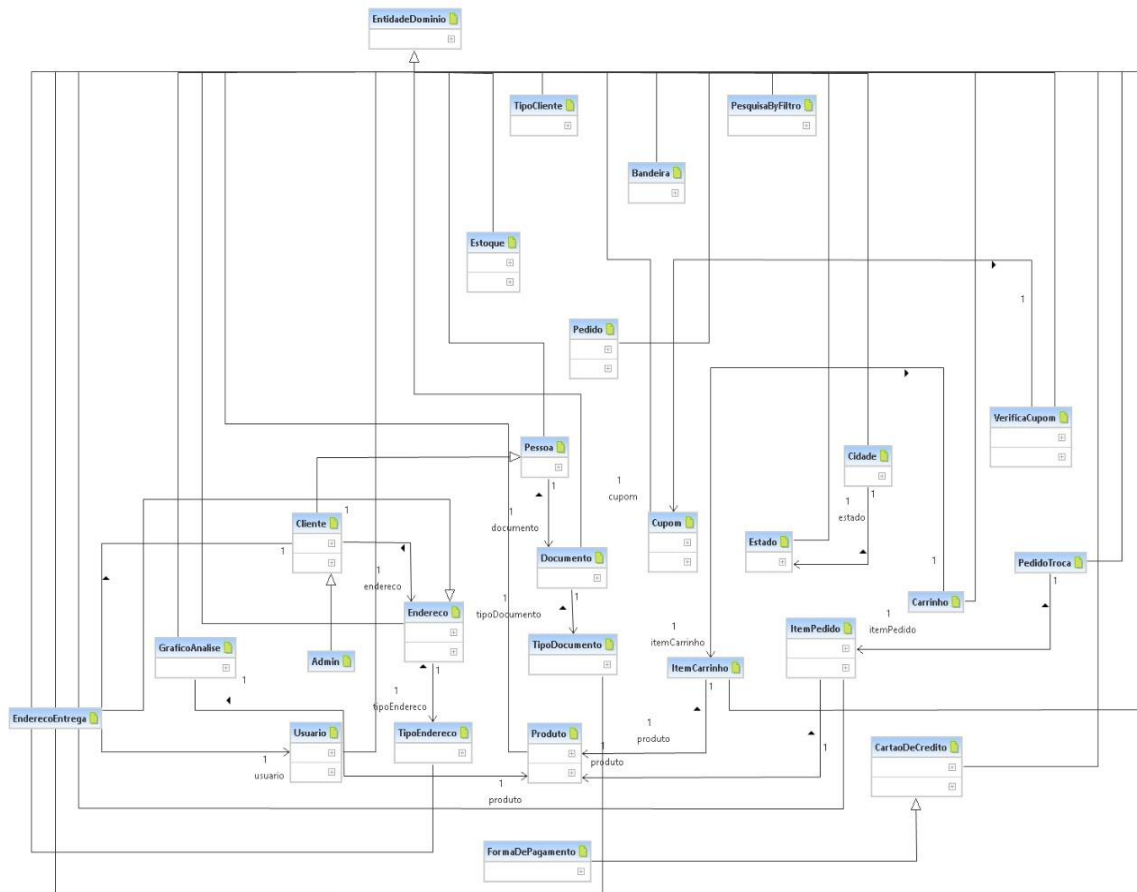


Figura 7: Classes de controle

### 7.3.4. PACOTE MODEL

A figura 8 ilustra as principais classes de domínio.



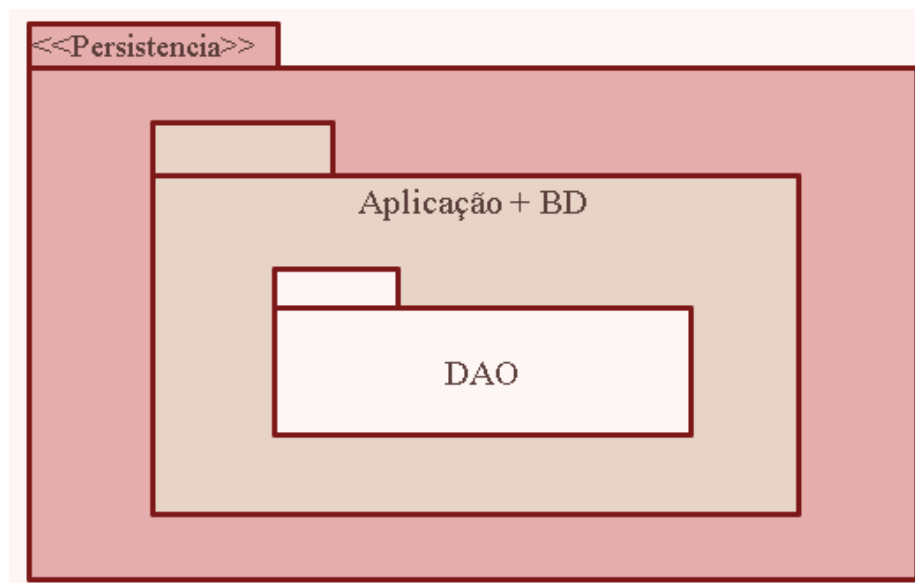
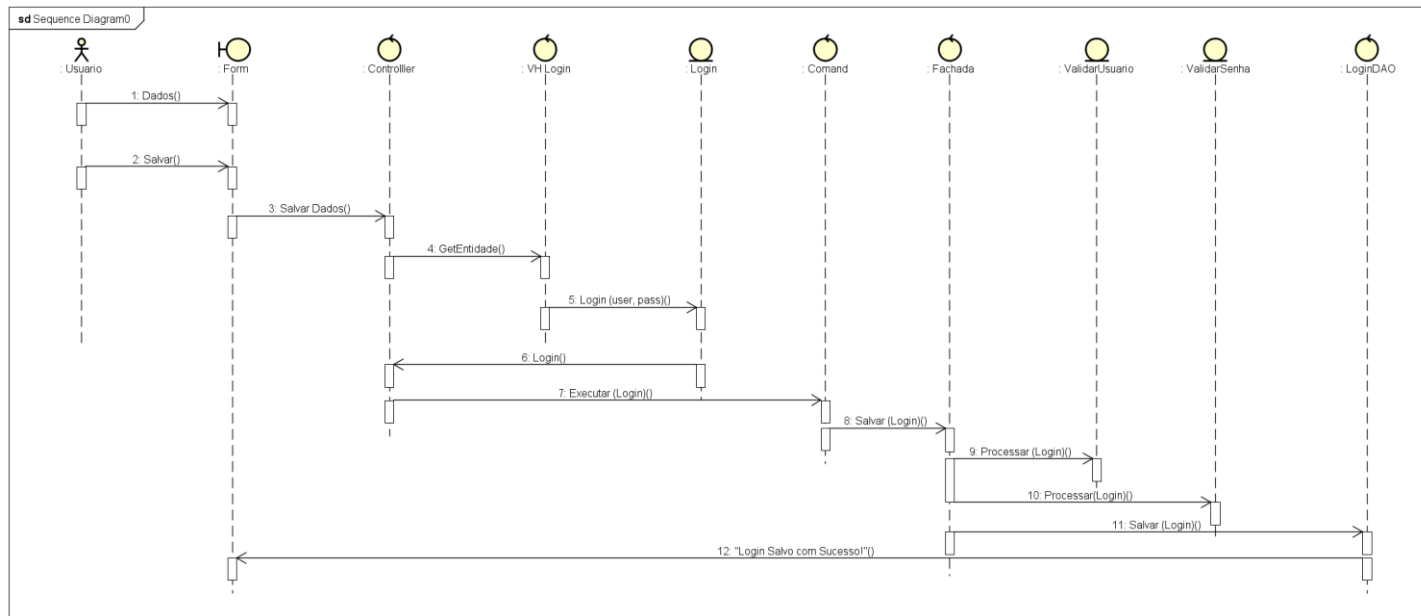


Figura 9: Camada de Persistência.

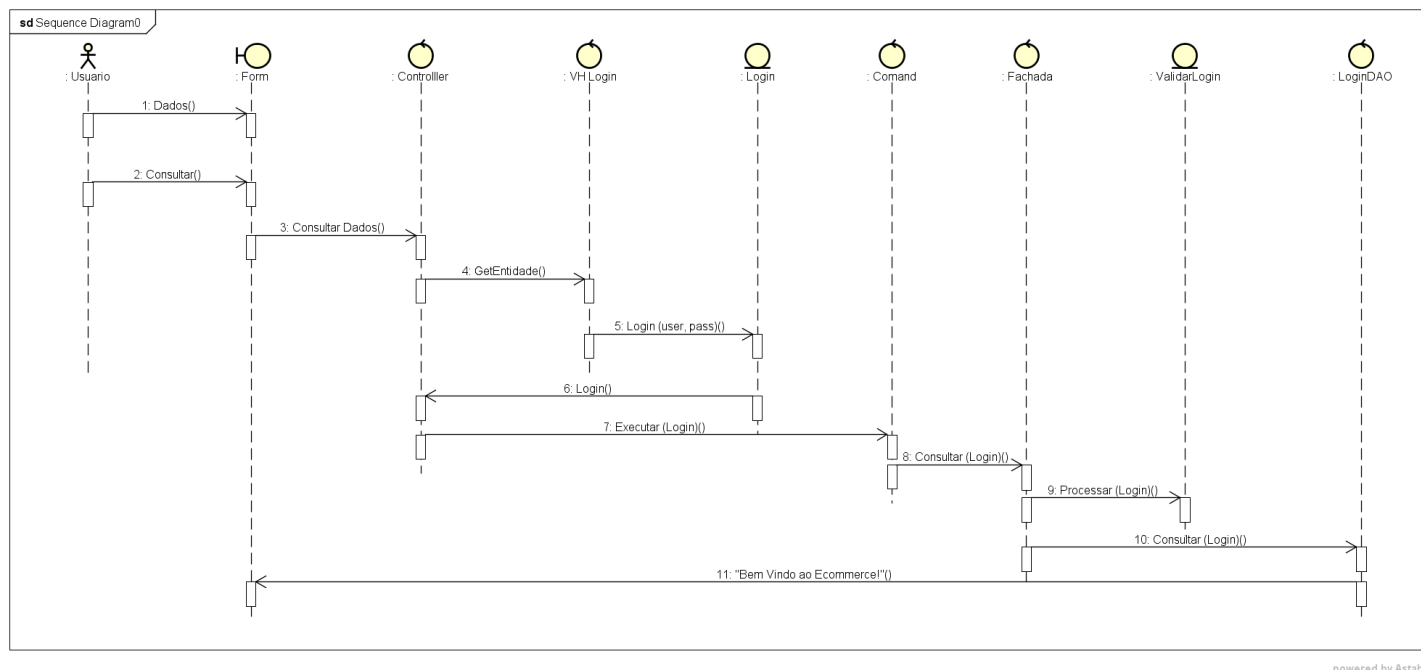


### 7.3.6. REALIZAÇÃO DOS CASOS DE USO SIGNIFICATIVOS

#### - Inserir conta do Usuário (Login)

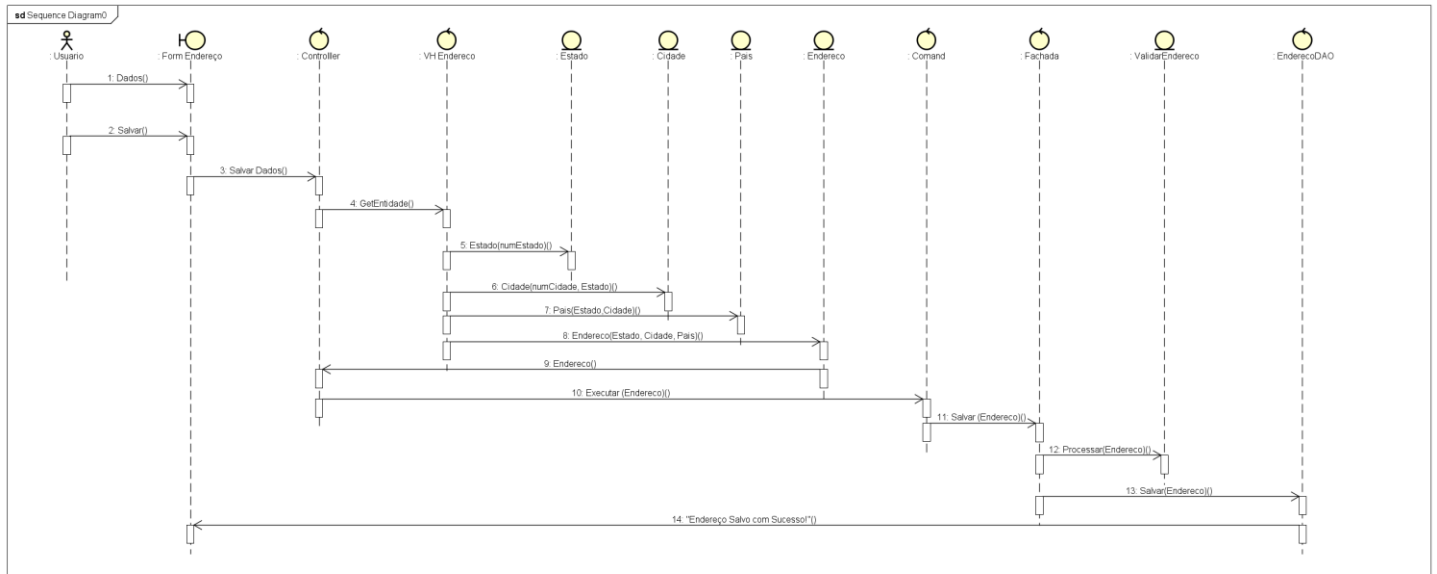


#### - Consultar Login

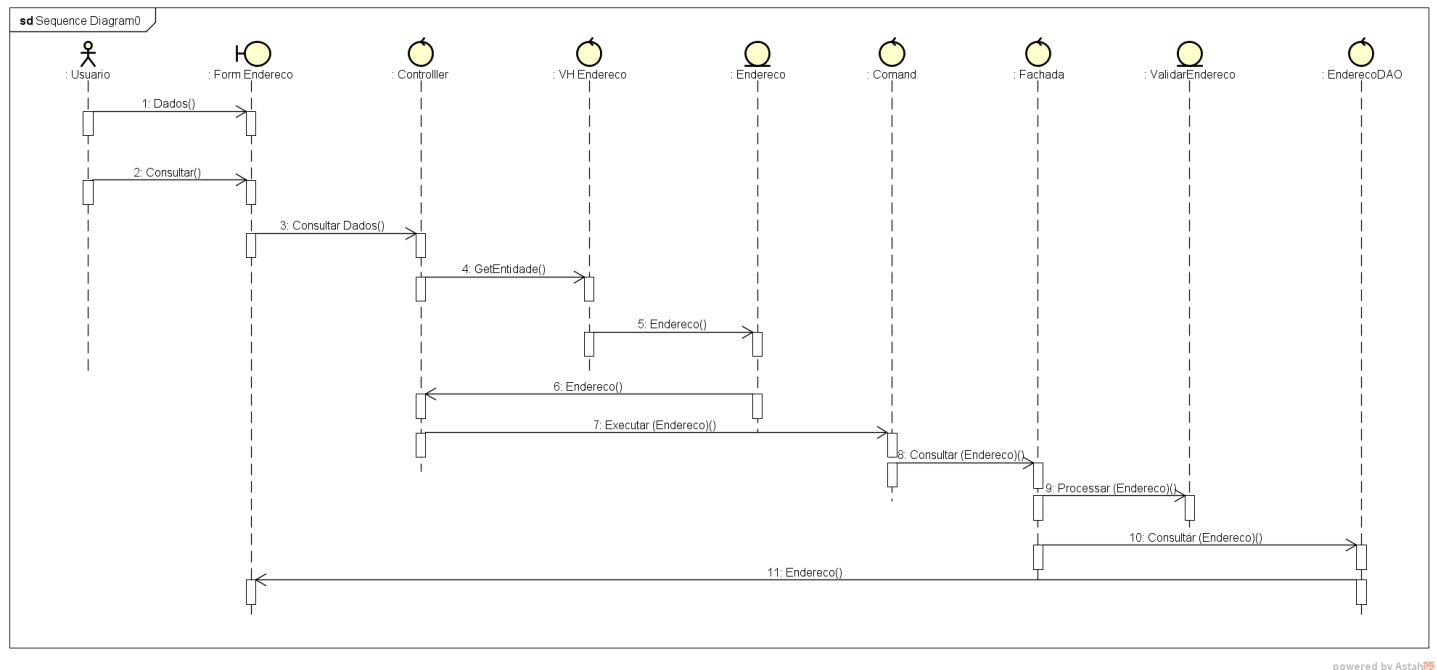




## - Inserir Endereço



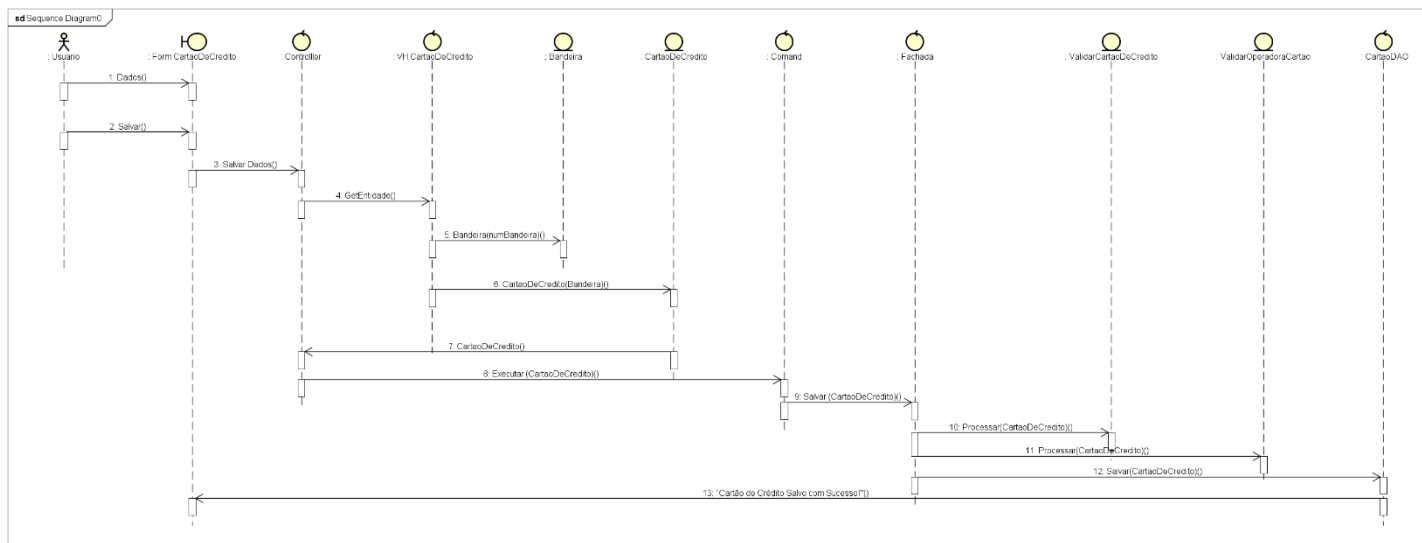
## - Consultar Endereço



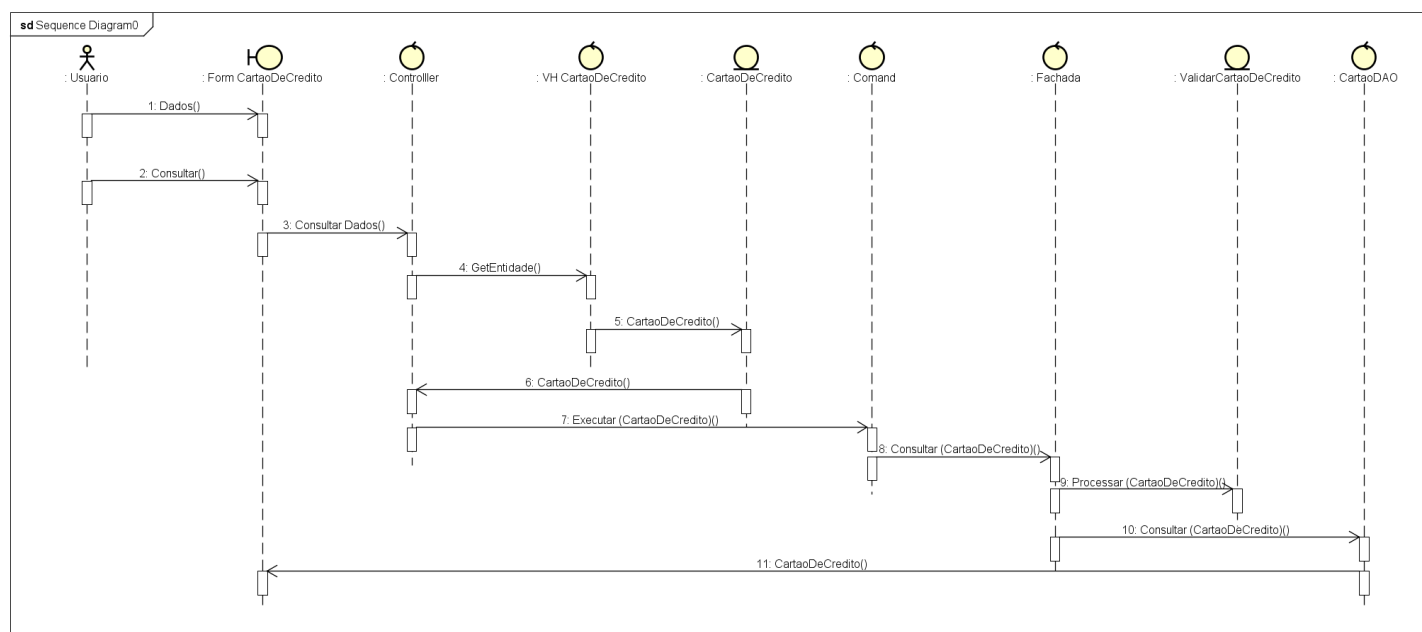




## - Inserir Cartão de Crédito



## - Consultar Cartão de Crédito





## 7.4. VISÃO DE IMPLANTAÇÃO

Esta seção descreve as configurações da rede física (hardware) na qual o Mirror Fashion será implantado e executado.

Trata-se de uma visão do Modelo de Implantação que, para a configuração em questão, indica os nós físicos (computadores, CPUs), que executarão o subsistema Mirror Fashion, e as respectivas interconexões (barramento, LAN, etc). A figura 10 ilustra o modelo de implantação para o e-commerce.

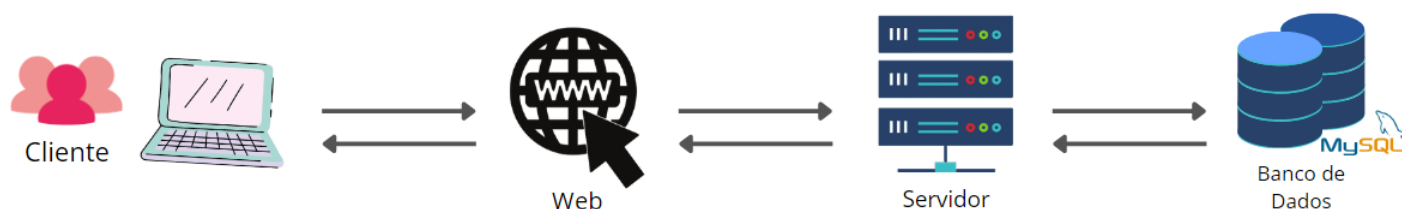


Figura 10: Visão de Implantação do Mirror Fashion

Na Figura 10 observa-se os seguintes nós físicos:

- **Web:** É a rede mundial que conecta esses clientes a outros pontos, como um servidor por exemplo.
- **Banco de dados MySQL:** Nó que contém o BD Central do Sistema Mirror Fashion.
- **Cliente:** Acesso de dispositivos desktop a aplicação hospedado no servidor de aplicação.

## 7.5. VISÃO DE IMPLEMENTAÇÃO

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

A estrutura geral de implementação para o Mirror Fashion é baseada na estrutura da Visão Lógica, assim, não há necessidade de detalhar os diagramas de camadas e pacotes de implementação, uma vez que são fortemente baseados naqueles desenvolvidos para Visão Lógica.

## 7.6. VISÃO DE DADOS

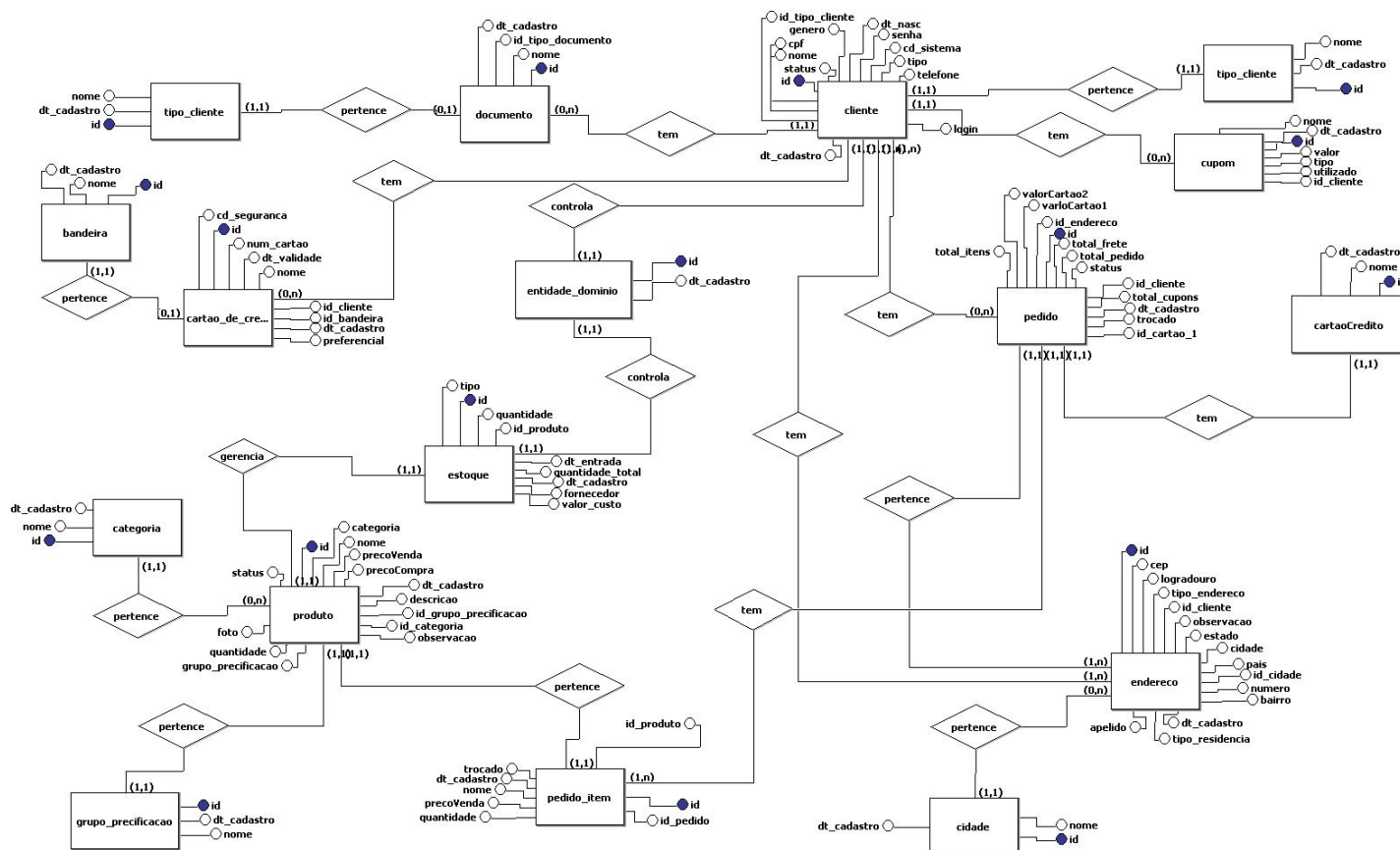


Figura 11 – Modelo Lógico

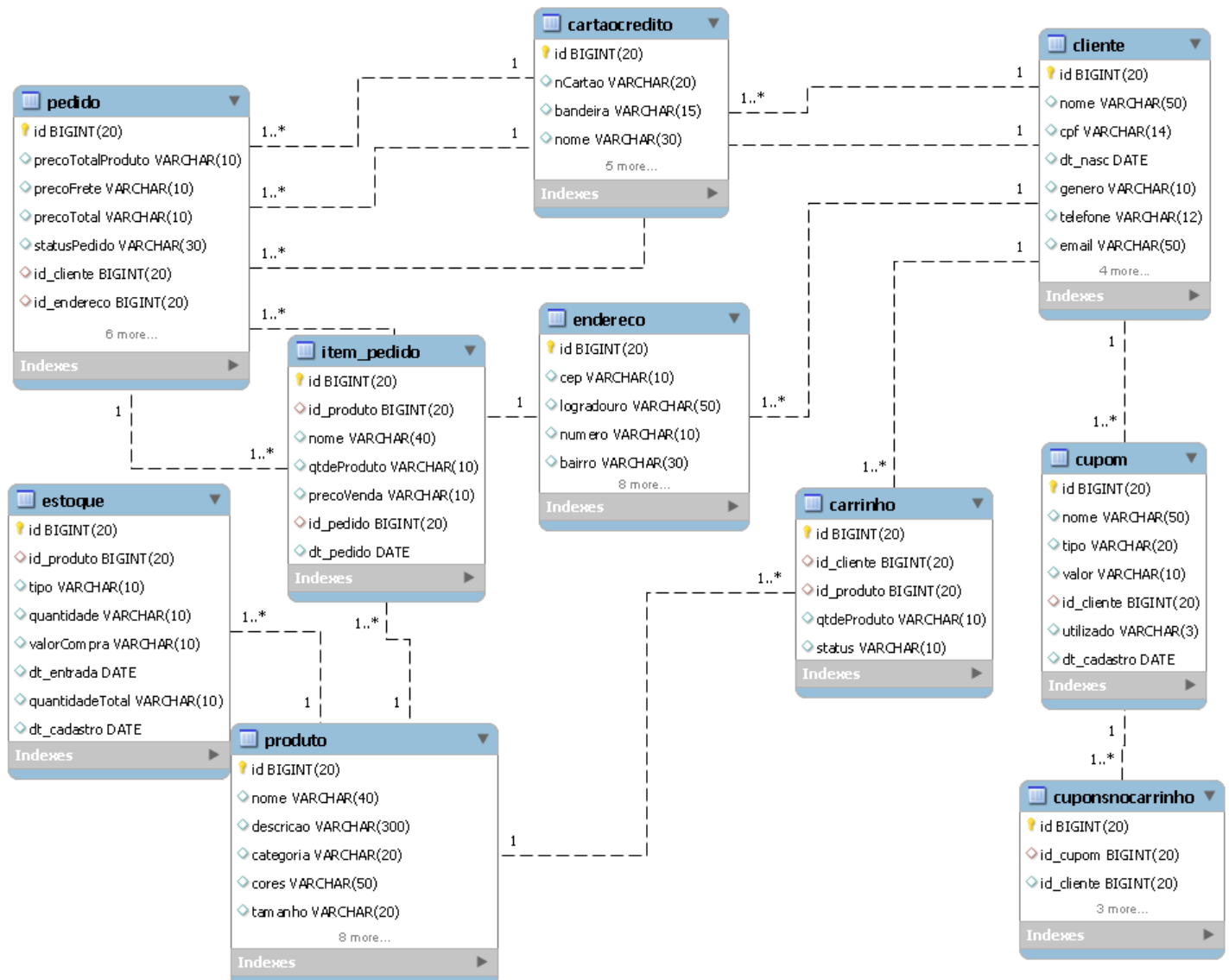


Figura 12 – Modelo Físico

A Tabela 1 define o mapeamento das principais classes de modelo para entidades do modelo lógico do BD Oracle.

Note que existem alguns campos nas entidades lógicas do BD que não estão mapeadas diretamente com as classes de modelo da Visão Lógica contidas neste documento:

Os campos que começam com "IN" são campos definidos através de domínios do BD. As classes de modelo que possuem os atributos que representam tais campos não constam neste documento, mas podem ser consultadas no documento [REF – Modelo de Design]



Tabela 1: Mapeamento Objeto-Relacional

Classe	Entidade
<p><b>Cliente</b></p> <ul style="list-style-type: none"> <li>telefone: String</li> <li>usuario : Usuario</li> <li>dt_nasc : String</li> <li>status : String</li> <li>tipo : String</li> <li>alteraCliente: String</li> </ul>	<p><b>cliente</b></p> <ul style="list-style-type: none"> <li>id BIGINT(20)</li> <li>nome VARCHAR(50)</li> <li>cpf VARCHAR(14)</li> <li>dt_nasc DATE</li> <li>genero VARCHAR(10)</li> <li>telefone VARCHAR(12)</li> <li>email VARCHAR(50)</li> <li>senha VARCHAR(20)</li> <li>status VARCHAR(10)</li> <li>tipo VARCHAR(10)</li> <li>dt_cadastro DATE</li> </ul> <p>Indexes</p>
<p><b>Endereco</b></p> <ul style="list-style-type: none"> <li>logradouro: String</li> <li>numero : String</li> <li>bairro: String</li> <li>cep: String</li> <li>cidade: String</li> <li>estado: String</li> <li>pais: String</li> <li>tipoResidencia: String</li> <li>observacoes: String</li> <li>tipoEndereco: String</li> <li>idCliente: String</li> <li>status : String</li> <li>alteraEnderecc: String</li> </ul>	<p><b>endereco</b></p> <ul style="list-style-type: none"> <li>id BIGINT(20)</li> <li>cep VARCHAR(10)</li> <li>logradouro VARCHAR(50)</li> <li>numero VARCHAR(10)</li> <li>bairro VARCHAR(30)</li> <li>cidade VARCHAR(50)</li> <li>estado VARCHAR(10)</li> <li>pais VARCHAR(20)</li> <li>tipoResidencia VARCHAR(20)</li> <li>observacoes VARCHAR(20)</li> <li>tipoEndereco VARCHAR(20)</li> <li>id_cliente BIGINT(20)</li> <li>dt_cadastro DATE</li> </ul> <p>Indexes</p>



**Produto**

- nome : String
- descricao : String
- categoria : String
- cores : String
- tamanho : String
- precoCompra : String
- precoVenda : String
- qtdeEstoque : String
- foto : String
- status : String
- observacao : String
- descricaoStatusProduto : String
- grupoPrecificacao : String
- alteraProduto : String

**produto**

- id BIGINT(20)
- nome VARCHAR(40)
- descricao VARCHAR(300)
- categoria VARCHAR(20)
- cores VARCHAR(50)
- tamanho VARCHAR(20)
- precoCompra VARCHAR(10)
- precoVenda VARCHAR(10)
- qtdeEstoque VARCHAR(10)
- foto VARCHAR(50)
- dt\_cadastro DATE
- status VARCHAR(10)
- grupoPrecificacao VARCHAR(30)
- descricaoStatusProduto VARCHAR(40)

Indexes

**Pedido**

- precoTotalProduto : String
- precoFrete : String
- precoTotal : String
- statusPedido : String
- id\_cliente : String
- id\_endereco : String
- cartao1 : String
- valorCartao1 : String
- cartao2 : String
- valorCartao2 : String
- desconto : String
- alteraPedido : String
- cliente : Cliente

**pedido**

- id BIGINT(20)
- precoTotalProduto VARCHAR(10)
- precoFrete VARCHAR(10)
- precoTotal VARCHAR(10)
- statusPedido VARCHAR(30)
- id\_cliente BIGINT(20)
- id\_endereco BIGINT(20)
- cartao1 BIGINT(20)
- valorCartao1 VARCHAR(10)
- cartao2 BIGINT(20)
- valorCartao2 VARCHAR(10)
- desconto VARCHAR(10)
- dt\_pedido DATE

Indexes



## **7.7. TAMANHO E PERFORMANCE**

O sistema Mirror Fashion será usado para o controle de vendas de roupas e consequentemente terá uma grande base.

As estimativas do número de usuários e de carga de utilização em períodos de pico de utilização, bem como maiores informações sobre questões relacionadas ao tamanho e desempenho do sistema Mirror Fashion podem ser obtidas no documento de requisitos não funcionais.

## **7.8. QUALIDADE**

O sistema Mirror Fashion será usado para alavancar vendas de roupas online, consequentemente tratando de altos volumes de pedidos e muitas operações de vendas/ acessos a loja diariamente.

## **7.9. 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.



## 7.10. REFERÊNCIAS

Unified Modeling Language: <http://www.omg.org/technology/documents/formal/uml.htm>  
RUP. Rational Unified Process.