**Documento de Visão de Projeto**



**Projeto de Livraria**

**Histórico de Versões**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versão - Git** | **Autor** | **Descrição** | **Data** |
| 2a2bf34 | PedroHenriqueFonsecaMelo | Initial commit | 15/02/2023 |
| 1678585 | PedroHenriqueFonsecaMelo | BaseV1 | 22/02/2023 |
| 4112d45 | PedroHenriqueFonsecaMelo | ConexaoBD e Inserir Cliente sucesso | 24/02/2023 |
| e2a5ea1 | PedroHenriqueFonsecaMelo | Shell Cartao e Shell Endereco + Endereco Form | 26/02/2023 |
| c03793a | PedroHenriqueFonsecaMelo | Shell Cartao e Shell Endereco + Endereco Form P.2.1 | 28/02/2023 |
| e749821 | PedroHenriqueFonsecaMelo | Shell Cartao e Shell Endereco + Endereco Form P.2 | 28/02/2023 |
| 59fca7f | PedroHenriqueFonsecaMelo | OFFLINE Maven test v.1 | 01/03/2023 |
| b6b2c6c | PedroHenriqueFonsecaMelo | Cartao Form e in user home P.3.0 | 03/03/2023 |
| dfff5b5 | PedroHenriqueFonsecaMelo | Cartao Form e in user home P.3.1 | 04/03/2023 |
| b671d30 | PedroHenriqueFonsecaMelo | P.3.2 | 06/03/2023 |
| 2cf4ddb | PedroHenriqueFonsecaMelo | cli update P.3.3 | 07/03/2023 |
| 2cf4ddb | PedroHenriqueFonsecaMelo | Shopping cart | 07/03/2023 |
| a7778a8 | PedroHenriqueFonsecaMelo | P.4.0 | 11/03/2023 |
| f8864de | PedroHenriqueFonsecaMelo | P.4.1 | 19/03/2023 |
| e7b80a4 | PedroHenriqueFonsecaMelo | P.4.1.1 | 23/03/2023 |
| f43fa85 | PedroHenriqueFonsecaMelo | P.4.1.2 | 26/03/2023 |
| 47011c6 | PedroHenriqueFonsecaMelo | P.4.1.3 | 28/03/2023 |
| ce72c18 | PedroHenriqueFonsecaMelo | Volta | 06/04/2023 |
| 975c97e | PedroHenriqueFonsecaMelo | P.4.1.4 | 08/04/2023 |
| bdf1fca | PedroHenriqueFonsecaMelo | P.4.1.4.5 | 10/04/2023 |
| 8f9a45c | PedroHenriqueFonsecaMelo | P.4.1.5 | 12/04/2023 |
| 196a5f4 | PedroHenriqueFonsecaMelo | P.4.1.6 | 15/04/2023 |

|  |  |
| --- | --- |
| **Cliente** | FATEC - Interno |
| **Documento** | Documento de Visão de Projeto: *ImoX* |
| **Data** | 15 de abril de 2023 |
| **Autor** | **Pedro Henrique Fonseca Melo**  alexmarcolongo@gmail.com |

**Página de Assinaturas**

|  |  |  |
| --- | --- | --- |
| Revisado e Aprovado por: |  |  |
|  |  | 15/04/2023 |

## Índice

Camada de Apresentação 7

Camada de Persistência 11

Diagrama de caso de uso geral do sistema 5

Escopo 4

Objetivo 4

Pacote Model 10

Qualidade 13

Realização dos Casos de Uso Significativos 11

Representação Arquitetural 4

Tamanho e Performance 13

Visão de Dados 13

Visão de Implementação 13

## Objetivo

Este documento trata principalmente da documentação das necessidades de negócios, 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.

## Escopo

O escopo deste documento trata do desenvolvimento de um sistema WEB que atenda todas as necessidades de uma loja de livros.

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.

## Referências

Listar documentos relacionados a este documento de visão. Podem ser tanto documentos que foram utilizados para confeccionar este documento ou fazem parte deste documento, como documentos que estendem este documento.

Para a construção deste documento foram utilizadas as seguintes referências:

* Reuniões informais entre o grupo xyz e Tour House.
* Documentos elaborados pela Tour House:

SISTEMA FRONT END - CRM 2.doc

Este documento influencia os seguintes documentos:

* Documento de Requisitos
* WBS

## Necessidades de Negócio

Exemplo 1: Um sistema informatizado para controle de uma biblioteca é necessário para que a instituição que utilizar esta solução consiga ter total controle de todos os livros que a mesma possui. Com o sistema, também é possível controlar todos os empréstimos e devoluções que são efetuadas. Além de tais funcionalidades, o sistema também poderá auxiliar no controle de empréstimos que estão em atraso, para que não ocorram perdas de livros.

Exemplo 2: Um sistema informatizado que a partir dos dados armazenados referentes as operações realizadas pela biblioteca gere relatórios que serão utilizados para o controle interno da mesma e para fornecer informações aos administradores da instituição de ensino na qual está localizada.

## Objetivo do Projeto

Exemplo 1: Desenvolver uma plataforma para soluções web capaz de:

* armazenar informações em uma base de dados
* utilizar o protocolo HTTP
* ser executado em qualquer navegador

Exemplo 2: Desenvolver uma solução para uma biblioteca, que possibilite:

* Controlar os livros presentes no acervo da biblioteca e entrada de novos livros.
* Gerenciamento das operações realizadas na biblioteca: empréstimo, devolução de livros
* Integração com o sistema utilizado pela instituição de ensino

Minimizando o tempo de atendimento dos alunos e professores que desejam emprestar algum livro do acervo e facilitando a produção de relatórios com base nos dados apresentados por tais operações.

## 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.

* 1. Descrição

Descrição narrativa do produto do projeto.

* 1. Produtos a serem entregues

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

* Sistema do módulo da biblioteca , 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.
  1. Requisitos

É importante salientar que o documento de visão estará sendo feito no momento de iniciação do projeto. No qual, ainda não se tem todo o detalhamento. Portanto, os requisitos devem descritos em alto nível e depois detalhados em outro documento, caso o projeto seja aprovado.

## Requisitos Funcionais

|  |  |  |
| --- | --- | --- |
| **ID** | **Nome** | **Descrição** |
| Grupo: Cadastro de Livros | | |
| RF0011 | Cadastrar livro | O sistema deve manter um cadastro único para livros. |
| RF0012 | Inativar cadastro de livro | O sistema deve possibilitar que livros sejam inativados. |
| RF0013 | Inativar livro de forma automática | O sistema deve inativar livros sem estoque e que não possuem venda com valor inferior a parâmetro predefinido no sistema. |
| RF0014 | Alterar cadastro de livro | O sistema deve possibilitar a alteração de dados cadastrais para os livros. |
| RF0015 | Consulta de livros | O sistema deve possibilitar que um livro seja consulta  do com base em um filtro definido pelo usuário. Todos os campos utilizados para identificação do livro podem ser utilizados como filtro, tanto de forma combinada como de forma isolada. |
| RF0016 | Ativar cadastro de livros | Deve ser possível ativar o cadastro de um livro. |
|  | | |
| Grupo: Cadastro de Clientes | | |
| 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 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. |
|  | | |
| Grupo: Gerenciar Vendas Eletrônicas | | |
| 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 o mesmo seja incorporado ao perfil do cliente. |
| RF0036 | Selecionar forma de pagamento | 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 o mesmo seja incorporado ao perfil do cliente.    O cliente também poderá utilizar um cupom de troca ou um cupom promocional válido.    Deve-se possibilitar que o pagamento seja feito utilizando tanto cupons de troca, promocionais e cartão de crédito. |
| 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 do mesmo. |
| 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 | O sistema deverá possibilitar que o administrador confirme o recebimento de pedidos de troca ou compra com status EM TROCA.    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. |
| 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 livros em estoque.  No registro de cada item, deve ser indicado o livro já previamente cadastrado e a quantidade de itens do livro. |
| 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 ao livro. |
| 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. |

## Requisitos Não Funcionais

|  |  |  |
| --- | --- | --- |
| **ID** | **Nome** | **Descrição** |
| 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 registado data, hora, usuário responsável além de manter os dados alterados. |
|  | | |
| Grupo: Cadastro de Livros | | |
| RNF0021 | Código de livro | Todo livro cadastrado deve receber um código único no sistema. |
| RNF0013 | Cadastro de domínios | Deve haver um script de implantação do sistema que insere todos registros de tabelas de domínio necessárias por ex: grupo de precificação, autor, editora, 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 da mesma. |
| RNF0033 | Senha criptografada | A senha deve ser criptografada |
| RF0034 | 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. |
| Grupo: Gerenciar Vendas Eletrônicas | | |
|  |  |  |
| RNF0042 | Apresentar itens retirados do carrinho | Deve ser apresentado na listagem de itens do carrinho os produtos removidos por atingirem o prazo determinado para finalização da compra (apresentar o tempo conforme parâmetro do sistema). Assim a opção comprar deve ser desabilitada e o itens deverão ser adicionados novamente no carrinho. |

## Regras de Negócio

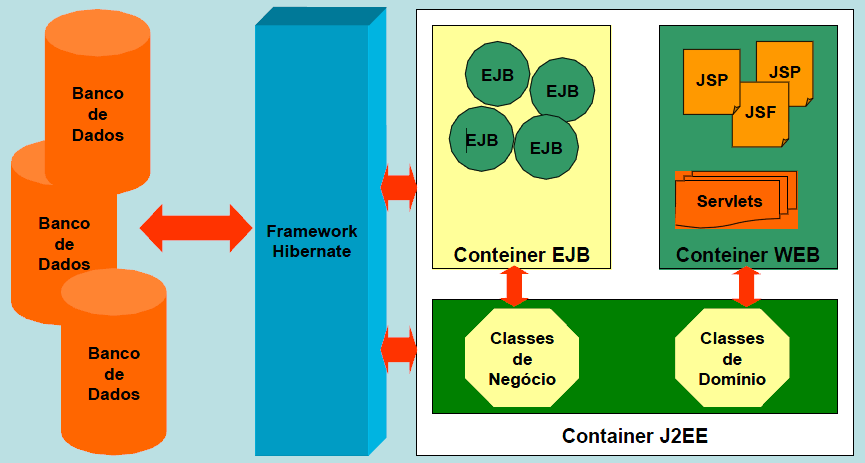
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Nome** | | **Descrição** |
| Grupo: Cadastro de Livros | | | |
| RN0011 | Dados obrigatórios para o cadastro de um livro | | Para todo livro cadastrado é obrigatório o cadastro dos seguintes dados: autor, categoria, ano, título, editora, edição, ISBN, número de páginas, sinopse, dimensões (Altura, largura, peso e profundidade), grupo de precificação e código de barras. |
| RN0012 | Associação com categorias | | Um livro pode estar associado com mais de uma categoria. |
| RN0013 | Definindo valor de venda | | Todo livro após cadastrado deverá ser associado a um grupo de precificação onde o valor deverá ter como base a margem de lucro parametrizado para o grupo definido no cadastro do livro. |
| RN0014 | Validar margem de lucro | | Um livro somente pode ter seu valor alterado se estiver dentro da margem de lucro definida pelo critério de grupo de precificação. Para um livro 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 | | Todo livro que for inativado manualmente deve ter uma justificativa e uma categoria de inativação associada. |
| RN0016 | Associar motivo de inativação automática | | Todo cadastro de livro inativado de forma automática deve ser categorizado como FORA DE MERCADO. |
| RN0017 | Associar motivo de ativação | | Todo livro 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: Nº 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 | | RN0028 | Validar retorno da operadora de cartão de crédito | Somente deve-se dar baixa no estoque de itens cuja compra tenha sido efetivada, isso significa que o status não é mais EM PROCESSAMENTO. Todo item que faça parte de uma compra não aprovada deve ser desbloqueado e mantido em estoque. | | | | |
| Grupo: Gerenciar Vendas Eletrônicas | | | |
| RN0031 | | Validar estoque para adição de itens no carrinho | Não deve ser permitido adicionar um item no carrinho de compra que não esteja disponível em estoque. Também deve ser validado a quantidade do item adicionado ao carrinho para que não seja adicionado mais itens do que o disponível em estoque. |
| ~~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 diversões 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 | Ao realizar pagamento utilizando cupons e cartões em conjunto, deve-se sempre considerar o valor máximo dos cupons.    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. |
| 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. |
| RN0037 | | Validar Forma de Pagamento para finalização de compra | Após a finalização da compra a forma de pagamento deve ser validada. Para tal deve-se validar a validade e veracidade dos cupons de troca e promocionais que porventura foram utilizados.    Também deve ser validado o aceite da compra pela respectiva operadora de cartão de crédito. |
| RN0038 | | Alterar status da compra conforme processo de aprovação de forma de pagamento | Caso as formas de pagamento tenham sido validadas com sucesso, a compra deve passar ter o status APROVADA.    Caso contrário deve passar a ter o status REPROVADA. |
| 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 | Todo item selecionado para troca deve gerar um pedido de troca. Este pedido deverá terá o status EM TROCA.    Caso o cliente solicite a troca de toda a compra o status do pedido deverá ser EM TROCA. |
| 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. |
| RN0044 | | Bloqueio de produtos | Ao adicionar o item no carrinho, este deverá ser temporiamente bloqueado para que novas compras não sejam solicitadas. Tal bloqueio só deve ser retirado no caso da compra que gerou tal status não ser efetivada ou aprovada em um prazo parametrizado, o prazo deve levar em consideração o momento do bloqueio. Obs.: O prazo parametrizado deve ser relativo ao último item incluído no carrinho. |
| RNF0045 | | Retirar item do carrinho | Toda vez que um item for desbloqueado todos 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 um determinado livro 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 itens deverão ser iguais, considerando então o maior valor de custo. |
| RN0061 | | Quantidade de itens | Não deve ser permitido que seja realizado a entrada de itens de livros com quantidade igual a zero. |
| RN0062 | | Valor de custo | Para todo item deve haver um valor de custo. |
| RNF0064 | | Data de entrada | Não deve ser permitido que itens sejam registrados sem que uma data de entrada seja registrada. |

## Premissas

* O projeto será orientado pelo professor Rodrigo Rocha.

## 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 do GoF e também nos padrões J2EE sendo executados dentro de um Servidor de Aplicações.



**Figura 1 - Modelo Arquitetural Genérico**

O Container WEB abrigará os componentes da arquitetura responsáveis pela camada de apresentação.

As Classes de Entidade são as classes que representam classes de manipulação de dados, contendo somente os atributos, os métodos getters/setters e métodos de tratamento de dados adaptados para cada classe.

As Classes de Controle representam as classes responsáveis por aplicar as regras de negócio do sistema como, por exemplo, Cadastrar Comarca.

## Restrições Arquiteturais

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

* Utilização do JDK 1.8 do Java;
* Utilização do framework Spring para desenvolvimento da camada de negócio e controle de transações;
* Utilização do SGBD H2.

## Visão de Use Case

Esta seção apresenta os Casos de Uso.

* 1. Diagrama de Caso de Manter Dados de Atributos

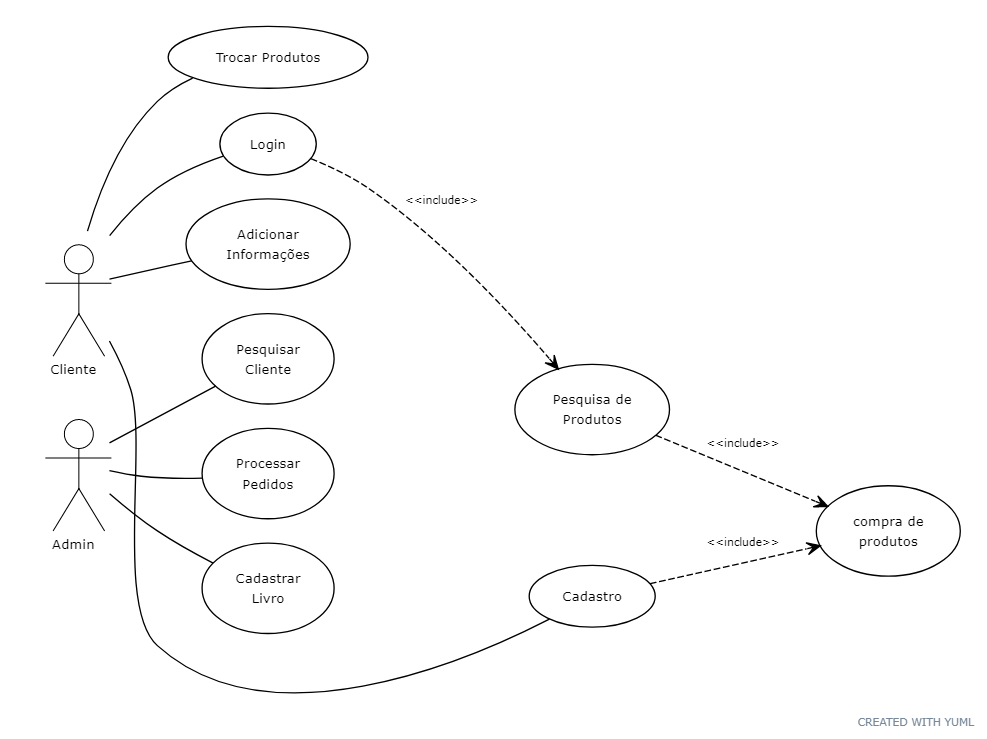


Figura 4.1 Diagrama de Caso de Uso de XXXXXXX

* 1. Descrição dos Casos de Uso Arquiteturalmente Significativos

Comprade Produtos

Este caso de uso se inicia opcionalmente após a consulta de produtos.

Login/Cadastro

Este caso de uso se inicia quando se faz login do cliente ou seu cadastro se não possuir conta no sistema.

Cadastrar Livro

Este caso de uso se inicia quando o admin fez o login e pretende cada testar um novo livro.

Pesquisar Cliente

Este caso de uso se inicia quando o admin feito o login decide procurar por uma informação específica de um ou mais usuários.

Processar Pedido

Este caso de ouro se inicia quando o admin verifica os pedidos feitos por clientes seja de troca ou compra e decide se vai aprová-los ou não

* 1. Visão de Lógica

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

O Diagrama com as camadas do sistema XXX é ilustrado na figura 5.1.



Figura 5.1 – Diagrama de camadas do XXXX

**Apresentação**: Contém classes para as interfaces gráficas com os usuários (GUI). Através destas interfaces os usuários conseguem interagir com o XXXX, com o intuito de incluir, alterar e excluir produtos.

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

**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.

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 helpers 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), dentre outros.

* 1. Camada de Apresentação

Nesta camada, temos o pacote form que contém todos os arquivos relacionados à exibição de informações para usuário, o que engloba páginas JSP e HTML, imagens, javascript, dentre outros. Já o pacote de controle desta camada, contém as Actions do Struts responsáveis pela comunicação com as classes da camada de negócio. Estes pacotes podem ser vistos na Figura 5.3.



Figura 5.3: Camada de Apresentação

## Camada de Negócio

Nesta camada, temos o pacote controle 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 contém as informações sobre o Sistema XXX. O pacote controller contém as classes de controle do negócio. O pacote complementador contém classes que auxiliam na complementação de informações e o pacote filtro contém as classes que auxiliam na filtragem de informações.

A imagem 4.4 ilustra os pacotes descritos.



Figura 5.4: Camada de Negócios

* 1. Pacote Controller

A figura 5.5 ilustra as principais classes de controle.

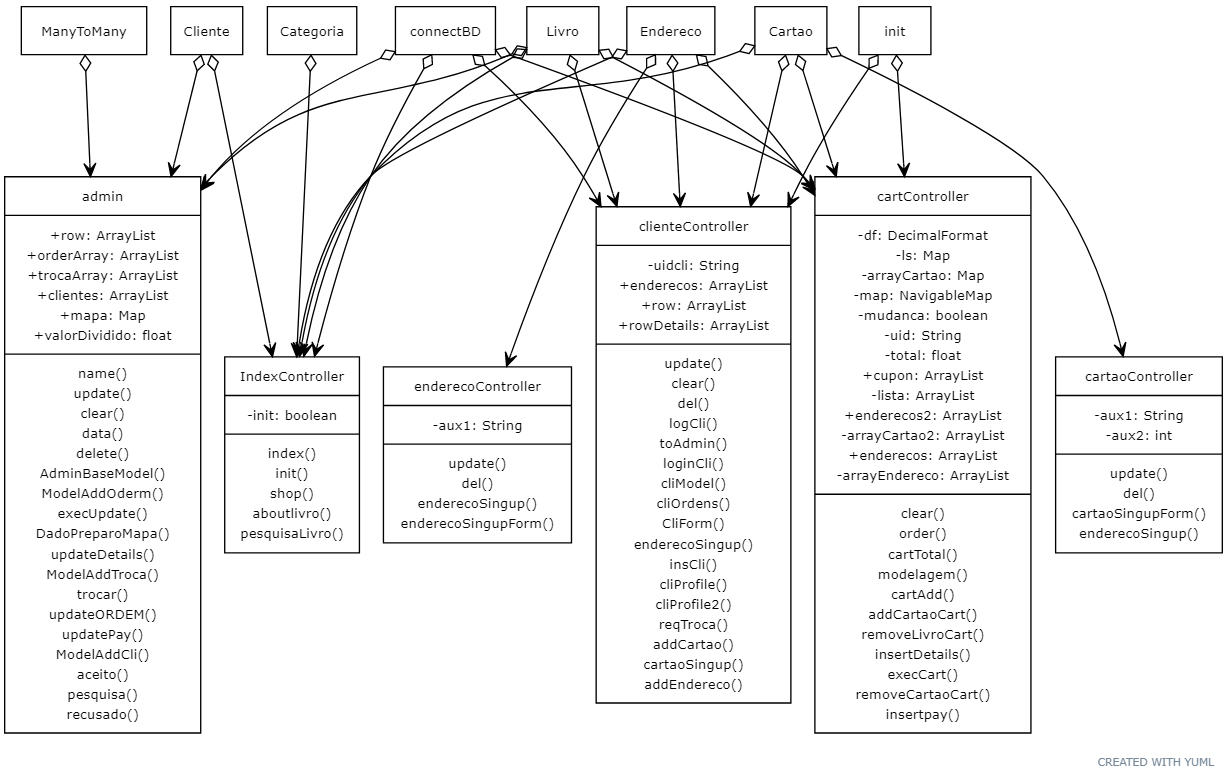


Figura 5.5: Classes de controle

* 1. Pacote Entidade

A figura 5.6. ilustra as principais classes do modelo.

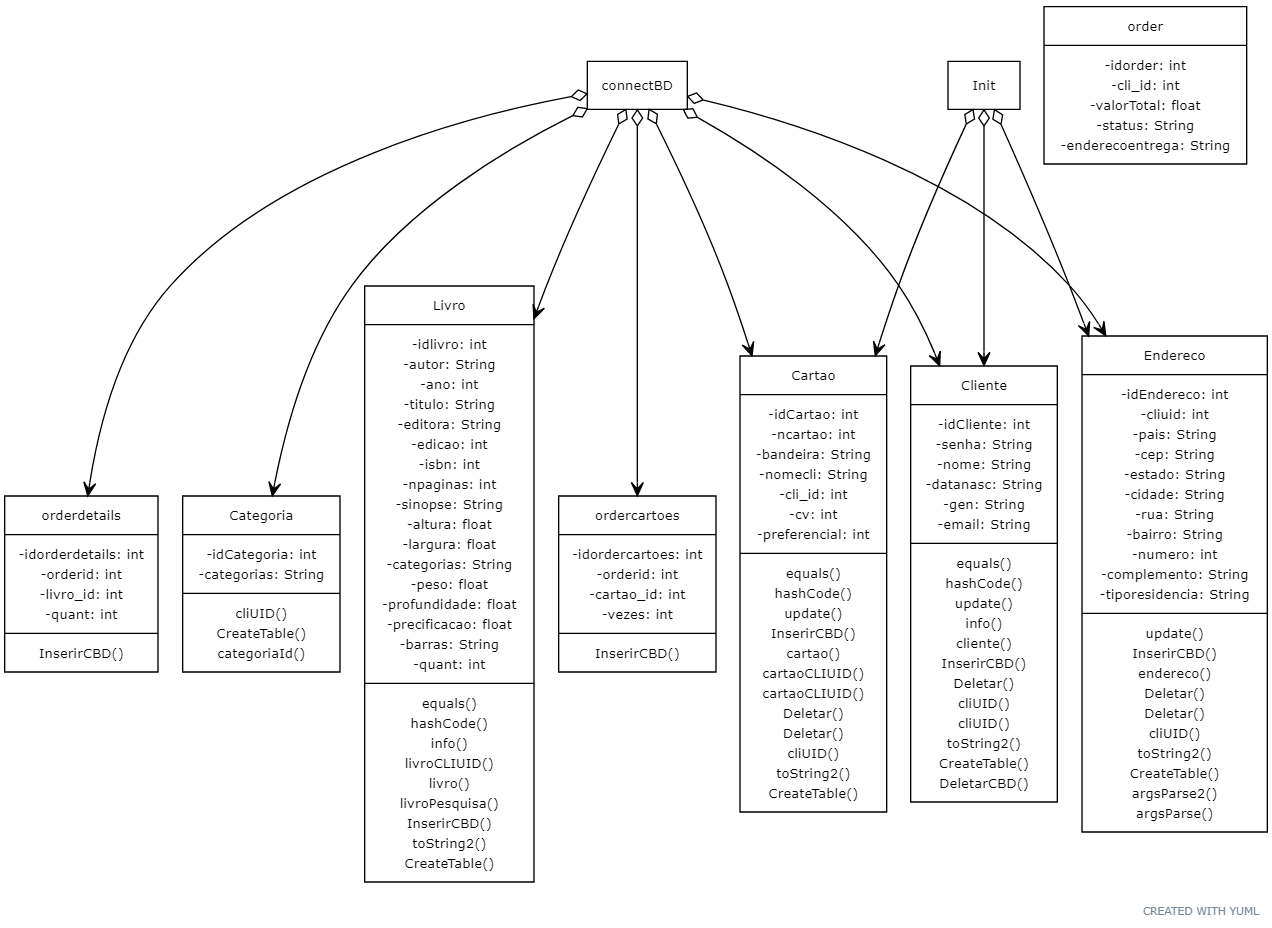
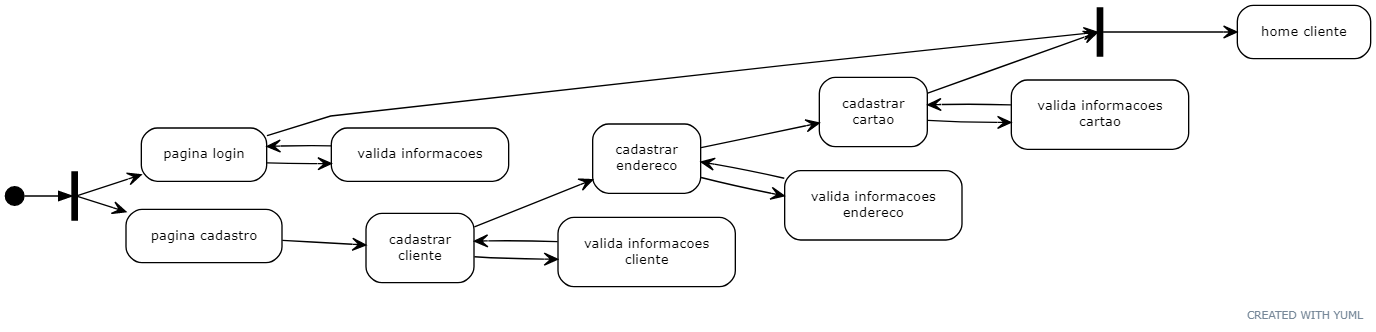


Figura 5.6: Classes do Modelo

## Realização dos Casos de Uso Significativos



## Visão de Implantação

Esta seção descreve as configurações da rede física (hardware) na qual o XXX 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 XXXX, e as respectivas interconexões (barramento, LAN, etc). A figura 6 ilustra o modelo de implantação para o XXXX.



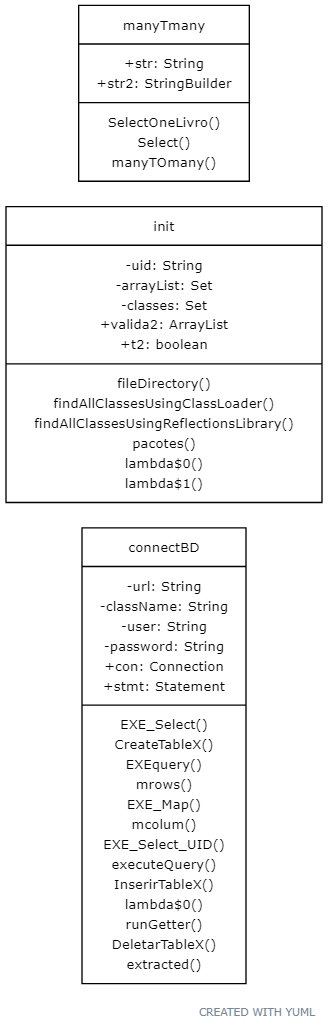
Figura 6: Visão de Implantação do XXXX

Na Figura 6 observa-se os seguintes nós físicos:

* **Web Client Application (WCA):** Aplicativos com interface de usuário via navegador, construídos com base no Framework Struts.
* **Servidor DB**: Nó que contém o BD Central do Sistema XXX.

## Visão de Dados

O mecanismo de persistência utilizado no sistema SISCAD utiliza-se o banco de dados Relacional Oracle juntamente com o framework para mapeamento objeto-relacional, Hibernate. O controle de transações adotado envolve a utilização do Spring Framework em conjunto com o Hibernate.



1. Referências

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

RUP. Rational Unified Process.