**Desenho Funcional**

**Vendas e Controle de Estoque**

Informações sobre o Documento

|  |  |
| --- | --- |
| **Nome do Documento** | **Tipo do Documento** |
| **Desenho\_Funcional\_It\_ContorlSales** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Data de Criação** | **Criado Por** | **Data de Aprovação** | **Aprovado por** |
| 18/07/2019 | Luis Otavio Oliveira |  |  |

Histórico de Revisões

|  |  |  |  |
| --- | --- | --- | --- |
| **Rev #** | **Data** | **Atualizado por** | **Descrição da Atualização** |
| **0** | 18/09/2019 | Luis Otavio Oliveira | Criação. |
| **1** |  |  | Revisão |
| **2** |  |  | Atualização |
| **3** |  |  | Aprovação |

Índice

[1. TERMINOLOGIA 4](#_Toc14386195)

[2. INTRODUÇÃO 5](#_Toc14386196)

[3. FUNCIONALIDADE 6](#_Toc14386197)

[3.1. Fluxo Principal 6](#_Toc14386198)

[3.2. Fluxos Alternativos 8](#_Toc14386199)

[3.2.1. acesso negado 8](#_Toc14386200)

[3.2.2. venda – pedido não selecionado 8](#_Toc14386201)

[3.2.3. venda – quantidade de produtos selecionados maior que a quantidade disponivel no estoque 8](#_Toc14386202)

[3.3. Wireframe 9](#_Toc14386203)

[4. BASE DE DADOS 13](#_Toc14386204)

[4.1. bd\_controle 13](#_Toc14386205)

[4.1.1. Configurações de base 13](#_Toc14386206)

[4.1.2. MODELO ER 14](#_Toc14386207)

[5. PREMISSAS 14](#_Toc14386208)

[6. DOCUMENTOS DE REFERÊNCIA 15](#_Toc14386209)

# TERMINOLOGIA

|  |  |
| --- | --- |
| **SIGLA** | **DEFINIÇÃO** |
| **FD** | Desenho Funcional |

# INTRODUÇÃO

Este documento descreve o comportamento funcional da funcionalidade ItControlSales.

# FUNCIONALIDADE

## Fluxo Principal

Fluxo se inicia quando o usuário acessa a funcionalidade “ItControlSales” com um login e senha;

Admin: admin/01 (ler e escrever)

User: user/01 (ler)

1. O sistema valida o tipo de usuário e atribui as permissões;
2. O sistema redireciona o usuário logado para a página de Dashboard.
3. Na página de dashboard é possível consultar informações do estoque e iniciar uma venda (conforme 1).
   1. O sistema exibe:
      1. Box “Vendas” (conforme Figura 1.1 - pesquisar e cadastra um pedido)
         1. Para efetuar uma venda é necessário criar um pedido de estoque do tipo saída.
         2. Selecionar:
            1. Filial (Obrigatório)
            2. Usuário (Obrigatório)
            3. Cliente (Obrigatório)
         3. Observação para entrega (Opcional)
         4. Regra para venda:
            1. Só é possível efetuar uma venda se houver um pedido do tipo saída no estoque.
         5. Cadastro dos itens do pedido:
            1. O usuário inicia o cadastro dos itens pesquisando por: código do produto, código de barras, descrição.
            2. O usuário digita a quantidade dos produtos desejados.

A quantidade deve ser superior a zero e inferior ou igual a quantidade de itens de produtos disponíveis para a filial selecionada.

Cada item possui:

Produto.

Status (ativo, cancelado ou processado).

Quantidade

Valor unitário

Valor Total

O valor total é calculado a partir da multiplicação da quantidade pelo valor unitário

* + - * 1. Todo novo item adicionado, possui o status de ativo e ao ser retirado do pedido, o status é atualizado para cancelado.
        2. Não pode existir produtos repetidos no mesmo pedido, nos status de ativo ou processado.
        3. A cada item adicionado no pedido, a quantidade total dos itens e o valor total do pedido é recalculado.
    1. box “Usuários” (conforme Figura 2 – pesquisa e cadastro de usuários )
       1. O usuário acessa a pagina de pesquisa e cadastro de usuários através do box “Usuários” na página de dashboard ou no Menu principal.
       2. O Usuário possui:
          1. Código (sequencial)
          2. Nome
          3. E-mail (identificados)
       3. Pesquisa de usuário
          1. A pesquisa do usuário pode ser feita por qualquer atributo pertencente a ele.
    2. box “Clientes” (conforme Figura 3– )
       1. O usuário acessa a página de pesquisa e cadastro de Clientes através do box “Clientes” na página de dashboard ou no Menu principal.
       2. O cliente possui:
          1. Código (sequencial)
          2. Nome
          3. E-mail (identificados)
       3. Pesquisa de cliente
          1. A pesquisa do cliente pode ser feita por qualquer atributo pertencente a ele.
    3. box “Filiais” (conforme Figura 4– filiais )
       1. O usuário acessa a página de pesquisa e cadastro de Filiais através do box “Filiais” na página de dashboard ou no Menu principal.
       2. A filial possui:
          1. Código (sequencial)
          2. Nome
          3. Cidade
          4. CNPJ
          5. Endereço
       3. Pesquisa de filiais
          1. A pesquisa de filial pode ser feita por qualquer atributo pertencente a ela.
    4. box “Produtos” (conforme Figura 5– produtos )
       1. O usuário acessa a página de pesquisa e cadastro de produtos através do box “produtos” na página de dashboard ou no Menu principal.
       2. O produto possui:
          1. Código (sequencial)
          2. Descrição
          3. Código de barras
          4. Valor
          5. Quantidade
          6. filial
       3. Pesquisa de produtos
          1. A pesquisa de produtos pode ser feita por qualquer atributo pertencente a ele.
    5. box “Pagamentos” (conforme Figura 6– pagamentos )
       1. O usuário acessa a página de pesquisa e cadastro de formas de pagamento através do box “pagamentos” na página de dashboard.
       2. A forma de Pagamento possui:
          1. Código (sequencial)
          2. Descrição
       3. Pesquisa de formas de pagamento
          1. A pesquisa pode ser feita por qualquer atributo pertencente a ele.
    6. box “Estoque” (conforme Figura 7– Estoque )
       1. O usuário acessa a página de pesquisa e cadastro de estoque através do box “pagamentos” na página de dashboard.
       2. O estoque possui:
          1. Código (sequencial)
          2. Código do produto
          3. Código da filial
          4. Quantidade mínima
          5. Quantidade
       3. Pesquisa de estoque
          1. A pesquisa pode ser feita por qualquer atributo pertencente a ele.
    7. box “Pedidos de Estoque” (conforme Figura 8– Pedidos de Estoque )
       1. O usuário acessa a página de pesquisa e cadastro de pedidos de estoque através do box “Pedidos de Estoque” na página de dashboard.
       2. O pedido de estoque possui:
          1. Código (sequencial)
          2. Tipo de receita
          3. Quantidade
       3. Pesquisa de pedidos de estoque
          1. A pesquisa pode ser feita por qualquer atributo pertencente a ele.

1. O usuário clica em “confirmar venda”;
2. Fim do fluxo.

## Fluxos Alternativos

### acesso negado

Passos:

A partir do passo 1 do fluxo principal;

1. O usuário tenta se logar com um usuário e senha invalida.
2. O sistema apresenta uma mensagem “Usuário e senha incorretas”
3. Fim do fluxo.

### venda – pedido não selecionado

Passo:

3.a.i.5 do fluxo principal;

1. O usuário cria um pedido.
2. O usuário não associa nenhum item ao pedido.
3. O botão de “confirmar venda” fica desabilitado.
4. Fim do fluxo.

### venda – quantidade de produtos selecionados maior que a quantidade disponivel no estoque

Passo:

3.a.i.5 do fluxo principal;

1. O usuário cria um pedido.
2. O usuário adiciona item ao pedido.
3. A quantidade de itens de produtos adicionados é maior que a quantidade em estoque.
4. O usuário clica em confirmar venda.
5. O sistema apresenta uma mensagem informando que “a quantidade de itens selecionados é maior que a quantidade de itens disponíveis”.
6. Fim do fluxo.

## Wireframe

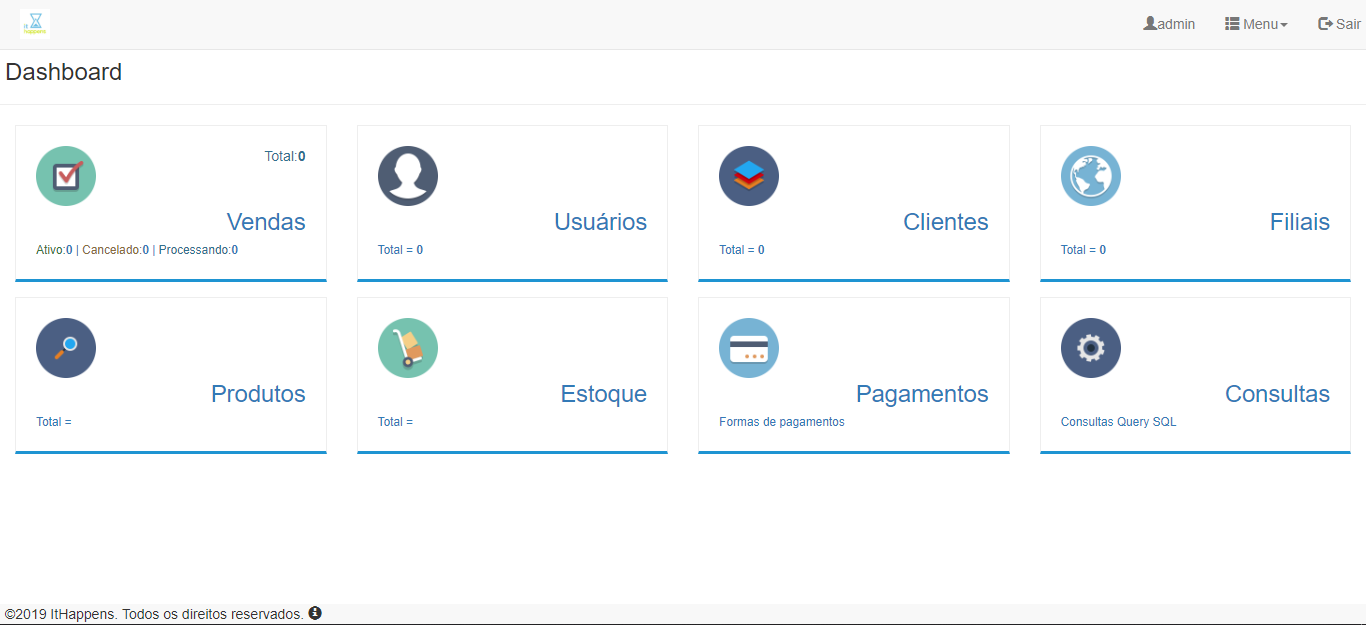
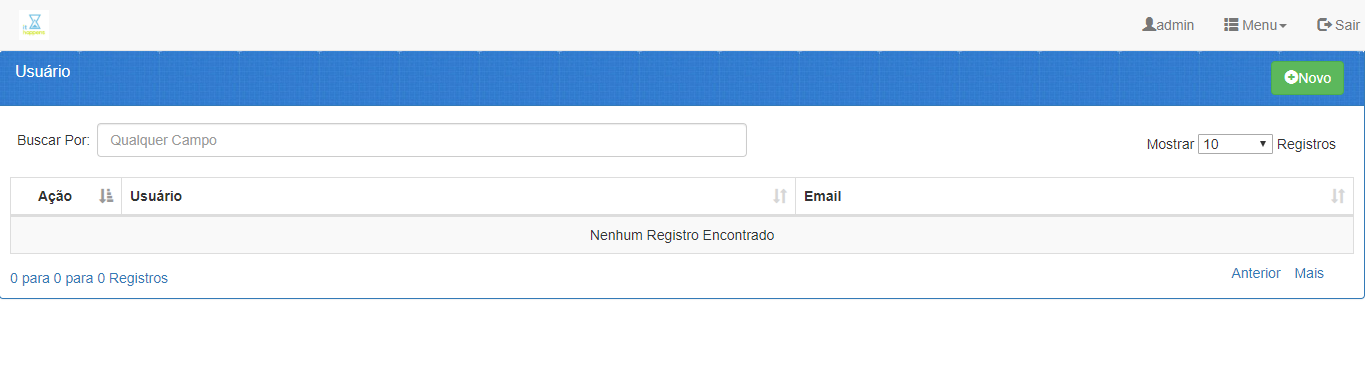


Figura - Dashboard



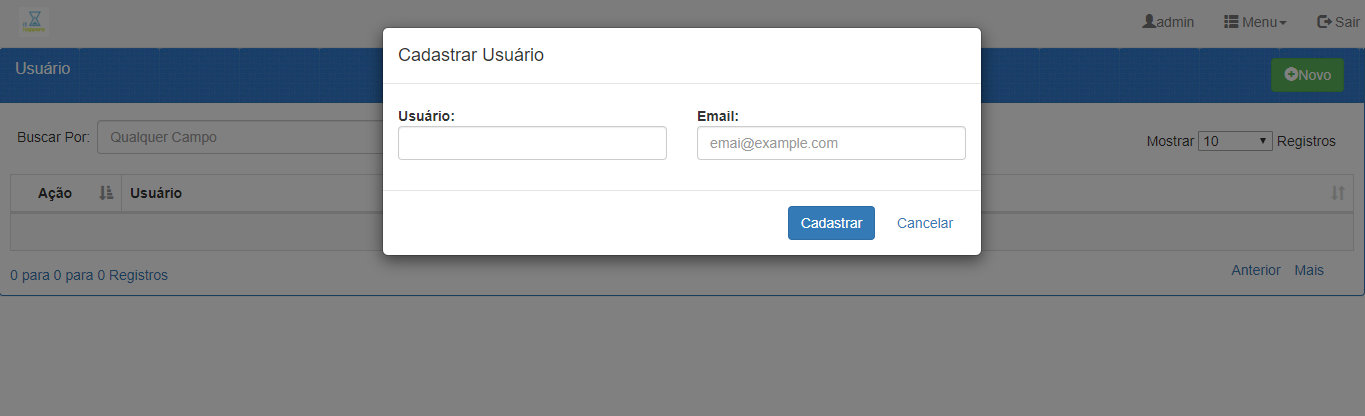


Figura 1.1 - Dashboard

# BASE DE DADOS

## bd\_controle

### Configurações de base

Banco H2 ou Postgres

Configuração de propriedades de base no arquivo: **application.properties**

spring.jpa.hibernate.ddl-auto=create-drop

spring.jpa.show-sql=true

#porta auxiliar

server.port=9096

Bd postgres:

#Configuração BD Postgres

spring.datasource.url=jdbc:postgresql://localhost/bd\_task

spring.datasource.username=postgres

spring.datasource.password=postgres

spring.jpa.hibernate.ddl-auto=update

#Bd H2 em memória:

spring.h2.console.enabled=true

spring.h2.console.path=/h2\_console

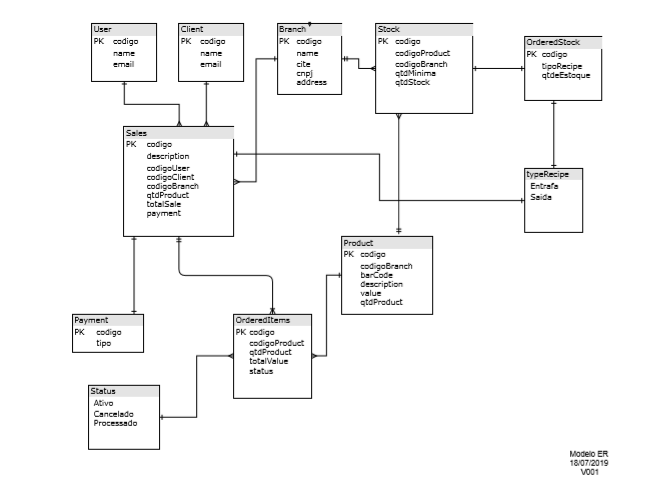
spring.datasource.url=jdbc:h2:file:~/h2/testdb

spring.datasource.username=sa

spring.datasource.password=sa

spring.datasource.driverClassName=org.h2.Driver

### MODELO ER



# PREMISSAS

1. O software foi desenvolvido na versão 1.0.
2. O software foi desenvolvido na plataforma web, utilizando o Spring boot (JAVA).
3. As regras de implementação seguem de acordo com as especificações do cliente;
4. A aprovação desse documento é por parte do cliente.

# DOCUMENTOS DE REFERÊNCIA

|  |  |
| --- | --- |
| Documento | Data |
| 1. prova-1202.pdf | 18/07/2019 |
|  |  |