**UNIVERSIDADE POSITIVO**

**GRADUAÇÃO - BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

JHONATAN PAULO NEVES

LUCAS EDUARDO LIMONI MARTINS

NÍCOLAS MANO TORRES

**DOCUMENTAÇÃO MÓDULO - VENDAS**

CURITIBA

JULHO/2018.

**1. INTRODUÇÃO.**

Este documento tem por objetivo introduzir, esclarecer e determinar como será e como funcionará o módulo de Vendas.

**1.1 Escopo**

O documento visa apresentar os requisitos funcionais, não-funcionais e casos de uso relacionados ao sistema e descreve detalhadamente cada um destes tópicos, não requerendo documentos adicionais.

**1.2 Público-alvo**

É destinado a Stakeholders, gerentes, analistas de negócios, analistas de sistemas, testadores, arquitetos e programadores

**2. Visão Geral**

O módulo de vendas é um web-service que tem por objetivo realizar uma ou várias vendas, vinculando um ou mais produtos a um cliente, disponibilizando as baixas desta venda para que o módulo de inventário possa realizar suas rotinas como: contagem de produtos e etc, bem como disponibilizar os dados que interessem a outros módulos, como o de Financeiro.

O módulo é uma aplicação Spring boot, onde todo o desenvolvimento é realizado em camadas, desta forma otimizamos a aplicação deixando com uma coesão muito grande, ou seja, funcionalidades que fazem apenas o que deveriam fazer.

O módulo de vendas irá disponibilizar os dados necessários para os outros módulos por rotas, pois é um web-service REST.

**2.1 Requisitos não funcionais:**

[RNF-001] **Arquitetura:** MVC em Camadas.

[RFN-002] **Tecnologia:** Java Spring Boot.

[RFN-003] **Banco de dados:** MySQL.

[RFN-004] **Versionamento:** GitLab e Bitbucket.

[RFN-005] **Web-Service:** REST.

[RFN-006] **Mensageria:** JMS

[RFN-007] **Servidor para Mensageria:** ActiveMQ.

**2.2 Requisitos Funcionais:**

[RF-001] Importação de dados do módulo de Pessoas, deve importar apenas dados relevantes para que seja possível realizar a venda.

[RF-002] Importação de dados do módulo de Produtos, deve importar apenas dados relevantes para que seja possível realizar a venda.

[RF-003] O sistema deverá permitir incluir um cliente no momento da venda, caso o cliente não esteja previamente cadastrado, deverá enviar uma mensagem assíncrona ao modulo de Pessoas, com todos os dados para que seja inserido na base.

[RF-004] O sistema deverá enviar uma mensagem assíncrona ao módulo de Inventário com a quantidade vendida do produto para que a base seja atualizada.

[RF-005] O sistema devera permitir inclusão de NFE-e.

**3.** **Especificação de Requisitos**:

3.1 [RNF-001] **Arquitetura**:

O módulo de vendas deverá ser desenvolvido em uma arquitetura em camadas, onde cada funcionalidade estará bem definida e isolada, representando unicamente o propósito onde ela foi pensada e construída.

O processo de desenvolvimento em camadas permite com que o sistema não possua muitas dependências de suas próprias funcionalidades, sendo altamente resistente a falhas, também nos permite uma maior segurança, onde cada camada conversa apenas com sua camada inferior, e não com a superior, assim dificultando possíveis vulnerabilidades.

Facilita a manutenabilidade do código e reaproveitamento do mesmo.

3.2 [RNF-002] **Tecnologia**:

O módulo de vendas deverá ser desenvolvido sobre a tecnologia **Spring Boot**, que trabalha em cima dos métodos de **Convenção sobre configuração**, facilitando o desenvolvimento.

3.3 [RNF-003] **Banco de dados**:

A base de dados para o módulo de vendas deverá ser estruturada e desenvolvida em MYSQL.

3.4 [RNF-004] **Versionamento**:

Todo o versionamento do módulo de Vendas deverá ser feito pelo Bitbucket:<https://bitbucket.org/JhoniDeveloper/vendas>

3.5 [RNF-005] **Web-Service**:

O módulo de vendas deverá ser desenvolvido com a dependência de web-service REST dentro do Spring-boot, para que seja possível realizar a integração com os demais módulos.

3.6 [RNF-006] **Mensageria**:

Deverá ser utilizado a tecnologia de **JMS** para mensageria pela alta compatibilidade com o **Spring boot**

**4.1 Caso de Uso: Adicionar item ao carrinho de compras**

Este caso de uso descreve o processo de adicionar um item a um carrinho de compras.

**Pré-condições**

Este use case pode iniciar somente se: O Cliente estiver autenticado no sistema.

**Sucesso - Pós-condições**

Permitir ao Cliente adicionar um item ao carrinho de compras.

**Falha – Pós-condições**

Apresentar mensagem de falha.

**Ator Primário**

Cliente

**Atores Secundários**

N/A

**Trigger**

Este use case deve começar quando o cliente acessar o site e clicar tentar adicionar um produto ao carrinho de compras.

**Fluxo Principal**

1) O Sistema apresenta a tela T1 para a listagem dos produtos.

2) Usuário seleciona o item desejado.

3) Sistema adiciona o item ao carrinho de compras. (A1)

4) Sistema redireciona usuário a tela T2 (carrinho de compras).

5) O Use Case é finalizado.

**Fluxo Alternativo**

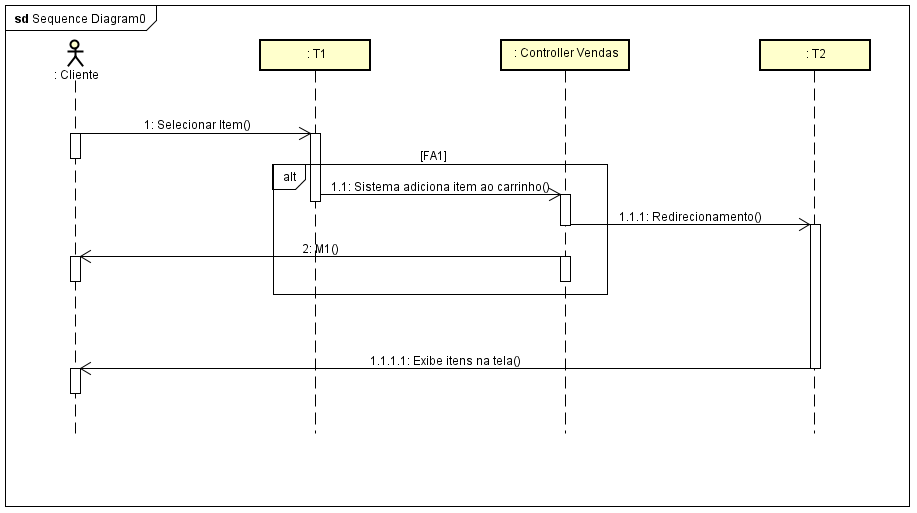
A1: Produto indisponível.

1. O Sistema apresenta a mensagem M1.

**Informações Relacionadas**

**M1**: Produto indisponível.

*4.1.1 Diagrama de Sequência*

******

**4.2 Caso de Uso: Realizar compra**

Este caso de uso descreve o processo de finalizar uma compra.

**Pré-condições**

Este use case pode iniciar somente se: O Cliente estiver autenticado no sistema e houver algo no carrinho de compras.

**Sucesso - Pós-condições**

Permitir ao Cliente realizar uma compra no sistema.

**Falha – Pós-condições**

Apresentar mensagem de falha.

**Ator Primário**

Cliente

**Atores Secundários**

N/A

**Trigger**

Este use case deve começar quando o cliente acessar seu carrinho e clicar tentar finalizar uma compra

**Fluxo Principal**

1) O Sistema apresenta a tela T1 de carrinho de compras.

2) Usuário clica em finalizar compra.

3) Sistema apresenta mensagem de confirmação M1.

4) Usuário clica em confirmar. (A1) (M4)

5) Sistema apresenta a tela T2, seleção de forma de pagamento.

6) Usuário seleciona forma de pagamento. (A2)

7) Sistema apresenta mensagem de confirmação M2.

8) Usuário clica em confirmar.

9) Sistema apresenta mensagem M3.

10) O Use Case é finalizado.

**Fluxo Alternativo**

A1: Usuário clica em cancelar.

1. O Sistema cancela a compra.

A2: Necessário cadastrar forma de pagamento.

2. O Sistema retorna para a tela de cadastro de forma de pagamento.

**Informações Relacionadas**

**M1**: Finalizar compra?

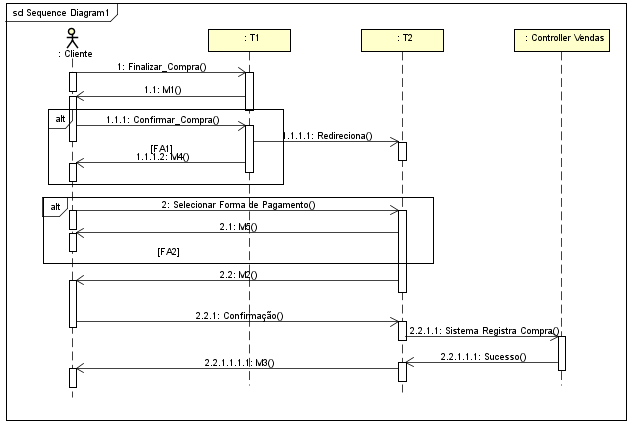
**M2**: Confirmar forma de pagamento?

**M3**: Compra finalizada.

**M4**: Compra cancelada.

**M5**: Você precisa cadastrar uma forma de pagamento

*4.2.1 Diagrama de Sequência*

**

***5. Informações de Integração.***

5.1 O Módulo de vendas deverá integrar com o módulo de produtos, inventário e pessoas.

5.2 Para o módulo de produtos, será necessário a integração para que seja possível identificar os produtos que serão vendidos.

5.3 Para o módulo de inventário, será necessário a integração para que seja possível a atualização da quantidade de produtos após a venda.

5.4 Para o modulo de pessoas será necessário a integração para que seja possível capturar o cliente previamente cadastrado e também para que seja possível atualizar a base de dados do módulo pessoas quando um cliente realizar uma compra e o mesmo não estiver previamente cadastrado.