2023-01

|  |  |
| --- | --- |
| **Professor:** | RADAMÉS PEREIRA |

Atributos de uma boa especificação de Projeto de Software:

* Clareza
* Não Ambígua
* Completa
* Simples
* Bem escrita

**UNIVERSIDADE COMUNITÁRIA REGIONAL DE CHAPECÓ - UNOCHAPECÓ**

**ÁREA DE CIÊNCIAS EXATAS E AMBIENTAIS**

**SISTEMAS DE INFORMAÇÃO**

Sistema Gourmet

Equipe: Guilherme Garghetti

Introdução

## Resumo do Projeto

Estudo de Caso: Sistema da Delícias Gourmet. Em uma indústria alimentícia chamada Delícias Gourmet, o controle de materiais é um processo-chave para garantir a qualidade e a eficiência dos produtos finais. Nesta história, vamos explorar como diferentes atores desempenham funções importantes em seus respectivos setores, interagindo e colaborando para manter o sistema de controle de materiais funcionando sem problemas. A história começa com João, o gerente de compras, que é responsável por adquirir os ingredientes e os materiais necessários para a produção dos alimentos. Ele trabalha em estreita colaboração com os fornecedores para garantir a qualidade dos produtos e negocia os melhores preços para reduzir os custos de produção. Maria, a supervisora de estoque, trabalha em conjunto com João. Ela é responsável por gerenciar o armazenamento dos materiais e garantir que todos os itens estejam devidamente organizados e estocados. Quando os ingredientes e materiais chegam ao armazém, Maria confere e verifica se tudo está em conformidade com os pedidos feitos por João. Dentro da fábrica, Pedro, o coordenador de produção, coordena a utilização dos ingredientes e materiais no processo de fabricação. Ele colabora com Maria para garantir que os itens necessários sejam entregues no momento certo e na quantidade adequada. Pedro também trabalha com os líderes de cada linha de produção para planejar e otimizar a utilização dos materiais. Na área de controle de qualidade, Ana, a analista de qualidade, monitora o processo de produção, verificando se todos os ingredientes e materiais estão dentro dos padrões exigidos. Ela também é responsável por conduzir inspeções regulares e garantir que a produção esteja em conformidade com as normas de segurança alimentar e regulamentações governamentais. Por fim, a história se completa com Carlos, o responsável pelas vendas e distribuição dos produtos. Ele coordena a entrega dos produtos acabados aos clientes e mantém um registro de todas as vendas realizadas. Carlos também fornece informações sobre a demanda do mercado e as preferências dos clientes, o que ajuda João na seleção de fornecedores e ingredientes para futuras compras. Esta história ilustra como os diferentes atores em uma indústria alimentícia colaboram e desempenham funções cruciais para manter o sistema manual de controle de materiais funcionando de forma eficiente. A interdependência entre as atividades de João, Maria, Pedro, Ana e Carlos é fundamental para garantir que a Delícias Gourmet; mantenha sua reputação de qualidade e satisfação do cliente.

## Plataforma de desenvolvimento

VSCode, PHP, Python

## Plataforma de operação

considerar que o sistema seja desenvolvido para operar em uma plataforma como web, desktop ou mobile.

## Definições e siglas

Descreve-se aqui a definição de todas as siglas, abreviações e termos usados.

## Perspectiva do produto

### Modos de operação

### Identificam-se aqui os modos requeridos de operação, tais como: Back-End-Front-End, Móvel, Stand-Alone, ...

### Requisitos de adaptação ao ambiente

Definem-se aqui possíveis requisitos de adaptação do produto aos ambientes particulares onde ele será implantado. Por exemplo, parâmetros e métodos de configuração requeridos para ambientes específicos devem ser descritos aqui.

|  |  |  |
| --- | --- | --- |
| Número de ordem | Requisito | Detalhes |
| 1 | Configuração de ticket de venda e da Nota Fiscal Eletrônica | Configuração dos campos de formulário com interface responsiva. |

## Funções do produto

Funções Básicas do sistema: R1.1 – Controle de materiais: Responsável por adquirir os ingredientes e materiais necessários para a produção de alimentos, garantindo a qualidade dos produtos em estreita colaboração com os fornecedores. R1.2 – Garantia de qualidade e eficiência dos produtos finais: Responsável por monitorar o processo de produção e garantir que todos os ingredientes e materiais estejam dentro dos padrões exigidos. R1.3 – Compra, armazenamento e controle de materiais: Responsável por comprar os materiais e ingredientes e gerenciar o armazenamento e controle deles. R1.4 – Produzir Alimentos: Responsável por coordenar a utilização dos ingredientes e materiais no processo de fabricação dos alimentos. R1.5 – Controlar a produção com otimização do uso de materiais: Responsável por trabalhar com os líderes de cada linha de produção para planejar e otimizar a utilização dos materiais. R1.6 – Controlar entrega de produtos e preferência de clientes: Responsável por coordenar a entrega dos produtos acabados aos clientes, manter um registro de todas as vendas realizadas e fornecer informações sobre a demanda do mercado e as preferências dos clientes. R1.7 - Negociação com fornecedores: Responsável por negociar os melhores preços com fornecedores para reduzir os custos de produção. R1.8 - Conformidade com normas de segurança alimentar e regulamentações governamentais: Responsável por garantir que a produção esteja em conformidade com as normas de segurança alimentar e regulamentações governamentais. R1.9 - Planejamento de compras futuras: Responsável por planejar futuras compras de ingredientes e materiais com base nas informações sobre a demanda do mercado e as preferências dos clientes.

## Características dos usuários

podemos inferir que os usuários incluem clientes, gerente de compras, supervisor de qualidade, dono, caixa, cozinheiro e garçom.

## Restrições

Descrevem-se aqui aspectos técnicos e gerenciais que possam limitar as opções dos desenvolvedores, tais como restrições legais.

## Hipóteses de trabalho

Descrevem-se aqui fatores que não são restrições limitativas do desempenho, como na subseção anterior, mas fatores cuja alteração requer modificações na ER, como, por exemplo, versão a ser utilizada do ambiente operacional ou plataforma de desenvolvimento.

# Requisitos específicos

## Interfaces externas

### Visão geral

Pode-se incluir ainda: um certo caso de uso e seus relacionamentos, todos os casos de uso para um certo ator.

### Requisitos para interfaces gráficas de usuário

Sugere-se, no caso de interfaces gráficas, a inclusão dos seguintes elementos:

* Um esboço do layout gráfico sugerido para a interface;
* Uma descrição dos relacionamentos com outras interfaces;
* Um diagrama de estados/atividades, caso necessário para melhor entender-se o comportamento requerido da interface;
* Uma lista dos campos de dados da interface;
* Uma lista dos comandos da interface;
* BPM;

## Requisitos funcionais

### Diagramas de casos de uso

@startuml

left to right direction

actor Cliente as C

actor "Gerente de Compras" as GComp

actor "Supervisor de Qualidade" as Qual

actor Dono as D

actor Caixa as Caixa

actor Cozinheiro as Coz

actor Garçom as Garcom

actor Almoxarifado as Almox

actor Financeiro as Fin

rectangle Restaurante {

(Fazer Reserva) <-- C

(Cancelar Reserva) <-- C

(Fazer Pedido) <-- C

(Modificar Pedido) <-- C

(Cancelar Pedido) <-- C

(Receber Pedido) <-- Garcom

(Enviar Pedido à Cozinha) <-- Garcom

(Preparar Pedido) <-- Coz

(Finalizar Pedido) <-- Coz

(Entregar Pedido) <-- Garcom

(Pagar Conta) <-- Caixa

(Gerenciar Estoque) <-- GComp

(Gerenciar Funcionários) <-- D

(Gerar Relatórios) <-- D

(Gerenciar Reservas) <-- D

(Supervisionar Qualidade) <-- Qual

(Receber Mercadorias) <-- Almox

(Armazenar Mercadorias) <-- Almox

(Solicitar Mercadorias) <-- Almox

(Controlar Inventário) <-- Almox

(Gerenciar Finanças) <-- Fin

}

C -- (Fazer Reserva)

C -- (Cancelar Reserva)

C -- (Fazer Pedido)

C -- (Modificar Pedido)

C -- (Cancelar Pedido)

Garcom -- (Receber Pedido)

Garcom -- (Enviar Pedido à Cozinha)

Coz -- (Preparar Pedido)

Coz -- (Finalizar Pedido)

Garcom -- (Entregar Pedido)

Caixa -- (Pagar Conta)

GComp -- (Gerenciar Estoque)

D -- (Gerenciar Funcionários)

D -- (Gerar Relatórios)

D -- (Gerenciar Reservas)

Qual -- (Supervisionar Qualidade)

Almox -- (Receber Mercadorias)

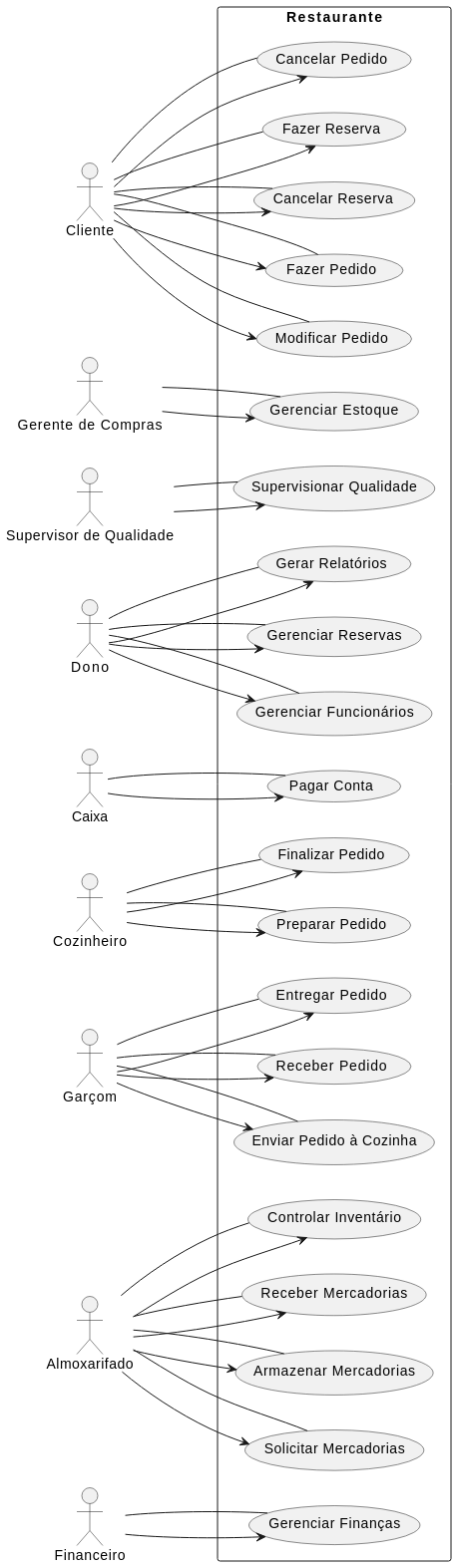
Almox -- (Armazenar Mercadorias)

Almox -- (Solicitar Mercadorias)

Almox -- (Controlar Inventário)

Fin -- (Gerenciar Finanças)

@enduml



UISITOS - ER

### Fluxos dos casos de uso

* Pré-condições para a realização do caso de uso;
* Fluxo principal do caso de uso (sucesso), descrito na forma de uma sequência de passos;
* Fluxos alternativos do caso de uso;
* Descrições mais formais, como diagramas de estado ou de atividade, se a complexidade do caso de uso exigir;
* Observações.

Pré-condições:

* O cliente precisa estar registrado no sistema.
* O produto desejado precisa estar disponível no estoque.

Fluxo principal do caso de uso:

1. O cliente seleciona os produtos desejados.
2. O sistema verifica a disponibilidade dos produtos no estoque.
3. Se os produtos estiverem disponíveis, o cliente prossegue.
4. O cliente informa a quantidade desejada de cada produto.
5. O sistema calcula o valor total da compra.
6. O cliente efetua o pagamento.
7. O sistema registra a compra.
8. O sistema atualiza o estoque, reduzindo a quantidade dos produtos comprados.
9. A compra é concluída com sucesso.

Fluxos alternativos do caso de uso:

* No passo 3, se os produtos não estiverem disponíveis, o sistema exibe uma mensagem informando a falta de estoque e o caso de uso é encerrado.
* No passo 6, se o cliente não efetuar o pagamento, o sistema cancela a compra e o caso de uso é encerrado.
* No passo 7, se ocorrer algum erro no registro da compra, o sistema exibe uma mensagem de erro e o caso de uso é encerrado.
* No passo 8, se ocorrer algum erro na atualização do estoque, o sistema exibe uma mensagem de erro e o caso de uso é encerrado.

Observações:

* O caso de uso assume que o sistema possui os mecanismos adequados para realizar o cálculo do valor total da compra, processar o pagamento e registrar as informações necessárias.

Caso de uso: Gerenciar Estoque

Pré-condições:

* O usuário precisa ter permissões de acesso ao módulo de gerenciamento de estoque.
* O sistema precisa estar em funcionamento.

Fluxo principal do caso de uso:

1. O usuário acessa o módulo de gerenciamento de estoque.
2. O sistema exibe a lista de produtos disponíveis no estoque.
3. O usuário seleciona um produto para gerenciar.
4. O sistema exibe as informações detalhadas do produto, como quantidade em estoque, data de entrada, fornecedor, etc.
5. O usuário realiza as ações desejadas, como adicionar mercadorias ao estoque, registrar saída de produtos, atualizar informações, etc.
6. O sistema atualiza as informações do estoque de acordo com as ações realizadas.
7. O usuário conclui as operações de gerenciamento de estoque.

Fluxos alternativos do caso de uso:

* No passo 2, se não houver produtos disponíveis no estoque, o sistema exibe uma mensagem informando a falta de produtos.
* No passo 3, se o usuário não selecionar nenhum produto, o sistema mantém a exibição das informações gerais do estoque.
* No passo 5, o usuário pode realizar diferentes ações de acordo com suas permissões e necessidades específicas.

Observações:

* O caso de uso pressupõe que o sistema possui mecanismos para realizar as ações de gerenciamento de estoque, como adicionar mercadorias, registrar saída de produtos, atualizar informações, etc.
* As permissões de acesso ao módulo de gerenciamento de estoque podem variar de acordo com o perfil do usuário (exemplo: gerente, supervisor, etc.).
* O caso de uso pode ser mais complexo dependendo das funcionalidades específicas do sistema de gerenciamento de estoque utilizado.

Caso 1: Mecânica:

@startuml

skinparam actor {

actor "Recepcionista" as Recepcionista

actor "Mecânico" as Mecanico

actor "Cliente" as Cliente

actor "Financeiro" as Financeiro

actor "Gerente" as Gerente

actor "Dono" as Dono

actor "Almoxarifado" as Almoxarifado

}

rectangle "Oficina Mecânica" {

Recepcionista -- Mecanico : Solicita Serviço

Mecanico -- Recepcionista : Conclui Serviço

Recepcionista -- Cliente : Atende Cliente

Cliente -- Recepcionista : Informa Problema

Cliente -- Financeiro : Paga Serviço

Financeiro -- Cliente : Emissão de Fatura

Gerente -- Recepcionista : Supervisiona

Gerente -- Mecanico : Acompanha Serviço

Dono -- Gerente : Gerencia Oficina

Almoxarifado -- Mecanico : Fornece Peças

note left of Recepcionista: Realiza o atendimento\ne agendamento dos clientes

note right of Mecanico: Executa os reparos e manutenções\nnos veículos dos clientes

note bottom of Financeiro: Realiza a cobrança e\ncontrole financeiro da oficina

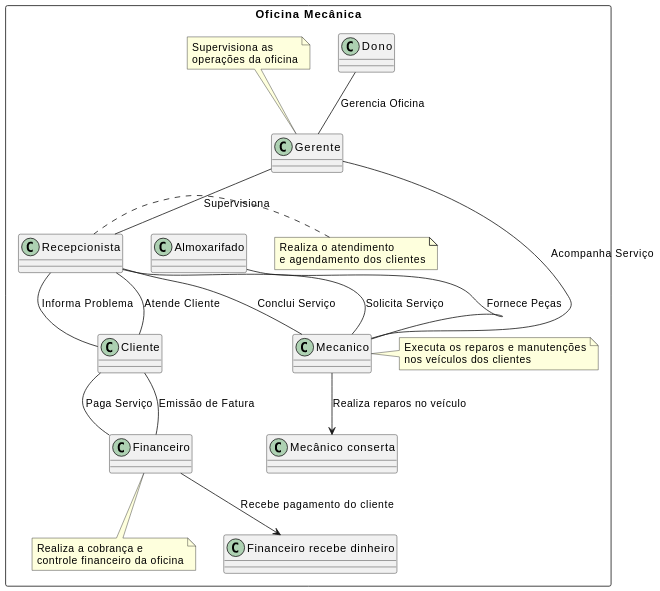
note top of Gerente: Supervisiona as\noperações da oficina

Mecanico --> "Mecânico conserta" : Realiza reparos no veículo

Financeiro --> "Financeiro recebe dinheiro" : Recebe pagamento do cliente

}

@enduml



Caso 2: Mercado:

@startuml

left to right direction

actor Cliente as C

actor Caixa as Caixa

actor Estoque as Estoque

actor Gerente as Gerente

rectangle "Mercado" {

rectangle "Área de Vendas" {

(Realizar Compra) <-- C

(Efetuar Pagamento) <-- Caixa

(Receber Pagamento) <-- Caixa

(Solicitar Reposição de Estoque) <-- Caixa

(Verificar Disponibilidade de Produtos) <-- C

(Empacotar Compras) <-- Caixa

}

rectangle "Área de Estoque" {

(Gerenciar Estoque) <-- Gerente

(Receber Mercadorias) <-- Estoque

(Armazenar Mercadorias) <-- Estoque

(Registrar Saída de Produtos) <-- Estoque

(Verificar Níveis de Estoque) <-- Gerente

}

C -- (Realizar Compra)

C -- (Verificar Disponibilidade de Produtos)

Caixa -- (Efetuar Pagamento)

Caixa -- (Receber Pagamento)

Caixa -- (Solicitar Reposição de Estoque)

Caixa -- (Empacotar Compras)

Gerente -- (Gerenciar Estoque)

Estoque -- (Receber Mercadorias)

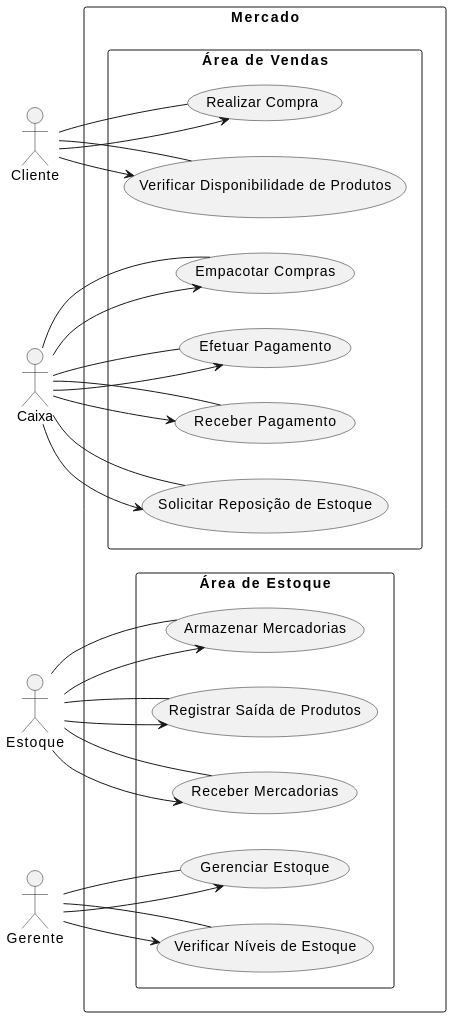
Estoque -- (Armazenar Mercadorias)

Estoque -- (Registrar Saída de Produtos)

Gerente -- (Verificar Níveis de Estoque)

}

@enduml



## Requisitos não-funcionais

### Requisitos de desempenho

Requisitos de desempenho devem ser especificados de forma quantitativa e mensurável.

### Requisitos de dados persistentes

Descrevem-se aqui estruturas lógicas de dados persistentes (que mantém seu valor após a execução do programa) que sejam usadas pelo produto. Cada estrutura de dados pode ser, por exemplo, um arquivo convencional ou uma tabela em um banco de dados.

INCLUIR AQUI O MODELO DE BANCO DE DADOS

### Restrições ao desenho

Restrições de projeto impostas por padrões externos, com influência da legislação..

### Atributos de Qualidade

Indica os atributos de qualidade, seguindo as características e subcaracterísticas recomendadas pela norma [ISO-9126](https://drive.google.com/open?id=14-OfxrvT5pD4sblFt234r8NPBNwt2HmG).

# Análise de UCP

As tabelas de escopo de valor do produto e tempo de desenvolvimento com Use Case Points - UCP.

Referências:

*IEEE Std. 830 – 1993. IEEE Recommended Practice for Software Requirements Specifications.*

*IEEE ISO/IEC/IEEE 29148 – 2011. IEEE Systems and software engineering — Life cycle processes — Requirements engineering*

**OBSERVAÇÃO: Os itens deste modelo de especificação, recomendado pela IEEE, poderão ser complementados com novos itens caso sejam justificáveis.**