Breve Descritivo do Sistema

**Sistema de Gerenciamento de Pedidos de Sorveteria**

Objetivo

O sistema de Gerenciamento de Pedidos de Sorveteria tem como o principal objetivo gerenciar todos os atendimentos de mesas. O sistema deve prever a realização de pedidos por mesa. As mesas e os produtos do cardápio deverão ser previamente cadastrados para agilizar as vendas e evitar eventuais erros.

Quando o pedido for feito pelo cliente, o garçom anota o pedido do cliente e o número da mesa em que o mesmo se encontra em uma comanda associada a mesa, onde o sistema enviara a comanda para a cozinha. Assim que o pedido ficar pronto, a cozinha informa o sistema que o pedido está pronto, e o mesmo notificara o garçom responsável pelo pedido para que o mesmo leve os produtos solicitados à mesa.

Regras de negocio

R.N 1 Todo cliente para fazer o pedido tem que estar cadastrado em uma mesa.

RN 2 Quando o pedido estiver pronto a cozinha obrigatoriamente avisa o garçom para servir o cliente.

RN3 quando o cliente fechar a mesa ele deve dirigir-se ao caixa para efetuar o pagamento da comanda.

Mini Especificação

Cadastro de pedidos

1. O funcionário abre a tela de pedido.
2. O funcionário informa o seu ID e o número da mesa.
3. O sistema valida o número da mesa e o ID do funcionário.
4. Caso o id ou mesa incorreta o sistema mostra a mensagem” id ou mesa invalida”
5. O funcionário informa nome do cliente e número de pedido.
6. Caso o número de pedido já existir o sistema mostra a mensagem “número de pedido já existente”
7. O sistema registra o nome do cliente e o número do pedido.
8. O funcionário seleciona os produtos.
9. O sistema valida os produtos.
10. Caso o produto não tiver em estoque o sistema manda uma mensagem “ produto não disponível “.
11. O sistema envia para a cozinha os dados da comanda.

Cadastro de Funcionário (inserir)

1. O Gerente abre a tela “cadastro de Funcionário”.
2. O Gerente informa os dados do Funcionário (nome, telefone, endereço).
3. O Gerente clica no botão salvar.
4. O sistema mostra a mensagem “Funcionário cadastrado com sucesso
5. Caso o gerente não coloque algum dado ou dados incorretos o sistema mostra uma mensagem “favor preencher corretamente os dados do Funcionário”

Cadastro de Funcionário (alterar)

1. O Gerente abre a tela “cadastro de Funcionário”.
2. O gerente seleciona o nome do Funcionário.
3. O Gerente clica em editar.
4. O Gerente faz as alterações de Cadastro do Funcionário.
5. O Gerente clica no botão Salvar.
6. O sistema mostra uma mensagem ” Cadastro de Funcionário alterado com sucesso”.
7. Caso o gerente não selecione o funcionário a ser alterado e clicar em editar o sistema mostra a mensagem” selecione o funcionário a ser alterado”.

Cadastro de Funcionário (Excluir)

1. O Gerente abre a tela “cadastro de Funcionário”.
2. O gerente seleciona o nome do Funcionário.
3. O Gerente clica no botão “excluir”.
4. O sistema mostra a mensagem “ Funcionário excluído com sucesso “,
5. Caso o Gerente não selecionar o funcionário e clicar excluir o sistema mostra uma mensagem ”Selecione o funcionário a ser excluído”.

Cadastro de Produtos (inserir)

1. O Gerente abre a tela “cadastro de Produtos”.
2. O Gerente informa os dados do Produto (nome, qtd).
3. O Gerente clica no botão salvar.
4. O sistema mostra a mensagem “Produto cadastrado com sucesso
5. Caso o gerente não coloque algum dado ou dados incorretos o sistema mostra uma mensagem “favor preencher corretamente os dados do Produto”

Cadastro de Produto (alterar)

1. O Gerente abre a tela “cadastro de Produto”.
2. O gerente seleciona o nome do Produto.
3. O Gerente clica em editar.
4. O Gerente faz as alterações no Cadastro de Produtos.
5. O Gerente clica no botão Salvar.
6. O sistema mostra uma mensagem ” Cadastro de Produtos alterado com sucesso”.
7. Caso o gerente não selecione o Produto a ser alterado, e clicar em editar o sistema mostra a mensagem” selecione o Produto a ser alterado”

Cadastro de Produto (Excluir)

1. O Gerente abre a tela “cadastro de Produto”.
2. O gerente seleciona o nome do Produto.
3. O Gerente clica no botão “excluir”.
4. O sistema mostra a mensagem “ Produto excluído com sucesso “,
5. Caso o Gerente não selecionar o Produto e clicar em excluir o sistema mostra uma mensagem ”Selecione o Produto a ser excluído”.

Diagrama de Casos de uso

Objetivo

O Diagrama de *Casos de Uso* tem o objetivo de auxiliar a comunicação entre os analistas e o cliente.

Um diagrama de Caso de Uso descreve um cenário que mostra as funcionalidades do sistema do ponto de vista do usuário.

O cliente deve ver no diagrama de Casos de Uso as principais funcionalidades de seu sistema.

Notação

O diagrama de Caso de Uso é representado por:

* atores;
* casos de uso;
* relacionamentos entre estes elementos.

Estes relacionamentos podem ser:

* associações entre atores e casos de uso;
* generalizações entre os atores;
* generalizações, *extends* e *includes* entre os casos de uso.

*casos de uso* podem opcionalmente estar envolvidos por um retângulo que representa os limites do sistema.

Em maiores detalhes

* ***Atores***

|  |  |
| --- | --- |
|  | Um ator é representado por um boneco e um rótulo com o nome do ator. Um ator é um usuário do sistema, que pode ser um usuário humano ou um outro sistema computacional. |

* ***Caso de uso***

|  |  |
| --- | --- |
| http://www.dsc.ufcg.edu.br/~sampaio/cursos/2007.1/Graduacao/SI-II/Uml/diagramas/usecases/images/image003.png | Um *caso de uso* é representado por uma elipse e um rótulo com o nome do *caso de uso*. Um *caso de uso*define uma grande função do sistema. A implicação é que uma função pode ser estruturada em outras funções e, portanto, um *caso de uso* pode ser estruturado. |

* ***Relacionamentos***

o        Ajudam a descrever *casos de uso*

* + Entre um ator e um *caso de uso*
    - Associação

|  |  |
| --- | --- |
| http://www.dsc.ufcg.edu.br/~sampaio/cursos/2007.1/Graduacao/SI-II/Uml/diagramas/usecases/images/image005.png | Define uma funcionalidade do sistema do ponto de vista do usuário. |

* + Entre atores
    - Generalização

|  |  |
| --- | --- |
| http://www.dsc.ufcg.edu.br/~sampaio/cursos/2007.1/Graduacao/SI-II/Uml/diagramas/usecases/images/image007.png | - Os *casos de uso* de B são também *casos de uso* de A  - A tem seus próprios *casos de uso* |

* + Entre *casos de uso*
    - *Include*

Um relacionamento *include* de um *caso de uso* A para um *caso de uso*B indica que B é essencial para o comportamento de A. Pode ser dito também que B *is\_part\_of* A.

* + - *Extend*

Um relacionamento *extend* de um *caso de uso* B para um *caso de uso*A indica que o *caso de uso* B pode ser acrescentado para descrever o comportamento de A (não é essencial). A extensão é inserida em um ponto de extensão do *caso de uso* A.

Ponto de extensão em um *caso de uso* é uma indicação de que outros *casos de uso* poderão ser adicionados a ele. Quando o caso de uso for invocado, ele verificará se suas extensões devem ou não serem invocadas.

Você entendeu?! Provavelmente, não. É que *extend* é unanimemente considerado um conceito obscuro.

Vamos a novas explicações.

Quando se especifica B *extends* A, a semântica é:

·        Dois *casos de uso* são definidos: **A** e **A *extended by* B**;

·        B é uma variação de A. Contém eventos adicionais, para certas condições;

·        Tem que ser especificado onde B é inserido em A.

* + - Generalização ou Especialização (é\_um)

*caso de uso* B é\_um *caso de uso* A (A é uma generalização de B, ou B é uma especialização de A).

Um relacionamento entre um caso de uso genérico para um mais específico, que herda todas as características de seu pai.

·        Sistema

* Limites do sistema: representado por um retângulo envolvendo os *casos de uso* que compõem o sistema.
* Nome do sistema: Localizado dentro do retângulo.

Use Case Da Sorveteria

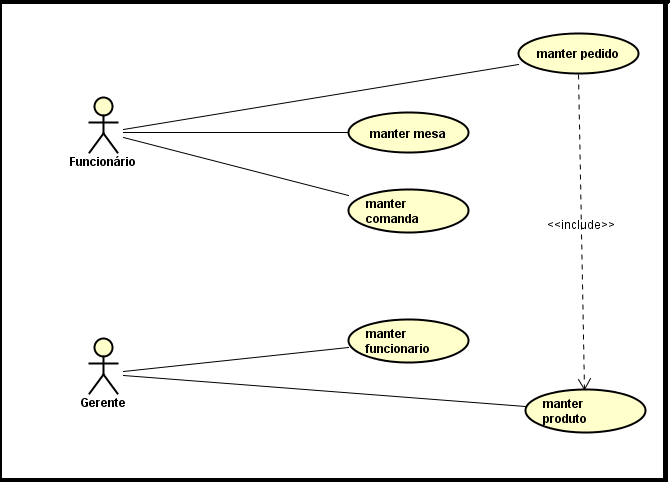


Diagrama de Classes

Objetivo

Descrever os vários tipos de objetos no sistema e o relacionamento entre eles.

Perspectivas

Um diagrama de classes pode oferecer três perspectivas, cada uma para um tipo de observador diferente. São elas:

* Conceitual ([exemplo](http://www.dsc.ufcg.edu.br/~jacques/cursos/map/html/uml/diagramas/classes/images/conceitual_exemplo_banco.GIF))
  + Representa os conceitos do domínio em estudo.
  + Perspectiva destinada ao cliente.
* Especificação ([exemplo](http://www.dsc.ufcg.edu.br/~jacques/cursos/map/html/uml/diagramas/classes/images/especificacao_exemplo_banco.GIF))
  + Tem foco nas principais interfaces da arquitetura, nos principais métodos, e não como eles irão ser implementados.
  + Perspectiva destinada as pessoas que não precisam saber detalhes de desenvolvimento, tais como gerentes de projeto.
* Implementação - *a mais utilizada de todas* ([exemplo](http://www.dsc.ufcg.edu.br/~jacques/cursos/map/html/uml/diagramas/classes/images/implementacao_exemplo_banco.GIF))
  + Aborda vários detalhes de implementação, tais como navegabilidade, *tipo* dos atributos, etc.
  + Perspectiva destinada ao time de desenvolvimento.

Diagrama de Classes da Sorveteria

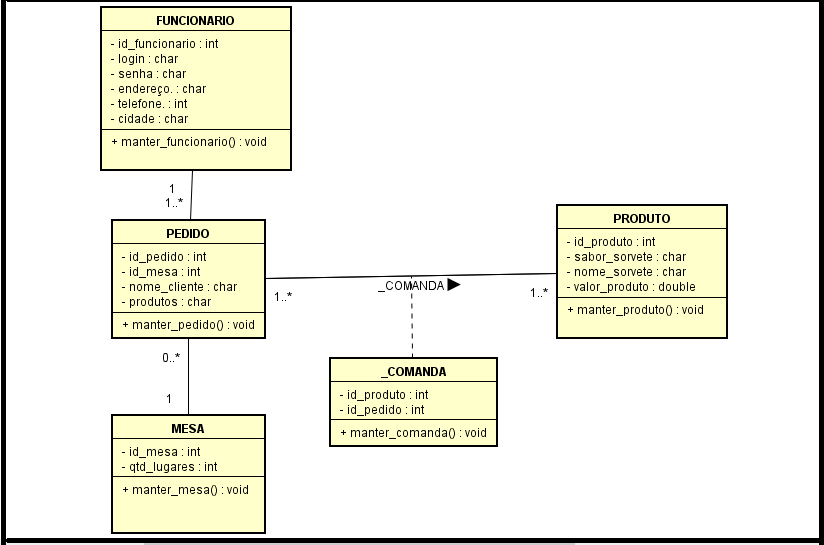


Diagrama Entidade Relacionamento (DER)

O**Diagrama Entidade-Relacionamento** descreve toda estrutura lógica do banco de dados. É possível construí-lo a partir de um MER (Modelo de Entidade e Relacionamento), identificando assim a partir de um conceito do mundo real como os dados serão armazenados de fato.

Diagrama Entidade Relacionamento da Sorveteria.

