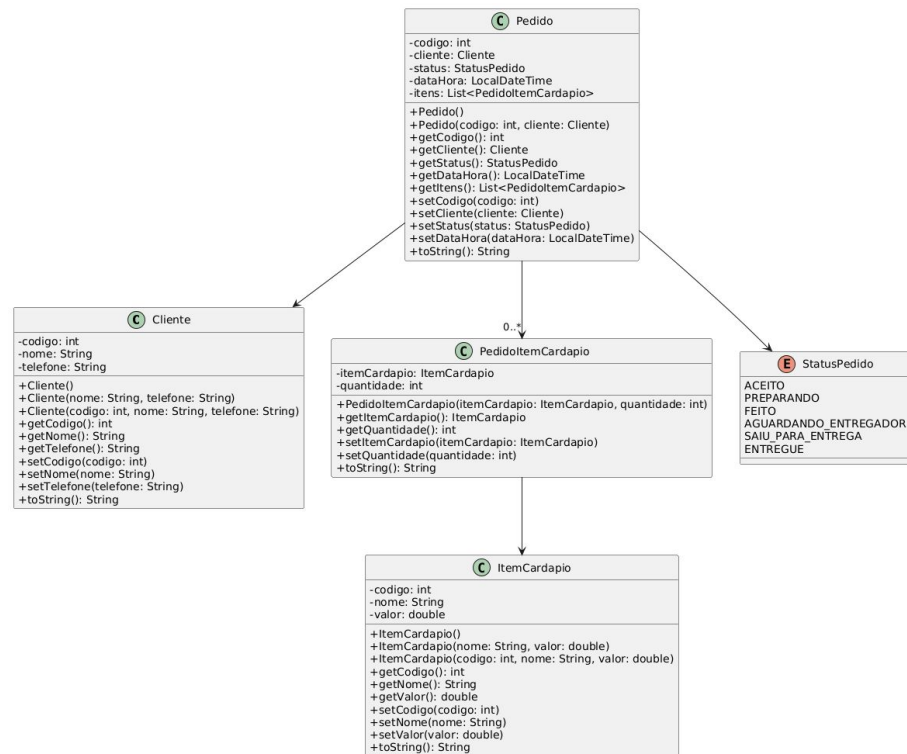


# **Aplicação CLI para FoodDelivery**

**Autores:** João Arthur  
Tony Cleriston  
Denilson Santos  
Vinícius Cerqueira  
Gabriel Arlisson

# Diagrama de Classes

- A imagem representa o diagrama feito depois de projetar as regras de negócio do sistema.
- Nele consta a solução projetada para o relacionamento de Pedido e itemCardapio.
- Como também os métodos utilizados para ter acesso aos dados privados da classe para listagem e alteração.



# Gerenciamento de Cardápio



- O sistema permite ao usuário cadastrar novos itens no cardápio, informando nome e valor.
- Cada item recebe automaticamente um código gerado pelo sistema para identificação.
- Os itens cadastrados são armazenados em uma estrutura de dados (lista/array).
- A opção de listar exibe todos os produtos, mostrando código, nome e valor formatado.

```
--- GERENCIAR CARDÁPIO ---
1 - Cadastrar Novo Item
2 - Listar Itens
3 - Voltar
Escolha uma opção: 1
Nome do item: refri500ml
Valor do item (ex: 12.50): 6.40
Item cadastrado com sucesso!

--- GERENCIAR CARDÁPIO ---
1 - Cadastrar Novo Item
2 - Listar Itens
3 - Voltar
Escolha uma opção: 2

--- ITENS DO CARDÁPIO ---
Código: 8589 | Nome: batata frita | Valor: R$ 4,50
Código: 6481 | Nome: x-burguer | Valor: R$ 11,00
Código: 6897 | Nome: refri500ml | Valor: R$ 6,40
```

# Gerenciamento de Clientes



- O sistema apresenta um menu interativo no console para gerenciar clientes.
- O usuário pode cadastrar novos clientes informando nome e telefone.
- Os dados são armazenados em uma estrutura (lista/array).
- A opção listar clientes exibe todos os registros cadastrados, com código, nome e telefone.
- O código é gerado automaticamente para identificar cada cliente.
- A opção voltar retorna ao menu anterior.

```
--- GERENCIAR CLIENTES ---
1 - Cadastrar Novo Cliente
2 - Listar Clientes
3 - Voltar
Escolha uma opção: 1
Nome do cliente: gabriel
Telefone do cliente (ex: 99 99999-9999): 76933345098
Cliente cadastrado com sucesso!

--- GERENCIAR CLIENTES ---
1 - Cadastrar Novo Cliente
2 - Listar Clientes
3 - Voltar
Escolha uma opção: 2

--- CLIENTES ---
Código: 5048 | Nome: tony | Telefone: 75912344321
Código: 1448 | Nome: joao arthur | Telefone: 749321377643
Código: 6556 | Nome: gabriel | Telefone: 76933345098
```

# Gerenciamento de Pedidos



- Cada pedido exibe: código, horário, status, cliente e itens escolhidos com valores e quantidades.
- No primeiro momento, ele seleciona um cliente e os itens pelo código, e informa a quantidade

```
--- GERENCIAR PEDIDOS ---
```

```
1 - Cadastrar Novo Pedido
```

```
2 - Atualizar Status do Pedido
```

```
3 - Consultar Pedido por Status
```

```
4 - Voltar
```

```
Escolha uma opção: 1
```

```
Código: 5048 | Nome: tony | Telefone: 75912344321
```

```
Código: 1448 | Nome: joao arthur | Telefone: 749321377643
```

```
Código: 6556 | Nome: gabriel | Telefone: 76933345098
```

```
Código do cliente: 1448
```

```
Pedido iniciado para joao arthur...
```

```
Código do item do cardápio: 6481
```

```
Informe a quantidade para x-burger: 1
```

```
Item "x-burger" (1x) adicionado com sucesso ao pedido.
```

# Gerenciamento de Pedidos



- Após a confirmação, o pedido é salvo no sistema e pode ser consultado/atualizado depois.
- Exibindo o horário exato na conclusão do pedido efetuado, e por padrão iniciando o status como "Aceito"

```
Adicionar um novo item (digite "n" para finalizar)? s
Código do item do cardápio: 6897
Informe a quantidade para refri500ml: 2
Item "refri500ml" (2x) adicionado com sucesso ao pedido.

Adicionar um novo item (digite "n" para finalizar)? n

-----

Pedido: 1 | Horário: 30/08/2025 10:06:30 | Status: ACEITO
Código: 1448 | Nome: joao arthur | Telefone: 749321377643
- x-burguer [R$ 11,00] (x1)
- refri500ml [R$ 6,40] (x2)

-----

Digite "s" para confirmar o pedido... s
Pedido cadastrado com sucesso.
```

# Atualizar Status

- O usuário do sistema pode atualizar o status do pedido informando o código.
- Automaticamente o status segue uma lógica de sua ordem, seguindo o enum.

```
--- GERENCIAR PEDIDOS ---
1 - Cadastrar Novo Pedido
2 - Atualizar Status do Pedido
3 - Consultar Pedido por Status
4 - Voltar
Escolha uma opção: 2
-----
Pedido: 1 | Horário: 30/08/2025 10:30:52 | Status: ACEITO
Código: 3066 | Nome: joao arthur | Telefone: 75952139473
- x-burguer [R$ 11,00] (x1)
- refri500ml [R$ 6,40] (x2)
-----
Código do pedido: 1
Status alterado de ACEITO para PREPARANDO.
```

```
package models;

public enum StatusPedido {
    ACEITO,
    PREPARANDO,
    FEITO,
    AGUARDANDO_ENTREGADOR,
    SAIU_PARA_ENTREGA,
    ENTREGUE
}
```



# Consultar Status



- O sistema permite consultar pedidos por status.
- O usuário seleciona um dos status disponíveis, e então busca e exibe apenas os pedidos que correspondem ao status escolhido.

```
Código do pedido: 1
Status alterado de PREPARANDO para FEITO.

--- GERENCIAR PEDIDOS ---
1 - Cadastrar Novo Pedido
2 - Atualizar Status do Pedido
3 - Consultar Pedido por Status
4 - Voltar
Escolha uma opção: 3
[1] ACEITO
[2] PREPARANDO
[3] FEITO
[4] AGUARDANDO_ENTREGADOR
[5] SAIU_PARA_ENTREGA
[6] ENTREGUE
Status a verificar: 3

-----
Pedido: 1 | Horário: 30/08/2025 10:30:52 | Status: FEITO
Código: 3066 | Nome: joao arthur | Telefone: 75952139473
- x-burguer [R$ 11,00] (x1)
- refri500ml [R$ 6,40] (x2)
-----
```



# Lógica do Status

- O método atualizarStatus é responsável por avançar o status de um pedido, ele obtém todos os status possíveis do enum StatusPedido. sendo possível localizar o índice do status atual do pedido.
- Ela verifica se não é o último status da lista e se já estiver no último status, nenhuma alteração é feita e retorna false. Essa lógica garante o fluxo sequencial dos pedidos.

```
public boolean atualizarStatus(Pedido pedido) {  
    List<StatusPedido> statusPossiveis = Arrays.asList(StatusPedido.values());  
    int indiceStatusAtual = statusPossiveis.indexOf(pedido.getStatus());  
    boolean ultimoStatusAlcancado = indiceStatusAtual == (statusPossiveis.size() - 1);  
    if (!ultimoStatusAlcancado) {  
        pedidoRepository.updateStatus(pedido, statusPossiveis.get(indiceStatusAtual + 1));  
        return true;  
    }  
    return false;  
}
```

# Gerar Relatórios

- O relatório simplificado exibe apenas a quantidade de pedidos e o valor total.
- Já o relatório detalhado mostra informações completas de cada pedido, incluindo: código e cliente, valor total e itens do pedido, pegando dados apenas do dia.

```
public List<RelatorioDetalhado> relatorioDetalhado() {  
    List<RelatorioDetalhado> dados = new ArrayList<>();  
  
    for (Pedido pedido : this.pedidoRepository.findAll()) {  
        if (pedido.getDataHora().toLocalDate().equals(LocalDate.now())) {  
            dados.add(new RelatorioDetalhado(pedido));  
        }  
    }  
  
    return dados;  
}
```

```
---- RELATÓRIO SIMPLIFICADO ----  
Quantidade: 1  
Valor Total: 23.8  
-----  
  
--- GERENCIAR RELATÓRIOS ---  
1 - Gerar Relatório Simplificado  
2 - Gerar Relatório Detalhado  
3 - Voltar  
Escolha uma opção: 2  
  
----- RELATÓRIO DETALHADO -----  
Pedido: 1  
Cliente: 6594 - joao arthur  
Valor Total: 23.8  
----- I T E M S -----  
        Produto: 1220 - x-burger  
        Quantidade: 1  
        Valor Unit.: 11.0  
        Valor Total: 11.0  
        - - - -  
        Produto: 2541 - refri500ml  
        Quantidade: 2  
        Valor Unit.: 6.4  
        Valor Total: 12.8  
        - - - -  
-----
```

# Separação de Responsabilidades

