

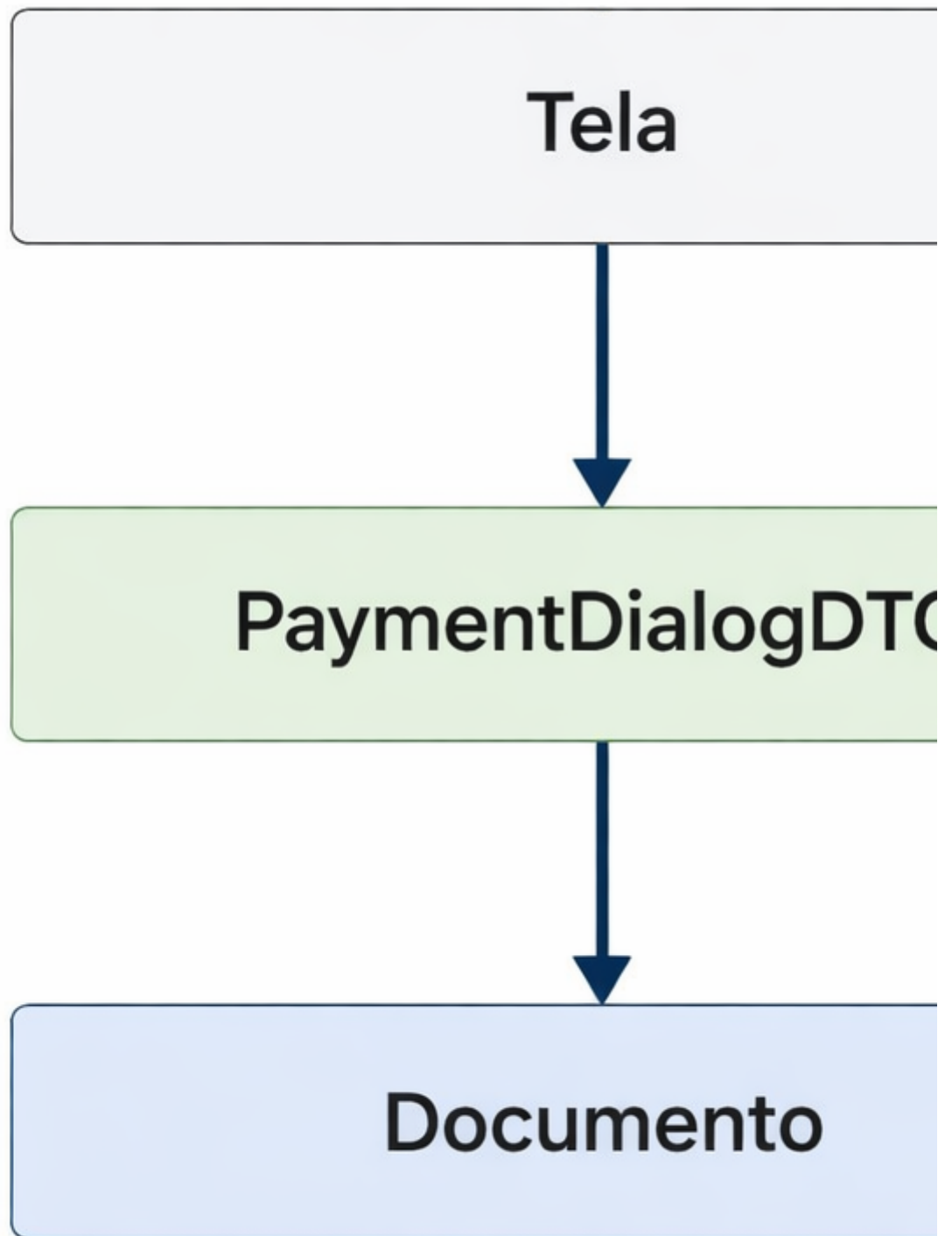
Pagamento Unificado - DTO genérico

	Projeto/Sistema: Pagamento Unificado - Gerencie Aqui	Versão do Template: 1.2
	 Processo: Pagamento Unificado - DTO genérico.	Versão do Documento: 1.0
	Responsável(eis): Luís Leite	Data: 05/01/2026

1. Introdução

Este processo visa exemplificar a criação de uma entidade genérica financeira que será utilizado em todo sistema.

2. Processo



I - Fluxograma

Processo	Envolvidos	Dados de Entrada	Dados de Saída
PROC001 Pagamento Unificado - DTO genérico	-	-	-

Especificação Funcional

PROC001 Pagamento Unificado - DTO genérico:

Regras de Negócio:

1. A tela de pagamento unificado será controlada como uma **entidade financeira genérica**, responsável por concentrar todas as informações relacionadas ao registro de recebimentos realizadas na interface do sistema.
2. A estrutura genérica será utilizada exclusivamente para controle das informações durante o uso da tela de pagamento. Após a confirmação da operação, os dados vão ser convertidos para a estrutura específica de acordo com o documento onde o financeiro será gerado.
3. Deve ser mapeado a estrutura genérica com as estruturas específicas.
4. O processo deverá ser realizado tanto para o envio de informações da tela para documento específico no momento da confirmação da operação; como para carregar informações já existentes do documento financeiro para a tela, quando o usuário abrir ou editar os pagamentos.

Questões Técnicas:

1. Deverá ser criado um **DTO genérico de pagamentos** com o objetivo de padronizar a comunicação entre a tela de registro de recebimentos e os diferentes módulos do sistema.

A tela deverá manipular exclusivamente esse DTO durante a interação com o usuário.

Para integração com os documentos financeiros específicos do sistema, deverão ser implementados **mapeamentos (mappers)** responsáveis por:

- converter os dados do documento financeiro para o DTO genérico no momento da abertura da tela;
- converter os dados do DTO genérico para a estrutura financeira específica do documento no momento da finalização da operação.

'formaPagamento' deve ser consultada das formas de pagamentos personalizadas cadastradas.

2. ENUMS:

```
public enum PaymentType {  
  
    DINHEIRO,  
    CREDITO,  
    DEBITO,  
    BOLETO,  
    PIX,  
    OUTROS
```

```
}

public enum PaymentPeriodicity {
    UNICO,          // pagamento único
    SEMANAL,
    QUINZENAL,
    MENSAL,
}
```

2. Exemplo DTO Principal:

```
public class PaymentDialogDTO {

    private LocalDateTime dataOperacao;

    // resumo da venda
    private BigDecimal totalCompra;
    private BigDecimal descontoValor;
    private BigDecimal descontoPercentual;
    private BigDecimal acrescimoValor;
    private BigDecimal acrescimoPercentual;
    private BigDecimal totalLiquido;

    // pagamentos adicionados
    private List<PaymentItemDTO> pagamentos;

    // resumo dos pagamentos
    private BigDecimal totalAPagar;
    private BigDecimal saldoAPagar;
    private BigDecimal totalPago;
    private BigDecimal troco;

}
```

3. Exemplo DTO pagamentos:

```
public class PaymentItemDTO {
```

```

private PaymentType tipoPagamento;
// DINHEIRO, CREDITO, DEBITO, BOLETO, PIX, OUTROS

private String formaPagamento;
// ex: Visa, MasterCard, etc B

private BigDecimal valorRecebimento;

private LocalDate dataVencimento;

private Integer numeroParcelas;

private PaymentPeriodicity periodicidade;

private String cnpjPSP;

private String cnpjRecebedor;

private String idTransacao;

private String codigoPix;

}

```

4. Mapeamento exemplo (não está no contexto do sistema):

```

public class OrdemServicoPaymentMapper {

    public static void mapPaymentsToOrdemServico(
        PaymentDialogDTO dto,
        OrdemServico ordemServico) {

        List<OrdemServicoRecebimento> recebimentos = new ArrayList<>();

        for (PaymentItemDTO item : dto.getPagamentos()) {

            OrdemServicoRecebimento recebimento = new OrdemServicoRecebimento();

            recebimento.setTipoPagamento(item.getTipoPagamento());
            recebimento.setValorRecebido(item.getValorRecebimento());

```

```
        recebimento.setDataVencimento(item.getDataVencimento());
        recebimento.setIdTransacao(item.getIdTransacao());

        recebimentos.add(recebimento);
    }

    ordemServico.setRecebimentos(recebimentos);
}

}
```

©SOFTEN SISTEMAS 2026

Revision #4

Created 11 March 2026 16:21:16 by Luís Leite

Updated 13 March 2026 11:23:32 by Luís Leite