

Alunos: Fábio Emanuell Pereira Milagres, João Pedro de Almeida Dupim e Pedro Henrique Santos.

Exemplo 1:

(a.1) Enunciado ECA (Evento-Condição-Ações)

Evento: O sistema precisa registrar o pagamento de uma conta a receber.

Condição: O registro de pagamento ocorre quando um usuário informa o código da conta e a data do pagamento.

Ações:

- Atualizar a data de pagamento da conta especificada na tabela ContaReceber.

(a.3) Documentação e Decisões de Projeto

1. Uso de Procedure: Criamos uma procedure para permitir o registro de pagamentos de forma eficiente e reutilizável.
2. Parâmetros de Entrada: A procedure recebe o código da conta (p_codigo) e a data do pagamento (p_data_pagamento).
3. Atualização Direta: O campo Data_Paga da conta especificada é atualizado na tabela ContaReceber.
4. Compatibilidade: Versões foram criadas para MySQL e PostgreSQL, adaptando a sintaxe conforme as necessidades de cada banco de dados.

(a.4) Casos de Teste Interessantes

Codigo	Data_Paga Inserida	Situação Antes	Situação Esperada Depois
1	2024-02-01	Conta não paga	Conta registrada como paga em 2024-02-01
2	2024-02-10	Conta já paga	Data_Paga atualizada para 2024-02-10
3	NULL	Conta não paga	Nenhuma mudança

1. Caso 1: Conta que estava em aberto é registrada como paga na data especificada.
2. Caso 2: Conta que já estava paga tem sua data de pagamento atualizada.
3. Caso 3: Teste com NULL para verificar comportamento quando nenhuma data é passada.

Exemplo 2:

(a.1) Enunciado ECA (Evento-Condição-Ações)

Evento: O sistema precisa registrar um item para uma nota fiscal de compra (NF_Compra) e atualizar seu valor total.

Condição: O registro ocorre quando um usuário informa o código da nota fiscal, o código do produto, a quantidade e o valor unitário do item.

Ações:

- Inserir o novo item na tabela Item_NF_Compra.
- Atualizar o valor total da NF_Compra com a soma de todos os itens cadastrados para ela.

(a.3) Documentação e Decisões de Projeto

1. Uso de Procedure: Criamos uma procedure para automatizar a inserção de itens em notas fiscais de compra e manter o valor total atualizado.
2. Parâmetros de Entrada: A procedure recebe o código da NF (p_codigo_nf), o código do produto (p_codigo_produto), a quantidade (p_quantidade) e o valor unitário (p_valor_uni).
3. Cálculo do Valor Total: Multiplicamos a quantidade pelo valor unitário para definir o valor total do item.
4. Geração de Código Automática: O novo item recebe um código sequencial automaticamente.
5. Atualização da NF_Compra: A soma de todos os itens da NF é calculada e o valor total da NF_Compra é atualizado.
6. Compatibilidade: Foram criadas versões para MySQL e PostgreSQL, adaptando a sintaxe conforme as necessidades de cada banco de dados.

(a.4) Casos de Teste Interessantes

Codigo_NF	Codigo_Produto	Quantidade	Valor_Uni	Valor_Total Esperado	Valor NF Antes	Valor NF Depois
1	1001	5	10.00	50.00	200.00	250.00
2	1002	3	15.50	46.50	0.00	46.50
3	1003	0	20.00	0.00	100.00	100.00

1. Caso 1: Um item é adicionado a uma NF que já tem itens, aumentando seu valor total.

2. Caso 2: Um item é adicionado a uma NF nova, iniciando seu valor total.
3. Caso 3: Um item com quantidade zero é inserido, sem afetar o valor da NF.