



Relatório Final

BANKSTORE – INTEGRATION

PROJETO DE INTEGRAÇÃO DE BANCO E LOJA, COM OPERAÇÕES

BANCÁRIAS E COMPRAS ONLINE.

Conteúdo

Introdução	2
Tecnologias e recursos	3
Bibliotecas utilizadas:	3
Ferramentas utilizadas:.....	4
Requisitos mínimos:.....	4
Implementação	4
Cronograma de desenvolvimento do projeto	4
Cronologia Simplificada	4
Fases de Implementação	7
Aspetos técnicos do desenvolvimento do projeto	8
Requisitos e funcionalidades do projeto:	8
Requisitos técnicos e/ou de performance	11
Interfaces	11
Estrutura da Base de Dados	20
Gatilhos Utilizados	22
Conclusão	24
Bibliografia	24
Anexos	24

Introdução

O projeto "**BankStore - Integration**" é um programa que integra um sistema bancário e uma loja de forma a facilitar as interações entre o utilizador, o banco e a loja, tudo em uma plataforma única.

O objetivo principal do projeto é permitir operações bancárias como depósito, levantamento e transferência de fundos, consultar as transações efetuadas e permitir compras simples e rápidas com uma loja que oferece um grande stock de itens variados e uma forma fácil e eficiente de pagar com a conta bancária.

Neste relatório vou relatar as tecnologias e recursos usados e o processo de implementação, incluindo o cronograma e aspetos técnicos do desenvolvimento do projeto.

Tecnologias e recursos

BIBLIOTECAS UTILIZADAS:

Microsoft.Data.SqlClient – Usada para aceder e manipular dados em SQL Server.

System.Data – Usada para gerir dados da base de dados

System.Security.Cryptography – Usada para encriptar as Palavras-passe e PINs dos utilizadores na base de dados

System.Text – Usada em conjunto com **System.Security.Cryptography**, usada para criar um buffer de bytes para a encriptação

```
/// <summary>
/// Classe de Criptografia
/// </summary>
/// </summary>
9 referências
public class Crypt
{
    /// <summary>
    /// Faz o hash do input e devolve o hash em base64
    /// </summary>
    /// <param name="input">string a fazer hash</param>
    /// <returns>hash do input em base64</returns>
    7 referências
    public static string Hash(string input)
    {
        using (SHA256 sha256 = SHA256.Create())
        {
            byte[] bytes = Encoding.UTF8.GetBytes(input);
            byte[] hashBytes = sha256.ComputeHash(bytes);
            return Convert.ToBase64String(hashBytes);
        }
    }
    /// <summary>
    /// Verifica se o hash do input corresponde ao hash guardado
    /// </summary>
    /// <param name="input">String a verificar</param>
    /// <param name="hash">Hash obtido da base de dados</param>
    /// <returns>True se corresponder, False se não corresponder</returns>
    3 referências
    public static bool Verify(string input, string hash)
    {
        string inputHash = Hash(input);
        return inputHash == hash;
    }
}
```

Figura 1 - Classe Crypt

FERRAMENTAS UTILIZADAS:

Visual Studio 2022 – Execução do projeto e gestão da base de dados

DbSchema 8.2.7 – Usada para fazer o esquema da base de dados para documentação.

Balsamiq Wireframes 4.7.5 – Usado para fazer o “rascunho” da estrutura dos forms.

Excalidraw – Usado para estruturar o projeto de forma visual

REQUISITOS MÍNIMOS:

CPU: Qualquer processador Intel ou AMD x64 bits

RAM: 2GB+

Armazenamento: 1Gb+

Implementação

CRONOGRAMA DE DESENVOLVIMENTO DO PROJETO

Cronologia Simplificada

1. *Início do Projeto*

- **Data:** 2 de maio de 2024
- **Descrição:** Criação inicial da documentação e organização do repositório.
- **Atividades:**
 - Criação de Docs/Originais

2. *Prototipagem e Planeamento*

- **Data:** 9 de maio de 2024
- **Descrição:** Desenvolvimento dos protótipos iniciais e início da execução do pré-projeto.
- **Atividades:**
 - Protótipos e início da execução do pré-projeto

3. Desenvolvimento da Base de Dados e Estrutura Inicial

- **Data:** 22 de maio de 2024 - 23 de maio de 2024
- **Descrição:** Estruturação da base de dados e criação dos formulários básicos.
- **Atividades:**
 - Criação da Base de Dados e Logótipos do Projeto
 - Criação de alguns forms
 - Criação de Diagrama de Forms e Organização do Projeto

4. Desenvolvimento de Funcionalidades Básicas

- **Data:** 23 de maio de 2024 - 31 de maio de 2024
- **Descrição:** Adição e desenvolvimento de formulários de criação de conta e início de sessão.
- **Atividades:**
 - Adição de Forms: Criação de Conta e Início de Sessão
 - Adição de Forms

5. *Expansão e Modificação dos Formulários*

- **Data:** 5 de junho de 2024 - 19 de junho de 2024
- **Descrição:** Alterações e adições aos formulários existentes, incluindo formulários bancários.
- **Atividades:**
 - Alteração de forms e Início dos Forms Banco
 - Conclusão de Forms

6. *Adição de Produtos Padrão*

- **Data:** 20 de junho de 2024
- **Descrição:** Inclusão de produtos padrão no sistema.
- **Atividades:**
 - Adição de Produtos Padrão

7. *Organização e Limpeza do Repositório*

- **Data:** 27 de junho de 2024
- **Descrição:** Organização final do repositório, incluindo a limpeza e criação do ficheiro .gitignore.
- **Atividades:**
 - Criação de .gitignore
 - Limpeza do Repositório
 - Reset src

8. *Conclusão do Código*

- **Data:** 27 de junho de 2024
- **Descrição:** Finalização e revisão do código do projeto.
- **Atividades:**
 - Conclusão do código

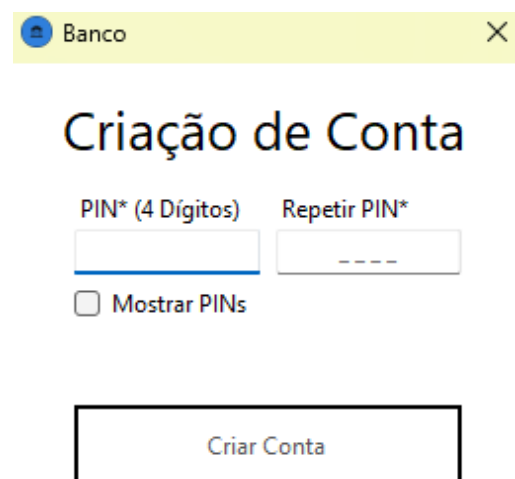
Fases de Implementação

1. **Início do Projeto:** 2 de maio de 2024
 - Criação inicial dos documentos e estruturação do repositório.
2. **Prototipagem e Planeamento:** 9 de maio de 2024
 - Desenvolvimento dos protótipos e início do pré-projeto.
3. **Desenvolvimento da Base de Dados e Estrutura Inicial:**
22 de maio de 2024 - 23 de maio de 2024
 - Estruturação da base de dados e criação de formulários iniciais.
4. **Desenvolvimento de Funcionalidades Básicas:**
23 de maio de 2024 - 31 de maio de 2024
 - Desenvolvimento de formulários de criação de conta e início de sessão.
5. **Expansão e Modificação dos Formulários:**
5 de junho de 2024 - 19 de junho de 2024
 - Adição e modificação de formulários, incluindo funcionalidades bancárias.
6. **Adição de Produtos Padrão:** 20 de junho de 2024
 - Inclusão de produtos padrão no sistema.
7. **Organização e Limpeza do Repositório:** 27 de junho de 2024
 - Limpeza do repositório e criação do .gitignore.
8. **Conclusão do Código:** 27 de junho de 2024
 - Finalização e revisão do código do projeto.

ASPETOS TÉCNICOS DO DESENVOLVIMENTO DO PROJETO

Requisitos e funcionalidades do projeto:

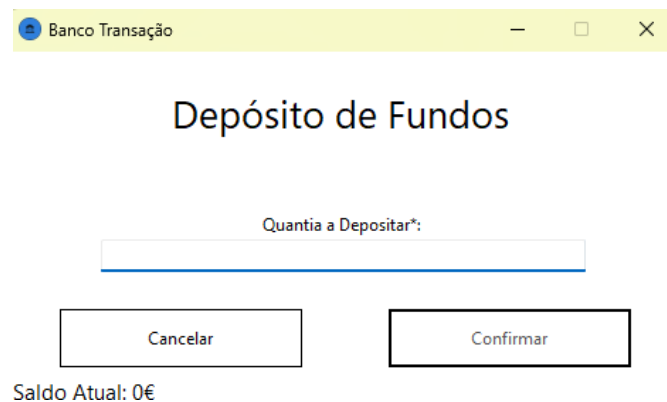
REQ0001 – Permitir a criação de uma conta bancária – **Concluído**



The screenshot shows a window titled 'Banco' with a close button (X). The main heading is 'Criação de Conta'. Below it, there are two input fields: 'PIN* (4 Dígitos)' and 'Repetir PIN*'. The first field is empty, and the second field contains four dashes. Below these fields is a checkbox labeled 'Mostrar PINs' which is unchecked. At the bottom, there is a button labeled 'Criar Conta'.

Figura 2 - Criação de Conta Bancária

REQ0002 – Permitir o depósito de fundos na conta bancária – **Concluído**



The screenshot shows a window titled 'Banco Transação' with standard window controls (minimize, maximize, close). The main heading is 'Depósito de Fundos'. Below it, there is a label 'Quantia a Depositar*:' followed by an empty input field. At the bottom, there are two buttons: 'Cancelar' and 'Confirmar'. Below the buttons, the text 'Saldo Atual: 0€' is displayed.

Figura 3 - Depósito de Fundos na Conta Bancária

REQ0003 – Permitir o levantamento de fundos da conta bancária – **Concluído**

Banco Transação

Levantamento de Fundos

Quantia a Levantar*:

— / —

Cancelar Confirmar

Saldo Atual: 0€

Figura 4 - Levantamento de Fundos na Conta Bancária

REQ0004 – Permitir a consulta das transações da conta bancária – **Concluído**

Banco Transações

Histórico de Transações

Data e Hora	Descritivo	Variação	Saldo Final
28/06/2024 11:08	Depósito de Fundos	+100,00 €	100,00 €
28/06/2024 11:08	Levantamento de Fundos	-10,00 €	90,00 €
28/06/2024 11:09	Transferência de fundos para PT50 1369 4200 2323 2362 7	-10,00 €	80,00 €

Voltar

Saldo Atual: 80€

Figura 5 - Consulta de transações da conta bancária

REQ0005 – Permitir a transferência de fundos entre contas bancárias – **Concluído**

Banco Transferência

Efetuar Transferência

Nome do Titular	IBAN
Martim Francisco	PT50 1369 4200 2323 2362 7
Martim Alves	PT50 1369 4200 9360 7378 6

IBAN destinatário*:

Quantia a Transferir*:

Cancelar

Confirmar

Saldo Atual: 80€

Figura 6 - Transferência de Fundos entre contas bancárias


REQ0006 – Permitir a compra de produtos na loja – **Concluído**

Loja

Sair

Categoria Seleccionada: Alimentos

Carrinho (0)




Carne de Javali 1kg

Carne de javali de alta qualidade, 1 kg.

25,50€

Quantidade Disponível: 30

0




Leite Mário Edição Limitada

Leite especial em edição limitada.

12,99€

Quantidade Disponível: 97

0



Croquetes de Carne

Deliciosos croquetes de carne.

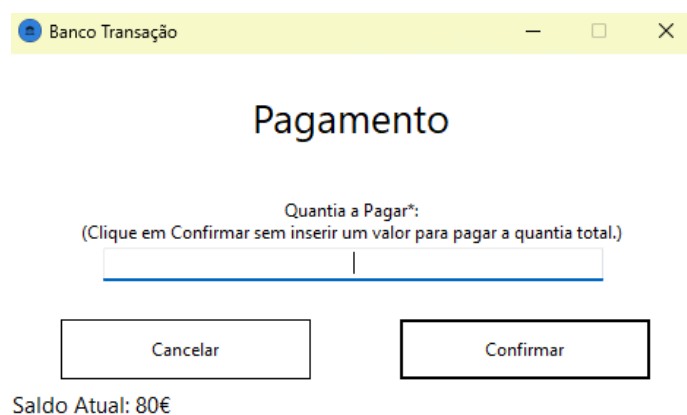
Figura 7 - Compra de Produtos na Loja

digital
escola
profissional

Rumos
Knowledge sharing

10

REQ0007 – Permitir o pagamento das compras através da conta bancária – **Concluído**



Banco Transação

Pagamento

Quantia a Pagar*:
(Clique em Confirmar sem inserir um valor para pagar a quantia total.)

Cancelar Confirmar

Saldo Atual: 80€

Figura 8 - Pagamento de compras com conta bancária

Requisitos técnicos e/ou de performance

- Utilização de criptografia para armazenamento de Palavras-passe e PINs
- Base de dados eficiente e bem organizada
- Rapidez e responsividade na execução do programa
- Modo de Administrador do programa

Interfaces

Menu Inicial



Definições



Selecione uma opção:

Criar Conta

Entrar na Conta

Sair

Definições do Programa

Definições

Aparência

Tema:

Redefinir Predefinições

Confirmar

Criação de Conta

Criação de Conta

Nome*

Nome de Utilizador*

Palavra-Passe*

Repetir Palavra-Passe*

☐ Mostrar Palavras-Passe

Criar Conta

Entrar na Conta

Iniciar Sessão

Nome de Utilizador*

Palavra-Passe*

☐ Mostrar Palavras-Passe

Iniciar Sessão

Menu Principal

Definições Conta

Bem Vindo Utilizador
Selecione um local:



Banco



Loja

Sair

Definições de Conta

Definições da Conta

Nome Atual: Utilizador

Alterar

Nome de Utilizador Atual: a

Alterar

Palavra-Passe

Repetir Palavra-Passe

Alterar

☐ Mostrar Palavras-Passe

Apagar Conta

Voltar

Banco – Menu Inicial



Banco

Selecione uma opção:

Criar Conta

Sair



Banco

Selecione uma opção:

Entrar na Conta

Sair

Banco – Criação de conta

Criação de Conta

PIN* (4 Dígitos)

Repetir PIN*

☐ Mostrar PINs

Criar Conta

Banco – Entrar na conta

Entrar na Conta

PIN*

☐ Mostrar PIN

Voltar

Entrar

Banco - Menu Principal



Banco

Bem Vindo Utilizador
Selecione uma operação:

Efetuar Depósito

Efetuar Levantamento

Efetuar Transferência

Consultar Transações

Alterar PIN

Sair

Saldo Atual: 0€

Banco – Depósitos/Levantamentos

Depósito de Fundos

Quantia a Depositar*:

Cancelar

Confirmar

Saldo Atual: 0€

Levantamento de Fundos

Quantia a Levantar*:

Cancelar

Confirmar

Saldo Atual: 0€

Banco – Transferências

Efetuar Transferência

Nome do Titular	IBAN
Martim Francisco	PT50 1369 4200 2323 2362 7
Martim Alves	PT50 1369 4200 9360 7378 6

IBAN destinatário*:

Quantia a Transferir*:

Cancelar

Confirmar

Saldo Atual: 0€

Banco – Histórico de Transações

Banco Transações	—	□	×
------------------	---	---	---

Histórico de Transações

Data e Hora	Descritivo	Variação	Saldo Final
28/06/2024 11:08	Depósito de Fundos	+100,00 €	100,00 €
28/06/2024 11:08	Levantamento de Fundos	-10,00 €	90,00 €
28/06/2024 11:09	Transferência de fundos para PT50 1369 4200 2323 2362 7	-10,00 €	80,00 €

Voltar

Saldo Atual: 80€

Banco – Alterar PIN

Alterar PIN

Novo PIN*
(4 Dígitos)

Repetir PIN*

☐ Mostrar PINs

Confirmar

Apagar Conta


Loja – Menu Principal

Sair

Categoria Seleccionada:

Tecnologia

Carrinho (0)

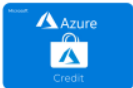


Calculadora gráfica Texas Instruments TI-82 STAT
Calculadora gráfica avançada TI-82 STAT.

189,99€

Quantidade Disponível: 14

0



Cartão presente Microsoft Azure 1 Mês
Cartão presente para 1 mês de serviços Microsoft Azure.





69,99€

Quantidade Disponível: 9

0

Smartphone SAMSUNG Galaxy S24 Ultra 256GB
Smartphone topo de gama Samsung Galaxy S24 Ultra com 256GB de

Loja – Carrinho

Nome do Produto	Qtd.	€/un	Total	
Tecnologia				
 Calculadora gráfica Texas Instruments TI-82 STAT	<input type="text" value="1"/>	189,99€	189,99€	<input checked="" type="checkbox"/>
 Cartão presente Microsoft Azure 1 Mês	<input type="text" value="1"/>	69,99€	69,99€	<input checked="" type="checkbox"/>
 Smartphone SAMSUNG Galaxy S24 Ultra 256GB	<input type="text" value="1"/>	1299,99€	1299,99€	<input checked="" type="checkbox"/>
 MacBook Pro M4 16	<input type="text" value="1"/>	5000,00€	5000,00€	<input checked="" type="checkbox"/>
			Total: 15941,57€	<input type="button" value="Pagamento"/>
				<input type="button" value="Voltar"/>

Loja – Pagamento

Pagamento

Total a Pagar:
15941,57€

Selecione o método de pagamento:

Conta Bancária

Dinheiro

Voltar

Loja/Banco – Pagamento com Conta Bancária

Pagamento

Quantia a Pagar*:
(Clique em Confirmar sem inserir um valor para pagar a quantia total.)

Cancelar

Confirmar

Saldo Atual: 0€

Loja – Pagamento com Dinheiro

Pagamento

Quantia a Pagar*:
(Clique em Confirmar sem inserir um valor para pagar a quantia total.)

Cancelar

Confirmar

Administrador – Gestão de Contas

Gestão de Contas

Conta Seleccionada: Martim

Voltar

Nome Atual: Martim Alves

Alterar

Nome de Utilizador Atual: Martim

Alterar

Palavra-Passe

Repetir Palavra-Passe

Alterar

Apagar Conta

☐ Mostrar Palavras-Passe

Administrador – Gestão de Contas Bancárias

Gestão Bancária

Conta Seleccionada: Martim

Voltar

Inserir Transação		Guardar Tabela	Apagar Transação Seleccionada	
Data e Hora	Descritivo		Variação	Saldo Final
26/06/2024 11:19	Depósito de Fundos		1500	1500
26/06/2024 11:19	Transferência de fundos para PT50 1369 4200 2323 2362 7		-1234	266
26/06/2024 11:20	Pagamento a Loja LLC.		-70	196
26/06/2024 11:22	Transferência de fundos de PT50 1369 4200 2644 9414 5		100	296

PIN* (4 Dígitos)

Repetir PIN*

Alterar PIN

Apagar Conta

☐ Mostrar PINs

Gestão da Loja

Nova Categoria:

Criar Categoria




Voltar

Categoria Seleccionada:

Alterar Nome da Categoria

Apagar Categoria

Novo Nome da Categoria:

Inserir Produto	Guardar Tabela			Apagar Produto Seleccionado	
	Imagem	Nome do Produto	Descrição	Stock	Preço
		Aspirador Robô	Aspirador robô	25	999,99
		Calculadora gráfica	Calculadora gráfica	14	189,99
		Cartão presente	Cartão presente para	9	69,99
		MacBook Pro M4 16	MacBook Pro de 16	5	5899,99
		Smartphone	Smartphone topo de	15	1299,99
		TV LG OLED83G45LW	Televisão LG OLED	10	5699,99

Estrutura da Base de Dados

A base de dados foi projetada para suportar o sistema integrado de banco e loja. O modelo de dados inclui as seguintes tabelas:

Tabela users

- Armazena informações dos utilizadores.
- Campos: id (chave primária, autoincremento), name, password, username.

Tabela bank_accounts

- Detalhes das contas bancárias associadas aos utilizadores.
- Campos: userid (chave estrangeira referenciando users.id), pin, saldo, IBAN.
- Restrições: Chave primária userid, chave estrangeira referenciando users.id com ação CASCADE.

Tabela bank_transactions

- Regista as transações financeiras realizadas.
- Campos: id (chave primária, autoincremento), userid (chave estrangeira referenciando bank_accounts.userid), time, description, variation, finalbalance.
- Restrições: Chave primária id, chave estrangeira referenciando bank_accounts.userid com ação CASCADE.

Tabela shop_categories

- Lista de categorias de produtos.
- Campos: id (chave primária, autoincremento), name.

Tabela shop_products

- Detalhes dos produtos disponíveis na loja.
- Campos: idprod (chave primária, autoincremento), idcat (chave estrangeira referenciando shop_categories.id), catpos, name, description, stock, price, imagepath.
- Restrições: Chave primária idprod, chave estrangeira referenciando shop_categories.id com ação CASCADE.

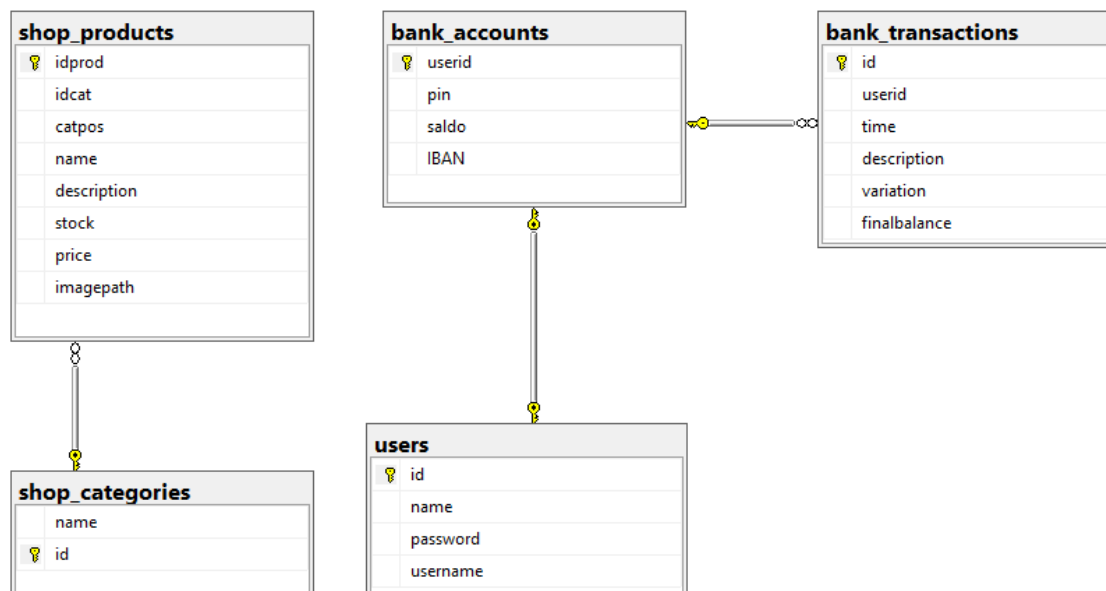


Figura 9 - Diagrama da Base de Dados do Projeto

Gatilhos Utilizados

Os gatilhos seguintes foram criados para manter a integridade e a consistência dos dados.

Trigger trg_maintain_finalbalance

Atualiza o finalbalance em bank_transactions e o saldo em bank_accounts após inserções, atualizações ou exclusões em bank_transactions.

```
IF NOT EXISTS (SELECT * FROM sys.triggers WHERE name = 'trg_maintain_finalbalance')
BEGIN
    EXEC ('CREATE TRIGGER trg_maintain_finalbalance
    ON [dbo].[bank_transactions]
    AFTER INSERT, UPDATE, DELETE
    AS
    BEGIN
        SET NOCOUNT ON;

        DECLARE @userid INT;

        -- Determine affected user ID
        IF EXISTS (SELECT 1 FROM inserted)
            SET @userid = (SELECT TOP 1 userid FROM inserted);
        ELSE
            SET @userid = (SELECT TOP 1 userid FROM deleted);

        -- Update finalbalance for all transactions of the affected userid
        WITH cte AS (
            SELECT
                id,
                userid,
                variation,
                SUM(variation) OVER (PARTITION BY userid ORDER BY [time], id ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS running_total
            FROM
                [dbo].[bank_transactions]
            WHERE
                userid = @userid
        )
        UPDATE bt
        SET bt.finalbalance = ROUND(cte.running_total, 2)
        FROM
            [dbo].[bank_transactions] bt
        JOIN
            cte ON bt.id = cte.id;

        -- Update saldo in bank_accounts table
        UPDATE ba
        SET ba.saldo = (
            SELECT ROUND(ISNULL(SUM(variation), 0), 2)
            FROM [dbo].[bank_transactions]
            WHERE userid = @userid
        )
        FROM [dbo].[bank_accounts] ba
        WHERE ba.userid = @userid;
    END');
END;
```

Figura 10 - Criação do trigger trg_maintain_finalbalance

Trigger *trg_OrderProductsByCategory*

Mantém a ordem alfabética dos produtos por categoria na tabela *shop_products* após inserções, atualizações ou exclusões.

```
IF NOT EXISTS (SELECT * FROM sys.triggers WHERE name = 'trg_OrderProductsByCategory')
BEGIN
    EXEC ('CREATE TRIGGER trg_OrderProductsByCategory
ON shop_products
AFTER INSERT, UPDATE, DELETE
AS
BEGIN
    SET NOCOUNT ON;

    ;WITH AffectedCategories AS (
        SELECT DISTINCT idcat
        FROM inserted
        UNION
        SELECT DISTINCT idcat
        FROM deleted
    ),

    CTE AS (
        SELECT
            sp.idprod,
            sp.idcat,
            ROW_NUMBER() OVER (PARTITION BY sp.idcat ORDER BY sp.name) AS RowNum
        FROM
            shop_products sp
        INNER JOIN
            AffectedCategories ac ON sp.idcat = ac.idcat
    )

    UPDATE sp
    SET sp.catpos = cte.RowNum
    FROM shop_products sp
    INNER JOIN CTE ON sp.idprod = CTE.idprod;
END');
END;
```

Figura 11 - Criação do trigger *trg_OrderProductsByCategory*

Conclusão

Concluindo, acho que “**BankStore – Integration**” correspondeu às minhas expectativas e fora algum desconhecimento inicial sobre SQL Server, acredito que não existiram dificuldades relevantes para aqui documentar, apenas acho que se tivesse sido dado em aula de base de dados a matéria relevante ao projeto teria mais conhecimento no início e mais facilidade ao longo do desenvolvimento do projeto.

Bibliografia

<https://stackoverflow.com>

<https://learn.microsoft.com/>

Anexos

Nada relevante foi encontrado para ser anexado.