

Prueba Técnica Bancolombia

Andrés David Hoyos Velásquez

Instrucciones

- Crea una base de datos con una tabla llamada "Movimientos". La tabla debe tener campos para almacenar información como el valor de la transacción, número de cuenta, fecha.
- Cree una tabla llamada "Cuentas" con la información de número de cuenta, tipo de cuenta, saldo y titular.
- Ingrese información Dummy en ambas tablas
- Implementar una funcionalidad que permita a los usuarios generar un archivo de Excel que contenga el resumen de todos los movimientos de su cuenta, con un diseño amigable para el usuario tipo extracto con los logos de la organización.
- El Archivo Excel(o pdf) generado debe estar bloqueado para modificaciones.
- El Archivo Excel(o pdf) debe tener contraseña de apertura

Tener en cuenta

- Considere la experiencia del usuario al desarrollar el formulario y las interacciones con la plataforma.
- Puede utilizar el lenguaje que considere más conveniente para las implementaciones.

Entregables

- Script de creación de la base de datos
- Script de inserción de la data dummy
- Código fuente de la aplicación y documento pdf con explicación del procedimiento realizado para llegar a la solución

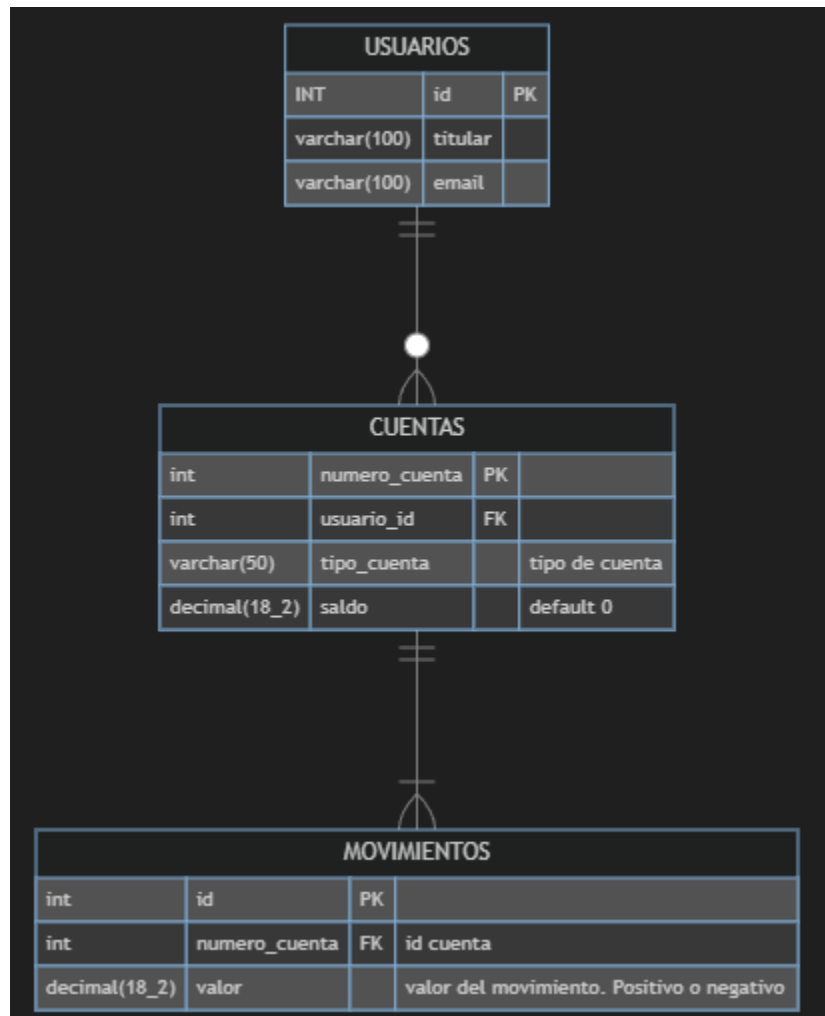
Procedimiento

Base de Datos

Se puede ver el código también en el archivo /docs/Bancolombia_prueba.sql

Para la base de datos se utilizó SQL Server 2022 Developer Edition.

Se crearon las tablas **cuentas** y **movimientos**. Para hacer un ejercicio más interesante se creó una tabla adicional **usuarios** y se reemplazó el campo *titular* de **cuentas** por **usuario_id**.



Se agrego un trigger a la tabla de **movimientos** para que cada vez que se agregue un movimiento a una cuenta, se actualice el saldo de la cuenta.

Dummy Data

Para la dummy data, se utilizo el sitio web <https://www.mockaroo.com/>.

Codigo SQL

```
-- crear base de datos - sql server
CREATE DATABASE prueba_bancolombia
GO

-----
--- crear tablas -----
-----

USE [prueba_bancolombia]
GO

CREATE TABLE [usuarios]
(
```

```

        [id]          INT PRIMARY KEY IDENTITY,
        [titular]     VARCHAR(100) NOT NULL,
        [email]       VARCHAR(100) NOT NULL UNIQUE,
    )
-- tabla cuentas
CREATE TABLE [cuentas]
(
    [numero_cuenta] INT PRIMARY KEY NOT NULL,
    [usuario_id]    INT              NOT NULL,
    [tipo_cuenta]   VARCHAR(50)      NOT NULL,
    [saldo]         DECIMAL(18, 2) DEFAULT 0,

    FOREIGN KEY (usuario_id) REFERENCES [usuarios] ([id])
)
-- tabla movimientos
CREATE TABLE [movimientos]
(
    [id]              INT PRIMARY KEY IDENTITY,
    [numero_cuenta] INT,
    [valor]           DECIMAL(18, 2) NOT NULL,
    [fecha]           DATETIME2      NOT NULL DEFAULT GETDATE(),

    FOREIGN KEY (numero_cuenta) REFERENCES [cuentas] ([numero_cuenta])
)
GO
-- trigger para actualizar el saldo de la cuenta -----
-----
-- para evitar problemas con bulk inserts, se utiliza la --
-- tabla temporal 'inserted' que representa todos los    --
-- registros insertados                                   --
-----
CREATE OR ALTER TRIGGER [TR_movimientos_actualizar_saldo_after_insert]
    ON [movimientos]
    AFTER INSERT
    AS
BEGIN
    UPDATE [c]
    SET [saldo] = [saldo] + [i].[valor]
    FROM [cuentas] [c]
        INNER JOIN [inserted] [i] ON [c].[numero_cuenta] =
[i].[numero_cuenta]
END
GO

```

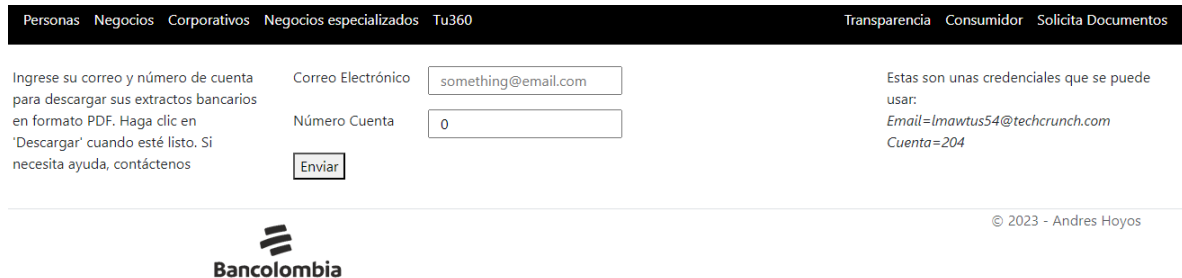
Extractos

Para el objetivo de los extractos, se decidió realizar una aplicación web utilizando ASP.NET Core MVC, y para la interacción con la base de datos se utilizó Entity Framework Core.

Como ya se tenía un esquema de la base de datos, se hizo Ingeniería Inversa a la base de datos (<https://learn.microsoft.com/en-us/ef/core/managing-schemas/scaffolding/>) para crear las entidades en el código.

Para la creación de los PDF se usó iText (<https://itextpdf.com/>).

Funcionamiento



The screenshot shows the Bancolombia website interface. At the top, there is a navigation bar with links: Personas, Negocios, Corporativos, Negocios especializados, Tu360, Transparencia, Consumidor, and Solicita Documentos. Below the navigation bar, there is a form for downloading bank statements. The form has two input fields: 'Correo Electrónico' (Email) with the value 'something@email.com' and 'Número Cuenta' (Account Number) with the value '0'. There is an 'Enviar' (Send) button. To the left of the form, there is text: 'Ingrese su correo y número de cuenta para descargar sus extractos bancarios en formato PDF. Haga clic en "Descargar" cuando esté listo. Si necesita ayuda, contáctenos'. To the right of the form, there is text: 'Estas son unas credenciales que se puede usar: Email=lmawtus54@techcrunch.com Cuenta=204'. At the bottom of the form, there is the Bancolombia logo and the text '© 2023 - Andres Hoyos'.

1. El usuario ingresa el correo y el numero de la cuenta. Si hay coincidencia en la base de datos, se puede generar el extracto, de lo contrario se le notifica al usuario que hay un error con las *credenciales*.
2. Se consulta la base de datos (usando EF Core) y se obtienen los movimientos de la cuenta del ultimo mes.
3. Se pasa la cuenta y sus movimientos al servicio que genera PDFs y este crea un stream que se le regresa al usuario como un archivo.
4. El pdf tiene como contraseña el correo del usuario (por facilidad en el momento)

A la izquierda de la pagina se pueden ver las instrucciones de como usarlo.

A la derecha, se pueden ver unas credenciales que se pueden usar, con el fin de que el examinador de la prueba pueda usarlos. Las credenciales que se muestran son aleatorias.

Requerimientos para correr

1. Tener instalado .NET.
2. Tener instalado una versión de SQL Server.
3. Abrir una consola en el directorio *"prueba-tecnica-bancolombia\BancolombiaExtractos"*
- 4.
5. Correr los comandos:
 - Instalar las herramientas
 - `dotnet tool install --global dotnet-ef`
 - Generar la base de datos
 - `dotnet ef database update`

- Correr la aplicación
 - `dotnet build`
 - `dotnet run`
- 6. Abrir el navegador en la ruta: <https://localhost:7086/> o la que aparezca en la consola

Código fuente

El código fuente y otros archivos se encuentran en el siguiente repositorio de GitHub:
[AndresH18/prueba-tecnica-bancolombia \(github.com\)](https://github.com/AndresH18/prueba-tecnica-bancolombia).