



MANUAL DE INSTALACIÓN Y USO DEL SISTEMA - CONTROL DE GASTOS

REQUISITOS DEL ENTORNO:

Tener instalado Visual Studio 2022 o superior con el paquete de desarrollo de ASP.NET Core y herramientas de Entity Framework. Tener instalado SQL Server 2019 o superior con SQL Server Management Studio (SSMS). Tener instalada la versión de .NET mas reciente preferiblemente la 8.0.

PASO 1 - ABRIR EL PROYECTO:

Descargar o clonar la carpeta del proyecto "ControlGastos". Abrir Visual Studio. Seleccionar "Abrir un proyecto o solución" y buscar el archivo ControlGastos.sln.

<https://github.com/alejandro270519/PruebaTecnicaSMART>

PASO 2 - CONFIGURAR LA CADENA DE CONEXIÓN A LA BASE DE DATOS:

Ir al archivo appsettings.json. Ubicar la sección "ConnectionStrings" y modificar "DefaultConnection" por el nombre del servidor SQL que se ve en SSMS. Ejemplo: "Server=DESKTOP-NOMBRE\SQLEXPRESS;Database= ControlGastosDB;Trusted_Connection = True;TrustServerCertificate=True;". Asegurarse de que el nombre del servidor coincida con el mostrado al conectarse a SQL Server Management Studio.

PASO 3 - ABRIR CONSOLA DEL ADMINISTRADOR DE PAQUETES NUGET:

Ir a Herramientas > Administrador de paquetes NuGet > Consola del Administrador de paquetes.

PASO 4 - VERIFICAR EL PROYECTO POR DEFECTO:

En la parte superior derecha de la consola, en "Proyecto predeterminado", seleccionar ControlGastos.

PASO 5 - INSTALAR EF CORE TOOLS (si no está instalado):

En la consola ejecutar: Install-Package Microsoft.EntityFrameworkCore.Tools

PASO 6 - EJECUTAR MIGRACIONES PARA CREAR LA BASE DE DATOS:

Ejecutar: Add-Migration InitialCreate y luego: Update-Database. Esto creará la base de datos y todas las tablas automáticamente.

PASO 7 - OPCIONAL: USO DE SQL SERVER DIRECTAMENTE:

Si se desea crear las tablas manualmente con scripts en SQL Server Management Studio, se pueden utilizar los siguientes comandos SQL para crear la base de datos y sus tablas (solo si no se usa Entity Framework): (Archivo en el git adjuntado)

NOTA: si se usa este método, no ejecutar migraciones (Add-Migration, Update-Database) para evitar errores de duplicación.

PASO 8 - EJECUTAR LA APLICACIÓN:

Presionar Ctrl + F5 o hacer clic en el botón “Iniciar sin depurar”. Se abrirá el navegador en <http://localhost:puerto>.

PASO 9 - REGISTRAR USUARIOS:

Desde el menú superior, hacer clic en “Registrarse”. El formulario solicita: Identificación, Nombre, Apellido, Fecha de Nacimiento, Dirección, Correo Electrónico y Contraseña. Al registrarse se guarda el usuario en la base de datos.

PASO 10 - AUTENTICARSE:

Luego de registrarse, iniciar sesión con el correo y contraseña para acceder al sistema.

PASO 11 - USO DEL SISTEMA:

Desde el menú puedes acceder a:

- Tipos de Gasto: agregar, editar y eliminar tipos de gasto.
- Fondos: registrar cuentas o cajas chicas.
- Presupuestos: indicar montos por tipo de gasto y mes.
- Gastos: registrar gastos con encabezado y detalle.
- Depósitos: ingresar fondos a cuentas.
- Consulta: mostrar todos los movimientos entre fechas.
- Gráfico: comparar presupuestado vs ejecutado con barras (Chart.js).
-

PASO 12 - CONSIDERACIONES DE FUNCIONALIDAD:

- Cada módulo tiene validación en el servidor.
- El encabezado de gasto no se guarda si no hay detalles.
- Se validan los montos respecto al presupuesto y se alerta si se supera.
- Todos los módulos están protegidos, requieren estar autenticado.
- El sistema es 100% Web y se ejecuta en el navegador.

OBSERVACIONES FINALES:

- No se requiere DevExpress (se reemplazó por Chart.js debido a licencias de DevExpress sin embargo sin eliminar funcionalidades).
- El gráfico es dinámico según las fechas seleccionadas.
- El diseño está adaptado para modo oscuro con inputs personalizados.
- La cadena de conexión debe adaptarse al equipo donde se ejecute.
- Si no ves los estilos o errores, limpiar el navegador con Ctrl+F5.

Desarrollado en ASP.NET Core MVC, SQL Server y Razor Views.

NOTA: SE ADJUNTA UN VIDEO COMPLETO DE FUNCIONAMIENTO Y EJECUCION DEL MISMO PARA LA CORRECTA VALIDACION