**Virtual Catalog**

Es una plataforma basada en la web diseñada para gestionar y mostrar eficientemente un catálogo de productos y categorías. Este proyecto incorpora control de acceso basado en roles (RBAC) y gestión de usuarios para simplificar las operaciones de productos y categorías. Construido con ASP.NET Core en el backend y React en el frontend, Virtual Catalog enfatiza la seguridad, escalabilidad y una experiencia de usuario fluida.

**Características**

* **Gestión de Productos y Categorías:** Crear, actualizar, visualizar y eliminar productos y categorías.
* **Control de Acceso Basado en Roles:** Roles de Administrador y Usuario con permisos específicos.
* **Autenticación JWT y Encriptación con SHA256:** Autenticación segura basada en tokens y encriptación unilateral de contraseñas.
* **API RESTful:** Backend escalable con ASP.NET Core.
* **Frontend Moderno:** Interfaz de usuario interactiva construida con React.
* **Gestión de Migraciones de Base de Datos:** Se utiliza Flyway para administrar las migraciones de la base de datos.

**Tecnologías Utilizadas**

**Backend**

* ASP.NET Core.
* Microsoft SQL Server.
* Flyway (para las migraciones de base de datos).
* ADO.NET

**Frontend**

* React.
* Vite.js.

**Instalación y Configuración**

**Requisitos Previos**

Asegurarse de contar con los siguientes programas instalados:

* .NET SDK 8.0 o superior.
* Node.js 18 o superior (incluye npm).
* Microsoft SQL Server.
* Flyway CLI.

**Datos Iniciales para Iniciar Sesión**

El sistema incluye datos iniciales para facilitar el acceso tanto a usuarios regulares como al administrador. Estos datos pueden ser utilizados para probar las funcionalidades de autenticación y gestión de roles.

**Administrador**

* **Correo Electrónico:** admin@example.com
* **Contraseña:** admin123

**Usuario Regular**

* **Correo Electrónico:** user@example.com
* **Contraseña:** user123

**Nota:** Estos datos iniciales son configurados automáticamente en la base de datos al ejecutar las migraciones de Flyway.

**Backend (VirtualCatalogAPI)**

1. Clona el repositorio del proyecto y navega al directorio del backend.
2. Configura la conexión a la base de datos en el archivo appsettings.json actualizando las credenciales de SQL Server. Asegúrate de incluir la información de tu servidor, usuario y contraseña.
3. Configura Flyway creando un archivo flyway.conf en el directorio raíz del backend. Este archivo debe incluir la URL de conexión a la base de datos, el usuario y la contraseña, además de la ubicación de las migraciones.
4. Ejecuta las migraciones de Flyway para inicializar la base de datos.
5. Restaura las dependencias del backend con el gestor de paquetes .NET.
6. Inicia el servidor del backend. Este estará disponible en la dirección https://localhost:7278.

**Frontend (Virtual-Catalog)**

1. Navega al directorio del frontend dentro del proyecto.
2. Instala las dependencias necesarias utilizando un gestor de paquetes.
3. Configura las variables de entorno creando un archivo .env en el directorio raíz del frontend. Define la URL base para las solicitudes a la API del backend, que debe apuntar a la dirección del servidor.
4. Inicia el servidor de desarrollo del frontend. Este estará disponible en la dirección <http://localhost:5173>.

**Diagrama de Base de Datos**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Diccionario de Datos**

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Column Name | Data Type | Constraints |
| Role | Id | BIGINT | Primary Key, Auto Increment (IDENTITY(1,1)) |
|  | Name | NVARCHAR(100) | Not Null |
| Privilege | Id | BIGINT | Primary Key, Auto Increment (IDENTITY(1,1)) |
|  | Name | NVARCHAR(100) | Not Null |
| User | Id | BIGINT | Primary Key, Auto Increment (IDENTITY(1,1)) |
|  | Name | NVARCHAR(100) | Not Null |
|  | LastName | NVARCHAR(100) | Not Null |
|  | Email | NVARCHAR(255) | Not Null, Unique |
|  | Identification | NVARCHAR(20) | Not Null |
|  | Password | NVARCHAR(255) | Not Null |
| UserRole | UserId | BIGINT | Foreign Key references User(Id), Part of Composite Primary Key |
|  | RoleId | BIGINT | Foreign Key references Role(Id), Part of Composite Primary Key |
| RolePrivilege | RoleId | BIGINT | Foreign Key references Role(Id), Part of Composite Primary Key |
|  | PrivilegeId | BIGINT | Foreign Key references Privilege(Id), Part of Composite Primary Key |
| Category | Id | BIGINT | Primary Key, Auto Increment (IDENTITY(1,1)) |
|  | Name | NVARCHAR(100) | Not Null |
| Product | Id | BIGINT | Primary Key, Auto Increment (IDENTITY(1,1)) |
|  | Name | NVARCHAR(100) | Not Null |
|  | Description | NVARCHAR(MAX) | Nullable |
|  | Price | DECIMAL(18,2) | Not Null |
|  | Stock | INT | Not Null |
|  | CategoryId | BIGINT | Foreign Key references Category(Id) |

**Anexos**

**Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente**

**User**

**Interfaz de usuario gráfica, Aplicación, Sitio web

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Aplicación, Sitio web

Descripción generada automáticamente**

**Interfaz de usuario gráfica

Descripción generada automáticamente**

**Admin**

**Interfaz de usuario gráfica, Aplicación, Sitio web

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Aplicación, Sitio web, Teams

Descripción generada automáticamente**

**Aplicación

Descripción generada automáticamente con confianza baja**

**Interfaz de usuario gráfica, Aplicación, Sitio web, Teams

Descripción generada automáticamente**