



**UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN**



**FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS**

**DISEÑO TÉCNICO DE LA APLICACIÓN**

# **Programación Web II**

---

**Docente: Enrique Tolentino García**

## **Equipo 4**

Abraham Reyes Muñiz.....1803707  
Samara Díaz Hirashi.....1662578  
Edgar Alejandro Niño Sánchez.....1800066  
Andrea Nahomi Rosas Sánchez .....1803204

---

**Grupo: 01**

---

**Abril de 2021**

---



# TIENDA DE MÚJICA

<b>DISEÑO DE BASE DE DATOS</b>	<b>3</b>
<b>DISEÑO DE DIAGRAMA TDM</b>	<b>5</b>
<b>ENDPOINTS</b>	<b>10</b>
<b>MODELOS</b>	<b>7</b>



## DISEÑO DE BASE DE DATOS

Definición: *Una base de datos es un programa capaz de almacenar gran cantidad de datos, relacionados y estructurados, que pueden ser consultados rápidamente de acuerdo con las características selectivas que se deseen.*

Usando: **MySQL**

USER
<b>Username</b> VARCHAR(15) NOT NULL, <b>[Name]</b> VARCHAR(50) NOT NULL, <b>[Password]</b> VARCHAR(15) NOT NULL, <b>Email</b> VARCHAR(30) NOT NULL, <b>PhoneNumber</b> VARCHAR(10) NULL, <b>Active</b> BIT NOT NULL, PRIMARY KEY( <b>Username</b> )

FORMAT
<b>Id</b> INT NOT NULL IDENTITY(1,1), <b>TypeCode</b> VARCHAR(3) NOT NULL, <b>[Type]</b> VARCHAR(15) NOT NULL,  PRIMARY KEY( <b>Id</b> )

GENRE
<b>Id</b> INT NOT NULL IDENTITY(1,1), <b>GenreName</b> VARCHAR(30) NOT NULL,  PRIMARY KEY( <b>Id</b> )

ARTIST
<b>Id</b> INT NOT NULL IDENTITY(1,1), <b>StageName</b> VARCHAR(50) NOT NULL, <b>RealName</b> VARCHAR(50) NOT NULL, <b>[Description]</b> VARCHAR(50) NOT NULL, <b>Active</b> BIT NOT NULL, PRIMARY KEY( <b>Id</b> )

PRODUCT
<b>IId</b> INT NOT NULL IDENTITY(1,1), <b>[Name]</b> VARCHAR(50) NOT NULL, <b>Price</b> DECIMAL NOT NULL, <b>[Description]</b> VARCHAR(MAX) NULL, <b>Active</b> BIT NOT NULL, <b>IdGenre</b> INT NOT NULL, CONSTRAINT FK_ProductGenre FOREIGN KEY (IdGenre) REFERENCES Genre(Id), PRIMARY KEY( <b>Id</b> )

PRODUCTFORMAT
<b>IId</b> INT NOT NULL IDENTITY(1,1), <b>IdProduct</b> INT NOT NULL, <b>IdFormat</b> INT NOT NULL, CONSTRAINT FK_ProductFormatProduct FOREIGN KEY (IdProduct) REFERENCES Product, CONSTRAINT FK_ProductFormatFormat FOREIGN KEY (IdFormat) REFERENCES [Format], PRIMARY KEY( <b>Id</b> )



# DISEÑO DE BASE DE DATOS

Definición: *Una base de datos es un programa capaz de almacenar gran cantidad de datos, relacionados y estructurados, que pueden ser consultados rápidamente de acuerdo con las características selectivas que se deseen.*

Usando: **MySQL**

●	SHOPPING CART	●
	<b>Id</b> INT NOT NULL IDENTITY(1,1), <b>IdProduct</b> INT NOT NULL, <b>Username</b> VARCHAR(15) NOT NULL, <b>CONSTRAINT FK_ShoppingCartProduct</b> <b>FOREIGN KEY</b> (IdProduct) REFERENCES Product(Id), <b>CONSTRAINT FK_ShoppingCartUser</b> <b>FOREIGN KEY</b> (Username) REFERENCES [User](Username), <b>PRIMARY KEY</b> (Id),	

●	ARTISTPRODUCT	●
	<b>Id</b> INT NOT NULL IDENTITY(1,1), <b>IdArtist</b> INT NOT NULL, <b>IdProduct</b> INT NOT NULL,  <b>CONSTRAINT FK_ArtistProductArtist</b> <b>FOREIGN KEY</b> (IdArtist) REFERENCES Artist(Id), <b>CONSTRAINT FK_ArtistProductProduct</b> <b>FOREIGN KEY</b> (IdProduct) REFERENCES Product(Id), <b>PRIMARY KEY</b> (Id)	

●	PHOTOS	●
	<b>Id</b> INT NOT NULL IDENTITY(1,1), <b>[Image]</b> VARBINARY(MAX) NOT NULL, <b>IdProduct</b> INT NOT NULL, <b>CONSTRAINT FK_PhotosProduct</b> <b>FOREIGN KEY</b> (IdProduct) <b>REFERENCES</b> Product(Id), <b>PRIMARY KEY</b> (Id),	

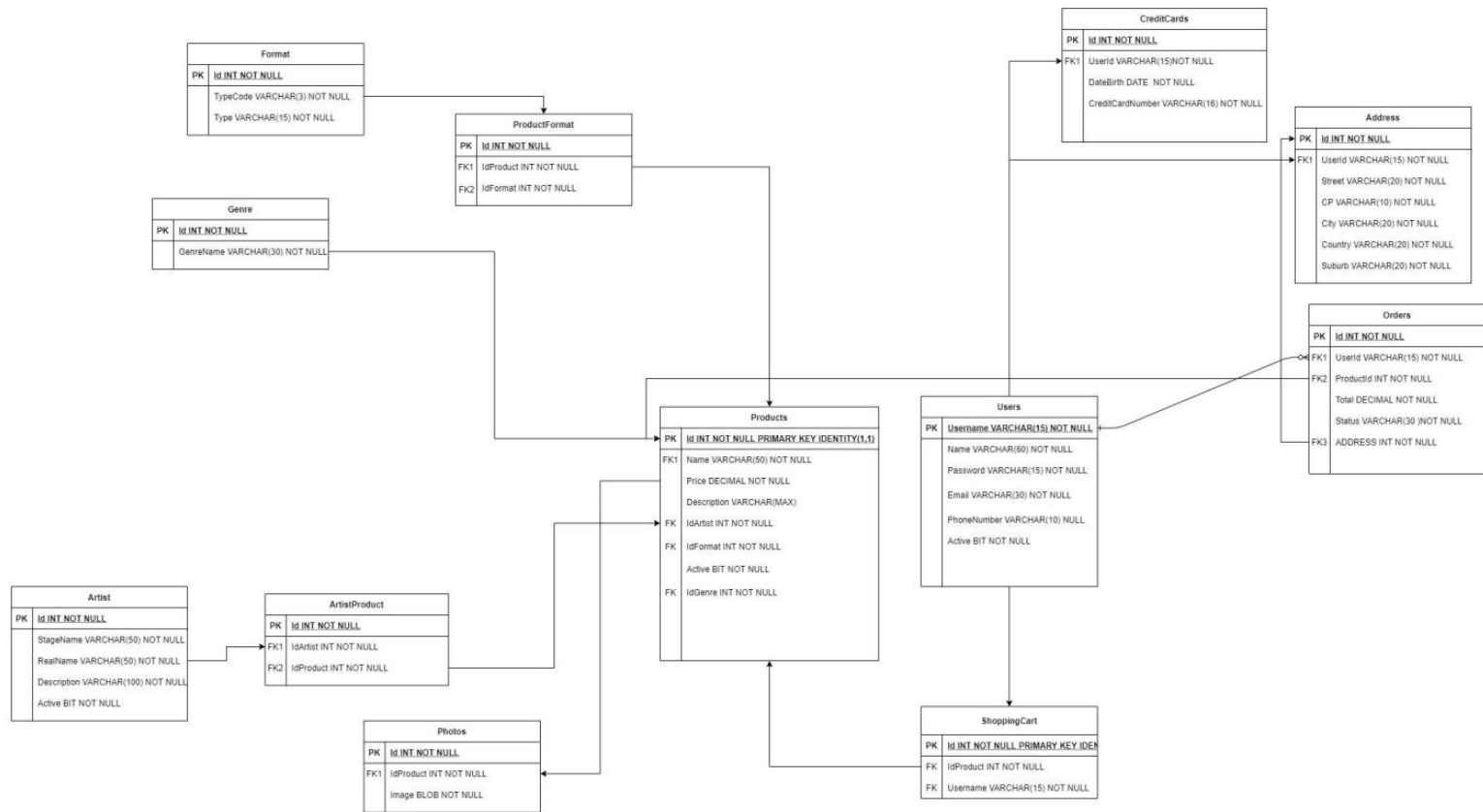
●	ADRESS	●
	<b>Id</b> INT NOT NULL, <b>Street</b> VARCHAR(20) NOT NULL, <b>CP</b> VARCHAR(10) NOT NULL, <b>City</b> VARCHAR(20) NOT NULL, <b>Country</b> VARCHAR(20) NOT NULL, <b>Suburb</b> VARCHAR(20) NOT NULL,  <b>Username</b> VARCHAR(15) NOT NULL, <b>CONSTRAINT FK_AddressUser</b> <b>FOREIGN KEY</b> (Username) REFERENCES [ser](Username),	

●	ORDER	●
	<b>Id</b> INT NOT NULL, <b>Total</b> DECIMAL NOT NULL, <b>[Status]</b> VARCHAR(30) NOT NULL,  <b>IdAddress</b> INT NOT NULL, <b>Username</b> VARCHAR(15) NOT NULL, <b>IdProduct</b> INT NOT NULL,  <b>CONSTRAINT FK_OrderUser</b> <b>FOREIGN KEY</b> (Username) REFERENCES [User] (Username),	

●	ORDERPRODUCT	●
	<b>Id</b> INT NOT NULL,  <b>IdProduct</b> INT NOT NULL, <b>IdOrder</b> INT NOT NULL,  <b>CONSTRAINT FK_OrderProductProduct</b> <b>FOREIGN KEY</b> (IdProduct) REFERENCES Product(Id),  <b>CONSTRAINT FK_OrderProductOrder</b>	



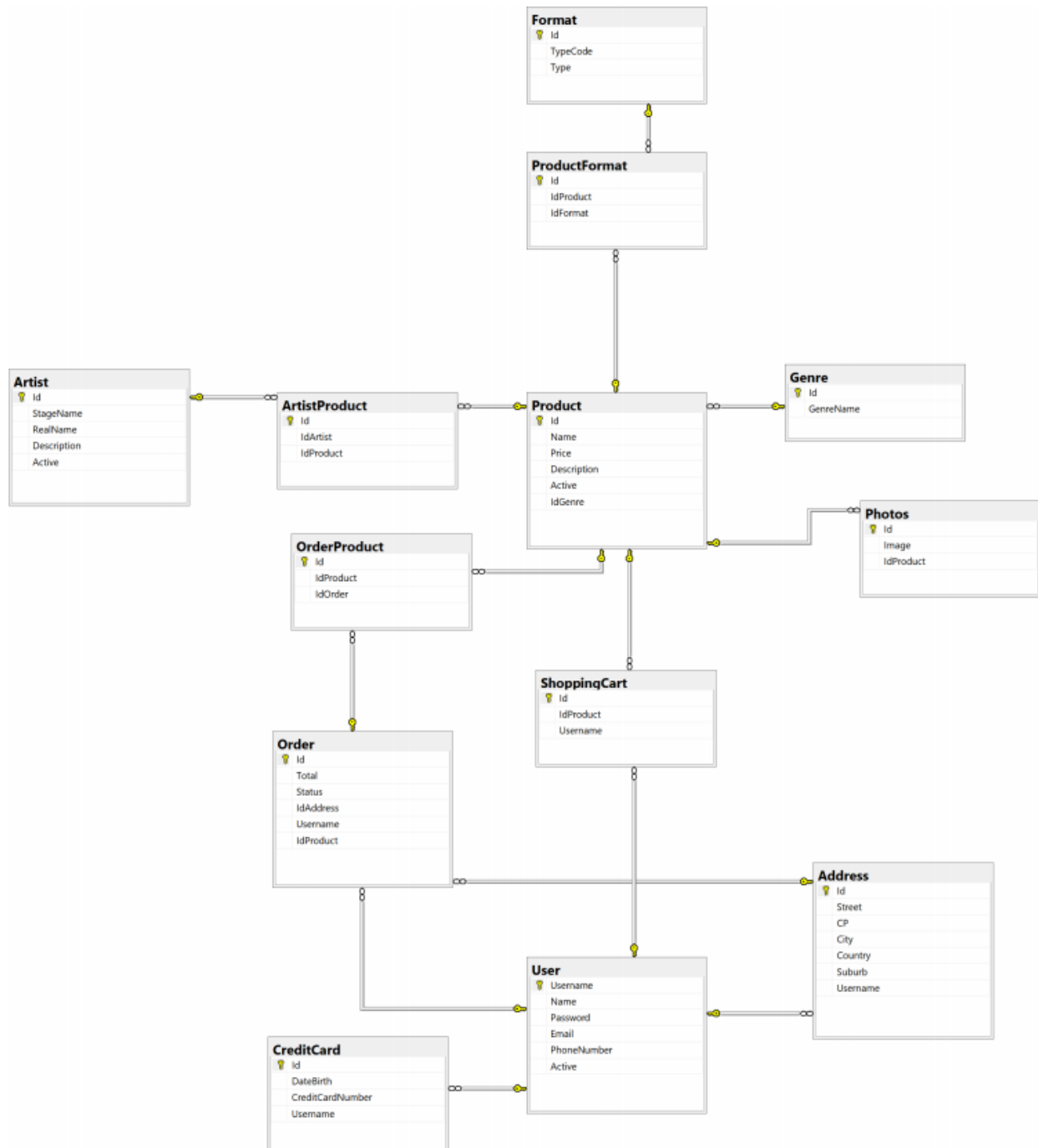
# DISEÑO DE DIAGRAMA TDM





# DISEÑO DE DIAGRAMA TDM

Usando: SQL





# MODELOS

Usando: Visual Studio Code

Address

```
namespace TiendaDeMujica.Models
{
    3referencias
    public class Address
    {
        1referencia
        public int Id { get; set; }
        1referencia
        public string Street { get; set; }
        1referencia
        public string CP { get; set; }
        1referencia
        public string City { get; set; }
        1referencia
        public string Country { get; set; }
        1referencia
        public string Suburb { get; set; }
        1referencia
        public virtual User Username { get; set; }
        //address
    }
}
```

Artist

```
namespace TiendaDeMujica.Models
{
    2referencias
    public class Artist
    {
        1referencia
        public int Id { get; set; }
        1referencia
        public string StageName { get; set; }
        1referencia
        public string RealName { get; set; }
        1referencia
        public string Description { get; set; }
        1referencia
        public bool Active { get; set; }
    }
}
```

ArtistProduct

```
namespace TiendaDeMujica.Models
{
    2referencias
    public class ArtistProduct
    {
        1referencia
        public int Id { get; set; }
        1referencia
        public int IdArtist { get; set; }
        1referencia
        public int IdProduct { get; set; }
        1referencia
        public virtual ICollection<Artist> Artist { get; set; }
        1referencia
        public virtual ICollection<Product> Product { get; set; }
    }
}
```

CreditCard

```
namespace TiendaDeMujica.Models
{
    1referencia
    public class CreditCard
    {
        //credit
        0referencias
        public int Id { get; set; }

        0referencias
        public DateTime DateBirth { get; set; }

        0referencias
        public string CreditCardNumber { get; set; }

        0referencias
        public virtual User Username { get; set; }
    }
}
```

Format

```
namespace TiendaDeMujica.Models
{
    2referencias
    public class Format
    {
        0referencias
        public int Id { get; set; }
        0referencias
        public string TypeCode { get; set; }
        0referencias
        public string Type { get; set; }

        0referencias
        public virtual ICollection<ProductFormat> ProductFormat { get; set; }
    }
}
```

Genre

```
namespace TiendaDeMujica.Models
{
    5referencias
    public class Genre
    {
        1referencia
        public int Id { get; set; }
        0referencias
        public string GenreName { get; set; }
    }
}
```





# MODELOS

Usando: Visual Studio Code

Order

```
namespace TiendaDeMujica.Models
{
    3 referencias
    public class Order
    {
        1 referencia
        public int Id { get; set; }

        1 referencia
        public decimal Total { get; set; }

        1 referencia
        public string Status { get; set; }

        0 referencias
        public virtual Address IdAddress { get; set; }

        1 referencia
        public virtual User Username { get; set; }

        //order
    }
}
```

OrderProduct

```
namespace TiendaDeMujica.Models
{
    1 referencia
    public class OrderProduct
    {
        0 referencias
        public int Id { get; set; }
        0 referencias
        public int IdProduct { get; set; }
        0 referencias
        public string IdOrder { get; set; }
        0 referencias
        public virtual Order Order { get; set; }
        0 referencias
        public virtual Product Product { get; set; }
    }
}
```

Photos

```
namespace TiendaDeMujica.Models
{
    2 referencias
    public class Photos
    {
        1 referencia
        public int Id { get; set; }
        1 referencia
        public int IdProduct { get; set; }
        1 referencia
        public string Image { get; set; }
        1 referencia
        public virtual Product Product { get; set; }
    }
}
```

Product

```
namespace TiendaDeMujica.Models
{
    4 referencias
    public class Product
    {
        0 referencias
        public int Id { get; set; }
        0 referencias
        public string Name { get; set; }
        0 referencias
        public float Price { get; set; }
        0 referencias
        public string? Description { get; set; }
        0 referencias
        public bool Active { get; set; }

        0 referencias
        public int IdGenre { get; set; }

        0 referencias
        public virtual ICollection<ShoppingCart> ShoppingCart { get; set; }
        0 referencias
        public virtual ICollection<OrderProduct> OrderProduct { get; set; }
        0 referencias
        public virtual ICollection<ArtistProduct> ArtistProduct { get; set; }
        0 referencias
        public virtual ICollection<ProductFormat> ProductFormat { get; set; }
        1 referencia
        public virtual ICollection<Photos> Photos { get; set; }

        0 referencias
        public virtual Genre Genre { get; set; }
    }
}
```

ProductFormat

```
namespace TiendaDeMujica.Models
{
    public class ProductFormat
    {
        public int Id { get; set; }
        0 referencias
        public int IdProduct { get; set; }
        0 referencias
        public int IdFormat { get; set; }

        0 referencias
        public virtual Product Product { get; set; }
        0 referencias
        public virtual Format Format { get; set; }
    }
}
```

ShoppingCart

```
namespace TiendaDeMujica.Models
{
    2 referencias
    public class ShoppingCart
    {
        0 referencias
        public int Id { get; set; }
        0 referencias
        public int IdProduct { get; set; }
        0 referencias
        public string Username { get; set; }
        0 referencias
        public virtual User User { get; set; }
        0 referencias
        public virtual User Product { get; set; }
    }
}
```





# MODELOS

Usando: Visual Studio Code

User

```
namespace TiendaDeMujica.Models
{
    public class User
    {
        public string Username { get; set; }

        public string Name { get; set; }
        0 referencias
        public string Password { get; set; }
        0 referencias
        public string Email { get; set; }
        0 referencias
        public string? PhoneNumber { get; set; }
        0 referencias
        public bool Active { get; set; }
        1 referencia
        public virtual ICollection<Address> Address { get; set; }
        1 referencia
        public virtual ICollection<Order> Order { get; set; }
        0 referencias
        public virtual ICollection<ShoppingCart> ShoppingCart { get; set; }
        0 referencias
        public virtual ICollection<CreditCard> CreditCard { get; set; }
    }
}
```

DbContext

```
6 referencias
public class TiendaDeMujicaDbContext : DbContext
{
    //public DbSet<> { get; set; }
    0 referencias
    public DbSet<Format> Format { get; set; }
    1 referencia
    public DbSet<Genre> Genre { get; set; }
    0 referencias
    public DbSet<ProductFormat> ProductFormat { get; set; }

    0 referencias
    public TiendaDeMujicaDbContext()
    {
    }

    0 referencias
    public TiendaDeMujicaDbContext(DbContextOptions<TiendaDeMujicaDbContext> options) : base(options) { }

    0 referencias
    protected override void OnModelCreating(ModelBuilder modelBuilder)
    {
        modelBuilder.Entity<Address>(entity =>{...});
        modelBuilder.Entity<Artist>(entity =>{...});
        modelBuilder.Entity<ArtistProduct>(entity =>{...});
        modelBuilder.Entity<CreditCard>(entity =>{...});
        modelBuilder.Entity<Format>(entity =>{...});
        modelBuilder.Entity<Genre>(entity =>{...});
        modelBuilder.Entity<Order>(entity =>{...});
        modelBuilder.Entity<OrderProduct>(entity =>{...});
        modelBuilder.Entity<Photos>(entity =>{...});
        modelBuilder.Entity<Product>(entity =>{...});
        modelBuilder.Entity<ProductFormat>(entity =>{...});
        modelBuilder.Entity<ShoppingCart>(entity =>{...});
        modelBuilder.Entity<User>(entity =>{...});
    }
}
```



## ENDPOINTS

### Definición

Usando: **Visual Studio Code**

### Address

<b>Ruta</b>	<code>https://localhost:44306/api/Address/Get</code>
<b>Descripción</b>	Trae la dirección de todos los usuarios, Address es la tabla de direcciones de usuario.
<b>Parámetros entrada</b>	*No aplica*
<b>Salida</b>	<pre>{   "Id": "int",   "Street": "string",   "CP": "string",   "City": "string",   "Country": "string",   "Suburb": "string",   "Username": "string" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<code>https://localhost:44306/api/Address/Get/{id}</code>
<b>Descripción</b>	Trae la dirección de un usuario en específico, Address es la tabla de direcciones de usuario.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "Street": "string",   "CP": "string",   "City": "string",   "Country": "string",   "Suburb": "string",   "Username": "string" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"



<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"
----------------------------	--

<b>Ruta</b>	<i>https://localhost:44306/api/Address/Create</i>
<b>Descripción</b>	<i>Crea una nueva dirección(Address) para el usuario. Address es la tabla de direcciones de usuario.</i>
<b>Parámetros entrada</b>	<pre>{   "Street": "string",   "CP": "string",   "City": "string",   "Country": "string",   "Suburb": "string" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "Street": "string",   "CP": "string",   "City": "string",   "Country": "string",   "Suburb": "string",   "Username": "string" }</pre>
<b>Respuestas de éxito</b>	200 - "Added new address successfully!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

<b>Ruta</b>	<i>https://localhost:44306/api/Address/Update/{id}</i>
<b>Descripción</b>	<i>Actualiza la dirección de un usuario específico, Address es la tabla de direcciones de usuario.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "Street": "string",   "CP": "string",   "City": "string",   "Country": "string",   "Suburb": "string",   "Username": "string" }</pre>



<b>Respuestas de éxito</b>	200 - "Address successfully edited!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

<b>Ruta</b>	<code>https://localhost:44306/api/Address/Delete/{id}</code>
<b>Descripción</b>	Elimina la dirección de un usuario específico, Address es la tabla de direcciones de usuario.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{  }</pre>
<b>Respuestas de éxito</b>	200 - "Address successfully deleted!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

## Artist

<b>Ruta</b>	<code>https://localhost:44306/api/Artist/GetAll</code>
<b>Descripción</b>	Trae la información de todos los artistas, Artist es la tabla que contiene la información de los artistas.
<b>Parámetros entrada</b>	*No aplica*
<b>Salida</b>	<pre>{   "Id": "int",   "StageName": "string",   "RealName": "string",   "Description": "string",   "Active": "bool" }</pre>
<b>Respuestas de éxito</b>	200 - "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

<b>Ruta</b>	<code>https://localhost:44306/api/Artist/Get/{id}</code>
-------------	--



<b>Descripción</b>	Trae la información del artista que coincida con el id, Artist es la tabla que contiene la información de los artistas.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "StageName": "string",   "RealName": "string",   "Description": "string",   "Active": "bool" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<a href="https://localhost:44306/api/Artist/Create">https://localhost:44306/api/Artist/Create</a>
<b>Descripción</b>	Crea un nuevo artista (Artist). Artist es la tabla que contiene la información de los artistas.
<b>Parámetros entrada</b>	<pre>{   "StageName": "string",   "RealName": "string",   "Description": "string",   "Active": "bool" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "StageName": "string",   "RealName": "string",   "Description": "string",   "Active": "bool" }</pre>
<b>Respuestas de éxito</b>	200 – "Added new artist successfully!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<a href="https://localhost:44306/api/Artist/Update/{id}">https://localhost:44306/api/Artist/Update/{id}</a>
-------------	---



<b>Descripción</b>	<i>Edita un artista que ya esté creado en la base de datos (Artist). Artist es la tabla que contiene la información de los artistas.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "StageName": "string",   "RealName": "string",   "Description": "string",   "Active": "bool" }</pre>
<b>Respuestas de éxito</b>	200 – "Artist successfully edited!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<i><a href="https://localhost:44306/api/Artist/Disable/{id}">https://localhost:44306/api/Artist/Disable/{id}</a></i>
<b>Descripción</b>	<i>Deshabilita la información de un artista específico, Artist es la tabla que contiene la información de los artistas.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{  }</pre>
<b>Respuestas de éxito</b>	200 – "Artist successfully disabled!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

## ArtistProduct

<b>Ruta</b>	<i><a href="https://localhost:44306/api/ArtistProduct/GetAll">https://localhost:44306/api/ArtistProduct/GetAll</a></i>
<b>Descripción</b>	<i>Trae la información de todos los productos de los artistas, ArtistProduct es la tabla que contiene los productos de los artistas.</i>
<b>Parámetros entrada</b>	<pre>{   *no aplica* }</pre>





<b>Salida</b>	<pre>{   "Id": "int",   "IdArtist": "int",   "IdProduct": "int" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<a href="https://localhost:44306/api/ArtistProduct/Get/{id}">https://localhost:44306/api/ArtistProduct/Get/{id}</a>
<b>Descripción</b>	Trae la información de un producto en específico de un artista. ArtistProduct es la tabla que contiene los productos de los artistas.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdArtist": "int",   "IdProduct": "int" }</pre>
<b>Respuestas de éxito</b>	200 – "Success!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<a href="https://localhost:44306/api/ArtistProduct/Create">https://localhost:44306/api/ArtistProduct/Create</a>
<b>Descripción</b>	Crea una nueva relación entre un producto y su artista, ArtistProduct es la tabla que contiene los productos de los artistas.
<b>Parámetros entrada</b>	<pre>{   "IdArtist": "int",   "IdProduct": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdArtist": "int",   "IdProduct": "int" }</pre>
<b>Respuestas de éxito</b>	200 – "Added new artist-product successfully!"



<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"
----------------------------	--

<b>Ruta</b>	<code>https://localhost:44306/api/ArtistProduct/Update/{id}</code>
<b>Descripción</b>	Edita la relación entre un producto y su artista. ArtistProduct es la tabla que contiene los productos de los artistas.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdArtist": "int",   "IdProduct": "int" }</pre>
<b>Respuestas de éxito</b>	200 - "Artist-product successfully edited!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

<b>Ruta</b>	<code>https://localhost:44306/api/ArtistProduct/Delete/{id}</code>
<b>Descripción</b>	Elimina la relación entre un artista y su producto, ArtistProduct es la tabla que contiene los productos de los artistas.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{ }</pre>
<b>Respuestas de éxito</b>	200 - "Artist-product successfully deleted!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

## CreditCard

<b>Ruta</b>	<code>https://localhost:44306/api/CreditCard/GetAll</code>
-------------	--



<b>Descripción</b>	<i>Trae todas las tarjetas de crédito de todos los usuarios. CreditCard es la tabla que contiene las tarjetas de crédito de cada usuario.</i>
<b>Parámetros entrada</b>	<pre>{   *no aplica* }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "DateBirth": "DateTime",   "CreditCardNumber": "string",   "Username": "string" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<i><a href="https://localhost:44306/api/CreditCard/Get/{id}">https://localhost:44306/api/CreditCard/Get/{id}</a></i>
<b>Descripción</b>	<i>Trae la información de las tarjetas de crédito de un usuario en específico. CreditCard es la tabla que contiene las tarjetas de crédito de cada usuario.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "DateBirth": "DateTime",   "CreditCardNumber": "string",   "Username": "string" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<i><a href="https://localhost:44306/api/CreditCard/Create">https://localhost:44306/api/CreditCard/Create</a></i>
<b>Descripción</b>	<i>Crea la información de una tarjeta de crédito de un usuario. CreditCard es la tabla que contiene las tarjetas de crédito de cada usuario.</i>
<b>Parámetros entrada</b>	<pre>{   "DateBirth": "DateTime",</pre>



	<pre>"CreditCardNumber": "string", "Username": "string" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "DateBirth": "DateTime",   "CreditCardNumber": "string",   "Username": "string" }</pre>
<b>Respuestas de éxito</b>	200 – "Added new credit card successfully!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<a href="https://localhost:44306/api/CreditCard/Update/{id}">https://localhost:44306/api/CreditCard/Update/{id}</a>
<b>Descripción</b>	Edita la información de la tarjeta de crédito del usuario. CreditCard es la tabla que contiene las tarjetas de crédito de cada usuario.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "DateBirth": "DateTime",   "CreditCardNumber": "string",   "Username": "string" }</pre>
<b>Respuestas de éxito</b>	200 – "Credit card successfully edited!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<a href="https://localhost:44306/api/CreditCard/Delete/{id}">https://localhost:44306/api/CreditCard/Delete/{id}</a>
<b>Descripción</b>	Elimina la información de un usuario específico. CreditCard es la tabla que contiene las tarjetas de crédito de cada usuario.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{ }</pre>
<b>Respuestas de éxito</b>	200 – "Credit card successfully deleted!"



<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"
----------------------------	--

## Format

<b>Ruta</b>	https://localhost:44306/api/Format/GetAll
<b>Descripción</b>	Trae todos los tipos de formatos de los productos. Format es la tabla que contiene la información del género del producto.
<b>Parámetros entrada</b>	{ *no aplica* }
<b>Salida</b>	{ "Id": "int", "IdProduct": "int", "IdFormat": "int" }
<b>Respuestas de éxito</b>	200 - "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

<b>Ruta</b>	https://localhost:44306/api/Format/Get/{id}
<b>Descripción</b>	Trae la información de un formato de un producto en específico. Format es la tabla que contiene la información del género del producto.
<b>Parámetros entrada</b>	{ "Id": "int" }
<b>Salida</b>	{ "Id": "int", "IdProduct": "int", "IdFormat": "int" }
<b>Respuestas de éxito</b>	200 - "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

<b>Ruta</b>	https://localhost:44306/api/Format/Create
-------------	---



<b>Descripción</b>	<i>Crea una nueva relación entre el producto y el tipo de formato. Format es la tabla que contiene la información del género del producto.</i>
<b>Parámetros entrada</b>	<pre>{   "IdProduct": "int",   "IdFormat": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "IdFormat": "int" }</pre>
<b>Respuestas de éxito</b>	200 – "Added new format successfully!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<i><a href="https://localhost:44306/api/Format/Update/{id}">https://localhost:44306/api/Format/Update/{id}</a></i>
<b>Descripción</b>	<i>Edita la información del formato de un producto. Format es la tabla que contiene la información del género del producto.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "IdFormat": "int" }</pre>
<b>Respuestas de éxito</b>	200 – "Format successfully edited!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<i><a href="https://localhost:44306/api/Format/Disable/{id}">https://localhost:44306/api/Format/Disable/{id}</a></i>
<b>Descripción</b>	<i>Elimina la información del formato de un producto. Format es la tabla que contiene la información del género del producto.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>





<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "IdFormat": "int" }</pre>
<b>Respuestas de éxito</b>	200 – "Format successfully disabled!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

## Genre

<b>Ruta</b>	<a href="https://localhost:44306/api/Genre/GetAll">https://localhost:44306/api/Genre/GetAll</a>
<b>Descripción</b>	Trae todos los tipos de géneros. Genre es la tabla que tiene los tipos de géneros del producto.
<b>Parámetros entrada</b>	<pre>{   *no aplica* }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "GenreName": "string",   "Active": "bool" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<a href="https://localhost:44306/api/Genre/Get/{id}">https://localhost:44306/api/Genre/Get/{id}</a>
<b>Descripción</b>	Trae el género específico de los tipos de géneros. Genre es la tabla que tiene los tipos de géneros del producto.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "GenreName": "string",   "Active": "bool" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"



<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"
----------------------------	--

<b>Ruta</b>	<i>https://localhost:44306/api/Genre/Create</i>
<b>Descripción</b>	<i>Crea la relación entre un género y su producto. Genre es la tabla que tiene los tipos de géneros del producto.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int",   "GenreName": "string",   "Active": "bool" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "GenreName": "string",   "Active": "true" }</pre>
<b>Respuestas de éxito</b>	200 - "Added new genre successfully!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

<b>Ruta</b>	<i>https://localhost:44306/api/Genre/Update/{id}</i>
<b>Descripción</b>	<i>Edita el tipo de género. Genre es la tabla que tiene los tipos de géneros del producto.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "GenreName": "string",   "Active": "bool" }</pre>
<b>Respuestas de éxito</b>	200 - "Genre successfully edited!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

<b>Ruta</b>	<i>https://localhost:44306/api/Genre/Disable/{id}</i>
<b>Descripción</b>	<i>Elimina el tipo de género. Genre es la tabla que tiene los tipos de géneros del producto.</i>



<b>Parámetros entrada</b>	{ "Id": "int" }
<b>Salida</b>	{ "Active": "false" }
<b>Respuestas de éxito</b>	200 – "Genre successfully disabled!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

## Order

<b>Ruta</b>	<a href="https://localhost:44306/api/Order/Get">https://localhost:44306/api/Order/Get</a>
<b>Descripción</b>	Trae todas las órdenes. Order es la tabla que tiene todas las órdenes.
<b>Parámetros entrada</b>	{ }
<b>Salida</b>	{ "Id": "int", "Total": "decimal", "Status": "string", "IdAdress": "int", "Username": "string" }
<b>Respuestas de éxito</b>	200 – "Success!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<a href="https://localhost:44306/api/Order/Get/{id}">https://localhost:44306/api/Order/Get/{id}</a>
<b>Descripción</b>	Trae las órdenes específicas que trae por id. Order es la tabla que tiene todas las órdenes.
<b>Parámetros entrada</b>	{ "Id": "int" }
<b>Salida</b>	{ "Id": "int", "Total": "decimal", "Status": "string", "IdAdress": "int", "Username": "string" }



<b>Respuestas de éxito</b>	200 – "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<a href="https://localhost:44306/api/Order/Create">https://localhost:44306/api/Order/Create</a>
<b>Descripción</b>	Crea una nueva entrada con información sobre una Orden. Order es la tabla que tiene todas las órdenes.
<b>Parámetros entrada</b>	<pre>{   "Id": "int",   "Total": "decimal",   "Status": "string",   "IdAdress": "int",   "Username": "string" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "Total": "decimal",   "Status": "string",   "IdAdress": "int",   "Username": "string" }</pre>
<b>Respuestas de éxito</b>	200 – "Added new order successfully!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<a href="https://localhost:44306/api/Order/Update/{id}">https://localhost:44306/api/Order/Update/{id}</a>
<b>Descripción</b>	Edita la información sobre una Orden. Order es la tabla que tiene todas las órdenes.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "Total": "decimal",   "Status": "string",   "IdAdress": "int",   "Username": "string" }</pre>
<b>Respuestas de éxito</b>	200 – "Order successfully edited!!"



<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"
----------------------------	--

<b>Ruta</b>	<i>https://localhost:44306/api/Order/Disable/{id}</i>
<b>Descripción</b>	<i>Elimina la información sobre una Orden. Order es la tabla que tiene todas las órdenes.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Status": "delivered" }</pre>
<b>Respuestas de éxito</b>	200 - "Order successfully disabled!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

## OrderProduct

<b>Ruta</b>	<i>https://localhost:44306/api/OrderProduct/Get</i>
<b>Descripción</b>	<i>Trae toda la información de la relación entre la orden y el producto. OrderProduct es la tabla que tiene todas los productos de una orden..</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "IdOrder": "int" }</pre>
<b>Respuestas de éxito</b>	200 - "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

<b>Ruta</b>	<i>https://localhost:44306/api/OrderProduct/Get/{id}</i>
-------------	--



<b>Descripción</b>	<i>Trae la información en específico de una relación entre la orden y el producto. OrderProduct es la tabla que tiene todas los productos de una orden.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "IdOrder": "int" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<i><a href="https://localhost:44306/api/OrderProduct/Create">https://localhost:44306/api/OrderProduct/Create</a></i>
<b>Descripción</b>	<i>Crea una nueva información en específico de una relación entre la orden y el producto. OrderProduct es la tabla que tiene todas los productos de una orden..</i>
<b>Parámetros entrada</b>	<pre>{   "IdProduct": "int",   "IdOrder": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "IdOrder": "int" }</pre>
<b>Respuestas de éxito</b>	200 – "Added new orderProduct successfully"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<i><a href="https://localhost:44306/api/OrderProduct/Update/{id}">https://localhost:44306/api/OrderProduct/Update/{id}</a></i>
<b>Descripción</b>	<i>Edita la información en específico de una relación entre la orden y el producto. OrderProduct es la tabla que tiene todas los productos de una orden..</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>





<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "IdOrder": "int" }</pre>
<b>Respuestas de éxito</b>	200 – "OrderProduct successfully edited!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<code>https://localhost:44306/api/OrderProduct/Delete/{id}</code>
<b>Descripción</b>	Elimina la información en específico de una relación entre la orden y el producto. OrderProduct es la tabla que tiene todas los productos de una orden.
<b>Parámetros entrada</b>	<pre>{   "IdOrder": "int" }</pre>
<b>Salida</b>	<pre>{ }</pre>
<b>Respuestas de éxito</b>	200 – "OrderProduct successfully deleted!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

## Product

<b>Ruta</b>	<code>https://localhost:44306/api/Product/Get</code>
<b>Descripción</b>	Trae toda la información de los productos. Product es la tabla que tiene todos los productos de la tienda.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "Name": "string",   "Price": "decimal",   "Description": "string",   "Active": "bool",   "IdGenre": "int" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"



<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

<b>Ruta</b>	https://localhost:44306/api/Product/Get/{id}
<b>Descripción</b>	Trae toda la información de los productos. Product es la tabla que tiene todos los productos de la tienda.
<b>Parámetros entrada</b>	{ }
<b>Salida</b>	{ "Id": "int", "Name": "string", "Price": "decimal", "Description": "string", "Active": bool, "IdGenre": int }
<b>Respuestas de éxito</b>	200 - "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

<b>Ruta</b>	https://localhost:44306/api/Product/GetProductFormats/{id}
<b>Descripción</b>	Trae toda la información de los productos con sus formatos. Product es la tabla que tiene todos los productos de la tienda.
<b>Parámetros entrada</b>	{ "Id": "int" }
<b>Salida</b>	{ "Id": "int", "Name": "string", "Price": "decimal", "Description": "string", "Active": bool, "IdGenre": int }
<b>Respuestas de éxito</b>	200 - "Success"



<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"
----------------------------	--

<b>Ruta</b>	<i>https://localhost:44306/api/Product/Create</i>
<b>Descripción</b>	<i>Crea la información de un nuevo producto. Product es la tabla que tiene todos los productos de la tienda.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int",   "Name": "string",   "Price": "decimal",   "Description": "string",   "Active": bool,   "IdGenre": int }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "Name": "string",   "Price": "decimal",   "Description": "string",   "Active": bool,   "IdGenre": int }</pre>
<b>Respuestas de éxito</b>	200 - "Added new product successfully!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

<b>Ruta</b>	<i>https://localhost:44306/api/Product/Update/{id}</i>
<b>Descripción</b>	<i>Edita la información de un producto. Product es la tabla que tiene todos los productos de la tienda.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "Name": "string",   "Price": "decimal",   "Description": "string",   "Active": bool,   "IdGenre": int }</pre>
<b>Respuestas de éxito</b>	200 - "Product successfully edited!"



<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"
<b>Ruta</b>	<i>https://localhost:44306/api/Product/Disable/{id}</i>
<b>Descripción</b>	<i>Elimina la información de un producto. Product es la tabla que tiene todos los productos de la tienda.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int",   "Active": "false" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "Name": "string",   "Price": "decimal",   "Description": "string",   "Active": "false",   "IdGenre": "int" }</pre>
<b>Respuestas de éxito</b>	200 - "Product successfully disabled!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"

## ProductFormat

<b>Ruta</b>	<i>https://localhost:44306/api/ProductFormat/GetAll</i>
<b>Descripción</b>	<i>Trae la información entre la relación de un producto y su formato. ProductFormat es la tabla que tiene la información del formato de cada producto.</i>
<b>Parámetros entrada</b>	<pre>{  }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "IdFormat": "int" }</pre>
<b>Respuestas de éxito</b>	200 - "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"



<b>Ruta</b>	<i>https://localhost:44306/api/ProductFormat/Get/{id}</i>
<b>Descripción</b>	<i>Trae la información la relación de un producto en específico y su formato. ProductFormat es la tabla que tiene la información del formato de cada producto.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "IdFormat": "int" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<i>https://localhost:44306/api/ProductFormat/Create</i>
<b>Descripción</b>	<i>Crea una nueva relación de un producto en específico y su formato. ProductFormat es la tabla que tiene la información del formato de cada producto.</i>
<b>Parámetros entrada</b>	<pre>{   "IdProduct": "int",   "IdFormat": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "IdFormat": "int" }</pre>
<b>Respuestas de éxito</b>	200 – "Added new orderProduct successfully!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<i>https://localhost:44306/api/ProductFormat/Update/{id}</i>
<b>Descripción</b>	<i>Edita la información la relación de un producto en específico y su formato. ProductFormat es la tabla que tiene la información del formato de cada producto.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>



<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "IdFormat": "int" }</pre>
<b>Respuestas de éxito</b>	200 – "Product-format successfully edited!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<a href="https://localhost:44306/api/ProductFormat/Delete/{id}">https://localhost:44306/api/ProductFormat/Delete/{id}</a>
<b>Descripción</b>	Elimina la información la relación de un producto en específico y su formato. ProductFormat es la tabla que tiene la información del formato de cada producto.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{ }</pre>
<b>Respuestas de éxito</b>	200 – "Product-format successfully deleted!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

## Security

<b>Ruta</b>	<a href="https://localhost:44306/api/Security/CreateUser">https://localhost:44306/api/Security/CreateUser</a>
<b>Descripción</b>	Crea la seguridad del usuario.
<b>Parámetros entrada</b>	<pre>{   "name": "string",   "userName": "string",   "phoneNumber": "string",   "email": "string",   "password": "string" }</pre>
<b>Salida</b>	<pre>{   "name": "string",   "userName": "string",   "phoneNumber": "string",   "email": "string",   "password": "string" }</pre>





<b>Respuestas de éxito</b>	200 – "Usuario creado exitosamente"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message" 500 – "Llena todos los campos" 500 – "Ingresa un email válido" 500 – "Ingresa una contraseña mayor a 4 caracteres" 500 – "Ingresa un número telefónico de 10 caracteres"

<b>Ruta</b>	<i>https://localhost:44306/api/Security/Login</i>
<b>Descripción</b>	<i>Permite al usuario entrar a la aplicación web.</i>
<b>Parámetros entrada</b>	{ "userName": "string", "password": "string" }
<b>Salida</b>	{ "userName": "string", "password": "string" }
<b>Respuestas de éxito</b>	200 – "Ok"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exception Message" 500 – "Usuario o contraseña inválidos" 500 – "El usuario no existe"

## ShoppingCart

<b>Ruta</b>	<i>https://localhost:44306/api/ShoppingCart/Get</i>
<b>Descripción</b>	<i>Trae la información de los elementos del carrito de compras. ShoppingCart es la tabla que tiene la información del carrito de compras.</i>
<b>Parámetros entrada</b>	{ }
<b>Salida</b>	{ "Id": "int", "IdProduct": "int", "Username": "string" }
<b>Respuestas de éxito</b>	200 – "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"



<b>Ruta</b>	<code>https://localhost:44306/api/ShoppingCart/Get/{id}</code>
<b>Descripción</b>	<i>Trae la información de elementos específicos del carrito de compras. ShoppingCart es la tabla que tiene la información del carrito de compras.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "Username": "string" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<code>https://localhost:44306/api/ShoppingCart/Create</code>
<b>Descripción</b>	<i>Crea la información de un nuevo elemento del carrito de compras. ShoppingCart es la tabla que tiene la información del carrito de compras.</i>
<b>Parámetros entrada</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "Username": "string" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "Username": "string" }</pre>
<b>Respuestas de éxito</b>	200 – "Added new shoppingCart successfully!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<code>https://localhost:44306/api/ShoppingCart/Update/{id}</code>
<b>Descripción</b>	<i>Edita la información de un elemento del carrito de compras. ShoppingCart es la tabla que tiene la información del carrito de compras.</i>
<b>Parámetros entrada</b>	<pre>{</pre>



	<pre>"Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Id": "int",   "IdProduct": "int",   "Username": "string" }</pre>
<b>Respuestas de éxito</b>	200 – "ShoppingCart successfully edited!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<a href="https://localhost:44306/api/ShoppingCart/Delete/{id}">https://localhost:44306/api/ShoppingCart/Delete/{id}</a>
<b>Descripción</b>	Elimina la información de un elemento del carrito de compras. ShoppingCart es la tabla que tiene la información del carrito de compras.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{ }</pre>
<b>Respuestas de éxito</b>	200 – "ShoppingCart successfully deleted!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

## User

<b>Ruta</b>	<a href="https://localhost:44306/api/User/Get">https://localhost:44306/api/User/Get</a>
<b>Descripción</b>	Trae la información de todos los elementos de los usuarios. User es la tabla que tiene la información de los usuarios.
<b>Parámetros entrada</b>	<pre>{ }</pre>
<b>Salida</b>	<pre>{   "Username": "string",   "Name": "string",   "Password": "string",   "Email": "string",   "PhoneNumber": "string", }</pre>



	<pre>"Active": "bool" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<code>https://localhost:44306/api/User/Get/{id}</code>
<b>Descripción</b>	Trae la información de un elemento en específico de los usuarios. User es la tabla que tiene la información de los usuarios.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Username": "string",   "Name": "string",   "Password": "string",   "Email": "string",   "PhoneNumber": "string",   "Active": "bool" }</pre>
<b>Respuestas de éxito</b>	200 – "Success"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 – "Exeption Message"

<b>Ruta</b>	<code>https://localhost:44306/api/User/Update/{id}</code>
<b>Descripción</b>	Edita la información de un elemento en específico de los usuarios. User es la tabla que tiene la información de los usuarios.
<b>Parámetros entrada</b>	<pre>{   "Id": "int" }</pre>
<b>Salida</b>	<pre>{   "Username": "string",   "Name": "string",   "Password": "string",   "Email": "string",   "PhoneNumber": "string",   "Active": "bool" }</pre>
<b>Respuestas de éxito</b>	200 – "User successfully edited!"

**Respuestas de Error**

400 - "Invalid Action"  
500 - "Exeption Message"

<b>Ruta</b>	https://localhost:44306/api/User/Disable/{id}
<b>Descripción</b>	Elimina la información de un elemento en específico de los usuarios. User es la tabla que tiene la información de los usuarios.
<b>Parámetros entrada</b>	{ "Id": "int" }
<b>Salida</b>	{ "Active": "false" }
<b>Respuestas de éxito</b>	200 - "User successfully disabled!"
<b>Respuestas de Error</b>	400 - "Invalid Action" 500 - "Exeption Message"