

UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA
UNAN – LEÓN
FACULTAD DE CIENCIAS Y TECNOLOGÍA
INGENIERÍA EN SISTEMAS DE LA INFORMACIÓN



AÑO LECTIVO: 2025
SEMESTRE: II

Componente Curricular: **PROGRAMACION
ORIENTADA A LA WEB II**

Grupo: 1

Sub-Grupo:

Profesor(a): JUAN CARLOS LEYTON BRIONES

Autor:

Luis Ignacio Hidalgo Molina.

León, Nicaragua, 2025.

“¡A la Libertad por la Universidad!”

Propósito y Alcance

Mundo GPU es una aplicación web para la gestión de inventario de GPUs, desarrollada con ASP.NET Core MVC. El sistema ofrece funcionalidades completas para administrar catálogos de GPU, cuentas de usuario e información de proveedores, con controles de acceso basados en roles y capacidades de exportación a Excel.

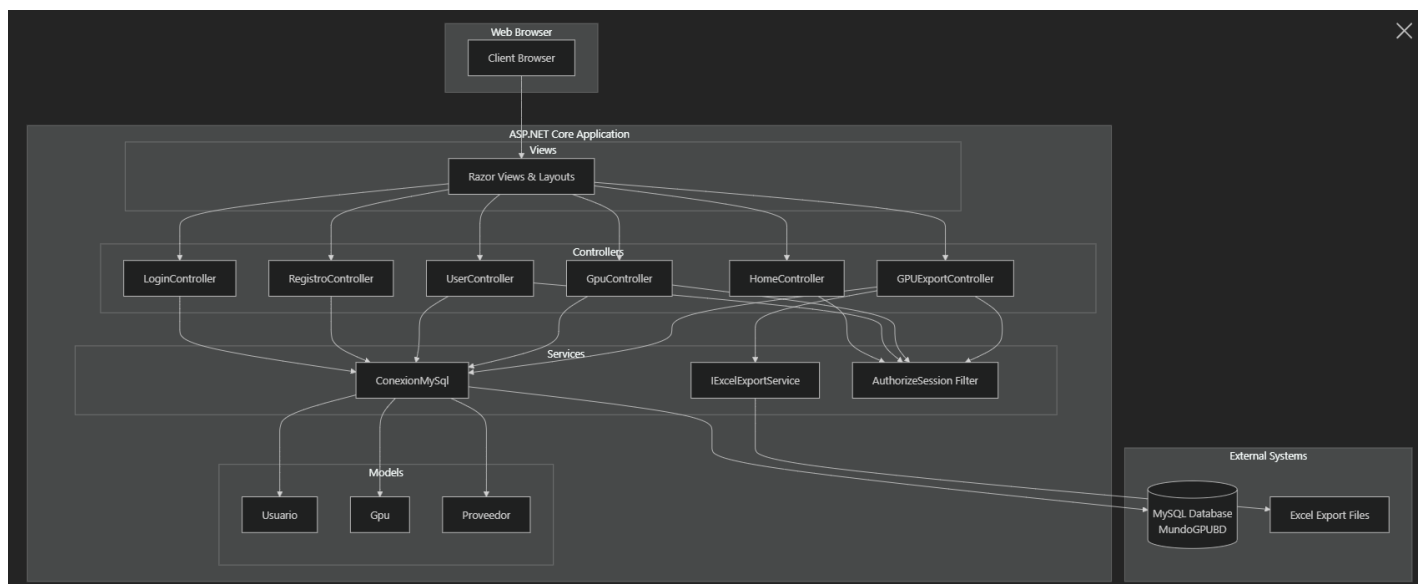
Este documento proporciona una introducción general a la arquitectura del sistema, los dominios de negocio principales y la pila tecnológica. Para información detallada sobre subsistemas específicos, consulta:

- **Arquitectura del Sistema:** patrones arquitectónicos
- **Diseño de Base de Datos:** modelado de datos
- **Gestión de Usuarios:** autenticación y autorización
- **Gestión de GPUs:** operaciones de inventario

Arquitectura del Sistema

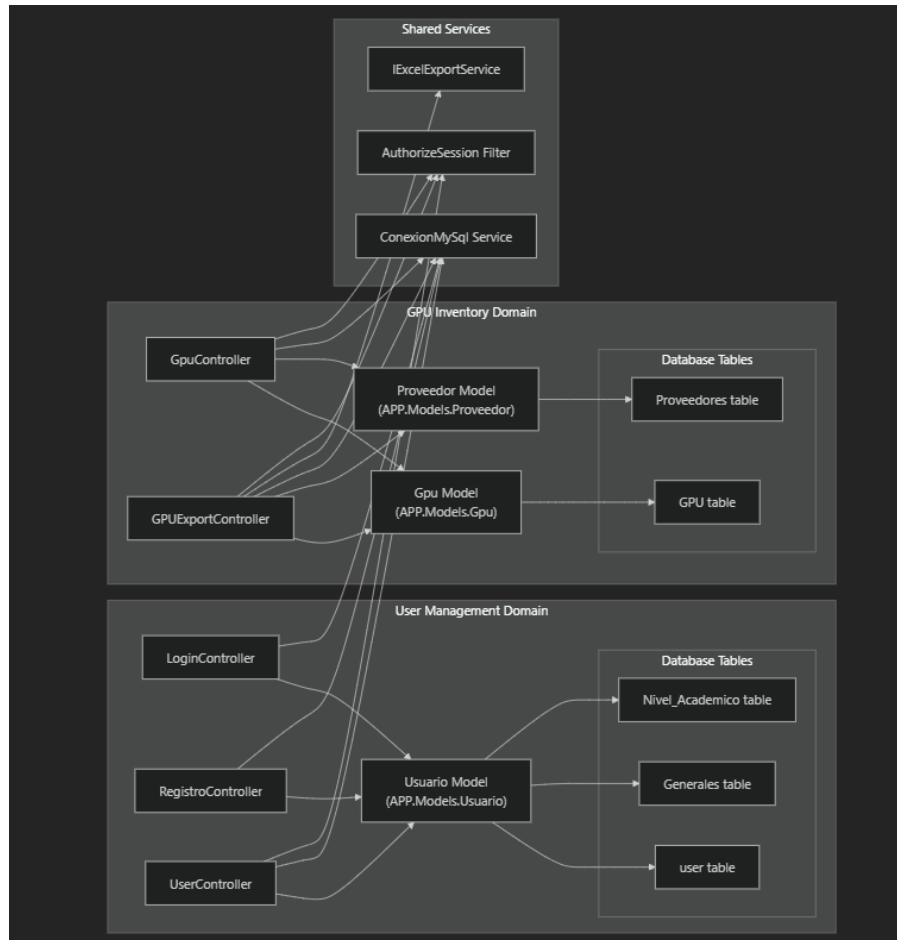
La aplicación sigue una arquitectura en capas basada en MVC, con una separación clara entre:

- Capa de presentación
- Lógica de negocio
- Acceso a datos



Dominios de Negocio Principales

El sistema se organiza en torno a dos dominios de negocio principales que se reflejan directamente en las entidades del código:



Pila Tecnológica y Dependencias

Componente	Tecnología	Ubicación de Configuración
Framework Web	ASP.NET Core 9.0 MVC	APP/Program.cs línea 12
Base de Datos	MySQL (esquema MundoGPUBD)	BD/BD2.sql línea 25
Acceso a Datos	Servicio personalizado ConexionMySQL	APP/Program.cs línea 18
Autenticación	Basada en sesión con filtro AuthorizeSession	APP/Program.cs líneas 24–30
Motor de Exportación	Implementación de IExcelExportService	APP/Program.cs línea 15

Componente	Tecnología	Ubicación de Configuración
Gestión de Sesiones	Caché distribuido en memoria	APP/Program.cs línea 24

Registro de Servicios Clave

Los servicios principales se configuran en el contenedor de inyección de dependencias:

- **Servicios MVC:** `AddControllersWithViews()` habilita el patrón MVC con vistas Razor
- **Exportación a Excel:** `AddScoped<IExcelExportService, ExcelExportService>()`
- **Acceso a Base de Datos:** `AddScoped<ConexionMySQL>()`
- **Estado de Sesión:** tiempo de espera de 30 minutos, cookies HTTP-only esenciales
- **Contexto HTTP:** `AddHttpContextAccessor()` para acceder al contexto de solicitud

Fuente: APP/Program.cs líneas 11–30

Roles de Usuario y Control de Acceso

El sistema implementa un control de acceso basado en roles con tres niveles definidos en el esquema de base de datos:

Rol	Valor en BD	Capacidades
Administrador	ADMIN	Acceso total, gestión de usuarios, CRUD de GPU, operaciones de borrado
Encargado	ENCARGADO	Gestión de GPU, exportación a Excel, operaciones limitadas de usuario
Empleado	EMPLEADO	Visualización del catálogo de GPU, operaciones básicas

La jerarquía de roles se aplica mediante:

- Restricción ENUM en BD: `ENUM('ADMIN', 'ENCARGADO', 'EMPLEADO')` en la tabla de usuarios
- Validación en modelo: atributo `RegularExpression` en `Usuario.Rol`
- Autorización en vistas: verificación de rol basada en sesión en Razor
- Filtro en controladores: `AuthorizeSession` para acciones protegidas

Fuentes:

- BD/BD2.sql línea 67

- APP/Models/Usuario.cs líneas 19–22
- APP/Views/GPU/index.cshtml líneas 10–19

Entidades de Datos Principales

El sistema gestiona tres entidades clave con las siguientes propiedades:

Entidad GPU (Gpu)

- **Especificaciones Técnicas:** Marca, Modelo, VRAM, NúcleosCuda, RayTracing
- **Datos Comerciales:** Precio, URL de imagen, relación con proveedor vía ProveedoresIdProveedor
- **Tabla en BD:** GPU con clave primaria idGPU auto-incremental

Entidad Usuario (Usuario)

- **Autenticación:** Username (máx. 16 caracteres), Password (máx. 32), Rol (enum)
- **Información Personal:** Nombre, Apellido, Sexo (enum M/F)
- **Información Académica:** NivelAcadémico, Institución, vinculado por NivelAcademicoId
- **Tablas en BD:** Normalizadas en usuario, generales y nivel_academico

Entidad Proveedor (Proveedor)

- **Contacto:** Nombre, Dirección, Teléfono, Email
- **Relación Comercial:** Relación uno-a-muchos con entidades GPU
- **Tabla en BD:** Proveedores con datos de fabricantes principales

Fuentes:

- APP/Models/GPUViewModel.cs líneas 15–31
- APP/Models/Usuario.cs líneas 5–52
- BD/BD2.sql líneas 31–116