

**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: **PROGRAMACIÓN**  
**ORIENTADA A LA WEB II**

**Grupo: 1**  
**Sub-Grupo:**  
**Profesor(a): Juan Carlos Leyton**  
**Autores:**

1. Kenner David Rodriguez González.

León, Nicaragua, 2025.

*“¡A la Libertad por la Universidad!”*

## Propósito y alcance

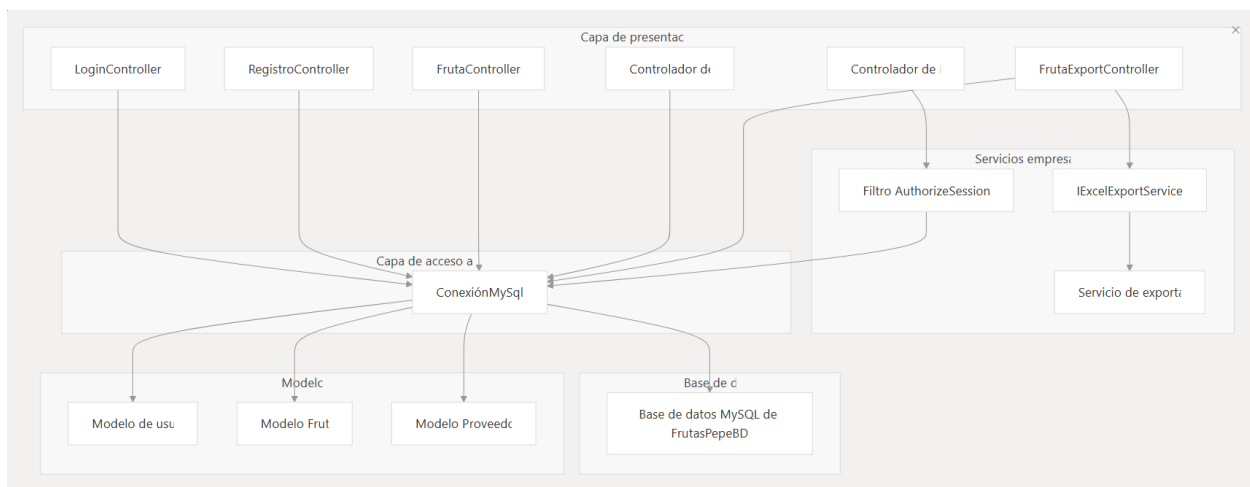
FrutaPepe es una aplicación web de ASP.NET Core diseñada para administrar el inventario de frutas, los proveedores y el acceso de los usuarios dentro de una organización. El sistema proporciona operaciones CRUD integrales para frutas y sus proveedores asociados, administración de usuarios basada en roles y capacidades de exportación de Excel para fines de informes.

Este documento proporciona una descripción general de alto nivel de la arquitectura del sistema, los dominios comerciales principales y los componentes técnicos. Para obtener información detallada sobre subsistemas específicos, consulte [Autenticación y gestión de usuarios](#), [Sistema de gestión de frutas](#), [Capa de datos](#) y [Sistema de interfaz de usuario](#).

## Arquitectura del sistema

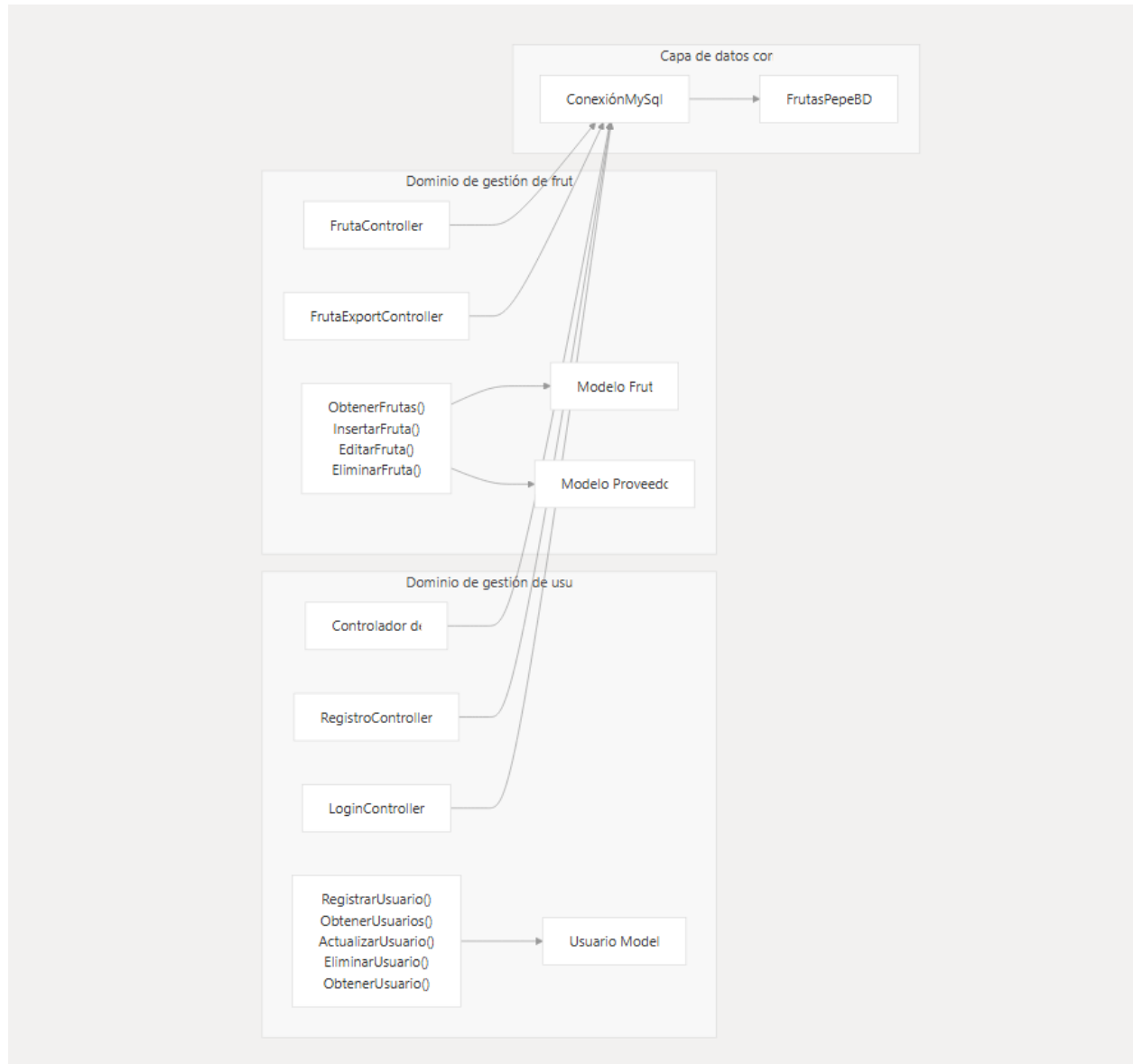
La aplicación sigue un patrón MVC (Modelo-Vista-Controlador) tradicional con una clara separación entre las capas de presentación, lógica empresarial y acceso a datos.

## Componentes de la arquitectura principal



## Dominios comerciales principales

La aplicación administra dos dominios empresariales principales con responsabilidades y patrones de acceso distintos.



## Descripción general de Technical Stack

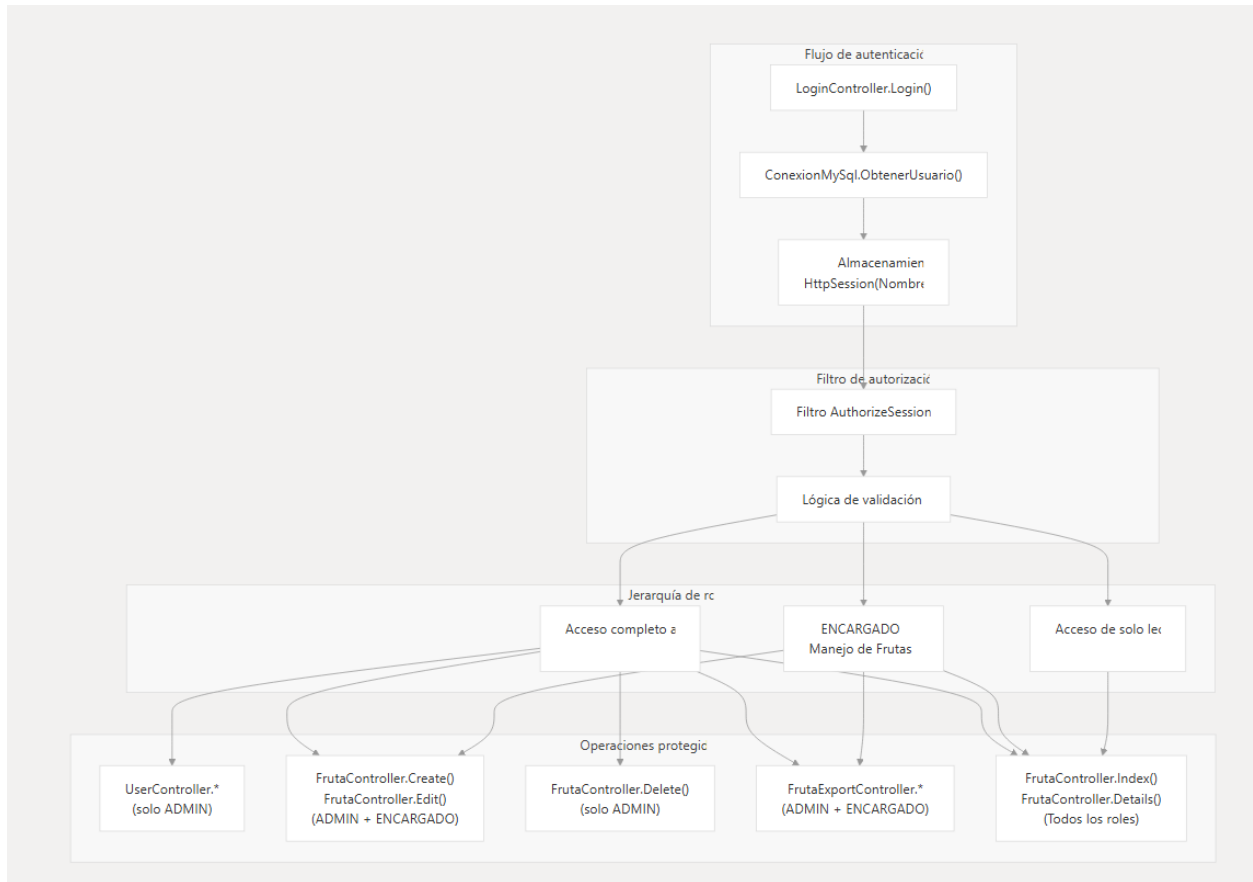
La aplicación se crea utilizando tecnologías .NET modernas con dependencias externas específicas para la conectividad de la base de datos y el procesamiento de Excel.

Componente	Tecnología	Implementación
Marco de referencia	ASP.NET núcleo MVC	Controladores con vistas de Razor
Base de datos	MySQL	Proveedor MySql.Data
Acceso a datos	ADO.NET personalizado	<b>ConexionMySQL</b> clase
Autenticación	Basado en sesiones	Filtro personalizado <b>AuthorizeSession</b>
Exportación de Excel	ClosedXML	<b>IExcelExportService</b> interfaz
Inyección de dependencias	DI incorporado	Configurado en <b>Program.cs</b>
Gestión de sesiones	Caché en memoria	Tiempo de espera de 30 minutos

## Sistema de control de acceso basado en roles

La aplicación implementa un sistema de roles jerárquico de tres niveles con autenticación basada en sesión y autorización de nivel de método.

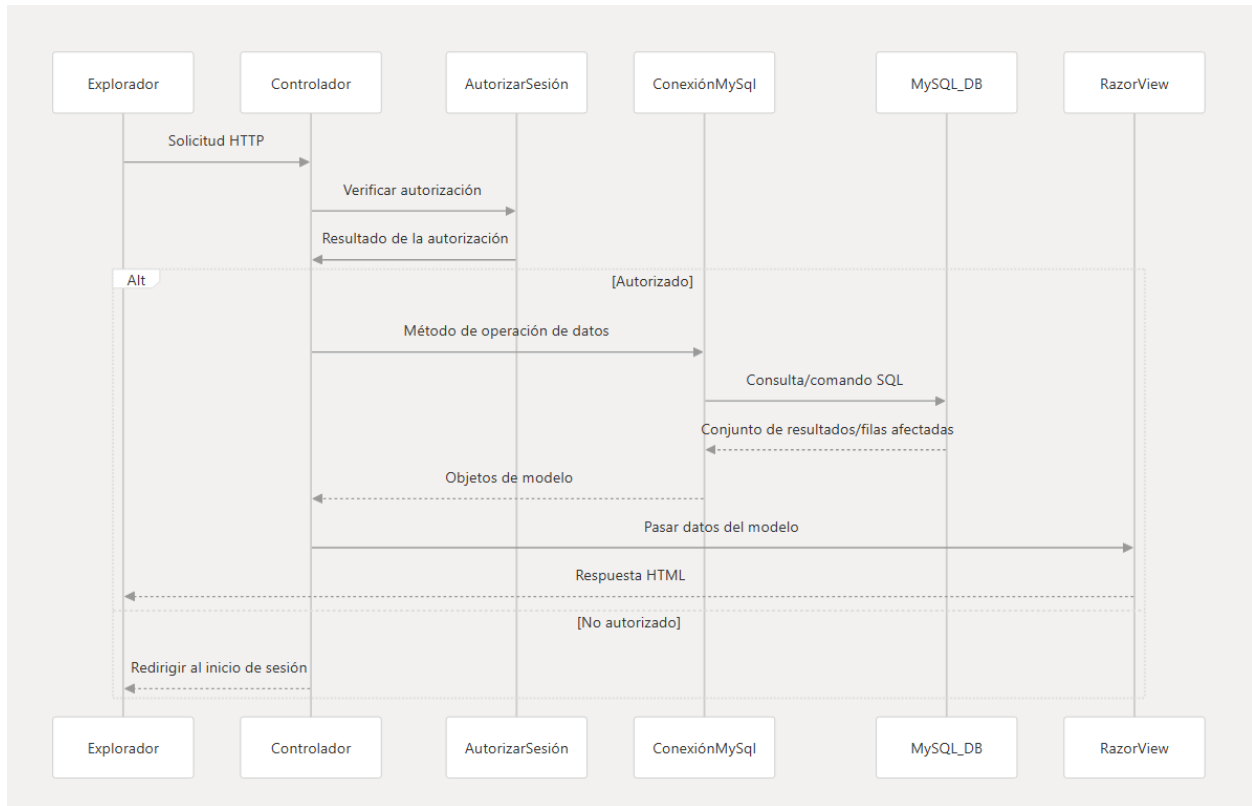
### Jerarquía de roles y permisos



## Arquitectura de flujo de datos

La aplicación sigue un patrón de solicitud-respuesta con acceso centralizado a los datos a través de la clase **ConexionMySQL**

### Flujo de solicitud típico



## **Características clave del sistema**

### **Acceso centralizado a los datos**

Todas las operaciones de base de datos se controlan a través de la clase única, que proporciona métodos para la autenticación de usuarios, la administración de frutas, las operaciones CRUD de usuarios y la recuperación de datos. La clase usa consultas parametrizadas y compatibilidad con transacciones para la integridad de los datos.**ConexionMySQL**

### **Seguridad basada en sesiones**

El estado de autenticación se mantiene en las sesiones del lado servidor con un tiempo de espera de 30 minutos. El atributo filter decora las acciones del controlador para aplicar el control de acceso basado en roles en el nivel de método.**AuthorizeSession**

### **Separación de dominios**

Al compartir la misma capa de datos, la administración de usuarios y la administración de frutas funcionan como dominios distintos con controladores, modelos y flujos de lógica empresarial separados.