

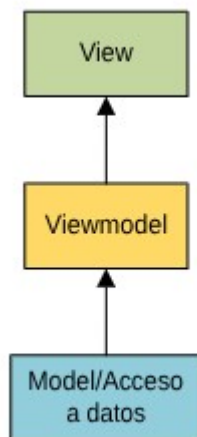
1. Arquitectura MVVM utilizada

La aplicación sigue el patrón Model–View–ViewModel (MVVM) de WPF.

- **Model:** contiene las entidades del dominio y el acceso a datos (repositorio + EF).
- **ViewModel:** expone propiedades y comandos para el enlace de datos con la vista.
- **View:** ventanas WPF (.xaml) que se enlazan al DataContext del ViewModel.

Este patrón separa la lógica de negocio de la interfaz gráfica, favoreciendo el mantenimiento y el testeo.

2. Diagrama de capas



3. Descripción de carpetas del proyecto

Estructura principal:

- /Model

Contiene entidades (Socio, Actividad, Reserva) y Repositorio para el acceso a datos.

- /ModelView

Contiene los ViewModels (SociosViewModel, ReservasViewModel, ActividadesViewModel) y la carpeta /Command con RelayCommand.

- /View

Vistas WPF (Socios.xaml, Reservas.xaml, Actividades.xaml)

- /Testing

Proyecto de pruebas unitarias

Ejemplo de estructura:

```
/Model  
/ModelView  
  /Command  
/View  
/Testing
```

4. Explicación de componentes

4.1.- Models

Representan las entidades del dominio:

- Socio (Id, Nombre, Email, Activo, etc.)
- Actividad (Id, Nombre, AforoMaximo, etc.)
- Reserva (Id, Socioid, ActividadId, Fecha, etc.)

Estas clases están vinculadas al modelo de datos (Entity Framework).

4.2.- ViewModels

Intermedian entre la vista y el modelo.

Ejemplos:

- SociosViewModel
- ActividadesViewModel
- ReservasViewModel

Exponen:

- Colecciones (ObservableCollection) para mostrar en DataGrid
- Objetos activos para formularios
- Comandos para operaciones CRUD

4.3.- Commands

Se utiliza RelayCommand para enlazar acciones de los botones. Ejemplos:

- GuardarCommand
- NuevoCommand
- EditarCommand
- EliminarCommand

Permite implementar CanExecute para habilitar/deshabilitar botones según selección.

4.4.- Acceso a datos

El acceso a datos se centraliza en Repositorio.

Funciones principales:

- SeleccionarSocios, SeleccionarActividades, SeleccionarReservas
- Guardar/Actualizar/Eliminar para cada entidad
- Validaciones (email, reservas duplicadas, aforo)

Internamente usa Entity Framework y el contexto CentroDeportivoEntities.

5. Diagrama de base de datos

