

*Descripción del proyecto.*  
**Entrega TFG 15-06-2022**

CinemasRafa



*Rafael Prieto Alvarez*

## Estructura de clases

### Persona

```
3 referencias
public abstract class Persona
{

    public int ID { get; set; }

    [Display(Name = "Apellidos")]
    [Required]
    [StringLength(80, ErrorMessage = "Apellido deberia de ser menor a 20 caracteres.")]
    36 referencias
    public string Surname { get; set; }

    [Display(Name = "Nombre")]
    [Required]
    [StringLength(20, ErrorMessage = "Nombre deberia de ser menor a 20 caracteres.")]
    36 referencias
    public string Name { get; set; }

    24 referencias
    public int CuentaUsuarioID { get; set; }

    [ForeignKey("CuentaUsuarioID")]
    27 referencias
    public CuentaUsuario CuentaUsuario { get; set; }

    [Required]
    [Display(Name = "Telefono")]
    [RegularExpression(@"^[0-9]{9}$", ErrorMessage = "Telefono no valido.")]
    35 referencias
    public string Phone { get; set; }

}
```

### Admin

```
namespace CinemasRafa.Models
{
    19 referencias
    public class Admin: Persona
    {
    }
}
```

### Worker

```
19 referencias
public class Worker: Persona
{
}
}
```

## Customer

```
19 referencias
public class Customer: Persona
{
    [Required]
    [Display(Name = "Direccion")]
    [StringLength(40, ErrorMessage = "La direccion no puede superar los 40 caracteres.")]
    12 referencias
    public string Address { get; set; }
}
```

## Role

```
27 referencias
public class Role
{
    21 referencias
    public int ID { get; set; }

    [Display(Name = "Rol")]
    [Required]
    20 referencias
    public string RoleName { get; set; }

    [Display(Name = "Descripcion")]
    [Required]

    21 referencias
    public string RoleDescription { get; set; }

    13 referencias
    public bool Enabled { get; set; }

    [InverseProperty("Role")]
    0 referencias
    public List<CuentaUsuario> CuentaUsuarios { get; set; }
    [InverseProperty("Role")]
    0 referencias
    public List<RoleHasMenu> RoleHasMenus { get; set; }
}
```

## RoleHasMenu

```
public class RoleHasMenu
{
    public int ID { get; set; }

    public int MenuID { get; set; }
    [ForeignKey("MenuID")]
    public Menu Menu { get; set; }

    public int RoleID { get; set; }
    [ForeignKey("RoleID")]
    public Role Role { get; set; }
}
```

## Menu

```
37 referencias
public class Menu
{
    26 referencias
    public int ID { get; set; }

    26 referencias
    public string Action { get; set; }

    27 referencias
    public string Label { get; set; }

    26 referencias
    public string Controller { get; set; }

    [InverseProperty("Menu")]
    0 referencias
    public List<RoleHasMenu> RoleHasMenus { get; set; }
}
```

## Películas

```
public class Peliculas
{
    public int Id { get; set; }
    public string Nombre { get; set; }
    public string Descripcion { get; set; }
    public string Director { get; set; }
    public int Duracion { get; set; }
    public DateTime FechaEstreno { get; set; }
    public string UrlDescarga { get; set; }
    public double Valoracion { get; set; }
    [ValidateNever]

    public string ImageUrl { get; set; }
    [Required]
    [Display(Name = "Categoria")]
    13 referencias
    public int CategoriaId { get; set; }
    [ValidateNever]
    9 referencias
    public Categoria Categoria { get; set; }
    [Required]
    [Display(Name = "Pegi")]
    13 referencias
    public int PegiId { get; set; }
    [ValidateNever]
    9 referencias
    public Pegi Pegi { get; set; }
}
```

## Series

```
public class Series
{
    public int Id { get; set; }
    public string Nombre { get; set; }
    public string Director { get; set; }
    public string Descripcion { get; set; }
    public int Temporada { get; set; }
    public DateTime FechaEstreno { get; set; }
    public bool HaTerminado { get; set; }
    public string UrlDescarga { get; set; }
    public double Valoracion { get; set; }

    [ValidateNever]

    public string ImageUrl { get; set; }
    [Required]
    [Display(Name = "Categoria")]
    11 referencias
    public int CategoriaId { get; set; }
    [ValidateNever]
    9 referencias
    public Categoria Categoria { get; set; }
    [Required]
    [Display(Name = "Pegi")]
    11 referencias
    public int PegiId { get; set; }
    [ValidateNever]
    9 referencias
    public Pegi Pegi { get; set; }
}
```

## Pegis

```
public class Pegi
{
    15 referencias
    public int Id { get; set; }
    19 referencias
    public string EdadMin { get; set; }
    15 referencias
    public int PegiEdad { get; set; }
}
```

## Categoria

```
23 referencias
public class Categoria
{
    15 referencias
    public int Id { get; set; }
    19 referencias
    public string Name { get; set; }
}
```

## CuentaUsuario

```
public class CuentaUsuario
{
    public int ID { get; set; }

    [Required]
    [Display(Name = "Nombre usuario")]
    [StringLength(10, MinimumLength = 2, ErrorMessage = "Error el usuario debe tener entre 2 y 10 caracteres.")]
    24 referencias
    public string Username { get; set; }

    [Required]
    [Display(Name = "Contraseña")]
    [DataType(DataType.Password)]
    [RegularExpression(@"^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[^\da-zA-Z]).{8,22}$",
        ErrorMessage = "Contraseña debe tener al menos una letra mayúscula, una minúscula, un número y un carácter especial.")]
    23 referencias
    public string Password { get; set; }

    [Required]
    [EmailAddress]
    [StringLength(50, ErrorMessage = "Email no puede superar 50 caracteres.")]

    30 referencias
    public string Email { get; set; }

    [Required]
    20 referencias
    public bool Active { get; set; }

    21 referencias
    public int RoleID { get; set; }

    [ForeignKey("RoleID")]
    9 referencias
    public Role Role { get; set; }
}
```

## Diagrama de tablas

