

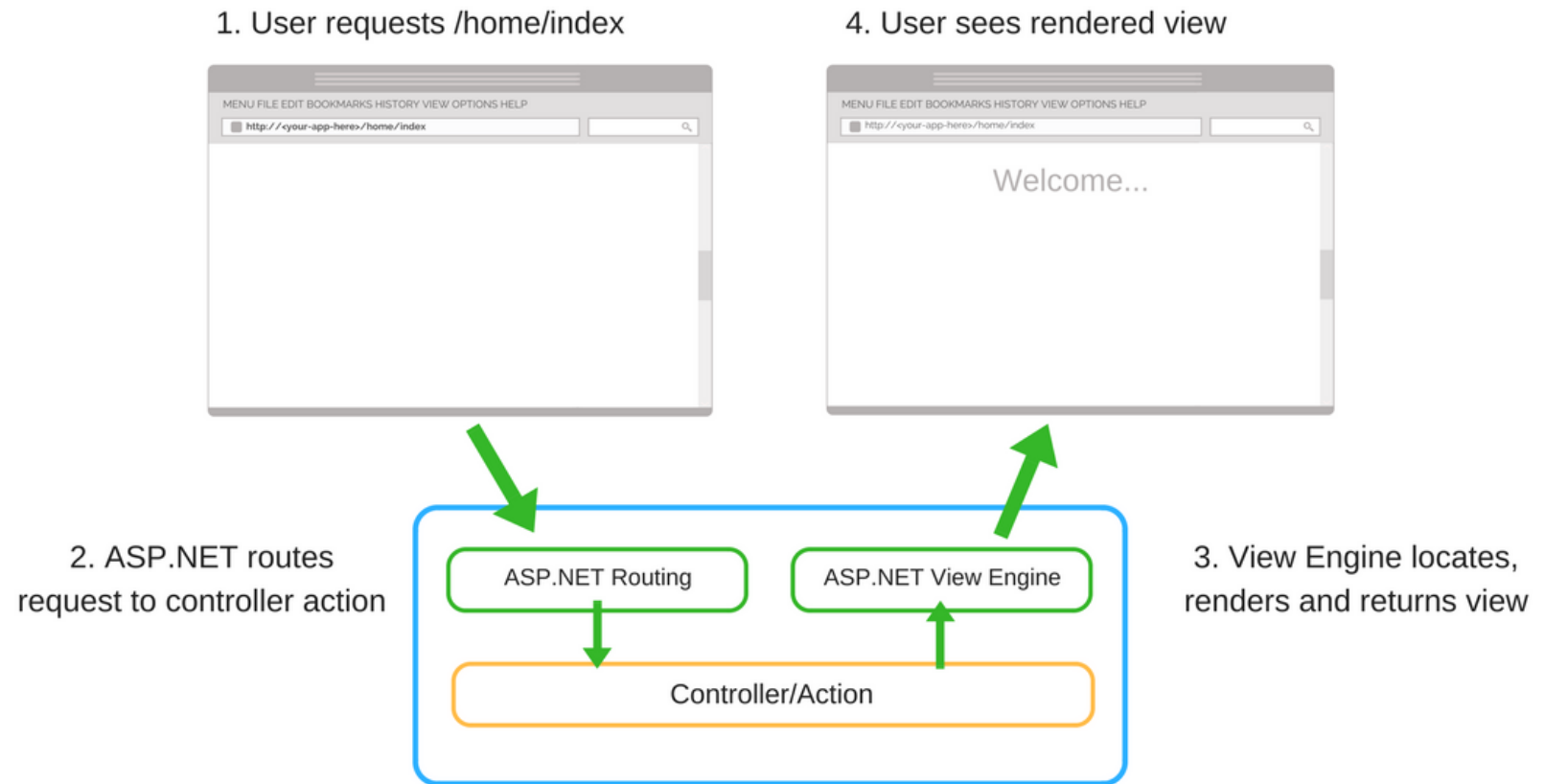
# Validación

- ¿Dónde validar?
- DRY
- Atributos de validación
- Librerías de validación
- ¿Dónde usar la validación?



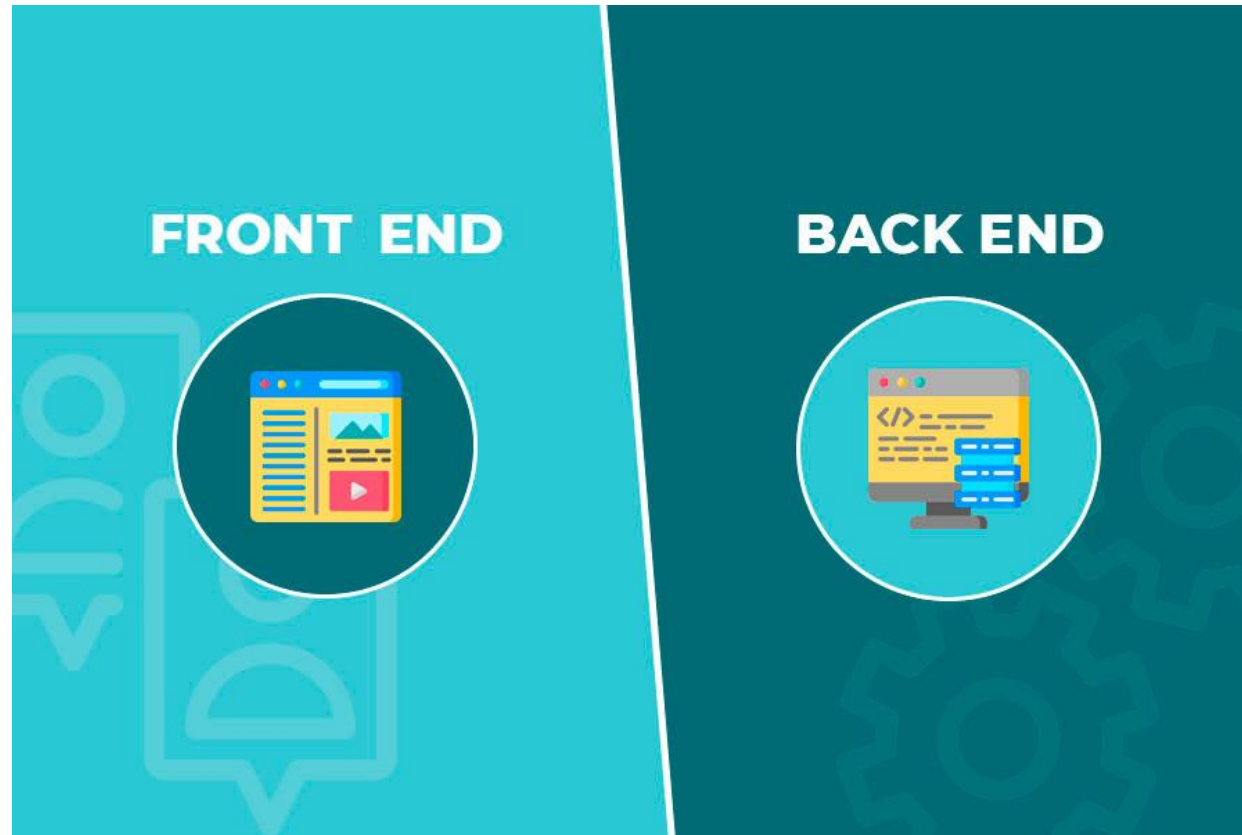
# Manejos de sesión

## Recordando



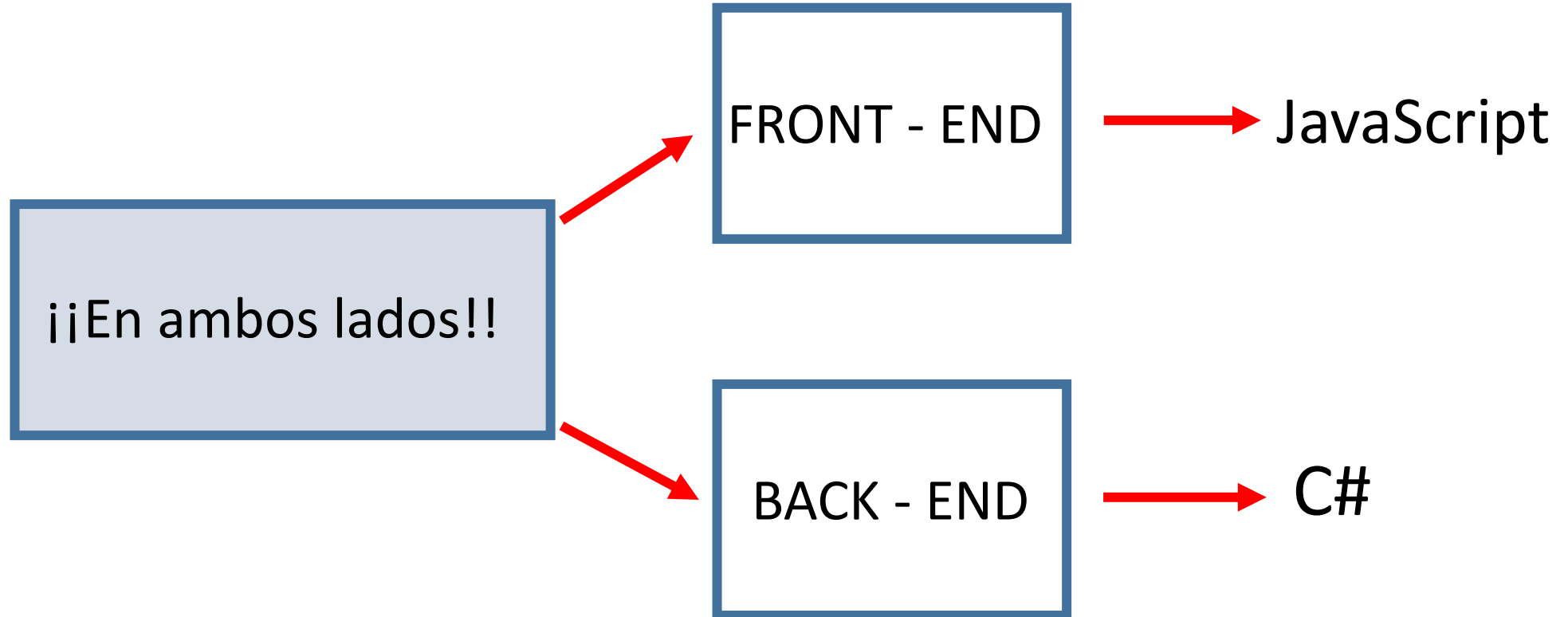
# Validación

¿Donde validar?



# Validación

Donde validar?



# Validación

Too much....

Que se termine el año



# Validación

Boy, DON'T REPEAT YOURSELF





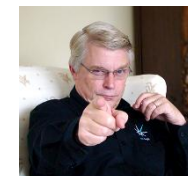
# Validación

## DON'T REPEAT YOURSELF

Según este principio toda "pieza de **información**" nunca debería ser duplicada debido a que la duplicación incrementa la dificultad en los cambios y evolución posterior, puede perjudicar la claridad y crear un espacio para posibles **inconsistencias**

El términos "pieza de información" son usados en un sentido amplio, abarcando:

- datos almacenados en una **base de datos**;
- **código fuente** de un programa de **software**;
- información textual o documentación.



Robert C. Martin

# Uniendo ideas



Validación

DRY

Atributos



# Validación

## DRY

Uno de los principios de diseño de MVC es [DRY](#) ("Una vez y solo una"). ASP.NET facilita especificar la funcionalidad o el comportamiento una sola vez y a que luego los refleje en el resto de la aplicación.

Esto reduce la cantidad de código que necesita escribir y hace que el código que se escribe sea menos propenso a errores, así como más fácil probar y de mantener.

# Validación

## Atributos de validación

Los atributos de validación permiten especificar reglas de validación para las propiedades del modelo. En el ejemplo siguiente de la aplicación de ejemplo se muestra una clase de modelo anotada con atributos de validación.

## Espacio de nombres

El espacio de nombres DataAnnotations proporciona un conjunto de atributos de validación integrados que se aplican mediante declaración a una clase o propiedad

`System.ComponentModel.DataAnnotations`

# Validación

## Atributos de validación

Estos son algunos de los atributos de validación integrados:

**[ValidateNever]:** ValidateNeverAttribute indica que una propiedad o parámetro debe excluirse de la validación.

**[Compare]:** valida que dos propiedades de un modelo coinciden.

**[EmailAddress]:** valida que la propiedad tiene un formato de correo electrónico.

**[Phone]:** valida que la propiedad tiene un formato de número de teléfono.

**[Range]:** valida que el valor de propiedad se encuentra dentro de un intervalo especificado.

**[RegularExpression]:** valida que el valor de propiedad coincide con una expresión regular especificada.

**[Required]:** valida que el campo no es NULL.

**[StringLength]:** valida que un valor de propiedad de cadena no supera un límite de longitud especificado.

**[Url]:** valida que la propiedad tiene un formato de dirección URL.

# Validación

## Atributos de validación

```
public class Movie
{
    public int Id { get; set; }
    [Required] [StringLength(100)]
    public string Title { get; set; }
    [ClassicMovie(1960)]
    [DataType(DataType.Date)]
    [DisplayName = "Release Date"]
    public DateTime ReleaseDate { get; set; }
    [Required] [StringLength(1000)]
    public string Description { get; set; }
    [Range(0, 999.99)]
    public decimal Price { get; set; }
    public Genre Genre { get; set; }
    public bool Preorder { get; set; }
}
```

# Validación

## Librerías de validación – En el Front End

Para agregar el conjunto de librerías de javascript que validan los datos con las reglas del servidor

```
@section Scripts { <partial name="_ValidationScriptsPartial" /> }
```

También se puede agregar directamente la librerías a la vistas si se quiere

```
@section Scripts
{
    <script src="~/lib/jquery-validation/dist/jquery.validate.min.js"></script>
    <script src="~/lib/jquery-validation-unobtrusive/jquery.validate.unobtrusive.min.js"></script>
}
```

# Validación

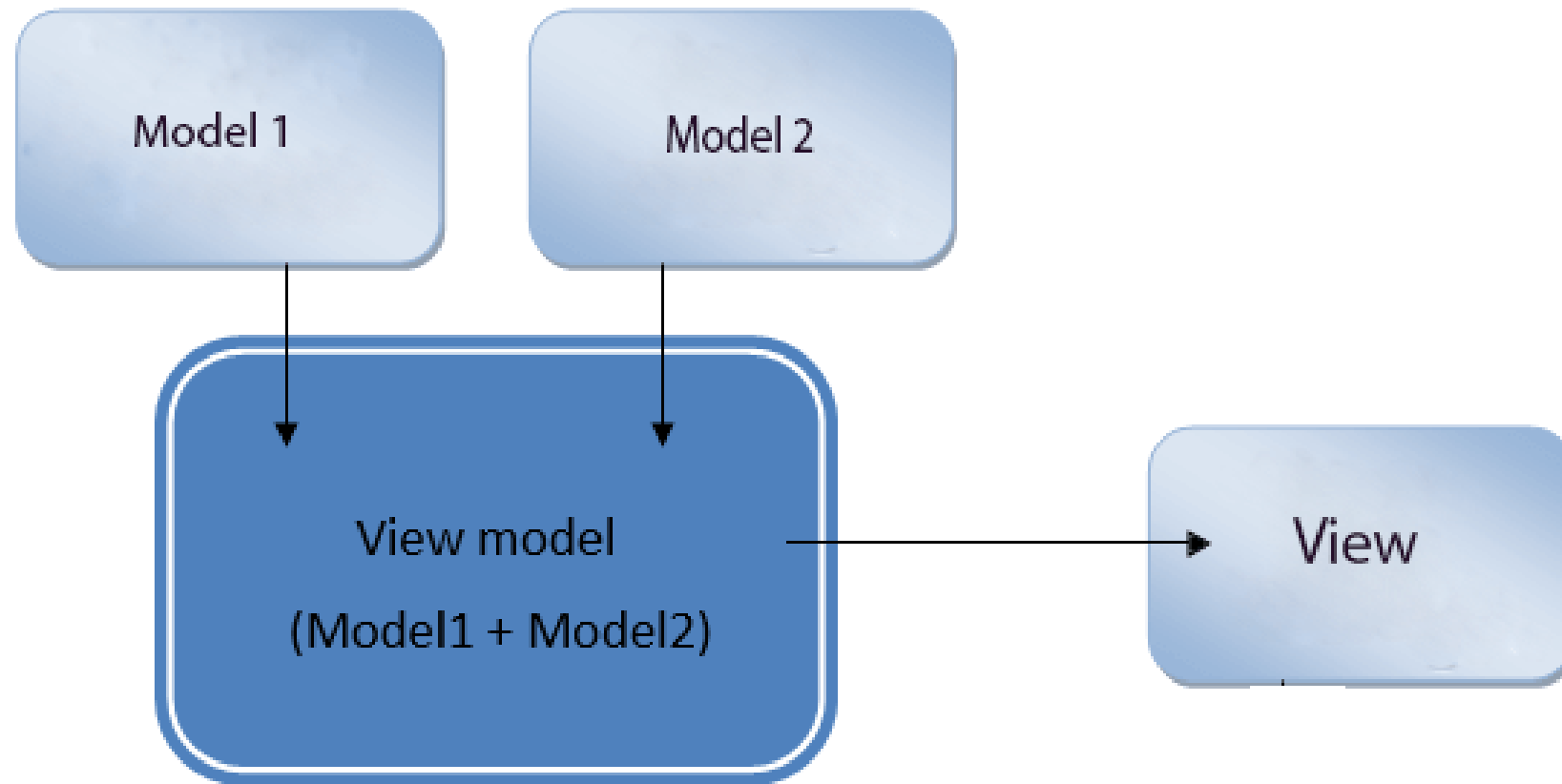
## Librerías de validación – En el Back End

```
if (ModelState.IsValid)
{
    [...] // lógica para modelo válido
}
```

¿Donde aplicar las validaciones?



# View models



# Bibliografía

<https://docs.microsoft.com/es-es/dotnet/api/system.componentmodel.dataannotations?view=net-5.0>

<https://wakeupandcode.com/validation-in-asp-net-core-3-1/>