PREGRADO

UNIDAD 3 | WEB SERVICES

WEB SERVICES INTERNATIONALIZATION

SI730 | Aplicaciones Web

Al finalizar la semana, el estudiante comunica resultados y proceso ágil colaborativo aplicado para la implementación de validación de datos, comportamiento y reglas del negocio, para una aplicación de lado servidor, con características innovadoras, bajo una arquitectura orientada a servicios y aplicando los principios RESTful utilizando el lenguaje C# y Microsoft .NET Framework.

AGENDA

INTRO

STRINGS

DATA ANNOTATIONS

MODEL BINDING

Intro

Una aplicación multi-idioma permite llegar a una audiencia más amplia. ASP.NET Core proporciona servicios y middleware para localizar en diferentes lenguajes y culturas.

Internationalization implica **Globalization** y **Localization**.

Globalization es el proceso de diseñar el app para múltiples culturas.

Localization es el proceso de adaptar un app globalizada, hacia un culture/locale particular.

AGENDA

INTRO

STRINGS

DATA ANNOTATIONS

MODEL BINDING

General Request Localization

Para ser capaz de enviar localized messages a los clientes, se requiere habilitar request localization.

Esto permite que el API recupere mensajes en el idioma deseado.

Startup

La configuración de localización se encuentra en el archivo Startup de la aplicación. Se debe modificar los métodos ConfigureServices y Configure.

Configure Services

En ConfigureServices se requiere agregar los servicios requeridos,

así como establecer default culture y supported cultures.

Se requiere la extensión **Microsoft.Extensions.Localization**.

```
public void ConfigureServices(IServiceCollection services)
{
    services.AddLocalization();
    services.AddRequestLocalization(x =>
    {
        x.DefaultRequestCulture = new RequestCulture("en");
        x.ApplyCurrentCultureToResponseHeaders = true;
        x.SupportedCultures = new List<CultureInfo> {new("es"), new("en")};
        x.SupportedUICultures = new List<CultureInfo> {new("es"), new("en")};
    });
    //...
}
```

Configure

Configure permite que se registre el RequestLocalizationMiddleware con el ASP.NET pipeline usando el método de registro proporcionado por ASP.NET Core.

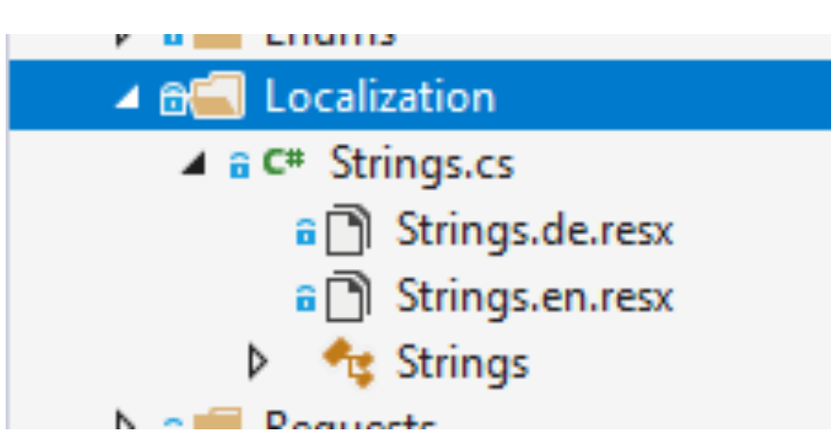
```
public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
{
    if (env.IsDevelopment()) {
        app.UseDeveloperExceptionPage();
        app.UseSwagger();
        app.UseSwaggerUI(c => c.SwaggerEndpoint("/swagger/v1/swagger.json", "My Demo API v1"));
    }


    app.UseRequestLocalization();
    app.UseHttpsRedirection();
    app.UseRouting();
    app.UseAuthorization();
    app.UseEndpoints(endpoints =>
    {
        endpoints.MapControllers();
    });
}
```

Localized Resources

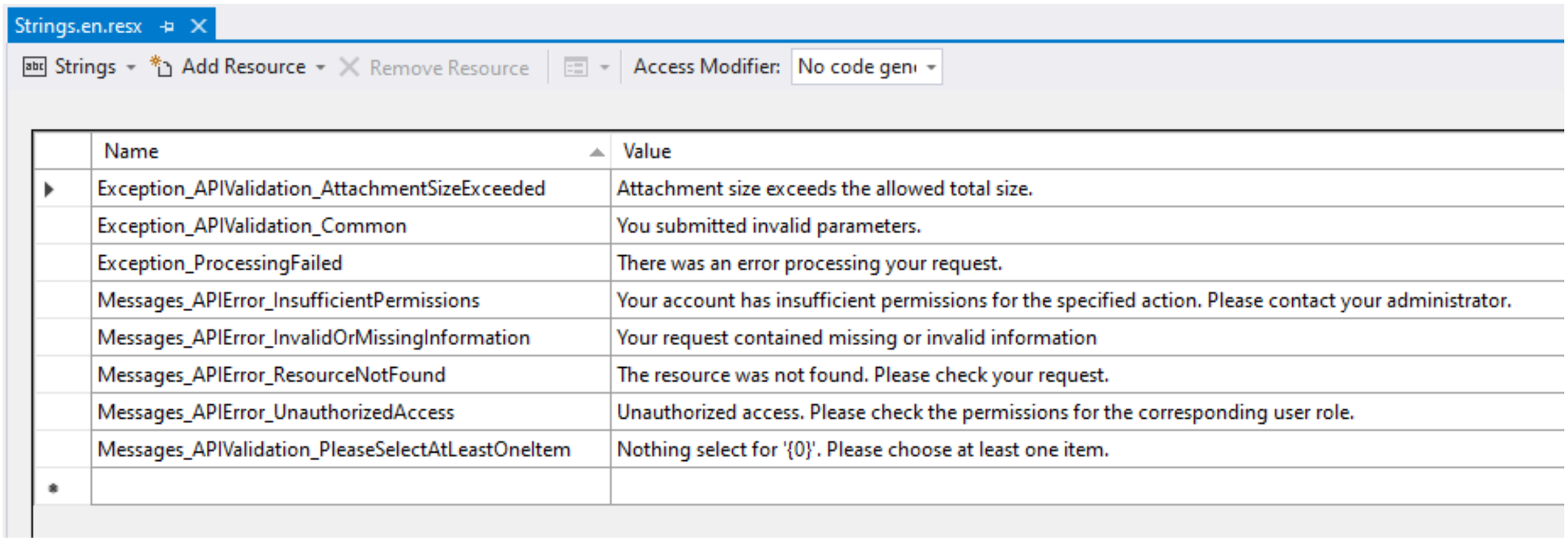
Se requiere resource files (.resx), uno por cada idioma, junto con un archivo .cs.

Por ejemplo, Strings.cs, Strings.en.resx, Strings.de.resx.



Localized Resources

Se agrega los textos a traducir en los resource files. Seleccionar **No code generation** en **Access Modifier**.



	Name	Value
▶	Exception_APIValidation_AttachmentSizeExceeded	Attachment size exceeds the allowed total size.
	Exception_APIValidation_Common	You submitted invalid parameters.
	Exception_ProcessingFailed	There was an error processing your request.
	Messages_APIError_InsufficientPermissions	Your account has insufficient permissions for the specified action. Please contact your administrator.
	Messages_APIError_InvalidOrMissingInformation	Your request contained missing or invalid information
	Messages_APIError_ResourceNotFound	The resource was not found. Please check your request.
	Messages_APIError_UnauthorizedAccess	Unauthorized access. Please check the permissions for the corresponding user role.
	Messages_APIValidation_PleaseSelectAtLeastOneItem	Nothing select for '{0}'. Please choose at least one item.
*		

Using Localized Resources

Solicitar un IStringLocalizer<T> de ASP.NET Core dependency injection, donde T es el tipo del resource group.

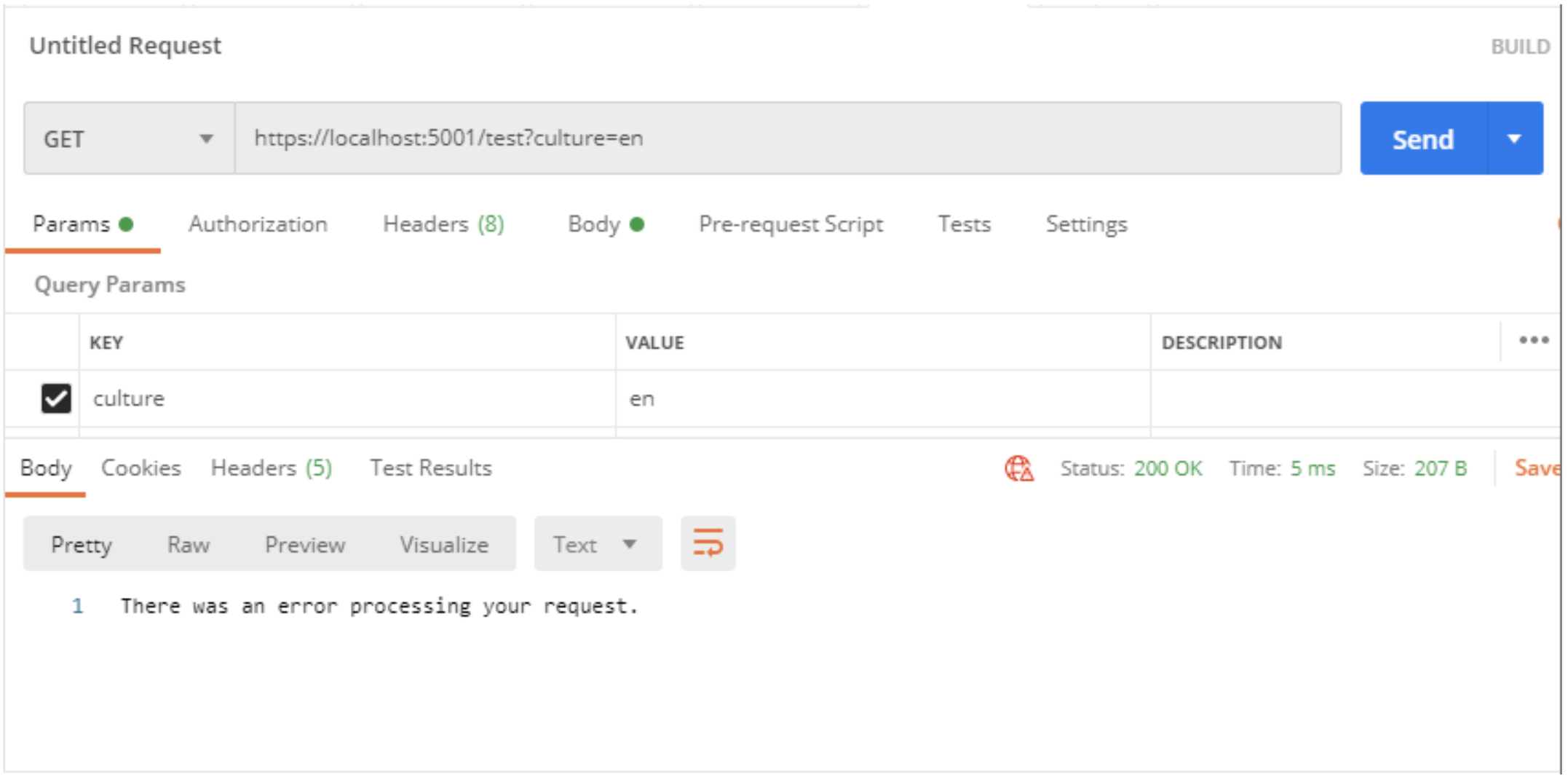
```
[ApiController]
[Route("[controller]")]
public class TestController : ControllerBase
{
    private readonly IStringLocalizer<Strings> _localizer;


    public TestController(IStringLocalizer<Strings> localizer) {
        _localizer = localizer;
    }


    [HttpGet(Name = nameof(GetLocalizedString))]
    [ProducesResponseType(typeof(string), (int)HttpStatusCode.OK)]
    public async Task<IActionResult> GetLocalizedString() {
        string msg = _localizer.GetString("Exception_ProcessingFailed");
        return Ok(msg);
    }
}
```

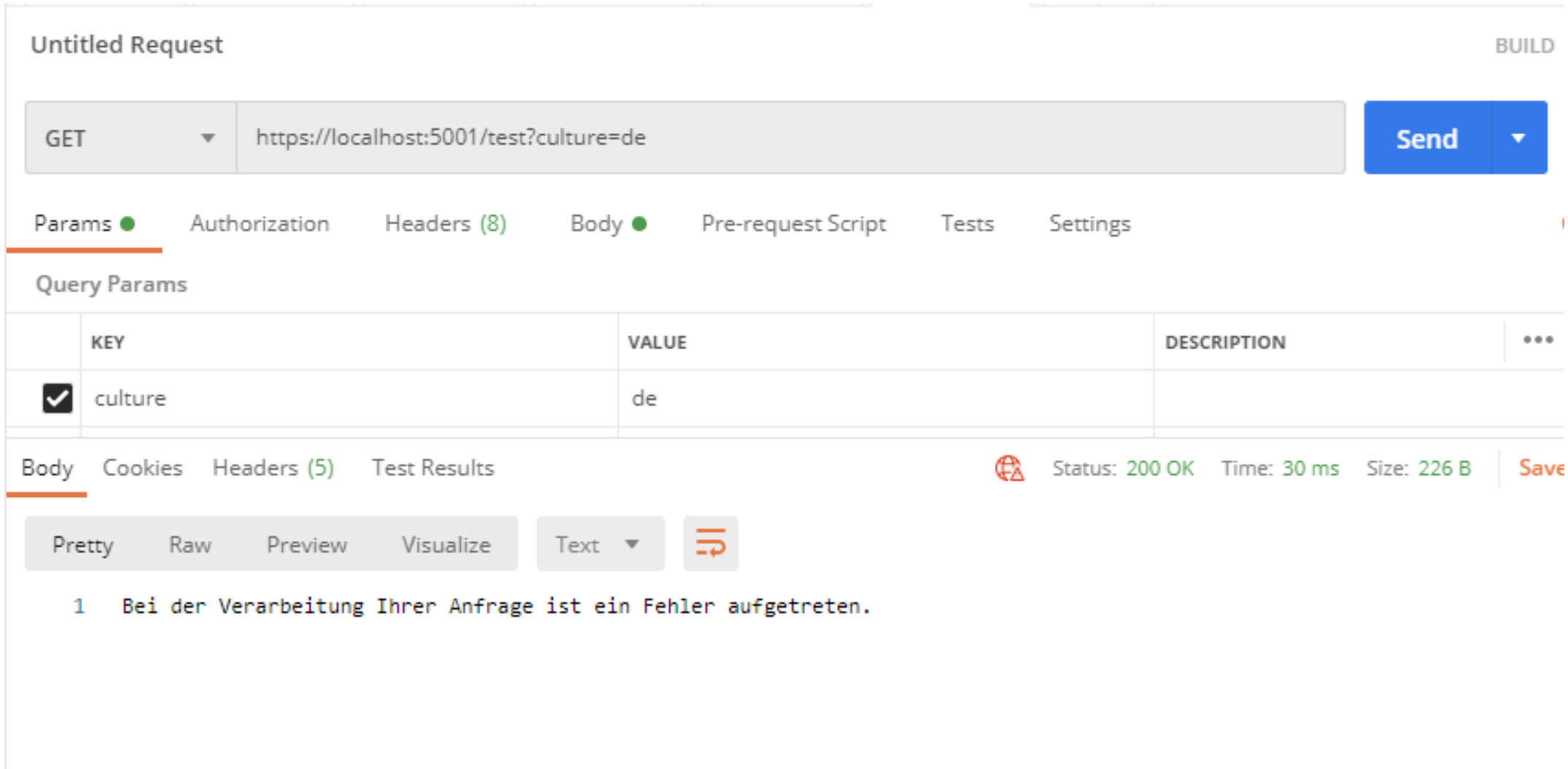
Using Localized Resources

El controller retorna el mensaje traducido



Using Localized Resources

El controller retorna el mensaje traducido



AGENDA

INTRO

STRINGS

DATA ANNOTATIONS

MODEL BINDING

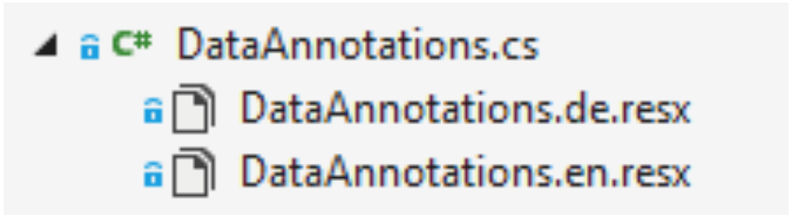
Data Annotation Localization

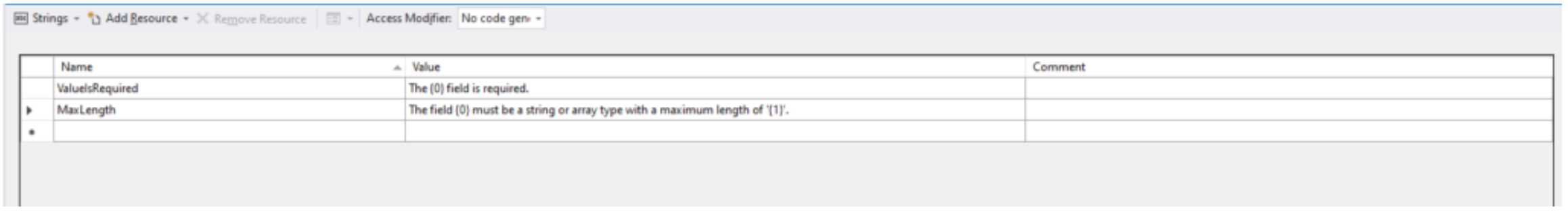
Se requiere cuando se utiliza model validation con ValidationAttribute (como RequiredAttribute, MaxLengthAttribute)

Se requiere modificar los Attributes existentes para que contengan un resource key.

Resources Setup

Preparar los resources.





Startup Services Configuration

Usar AddDataAnnotationsLocalization extensión method para registrar IStringLocalizer<DataAnnotations>.

```
services.AddLocalization();
services.AddRequestLocalization(x =>
{
    //...
});


services.AddControllers()
    .AddDataAnnotationsLocalization(o =>
    {
        o.DataAnnotationLocalizerProvider = (type, factory) => factory.Create(typeof(DataAnnotations));
    });
```

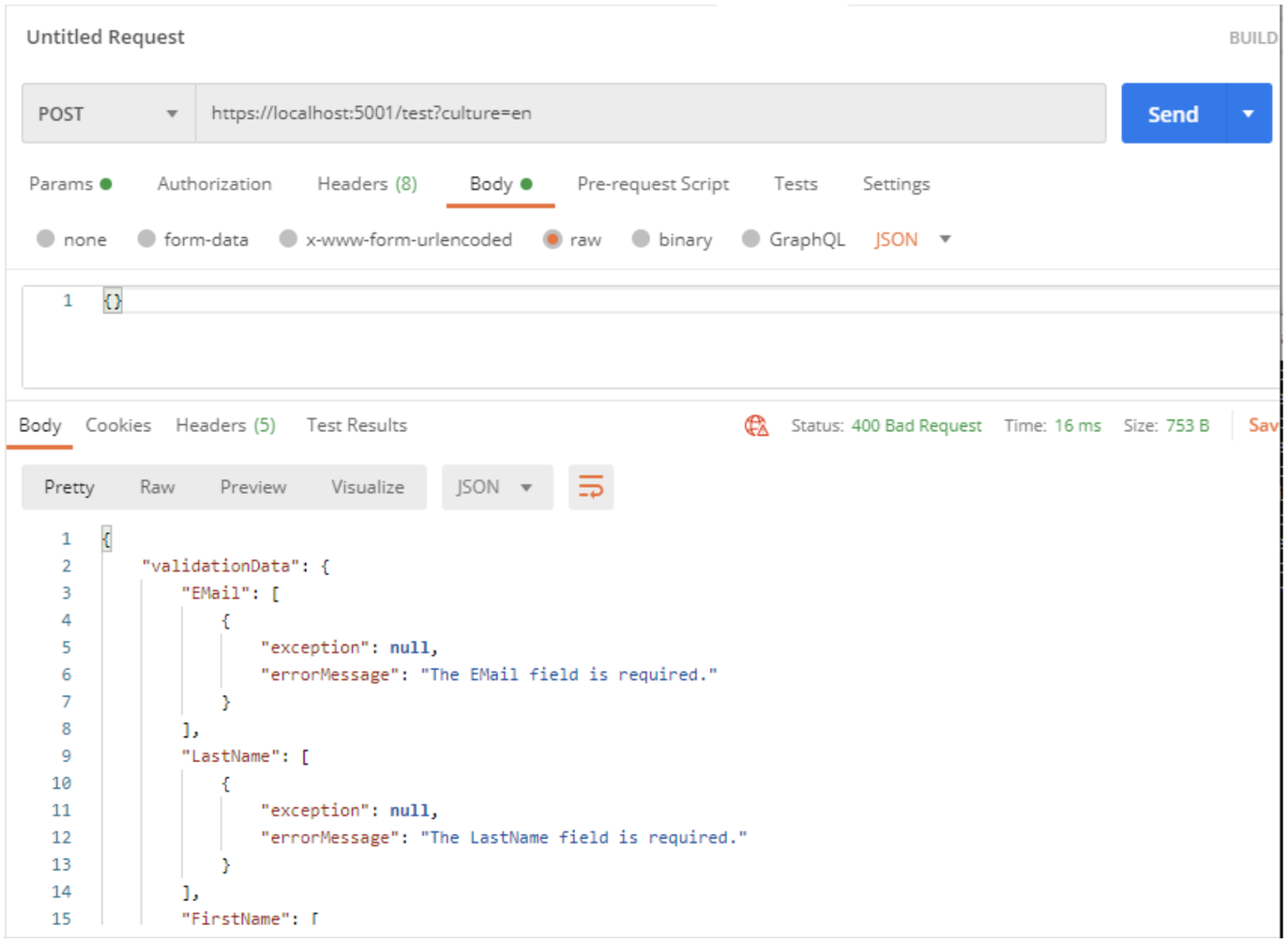
DataAnnotation Attributes Translation

Utilizar ErrorMessage property para especificar un localization key para cada attribute.

```
public record CreateContactRequest
{
    [Required(ErrorMessage = "ValueIsRequired")]
    public string FirstName { get; set; }
    [Required(ErrorMessage = "ValueIsRequired")]
    public string LastName { get; set; }
    [Required(ErrorMessage = "ValueIsRequired")]
    public string EMail { get; set; }
}
```

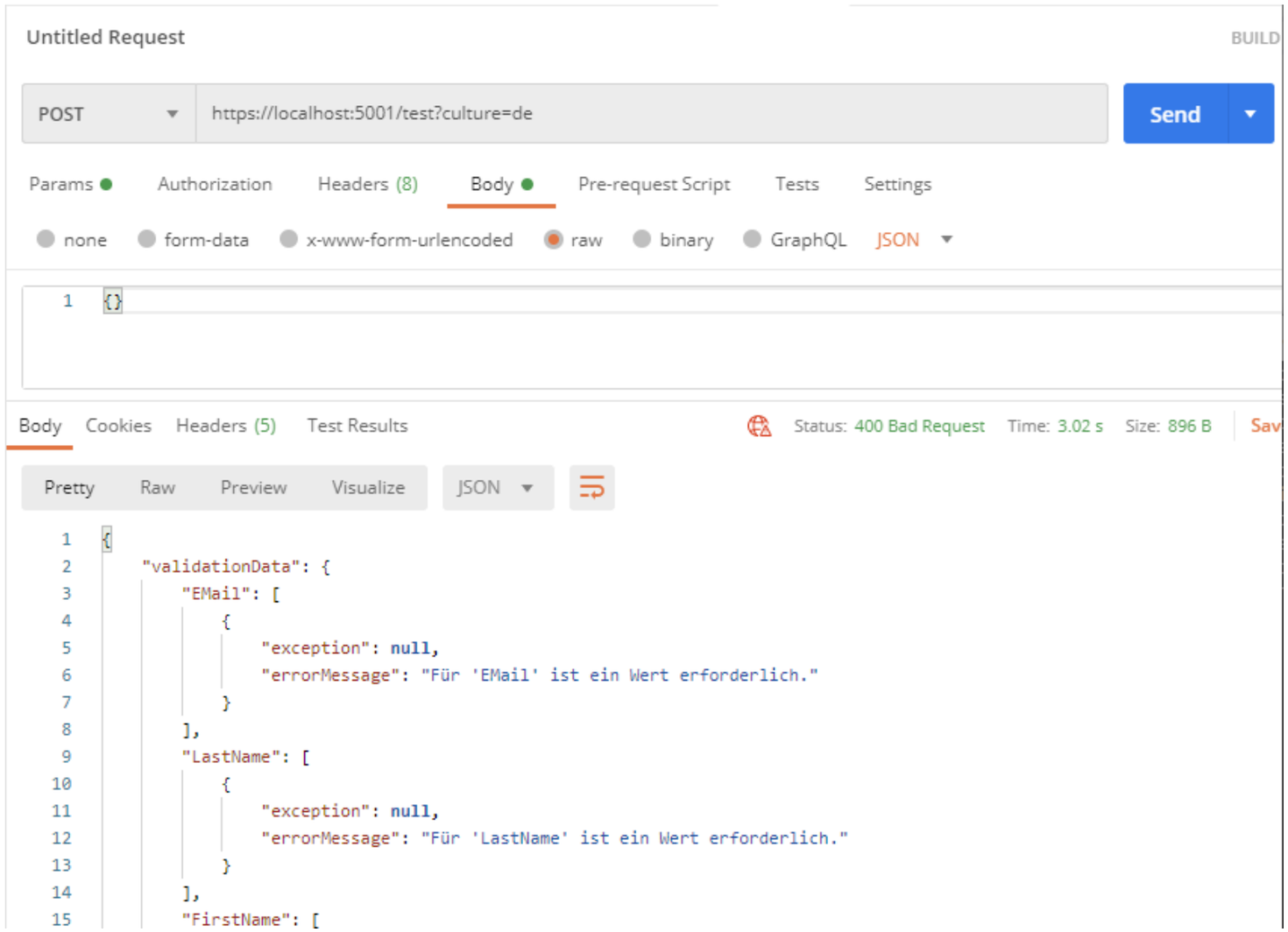
DataAnnotation Attributes Translation

Utilizar ErrorMessage property para especificar un localization key para cada attribute.



DataAnnotation Attributes Translation

Utilizar ErrorMessage property para especificar un localization key para cada attribute.



AGENDA

INTRO

STRINGS

DATA ANNOTATIONS

MODEL BINDING

Model Binding Localization

Model binding es utilizado cuando ASP.NET Core MVC trata de hacer un mapeo de datos del request a parámetros de action method.

Por ejemplo se genera model binding errors cuando:

Caller incluye valor no esperado en route property (string en vez de integer para un id por ejemplo)

Resource Files preparation

Crear los .resx necesarios y el archivo .cs con una clase vacía en él.

MVC Configuration with IConfigureOptions<T>

Implementar la interfaz IConfigureOptions<MvcOptions>.

```
public class MvcConfiguration : IConfigureOptions<MvcOptions>
{
    private readonly IStringLocalizer<ModelBinding> _stringLocalizer;


    public MvcConfiguration(IStringLocalizer<ModelBinding> stringLocalizer)
    {
        _stringLocalizer = stringLocalizer;
    }
    public void Configure(MvcOptions options)
    {
        options.ModelBindingMessageProvider.SetValueIsInvalidAccessor(
              x => _stringLocalizer.GetString("ValueIsInvalidAccessor", x));
        options.ModelBindingMessageProvider.SetValueMustBeANumberAccessor(
              x => _stringLocalizer.GetString("ValueMustBeANumberAccessor", x));
        options.ModelBindingMessageProvider.SetMissingBindRequiredValueAccessor(
              x => _stringLocalizer.GetString("MissingBindRequiredValueAccessor", x));
        options.ModelBindingMessageProvider.SetAttemptedValueIsInvalidAccessor(
              (x,y) => _stringLocalizer.GetString("AttemptedValueIsInvalidAccessor", x,y));
        options.ModelBindingMessageProvider.SetMissingKeyOrValueAccessor(
              () => _stringLocalizer.GetString("MissingKeyOrValueAccessor"));
        options.ModelBindingMessageProvider.SetUnknownValueIsInvalidAccessor(
              x => _stringLocalizer.GetString("UnknownValueIsInvalidAccessor", x));
        options.ModelBindingMessageProvider.SetValueMustNotBeNullAccessor(
              x => _stringLocalizer.GetString("ValueMustNotBeNullAccessor", x));
        options.ModelBindingMessageProvider.SetNonPropertyAttemptedValueIsInvalidAccessor(
              x => _stringLocalizer.GetString("NonPropertyAttemptedValueIsInvalidAccessor", x));
        options.ModelBindingMessageProvider.SetNonPropertyUnknownValueIsInvalidAccessor(
              () => _stringLocalizer.GetString("UnknownValueIsInvalidAccessor"));
        options.ModelBindingMessageProvider.SetNonPropertyValueMustBeANumberAccessor(
              () => _stringLocalizer.GetString("NonPropertyValueMustBeANumberAccessor"));
        options.ModelBindingMessageProvider.SetMissingRequestBodyRequiredValueAccessor(
              () => _stringLocalizer.GetString("MissingRequestBodyRequiredValueAccessor"));
    }
}
```

MVC Configuration with IConfigureOptions<T>

Establecer accessor para translatable messages

AttemptedValueIsInvalidAccessor	The value ‘{0}’ is not valid for {1}.
MissingBindRequiredValueAccessor	A value for the ‘{0}’ property was not provided.
MissingKeyOrValueAccessor	A value is required.
MissingRequestBodyRequiredValueAccessor	A non-empty request body is required.
NonPropertyAttemptedValueIsInvalidAccessor	The value ‘{0}’ is not valid.
NonPropertyUnknownValueIsInvalidAccessor	The supplied value is invalid.
NonPropertyValueMustBeANumberAccessor	The field must be a number.
UnknownValueIsInvalidAccessor	The supplied value is invalid for {0}.
ValueIsInvalidAccessor	The value ‘{0}’ is invalid.
ValueMustBeANumberAccessor	The field {0} must be a number.
ValueMustNotBeNullAccessor	The value ‘{0}’ is invalid.

Update Startup ConfigureServices

Registrar singleton en método ConfigureServices

```
services.AddLocalization();
services.AddRequestLocalization(x =>
{
    //...
});

services.AddSingleton<IConfigureOptions<MvcOptions>, MvcConfiguration>();
```

RESUMEN

Recordemos

Globalization

Localization

REFERENCIAS

Para profundizar

https://docs.microsoft.com/en-us/aspnet/core/fundamentals/localization?view=aspnetcore-5.0

https://www.syncfusion.com/blogs/post/how-to-use-localization-in-an-asp-net-core-web-api.aspx

PREGRADO

Ingeniería de Software

Escuela de Ingeniería de Sistemas y Computación | Facultad de Ingeniería

UPC
Universidad Peruana
de Ciencias Aplicadas

Prolongación Primavera 2390,
Monterrico, Santiago de Surco
Lima 33 - Perú
T 511 313 3333
https://www.upc.edu.pe

exígete, innova