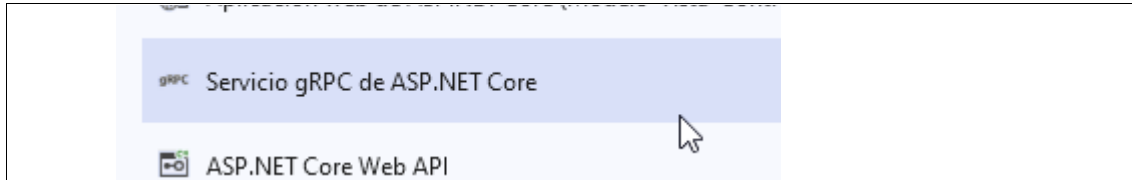


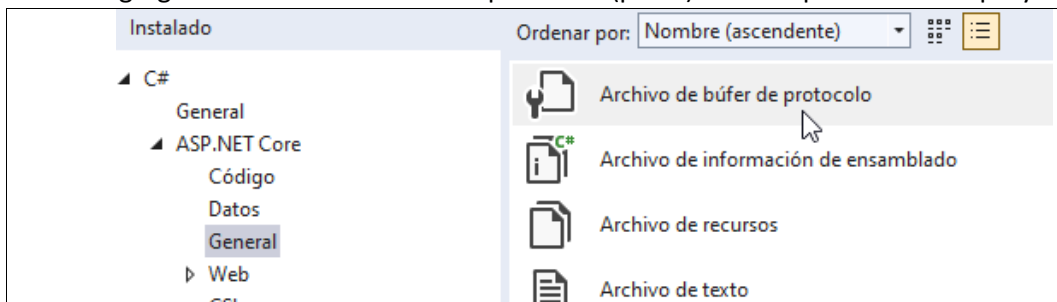
GUIA RAPIDA – SERVICIOS gRPC (Google Remote Procedure Call)

EN EL PROYECTO DEL SERVICIO gRPC

1. Iniciar un proyecto de tipo gRPC Asp Net Core



2. Agregar un archivo de buffer de protocolo (proto) en la carpeta Proto del proyecto



3. Desarrollar el “contrato” del servicio gRPC en el archivo Proto. No olvide poner un namespace que luego lo empleara al momento de implementar el servicio

```
syntax = "proto3";  
  
option csharp_namespace = "GrpcServicePromediar";  
  
package promediar;  
  
// Definición del servicio  
service PromediarNotas {  
  rpc CalcularPromedio (NotasRequest) returns (PromedioReply);  
}  
  
// Que se va a solicitar para en la invocacion del servicio.  
message NotasRequest {  
  int32 nota1 = 1;  
  int32 nota2 = 2;  
  int32 nota3 = 3;  
}  
  
// Que se va a responder.  
message PromedioReply {  
  float promedio = 1;  
}
```

4. Agregar al proyecto la configuración del archivo proto con el contrato de servicio (doble click al nombre del proyecto de servicio en el Explorador de proyectos)

```
<!--Agregar esto por cada proto -->  
<!--proto greet-->  
<ItemGroup>  
  <Protobuf Include="Protos\greet.proto" GrpcServices="Server" />  
</ItemGroup>  
<!--proto promediar-->  
<ItemGroup>  
  <Protobuf Include="Protos\promediar.proto" GrpcServices="Server" />  
</ItemGroup>
```

5. Implementar el servicio definido en el contrato del archivo Proto. Agregar una clase en la carpeta Services que deberá implementar lo indicado en el contrato (archivo proto).

Verificar que en dicha clase figure el namespace definido en el archivo proto, hacer las importaciones del caso e implementar lo definido en el contrato

```
using Grpc.Core;
using GrpcServicePromediar; //namespace definido en el contrato
namespace GrpcNotas.Services
{
    public class PromediarNotasServices: PromediarNotas.PromediarNotasBase
    {
        // Implementar una instancia de la clase Logger para que nos ayude a
        // verificar algunos sucesos en la ejecución del servicio
        private readonly ILogger<PromediarNotasServices> _logger;

        public PromediarNotasServices(ILogger<PromediarNotasServices> logger)
        {
            _logger = logger;
        }

        // Se implementa el metodo definido en el contrato
        public override Task<PromedioReply> CalcularPromedio(NotasRequest request)
```

6. Registrar el servicio en la clase Program del proyecto del servicio

```
// Registro del servicio
app.MapGrpcService<GrpcNotas.Services.PromediarNotasServices>();
app.MapGet("/", () => "La comunicación con los puntos finales (endpoints) de gRPC debe realizarse a través de un cliente gRPC");
app.Run();
```

7. Ejecutar el proyecto de servicio gRPC y en la consola de hospedaje capturar el puerto donde se aloja el servicio

```
Selecciónar E:\Cibertec\DESARROLLO DE APLICACIONES ASP NET CORE Y SERVICIOS\DESARROLLO TEMAS (CLASES)\INSTRUCTOR\TEMA 04 INSTRUCT...
Info: Microsoft.Hosting.Lifetime[14]
Now listening on: https://localhost:7176
Info: Microsoft.Hosting.Lifetime[14]
Now listening on: http://localhost:5296
Info: Microsoft.Hosting.Lifetime[0]
Application started. Press Ctrl+C to shut down.
Info: Microsoft.Hosting.Lifetime[0]
Hosting environment: Development
Info: Microsoft.Hosting.Lifetime[0]
Content root path: E:\Cibertec\DESARROLLO DE APLICACIONES ASP NET CORE Y SERVICIOS\DESARROLLO TEMAS (CLASES)\INSTRUCTOR\TEMA 04 INSTRUCTOR\GrpcServiceRegistro\GrpcServiceRegistro
```

PROYECTO CLIENTE

1. Agregar a la solución un proyecto cliente MVC
2. Agregar al proyecto cliente la referencia al servicio gRPC, accediendo al archivo proto del contrato del servicio y estableciendo el tipo de referencia como Cliente.
3. Verificar, dando doble click al archivo del proyecto cliente en el Explorador de soluciones, el registro de la referencia al servicio:

```
<Protobuf Include="..\GrpcServiceRegistro\Protos\promediar.proto" GrpcServices="Client">
  <Link>Protos\promediar.proto</Link>
</Protobuf>
</ItemGroup>
```

4. Desde el administrador de paquetes Nuguet instalar el paquete Grpc.Net.Client
5. Agregar un controlador al proyecto cliente, el cual implementara los IActionResult que invoquen a los métodos del servicio al cual se desea invocar. No olvide agregar las importaciones necesarias en dicho controlador

```

using Microsoft.AspNetCore.Mvc;
// Agregar
using GrpcServicePromediar; //referencia al servicio
using Grpc.Net.Client;

namespace app_Core5.Controllers
{
    public class PromedioController : Controller
    {
        private PromediarNotas.PromediarNotasClient _cliente;

        //Index : Segmento Get (Vista para ingresar las notas a promediar)
        public IActionResult Index()
        {
            ViewBag.n1 = 0;
            ViewBag.n2 = 0;
        }
    }
}

```

6. Implementar las vistas de los IActionResult implementados
7. Para la prueba del cliente, debe ejecutar los 2 proyectos en simultaneo. Recuerde que, sino ejecuta el proyecto del servicio gRPC, el servicio no estará hospedado y el proyecto cliente no podrá invocarlo. Para ello debe configurar su solución con la opción "Configurar proyectos de inicio" para que se ejecuten los 2 proyectos en simultaneo:

