

Technologia .NET Remoting pozwala na tworzenie usług analogicznych do komunikacji RPC (DCOM), Java RMI, Corba itp.

APLIKACJA SERWEROWA

```
using System;
using System.Runtime.Remoting;
using System.Runtime.Remoting.Channels;
// trzeba dodac do projektu referencje NET -> System.Runtime.Remoting
using System.Runtime.Remoting.Channels.Tcp;

// namespace z obiektem musi sie tak samo nazywac i po stronie serwera, i po stronie klienta
// najlepiej, jesli interfejs jest zdefiniowany w jednym pliku, ktorego kopie sa w obu projektach

namespace ObZdalny
{
    public interface IAPPrzyklad
    {
        int DodajNapis(string napis);          // public ...
    }

    public class CAPPrzyklad : MarshalByRefObject, IAPPrzyklad
    {
        public static string calynapis = "";

        public int DodajNapis(string napis)
        {
            calynapis += napis + "\r\n";
            return 1;
        }
    }
}

namespace NETRemSerwer
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Wcisnij dowolny klawisz aby uruchomic serwer");
            Console.ReadKey();

            TcpChannel tcpchl = new TcpChannel(5000);

            ChannelServices.RegisterChannel(tcpchl, false);

            Type typserwerar = typeof(ObZdalny.CAPPrzyklad);

            // SingleCall - model, w ktorym kazde zapytanie od aktywatora ma svoj wlasny obiekt
            // Singleton - model, w ktorym zapytania odwołują się do jednego obiektu

            RemotingConfiguration.RegisterWellKnownServiceType(typserwerar, "Zdalny",
                                                                WellKnownObjectMode.Singleton);

            Console.WriteLine("Wcisnij dowolny klawisz aby wyswietlic otrzymane napisy");
            Console.ReadKey();
            Console.Write(ObZdalny.CAPPrzyklad.calynapis);

            Console.WriteLine("Wcisnij dowolny klawisz aby zakonczyc");
            Console.ReadKey();
        }
    }
}
```

APLIKACJA KLIENCKA

```
using System;
using System.Runtime.Remoting;
using System.Runtime.Remoting.Channels;
// trzeba dodac do projektu referencje NET -> System.Runtime.Remoting
using System.Runtime.Remoting.Channels.Tcp;

// namespace z obiektem musi sie tak samo nazywac i po stronie serwera, i po stronie klienta
// najlepiej, jesli interfejs jest zdefiniowany w jednym pliku, ktorego kopie sa w obu projektach

namespace ObZdalny
{
    public interface IAPPrzyklad
    {
        int DodajNapis(string napis);        // public ...
    }
}

namespace NETRemKlient
{
    class Program
    {
        static void Main(string[] args)
        {
            Type SerwerTyp = typeof(ObZdalny.IAPPrzyklad);

            string adres = "tcp://" + "localhost" + ":5000/Zdalny";

            object obj = Activator.GetObject(SerwerTyp, adres);
            ObZdalny.IAPPrzyklad iprz = (ObZdalny.IAPPrzyklad)obj;

            Console.WriteLine("napisz tekst");
            string tekst = Console.ReadLine();
            iprz.DodajNapis(tekst);
        }
    }
}
```

