Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут ім. Ігоря Сікорського”

Кафедра автоматизації проектування енергетичних процесів і систем

Звіт

з лабораторної роботи №4

з дисципліни:

«Проектування систем з розподіленими бащами даних»

Виконали:

група ТР-01мп  
Кривошеев Е.В.

Лєсніченко Р.А.

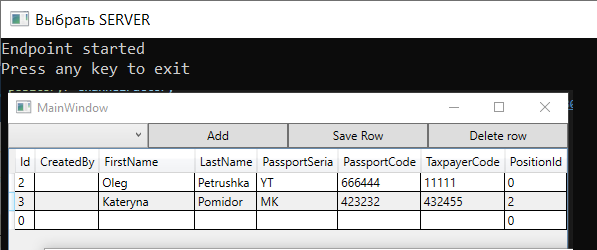
Перевірив:

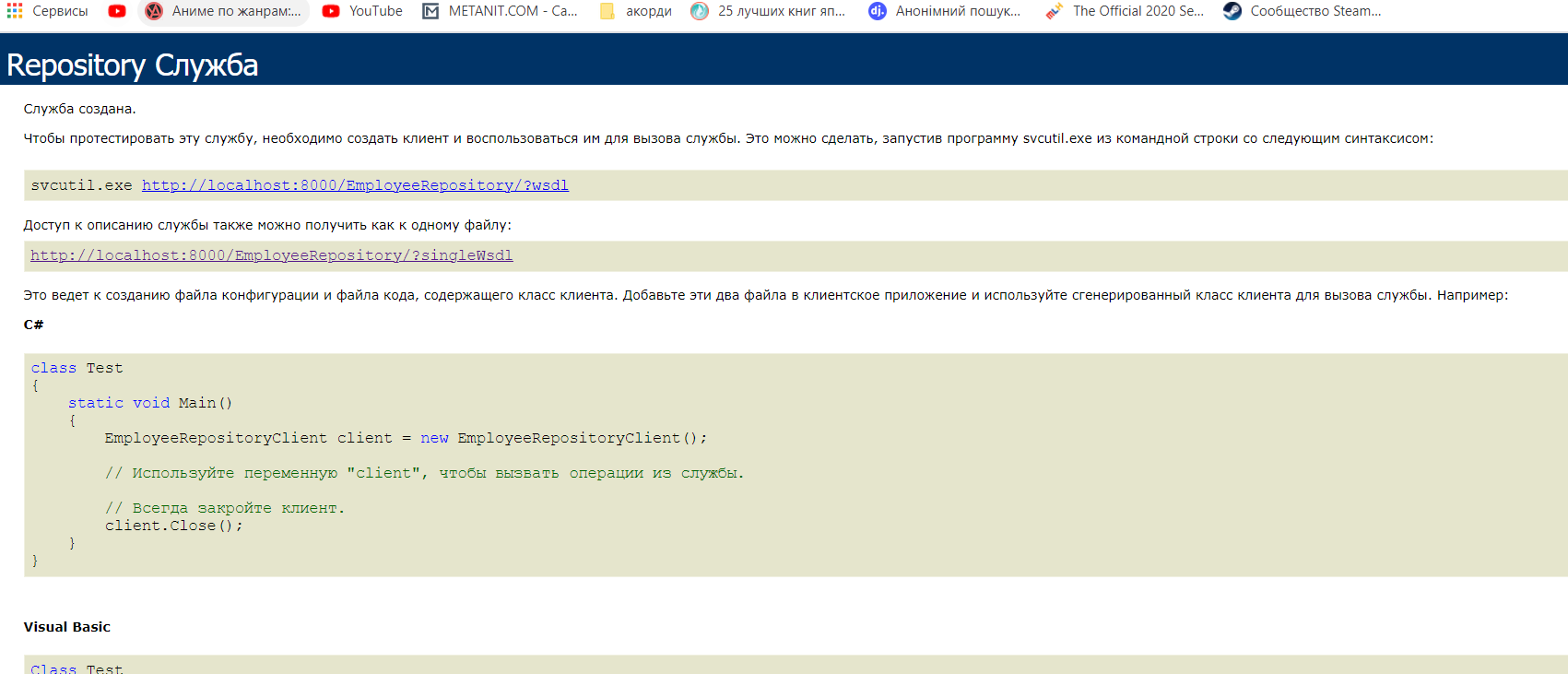
Варава І.А.

Київ – 2020

Завдання: Триланкова система .Net Remoting (конфігурація в коді та конфігураційному файлі)Умовна схема системи:

Скріншот роботи:





WSDL mex-точки

WCF Test Client , на жаль, не було знайдено в папці Visual Studio 2019

Код сервера:

static void Main(string[] args)

{

Console.Title = "SERVER";

//Configuration from file (should be uncommented also app.config file)

//ServiceHost host = new ServiceHost(typeof(Repository));

//configuration from code

ServiceHost host = new ServiceHost(typeof(Repository), new Uri("http://localhost:8000/EmployeeRepository/"));

BasicHttpBinding binding = new BasicHttpBinding();

binding.CloseTimeout = new TimeSpan(0, 1, 0);

binding.ReceiveTimeout = new TimeSpan(0, 1, 0);

host.AddServiceEndpoint(typeof(IEmployeeRepository), binding, "");

ServiceMetadataBehavior smb = new ServiceMetadataBehavior();

smb.HttpGetEnabled = true;

host.Description.Behaviors.Add(smb);

host.Open();

Console.WriteLine("Endpoint started");

Console.WriteLine("Press any key to exit");

Console.ReadKey();

}

Код конфігураційного файлу

<system.serviceModel>

<services>

<service name="WcfServer.Repository">

<host>

<baseAddresses>

<add baseAddress="http://localhost:8000/EmployeeRepository/" />

</baseAddresses>

</host>

<endpoint address="" binding="basicHttpBinding" contract="Models.IEmployeeRepository" />

</service>

</services>

</system.serviceModel>

public MainViewModel()

{

try

{

AddNewRowCommand = new RelayCommand(AddNew);

SaveSelectedRowCommand = new RelayCommand(SaveSelected);

DeleteSelectedRowCommand = new RelayCommand(DeleteSelected);

ChannelFactory<IEmployeeRepository> channelFactory =

new ChannelFactory<IEmployeeRepository>(new BasicHttpBinding(), new EndpointAddress("http://localhost:8000/EmployeeRepository/"));

\_repository = channelFactory.CreateChannel();

Rows = new ObservableCollection<Employee>( \_repository.GetObjects());

}

catch(Exception ex)

{

MessageBox.Show(ex.Message);

}

}

public class Repository : IEmployeeRepository

{

private string \_connectionString;

private DbContextOptions<LvivFilialDBContext> dbOptions;

public Repository()

{

\_connectionString = "Server = localhost\\SQLExpress; Database = LvivFilialDB; Trusted\_Connection = True;";

dbOptions = new DbContextOptionsBuilder<LvivFilialDBContext>()

.UseSqlServer(\_connectionString).Options;

}

public void Create(Employee obj)

{

if (obj == null) throw new ArgumentNullException();

using (var context = new LvivFilialDBContext(dbOptions))

{

context.Add(obj);

context.SaveChanges();

}

}

public void Remove(Employee obj)

{

if (obj == null) throw new ArgumentNullException();

using (var context = new LvivFilialDBContext(dbOptions))

{

context.Remove(obj);

context.SaveChanges();

}

}

public IList<Employee> GetObjects()

{

using (var context = new LvivFilialDBContext(dbOptions))

{

return context.Employee.ToList();

}

}

public void Update(Employee obj)

{

if (obj == null) throw new ArgumentNullException();

using (var context = new LvivFilialDBContext(dbOptions))

{

context.Update(obj);

context.SaveChanges();

}

}

}

}

Зміна конфігурацій відбувається за допомогою коментування-розкоментування коду