Lesson 2

# How to start creating a VoIP solution with Ozeki VoIP SIP SDK

2012. 01. 12.

## Abstract

The second lesson of Ozeki VoIP Training will show you all the basic steps of starting VoIP application programming with Ozeki VoIP SIP SDK. You will find all the necessary information about the tools and requirements related to the topic. This material also contains a detailed guide for starting a Visual Studio Project and for registering Ozeki VoIP SIP SDK to the project, that needs to be done in case of every VoIP application.

## Introduction

Ozeki VoIP SIP SDK is an excellent development kit that helps you to create a professional voice over IP communication software even if you have not much experience in C# programming. If you know the basic programming techniques like method calls, parameter settings, declarations you will be able to write a VoIP program too.

This course material covers all the things you need to do in case of any VoIP application if you use Ozeki VoIP SIP SDK. You will find that after some basic settings a great set of namespaces, classes and methods will be available for you that will make your work really easy.

In the following sections you will find an illustrated guide about creating a C# Windows Forms Application and registering Ozeki VoIP SIP SDK as reference to the project. You will also find a guide that shows how you can built the user interface of your VoIP solution in Visual Studio

## Basic Tools

Before you start programming, you will need some basic software and hardware requirements like in case of any programs. Ozeki VoIP SIP SDK basically needs a Windows XP+ environment with Visual Studio 2010 and .NET Framework 3.5 or 4.0, and 1GHz+ processor with 1 GB (32 bit) or 2 GB (64 bit) RAM. In case of some solutions, you may also need Adobe Flash Builder too.

First of all you will need to install the programming tools and Ozeki VoIP SIP SDK on your computer. You can download the SDK on the
http://voip-sip-sdk.com/p_21-download-ozeki-voip-sip-sdk-voip.html page. You are suggested to install Ozeki VoIP SIP SDK without interaction. In that case the SDK .dll will be found at "c:\Program Files\Ozeki\VoIP SIP SDK\SDK\VoIPSDK.dll" location. This will be important to know in the future.
In case of a VoIP solution you will also need a Private Branch Exchange (PBX) system too for

ensuring the connection establishment between your clients. Ozeki VoIP SIP SDK supports a lot of PBY systems, some of them are also free or open source solutions. You can install one of the PBXs on your computer or virtual machine, but you can also look for an online PBX service provider to work with.

**Note:** If you are not familiar with the term PBX, please check Ozeki VoIP Training Lesson 1 about the topic.

If you choose to install your own PBX, you can check the necessary configuration steps for all supported PBX systems at http://voip-sip-sdk.com/p_103-voip-pbx-sdk-compatibility-voip.html.

# Getting Started

The first step in C# programming is to create a new Visual Studio project for your needs. In this first case let's start with a simple softphone solution. That will be a desktop application and let's use Windows Forms for the graphical user interface. In this case the application will look like a standard Windows application.

In the following paragraphs you can see an illustrated guide about how you can create your first Ozeki VoIP SIP SDK supported WinForms application. These steps are basically the same in case of any windows Forms application, the project name and location may differ.
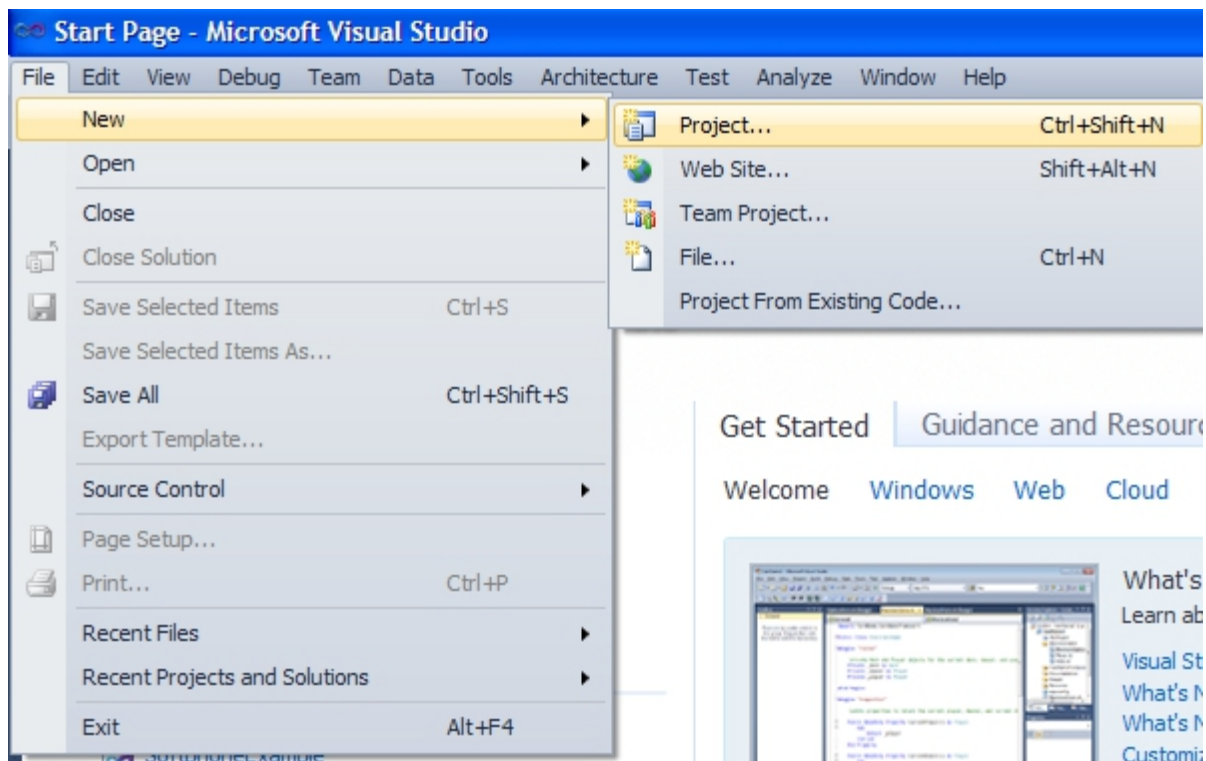


**Figure 1** - Creating a new project in Visual Studio

The very first step after starting Visual Studio 2010 is to create a new C# project. This can be done by using the File->New->Project... menu as it is shown in Figure 1.

(You can also use the File menu for Opening existing Visual Studio Projects on your hard drive.)

When you choose the File->New->Project... menu the New Project window will pop up where you can specify the project you want to create.
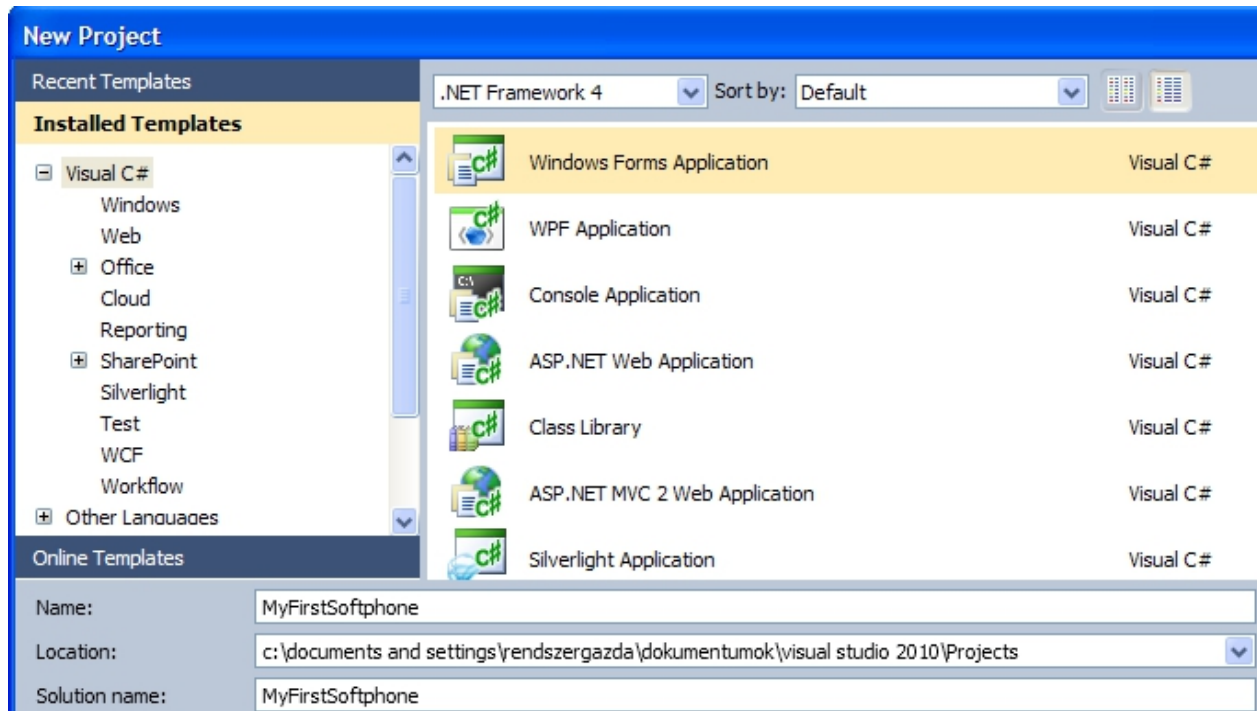


**Figure 2** - The pop-up window to define the project properties

The New Project window contains more panels with their own function. At first you need to specify the solution type you will use. You can choose from the template list on the left side of the window. In case of this VoIP application, you need to choose C# as the programming language.

The list on the right panel that shows the possible application types vary according to the template selection. In case of C# programming the first option is Windows Forms Application that is the one you will need to choose now.

If you want to choose other solutions you can be more specific about the template selection to have a shorter solution list or you can browse the whole list for the preferred solution.

On the bottom panel you can specify the name and the location of your new project. Visual Studio always give a generated name to a new project that you can modify according to your wish. You can also choose a different project folder if you want.

In the case of the example I have modified the name of the project to MyFirstSoftphone but kept the location folder Visual Studio offered. The location folder will be important to know if you want to copy your solution somewhere else.

Ozeki VoIP SIP SDK also provides support for webphone client and server applications that

are usually not desktop applications. In case of a webphone server, you are suggested to use console application type and as for a webphone client you can use a Silverlight solution. These application types can also be chosen on the New Project window.

After you specified all the basic properties of your project press Enter or click on the OK button and your C# project will be created by Visual Studio.

This is the point when you can make your work as easy as possible by getting the support of Ozeki VoIP SIP SDK. Visual Studio provides you the possibility to use third-party SDK tools as references in your projects. You only need to add the SDK .dll file to the project and you can instantly use all the tools it defines.

Figure 3 shows how you can add an SDK to the References of the project.
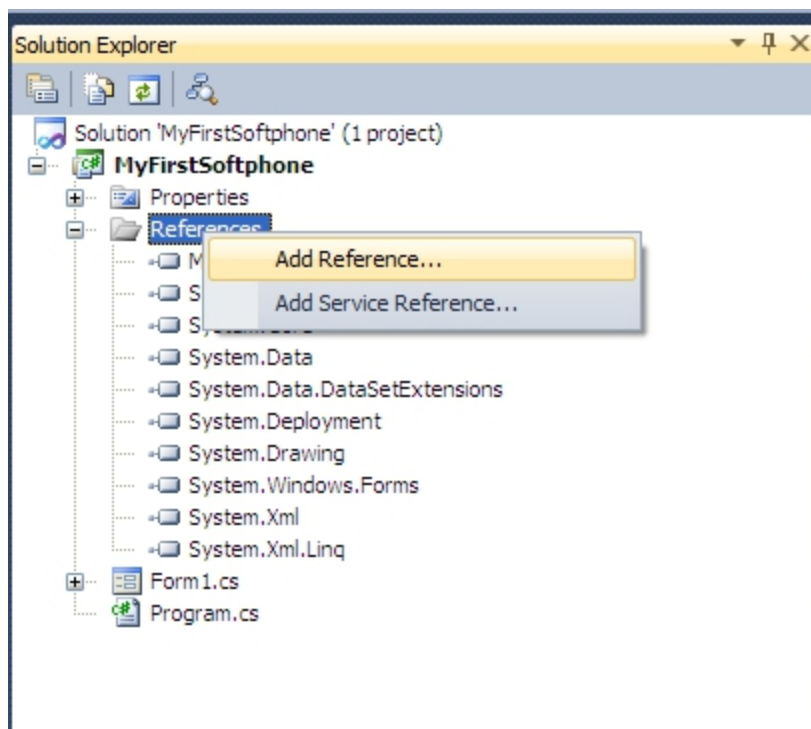


**Figure 3** - The Solution Explorer Panel

The Solution Explorer panel that can be found on the right side of the Visual Studio window contains all the tools that are defined in or registered to your project. When you want to register a new SDK to your project you need to right click on the References label and choose Add Reference... from the list.

When you click on the Add Reference item, the window shown in Figure 4 appears where you can specify the .dll you want to register to your project.
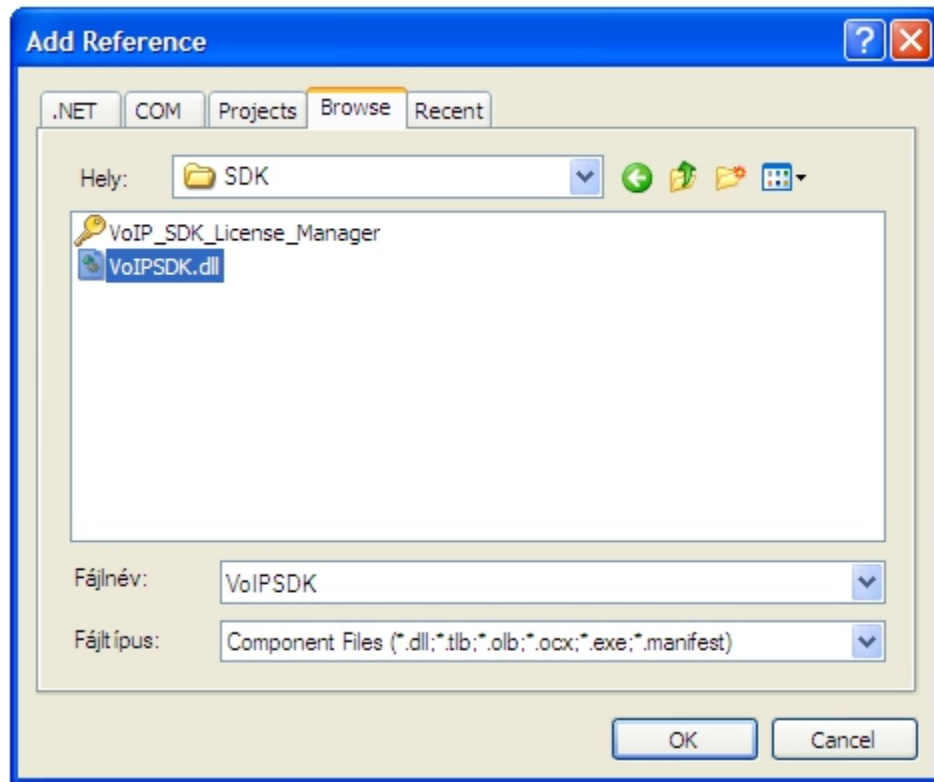
**Figure 4** - You can specify the SDK .dll reference file on this window

On the Add Reference window you can choose from more solutions to specify a reference. In this case you should choose the Browse tab and you need to look for the Ozeki VoIP SIP SDK .dll file on your hard drive.

If you have installed Ozeki VoIP SIP SDK without interaction, the .dll file will be located at "c:\Program Files\Ozeki\VoIP SIP SDK\SDK\" folder.

You can also use a .dll file from other Visual Studio projects by choosing the Projects tab, or you can use the resources provided by the programming environment on the .NET and Com tabs or you can simply choose from the recently used .dll files on the Recent tab.

When you have chosen the SDK .dll file, you need to press the OK button and the SDK will be registered to your project as a reference. This means that from this point you can reach all the namespaces, classes and methods that are provided in Ozeki VoIP SIP SDK.

If you want to check the tools Ozeki VoIP SIP SDK provides, you can do it by double clicking on the VoIPSDK label that appeared under the References on the Solution Explorer panel. When you double click on the VoIPSDK, the Object Browser will appear as a panel on the left side of the Visual Studio window and you can check the namespaces the SDK provides (Figure 5).
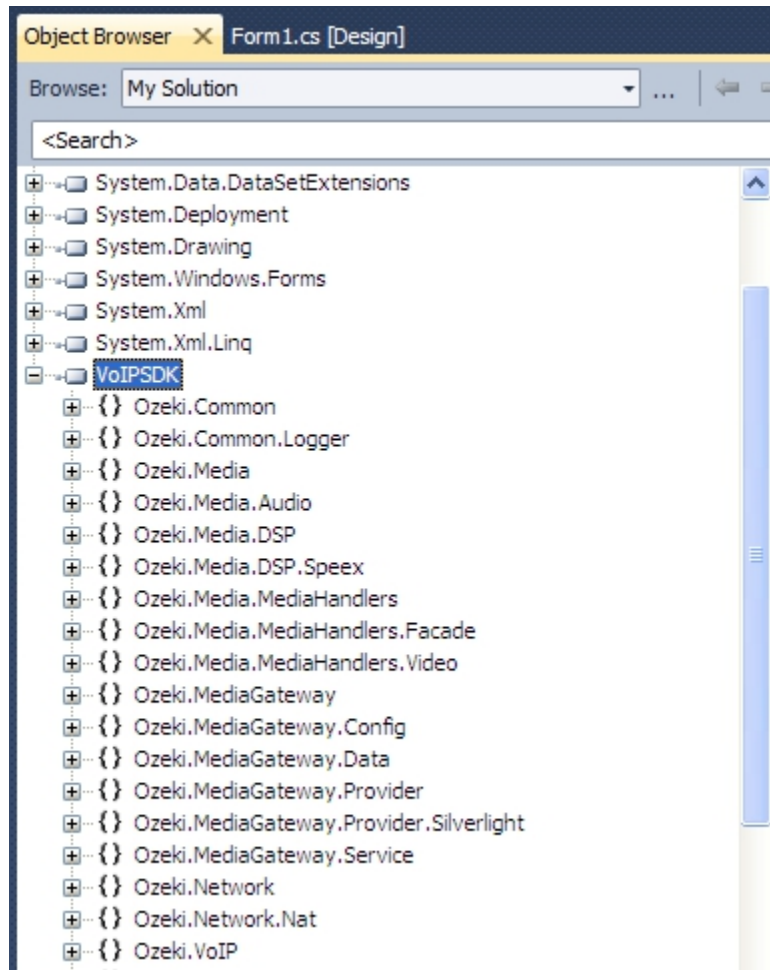
**Figure 5** - The Object Browser lists all the tools that can be used in your project

Any time you want to check the tools you can use for a solution, you are free to look them up on the Object Browser panel. You may also find descriptions for some tools here.

# Creating a Simple Windows Forms GUI

Windows Forms is a simple to use professional tool that can be used for creating the graphical user interface (GUI) for a desktop application. It is designed to create standard Windows applications that behave the same way as any other desktop apps, like the window can be closed by pressing the X button in the top right corner.

When you create a new WinForms application in Visual Studio, you get a simple blank window with the basic close, minimize and maximize function buttons in the top right corner. This is the main window of your application that can be extended with all the functions you want to use.

The GUI elements that can be added onto a WinForms window are listed on the Toolbox panel (Figure 6). Here you can choose the needed elements and put them onto the GUI by using drag and drop technology.
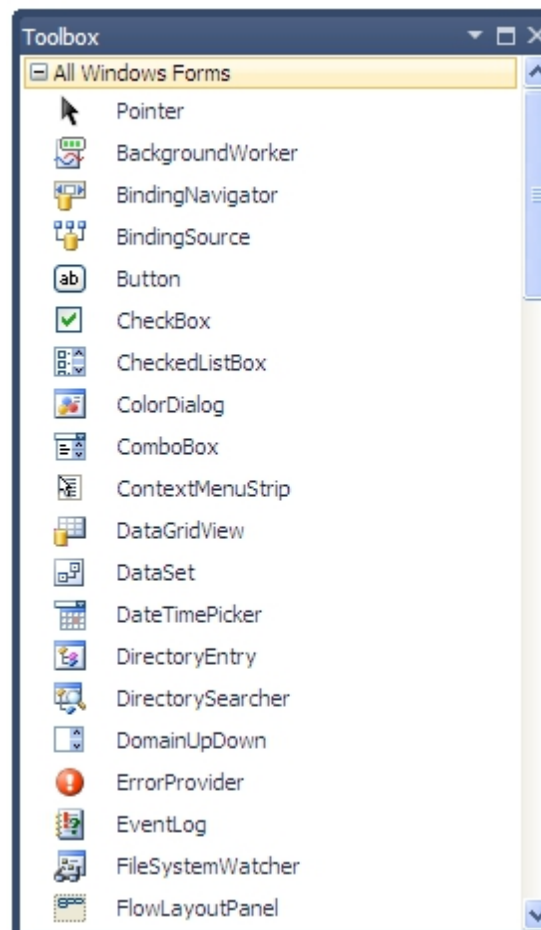
**Figure 6** - You can choose the GUI elements for your application on the Toolbox panel

When you put a GUI element on the application window, you can modify its properties like the size, location, displayed text etc. according to the element type. This can be done on the Properties panel (Figure 7).
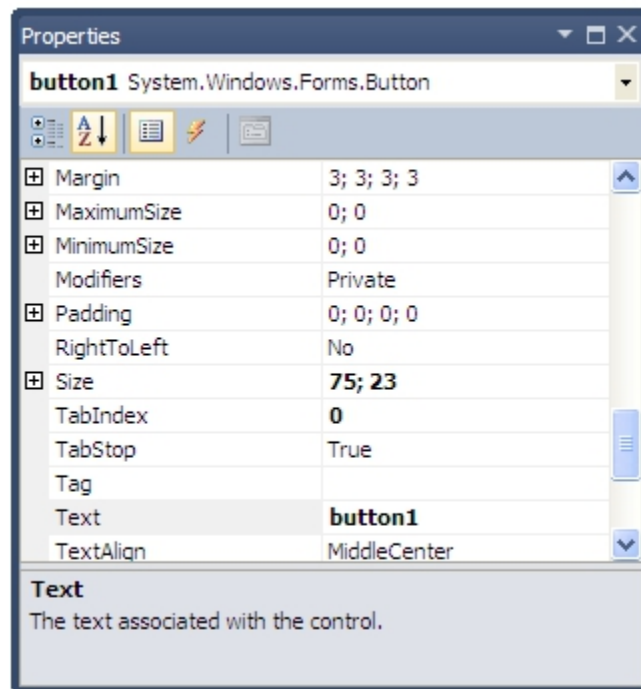
**Figure 7** - The properties of a GUI element can be modified on the Properties panel

The Properties panel can be opened from the View->Properties Window menu or by right clicking on one of the GUI element and choosing Properties from the pop-up list as it is shown in Figure 8.
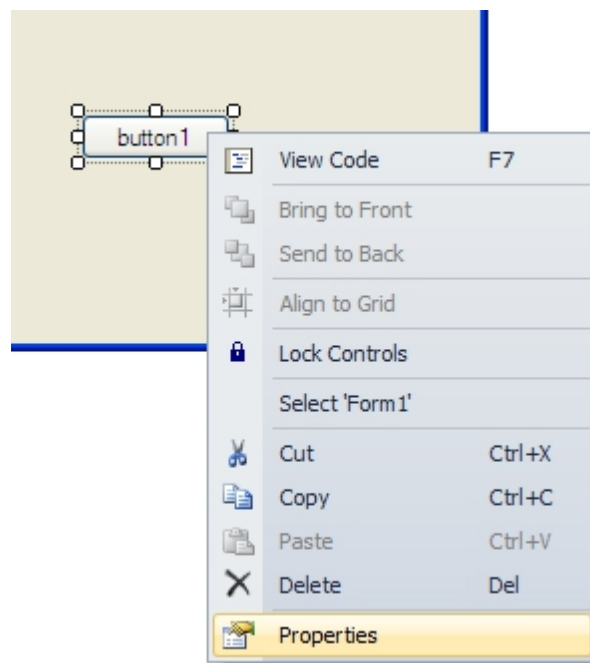


**Figure 8** - Opening the Properties panel

The Properties panel contains the modifiable properties according to the selected GUI element.

On Figure 7 a Button object was selected, therefore the Properties panel showed the relevant settings. The properties that are shown in bold are the ones that are usually customized in case of a certain element.

When you make a GUI design you need to think over the functionalities you want to put onto the user interface. You can use a lot of GUI elements, standard buttons, menus, text boxes, the decision should be made according to the purpose of the application you make.

A softphone GUI for example should look like a traditional telephone with a keypad, some labels and a text box for the number you want to dial. If you want to build a softphone that is capable for video phoning, you also need to put video displays on the GUI. The possibilities are only limited to your imagination.

# Summary

This second lesson of Ozeki VoIP Training introduced you the basic first steps that are needed when you want to create a VoIP application using the background support of Ozeki VoIP SIP SDK. If you read this guide carefully, you can create a Visual Studio project and register Ozeki VoIP SIP SDK as a reference to it by now. Try to answer the knowledge check questions to see if you have understood all the topics that were covered in this material.

# Knowledge Check Questions

1. How can you specify the program type you want to create in Visual Studio?
2. What kind of application types should be used for an Ozeki VoIP SIP SDK supported VoIP solution?
3. Where can you register your Ozeki VoIP SIP SDK to your C# project
4. What kind of GUI elements are needed for a simple softphone application?

# Correct Answers for Knowledge Check Questions

1. The program type of an application in Visual Studio can be specified during the creation process. You can choose the main type from a template, like C# language and then you can choose from the listed application types on the New Project window.

2. You can make different VoIP solutions with Ozeki VoIP SIP SDK. A simple desktop softphone can be created for example as a WinForms application. A webphone client is usually a simple console application, while a webphone client can be defines as a Silverlight solution. These are, of course, the simplest examples only, there are more possibilities you can use.

3. You need to right click on the References label on the Solution Explorer panel in Visual Studio if you want to register your SDK to the project you have created. You need to choose Add Reference from the pop-up menu and you can specify the .dll file to use on the appearing window.

4. As a simple softphone application is the software model of an original telephone set is basically need to look like one. This means that it needs a keypad and some notification labels, and maybe a text box where you can type in the dialed number.