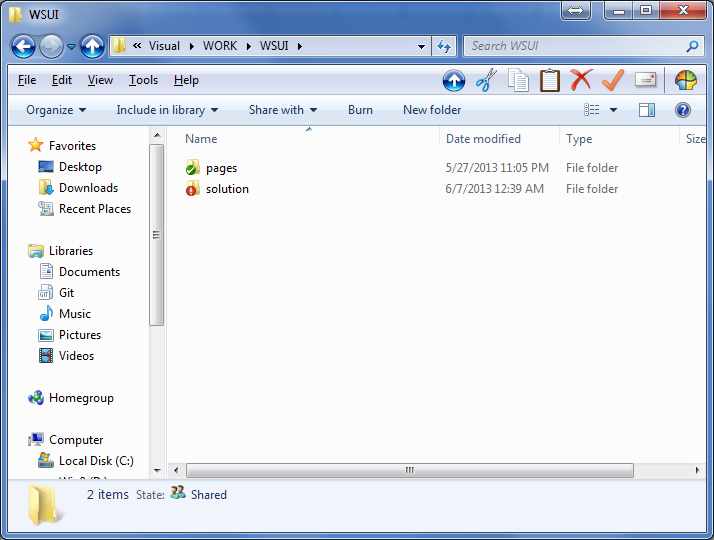
**The project should had next structure:**



Main folder should be named – “**WSUI**” or something like other, but inner folders must be named as “**solution**” – for storing “***main***” branch of the project;

“**pages**” – for storing “***gt-pages***” branch, where situated site of the project.

***For deploying new version of the project we should:***

**1.** Enter to the “Toolkits” folder in “solution” folder.

**2.** Open file – “wsgen.exe.config”.

**3.** Change version of project - settings named - > “**BuildNumber**”. For example:

*Before:*

**<setting name="BuildNumber" serializeAs="String">**

**<value>1.0.1</value>**

**</setting>**

*After:*

**<setting name="BuildNumber" serializeAs="String">**

**<value>1.0.2</value>**

**</setting>**

**4.** Run “deploy.bat”.

**Decribing files.**

**wsui.build** – the main file for creating folders, compile solution, making clicktwice installer and deploying to the github.

**deploy.bat** - bat-file which run deploying process. It run run.bat with parameter which mean what action should be start during deploying.

**go.bat** - run nant.exe with wsui.build.

**wsgen.exe** – generate new Version file for subscribing dlls and update version in Installer project.

**wsadxpublisher.exe** – it sets up and clears settings for creating clicktwice installer.

**wsgen.exe.config** – settings for wsgen.exe.

**wsadxpublisher.exe.config** – settings for wsadxpublisher.exe.

**Describing parameters in settings files.**

Sample of **wsgen.exe.config** with comments for each parameters:

<?xml version="1.0"?>

<configuration>

    <configSections>

        <sectionGroup name="applicationSettings" type="System.Configuration.ApplicationSettingsGroup, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089">

            <section name="WSGen.Properties.Settings" type="System.Configuration.ClientSettingsSection, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" requirePermission="false"/>

        </sectionGroup>

        <sectionGroup name="userSettings" type="System.Configuration.UserSettingsGroup, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089">

            <section name="WSGen.Properties.Settings" type="System.Configuration.ClientSettingsSection, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" allowExeDefinition="MachineToLocalUser" requirePermission="false"/>

        </sectionGroup>

    </configSections>

    <applicationSettings>

        <WSGen.Properties.Settings>

            <!-- filename where should be saved new version (not required) -->

            <setting name="BulidNumberFilename" serializeAs="String">

                <value>wsui.number</value>

            </setting>

            <!-- relative path to instakller project. App will update version and GUID for installer (required) -->

            <setting name="SetupProjectFile" serializeAs="String">

                <value>\WSUI.OutlookPlugin\WSUI.OutlookPluginSetup\Product.wxs</value>

            </setting>

            <!-- new version of project. It should be changed before deploying (required)-->

            <setting name="BuildNumber" serializeAs="String">

                <value>1.0.1</value>

            </setting>

            <!-- Name of file which should be generated by app. For subscribing dlls in the project (required)-->

            <setting name="VersionFilename" serializeAs="String">

                <value>Version.cs</value>

            </setting>

            <!-- patter for finding version in Visual Setup Project (not required) -->

            <setting name="ProductVersion" serializeAs="String">

                <value>"ProductVersion" = "8:[1-9]{1}\.[0-9]{1}\.[0-9]{1}"</value>

            </setting>

             <!-- patter for finding GUID in Visual Setup Project (not required) -->

            <setting name="ProductCode" serializeAs="String">

                <value>"ProductCode" = "8:\{[0-9a-fA-F]{8}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{12}\}"</value>

            </setting>

          <!-- patter for adding version in Visual Setup Project (not required) -->

            <setting name="ProductVersionTemplate" serializeAs="String">

                <value>"ProductVersion" = "8:{0}"</value>

            </setting>

        </WSGen.Properties.Settings>

    </applicationSettings>

    <userSettings>

        <WSGen.Properties.Settings>

            <setting name="ProductCodeTemplate" serializeAs="String">

                <value>"ProductCode" = "8:{0}"</value>

            </setting>

        </WSGen.Properties.Settings>

    </userSettings>

<startup><supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.0"/></startup></configuration>

Sample of **wsadxpublisher.exe.config** with comments for each parameters:

<?xml version="1.0"?>

<configuration>

<configSections>

    <sectionGroup name="applicationSettings" type="System.Configuration.ApplicationSettingsGroup, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" >

        <section name="WSADXPublisher.Properties.Settings" type="System.Configuration.ClientSettingsSection, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" requirePermission="false" />

    </sectionGroup>

</configSections>

<startup><supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.0"/></startup><applicationSettings>

        <WSADXPublisher.Properties.Settings>

            <!--Absolute path to MSI installer (required)-->

            <setting name="installerFile" serializeAs="String">

                <value>h:\Visual\WORK\WSUI\solution\ROutput\WSUI.OutlookPluginSetup.msi</value>

            </setting>

            <!--Absolute path to folder, where we store temp files for deploying (required)-->

            <setting name="publishingLocation" serializeAs="String">

                <value>h:\Visual\WORK\WSUI\solution\Clicktwice</value>

            </setting>

            <!-- url where we will deploy out project (required) -->

            <setting name="installationUrl" serializeAs="String">

                <value>http://fruitfactory.github.io/WSSQL\_GUI/downloads/clicktwice/</value>

            </setting>

            <!-- Absolute path to installer sertificate (required) -->

            <setting name="certificateFile" serializeAs="String">

                <value>h:\Visual\WORK\WSUI\solution\WSUI.OutlookPlugin\wsuiclicktwice.pfx</value>

            </setting>

            <!-- absolute path to icon for installer (required)-->

            <setting name="iconFileName" serializeAs="String">

                <value>h:\Visual\WORK\WSUI\solution\WSUI\Search.ico</value>

            </setting>

            <!-- If UI while installing will be shown (required) False - will be shown, False - otherwise -->

            <setting name="quietModeDuringInstall" serializeAs="String">

                <value>False</value>

            </setting>

            <!-- IF Full UI will be shown while unistall (True - quiet mode (not shown), False - will be shown) (required)-->

            <setting name="quietModeDuringUninstall" serializeAs="String">

                <value>True</value>

            </setting>

            <!-- If download window will be shown (required)-->

            <setting name="showDownloaderWindow" serializeAs="String">

                <value>False</value>

            </setting>

            <!--Absolute path to folder where was installed Add-In Express toolkit (required)-->

            <setting name="AddInExpressBinFolderPath" serializeAs="String">

                <value>C:\Program Files (x86)\Add-in Express\Add-in Express for .NET\Bin\</value>

            </setting>

            <!-- Name of settings file, which we should set up before deploying and clear after it (required)-->

            <setting name="ADXPublisherConfigFilename" serializeAs="String">

                <value>adxpublisher.exe.config</value>

            </setting>

            <!--  Downloader filename (required) -->

            <setting name="downloaderFileName" serializeAs="String">

                <value>wsui.outlookplugin.exe</value>

            </setting>

            <!-- target framework which use out project (required)  -->

            <setting name="targetFrameworkVersion" serializeAs="String">

                <value>v4.0.30319</value>

            </setting>

            <!-- Programs name who will be closed durring install (required) -->

            <setting name="targetApplicationNames" serializeAs="String">

              <value>Outlook</value>

            </setting>

            <!--show warnings durring installation (required)-->

            <setting name="showRunningApplicationsWarning" serializeAs="String">

                <value>False</value>

            </setting>

        </WSADXPublisher.Properties.Settings>

    </applicationSettings>

</configuration>

File wsui.build:

  <!-- clear folder where we store builded files -->

  <target name="cleanROutput" unless="${target::has-executed('cleanROutput')}">

    <echo message="Clean Routput directory..." />

    <delete dir="${bin.dir}" failonerror="false"  if="${directory::exists(bin.dir)}"/>

  </target>

  <!-- create new empty folder for storing files -->

  <target name="createROutput" unless="${target::has-executed('createROutput')}">

      <echo message="Create ROutput directory..." />

      <mkdir dir="${bin.dir}" failonerror="true" unless="${directory::exists(bin.dir)}"/>

  </target>

  <!-- run updating version and GUID for installer project and generate Version.cs -->

  <target name="updatedVersionInfo" unless="${target::has-executed('updatedVersionInfo')}" >

    <echo message="Update version info..." />

    <exec program="wsgen.exe" basedir="${build.tools.dir}" commandline='"${source.directory}"' workingdir="." />

  </target>

  <!-- compile solution by Visual Studio and write buid log into ROutput folder -->

  <target name="compilesolution" depends="updatedVersionInfo, cleanROutput, createROutput">

    <echo message="Compiling Solution..." />

    <exec program="devenv.com"  basedir="C:\Program Files (x86)\Microsoft Visual Studio 10.0\Common7\IDE"

  commandline='"${source.directory}\${solution}" /Build "${project.build.type}|Mixed Platforms" /Out "${bin.dir}\Build\_WSUI\_Solution.log" ' workingdir="." />

  </target>

  <!-- set up settings for creating clicktwice installer -->

  <target name="setSettingsADX" depends="compilesolution">

    <echo message="Set publishing settings..." />

    <exec program="wsadxpublisher.exe" basedir="${build.tools.dir}" commandline='/set' workingdir="." />

  </target>

  <!-- make clicktwice installer, generate version\_file.xml and copy fo Clicktwice folder -->

  <target name="publishClickTwice" depends="setSettingsADX">

    <echo message="Publish..." />

    <exec program="adxpublisher.exe" basedir="C:\Program Files (x86)\Add-in Express\Add-in Express for .NET\Bin" commandline='/OutputType=ClickTwice' workingdir="." />

  </target>

  <!-- clear settings which we set on  setSettingsADX step-->

  <target name="clearSettingsADX" depends="publishClickTwice">

    <echo message="Clear publishing settings..." />

    <exec program="wsadxpublisher.exe" basedir="${build.tools.dir}" commandline='/clear' workingdir="." />

  </target>

  <!-- copy ready files to folder pages\downloads\clicktwice-->

  <target name="copyFiles" depends="clearSettingsADX">

    <exec program="copy.bat" verbose="true">

    </exec>

  </target>

  <!-- run script which commit and push out installer on GitHub -->

  <target name="deploy" depends="copyFiles">

    <exec program="bash.exe" basedir="C:\Program Files (x86)\Git\bin\" verbose="true">

      <environment>

        <variable name="HOME" value="${environment::get-variable('userprofile')}"/>

        <variable name="ADDPATH" value=".:/usr/local/bin:/mingw/bin:/bin:/bin"/>

      </environment>

      <arg file="deploy.sh" />

    </exec>

  </target>