Sage 300 Web Screens SDK

Solution Wizard

August 2021

The MIT License (MIT)

Copyright © 2021 The Sage Group plc or its licensors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Contents

[1. Overview 4](#_Toc80358546)

[2. Accessing the Solution Wizard 5](#_Toc80358547)

[3. Solution Wizard Inputs 6](#_Toc80358548)

[3.1 Solution Information 6](#_Toc80358549)

[3.2 Kendo UI Information 7](#_Toc80358550)

[3.3 Resource Files 8](#_Toc80358551)

[3.4 Generate Solution 9](#_Toc80358552)

[3.5 Set Login.aspx as the Start Page 9](#_Toc80358553)

[4. Examining the Solution 10](#_Toc80358554)

[4.1 Solution Explorer 10](#_Toc80358555)

[4.2 Business Repository Project 10](#_Toc80358556)

[4.3 Interfaces Project 11](#_Toc80358557)

[4.4 Models Project 12](#_Toc80358558)

[4.5 Resources Project 12](#_Toc80358559)

[4.6 Services Project 13](#_Toc80358560)

[4.7 Web Project 14](#_Toc80358561)

1. Overview

This document is intended to serve as a guide for illustrating how to use the Solution Wizard to create a Visual Studio Solution for Sage 300 Web User Interfaces (“UIs”).

The Solution Wizard is a Visual Studio Plugin and is compatible with Visual Studio 2019.

The intent of the wizard is to create a solution for Sage 300 Web UIs. The wizard will provide the necessary scaffolding and structures to accommodate the new Sage 300 Web UIs whether they are created manually or generated via the Sage 300 Code Generation Wizard.

The wizard will create the required projects in preparation for creating screens, reports, inquiries, etc. The projects created are:

* Business Repository

Contains the code files for the Entity Repositories, Entity Mappers, Menu Navigation XML, Security Constants, and so on.

* Interfaces

Contains the code files for the Entity Repository/Service Interfaces.

* Models

Contains the code files for the Models, Model Enumerations, and Model Fields.

* Resources

Contains the code files for the Resources (Resx Files) for supported languages.

* Services

Contains the code files for the Services.

* Web

Contains the standard code files based upon MVC patterns (Controllers, View Models, JavaScript, Razor Views, Configuration files, and so on.

The solution created is aligned with how the ASP.Net tools create solutions and allows the wizard to leverage everything built into Visual Studio. Therefore, instead of a separate utility, the wizard is embedded directly into the Visual Studio IDE.

1. Accessing the Solution Wizard

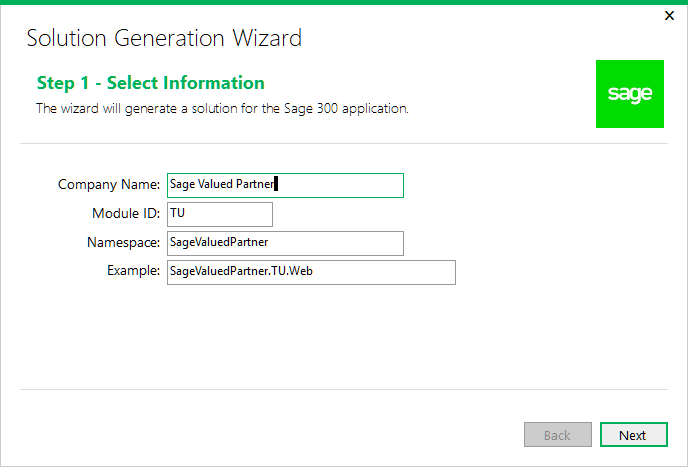
Because the wizard is embedded in Visual Studio as a plugin, when selecting the option to create a new project, you will see the Sage 300 Solution Wizard option:



Simply select the name that is to be used for the solution, and then specify any other options as needed in the New Project Dialog.

1. Solution Wizard Inputs
   1. Solution Information

After supplying the information required to create a new project (solution), the following dialog box appears:



The following information is required by the wizard:

* Company Name

Used for copyright information and to provide a default value for the Namespace field.

* Module ID

Used as a Module designator that will be used to segregate the projects, and will also become part of the namespace.

* Namespace

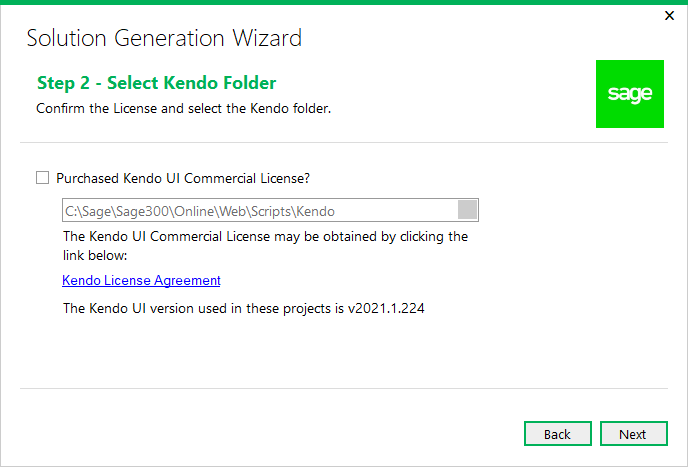
Used as the base Namespace value. The namespace will become a composite of this field and the Module ID. This field is defaulted from the Company Name field but may be overridden.

* Example

This is a readonly field to give the end user an example of what a full namespace will look like based on the entered information.

Click X to exit the wizard, or click Next to proceed.

* 1. Kendo UI Information



The following information is required by the wizard:

* Purchased Kendo Commercial License

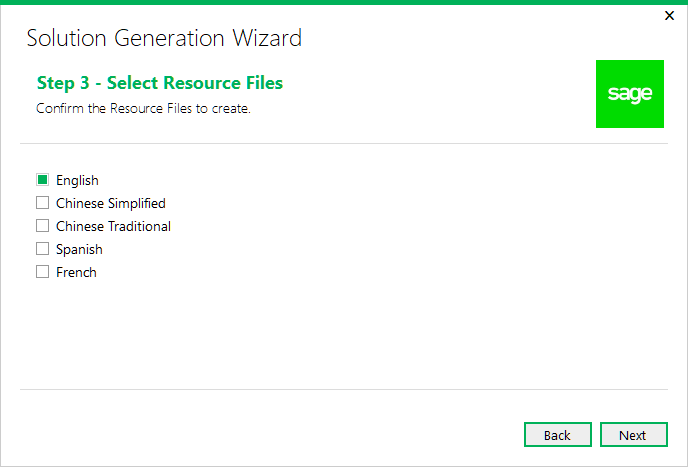
This checkbox must be selected or the Solution Wizard will not proceed. This is the acknowledgment that the user/consumer has purchased a Kendo License since Sage does not distribute the Kendo files for development purposes.

* Kendo Folder

Defaults to the location of the Sage 300 Web installation in order to consume that version of the Kendo file. An alternate folder may be entered, but the Kendo folder must have the version specified as being compatible with Sage 300c Web Screens

Click Back to return to the previous step, or click Next to proceed.

* 1. Resource Files



The following information is required by the wizard:

* English

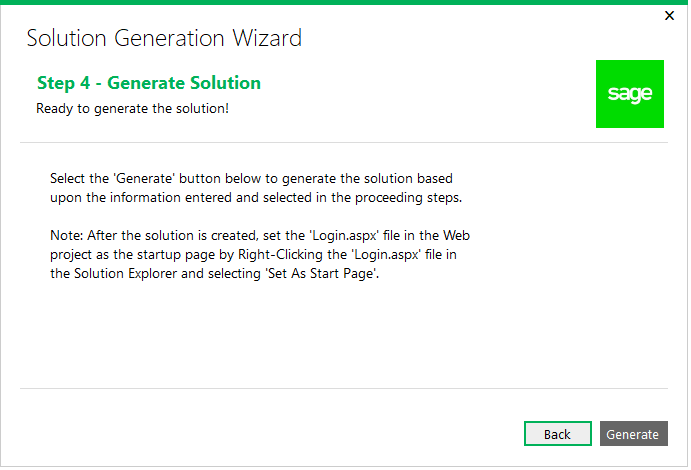
This checkbox is selected by default and cannot be unselected in order to ensure that at least the English Resource Files will be generated in the Resources Project.

* Chinese Simplified, Chinese Traditional, Spanish and French

Select these checkboxes to generate the appropriate language files in the Resources Project. For these optional languages, the keys will be generated with blank values. These blank values must be translated prior to deployment.

Click Back to return to the previous step, or click Next to proceed.

* 1. Generate Solution



Click Back to return to the previous step, or click Generate to generate the solution.

* 1. Set Login.aspx as the Start Page

In Sage 300 version 2022, a login page was added to the SDK Samples and generated partner solutions, which replaced hardcoded login credentials in the Global.asax.cs file.

In the Solution Explorer, Right-Click the **Login.aspx** file in the Web project and select **Set As Start Page**.

Failure to set this ASPX file for the solution will result in a failure when running the solution in Debug mode.

1. Examining the Solution

It is time to examine the solution and get familiar with the class and folder scaffolding.

* 1. Solution Explorer



Note the solution name supplied in the New Project dialog box.

Note the namespace of the projects as specified in the Sage 300 Solution Wizard dialog box.

* 1. Business Repository Project



The wizard has already created the required folders and a few classes that will be used by the Code Generation Wizard.

Note the presence of the Process and Reports folders. In the Code Generation Wizard, if a Code Type of Process or Reports is selected, the appropriate Mapper and Repository class will be generated in these subfolders as opposed to the root folders.

The TUMenuModuleHelper.cs file already has generated code for Menu Navigation.

The Security.cs file already has constants generated for the Import and Export constants for the generated Module ID.

* 1. Interfaces Project



The wizard has already created the required folders that will be used by the Code Generation Wizard.

Note the presence of the Process and Reports folders. In the Code Generation Wizard, if a Code Type of Process or Reports is selected, the appropriate Interface classes will be generated in these subfolders as opposed to the root folders.

* 1. Models Project



The wizard has already created the required folders that will be used by the Code Generation Wizard.

Note the presence of the Process and Reports folders. In the Code Generation Wizard, if a Code Type of Process or Reports is selected, the appropriate Enumerations, Fields, and Model classes will be generated in these subfolders as opposed to the root folders.

* 1. Resources Project



The wizard has already created the required folders that will be used by the Code Generation Wizard.

Note the presence of the Forms, Process, and Reports folders. In the Code Generation Wizard, if a Code Type of Process or Reports is selected, the appropriate Resx files will be generated in these subfolders as opposed to the Forms folders.

Note the generated Resx files:

* Any Resx file that is common to all screens or reports in a module is to be placed in the root folder.
* Only the English Resx file is marked as Public. All other Resx files are marked as No Code Generation.
* MenuResx.resx is the English resource.
* MenuResx.es.resx is the Spanish resource, if selected in the Resource Files Step.
* MenuResx.fr.resx is the French resource, if selected in the Resource Files Step.
* MenuResx.zh-Hans.resx is the Chinese Simplified resource, if selected in the Resource Files Step.
* MenuResx.zh-Hant.resx is the Chinese Traditional resource, if selected in the Resource Files Step.
  1. Services Project



The wizard has already created the required folders and a class that will be used by the Code Generation Wizard.

Note the presence of the Process and Reports folders. In the Code Generation Wizard, if a Code Type of Process or Reports is selected, the Service class will be generated in these subfolders as opposed to the root folders.

The TUBootstrapper.cs file already has generated code for Unity Dependency Injection which will be added to by the Code Generation Wizard.

* 1. Web Project



The wizard has already created the required folders, files and classes that will be used by the Code Generation Wizard.

Note the presence of the Finder, Process, and Reports folders. In the Code Generation Wizard, if a Code Type of Process or Reports is selected, the appropriate classes will be generated in these subfolders as opposed to the root folders. If selected in the Code Generation Wizard Options, the Finder will be created in the Finder folder.

The TUBootstrapper.cs, TUAreaRegistration.cs, TUMenuDetails.xml, and TUWebBootstrapper.cs files already have generated code for Unity Dependency Injection which will be added to by the Code Generation Wizard.