

# Developer Reference

## Zodiac.NET Survey Engine Toolkit Release 1.2

© 2009 - 2010 MentorLogic Ltd.  
All rights reserved.

MentorLogic is a leading software provider for software utilities, reusable components and libraries.

This document is part of the company products documentation series.

For more information or ordering of this product or others products, please contact:  
support@mentor-logic.com

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electrical, mechanical, photocopying, recording, scanning or otherwise except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher.

Designations used by companies to distinguish their products are often claimed as trademarks. In all instances where MentorLogic Ltd. is aware of a claim, the product names appear in initial Capital or ALL CAPITAL LETTERS.

Readers, however, should contact the appropriate companies for more complete information regarding trademarks and registration.

# Contents

Scope .....	5
Zodiac.NET Overview .....	5
1. Installation.....	5
1.1 Include References.....	5
1.2 Creating Folders .....	6
1.3 Using Zodiac.NET Web Controls .....	6
1.3.1 Survey Viewer Web Control.....	6
1.3.2 Survey Builder Web Control.....	9
1.3.3 Survey Response List Web Control.....	10
1.3.4 Survey Report Web Control.....	10
About MentorLogic .....	12



## Scope

This document was designed for the purpose of providing a reference for software engineers and developers wishing to develop and customize Zodiac.NET Survey Engine in their own projects and applications. It is recommended that the reader of this document have a reasonable grasp of Microsoft .NET technology, ASP.NET, XML and Client Scripting.

## Zodiac.NET Overview

Zodiac.NET is an ASP.NET web based survey and form engine component for Microsoft .NET written in pure managed C#. It helps .NET developers to integrate web-based surveys and questionnaires in ASP.NET projects using state of the art Microsoft ASP.NET. The engine provides several types of questions to fit the different needs of the survey makers.

The engine is customizable to provide better solutions and unified experience for your users. Using the web controls properties, events and styles, .NET developers are able to simply change, interact and integrate the engine with their existing code and applications.

Zodiac.NET combines XML and ASP.NET as core technologies for delivering its features. XML has already proved to be the right choice for building integrated components and libraries. In Zodiac.NET, the survey definition files and responses are fully represented as XML. Zodiac.NET doesn't use databases to store or retrieve the survey definition or responses which give easier setup experience with almost zero-footprint installation and setup.

## 1. Installation

Follow the following steps to use Zodiac.NET in .NET web projects. The steps are applicable for Visual Studio 2005/2008.

### 1.1 Include References

- i. Create a new ASP .NET Web Project.
- ii. Right click on the project in the Solution Explorer and click "Add Reference".
- iii. Browse the bin directory of Zodiac.NET downloaded software package and select the following assemblies:

MentorLogic.Common.dll  
MentorLogic.Engines.Zodiac.dll

- iv. Click OK. The two assemblies should be added to the bin directory of the web application.
- v. In case you have a license, then copy the license file to the bin directory of your web application.

## 1.2 Creating Folders

Create the following folders in your web project:

- **Surveys:** This folder will store the Surveys XML files that contain the definition files of your surveys.
- **Responses:** This folder will store the user responses to the survey. The responses will be saved as XML files.

Note: You can find a sample survey XML file in the following path:

```
<package_root>\Sample 1\ZodiacSample\surveys\sample.xml
```

## 1.3 Using Zodiac.NET Web Controls

Add the register tag at the top of your page to register the assembly.

```
<%@ Register Assembly="MentorLogic.Engines.Zodiac"  
Namespace="MentorLogic.Engines.Zodiac.Web" TagPrefix="Zodiac" %>
```

NOTE: You can register the controls in the application web.config so that you don't have to register it for each page in the web project. The following sample show you how to register the controls in the web.config file:

```
<system.web>  
  <pages>  
    <controls>  
      <add assembly="MentorLogic.Engines.Zodiac"  
        namespace="MentorLogic.Engines.Zodiac.Web"  
        tagPrefix="Zodiac"></add>  
    </controls>  
  </pages>  
</system.web>
```

### 1.3.1 Survey Viewer Web Control

To view and render the survey to the end users, add the survey viewer user control to your page. The following is a sample definition for the web control:

```
<Zodiac:SurveyViewerWebControl ID="ucSurveyWebControl"  
  CssClass="bluesky"  
  XmlFilePath="<<XML_File_Path>>"  
  OutputFolder="<<Responses_Folder_Path>>"  
  runat="server">  
</Zodiac:SurveyViewerWebControl>
```

The following is a list of the web control properties:

Property	Description
ID	The web control unique ID
XmlFilePath	The file path of the XML definition file.
Xml	The XML definition of the survey. Use this property to directly set the XML to the control without the need of using physical XML file.
OutputFolder	The folder path to save the user responses to the survey.
CssClass	The CSS class name to be applied to the survey viewer web control.
AuthType	The authentication method used to ensure that a single user cannot take the survey more than once. This can take values: None, Sessions, Cookies or All.
CurrentPageIndex	Gets the current active survey page index.
PagesCount	Gets the number of the survey pages.
NextButtonText	Gets or sets the Next button text. The default text is "Next".
PreviousButtonText	Gets or sets the Previous button text. The default text is "Previous".
FinishButtonText	Gets or sets the Finish button text. The default text is "Finish".
ResponsesCount	Gets or override the number of the survey responses. Use this property if you don't provide responses folder path to the control and you want the web control to check the survey quota to allow/prevent any more responses to the survey.
PreviewMode	Gets or sets if the viewer is in the preview mode or not. If true, the control will not collect any responses (i.e. no responses will be generated or saved). Use this property for adding preview functionality to your surveys.
MediaUploadPath	Gets or sets the path where the media files will be uploaded.
SkipStartPage	Force the web control to skip the first page of the survey when displaying it to the end user.
IsStartPage	Is the current active survey page is the first page.
IsFinishPage	Is the current active survey page is the finish/end page.
IsStartOrFinishPage	Is the current active survey page is the start or end page.
IsLastSurveyPageBeforeFinish	Is the current active survey page is the last end page.

The following is a list of the web control events:

Event	Description	Arguments
OnSurveyStart	This event is fired when the survey is started.	EventArgs: Standard event argument.
OnMoveNextClicked	This event is fired when the user clicks on the Next button.	SurveyPageArgs: This arguments contains information about the new survey page that result after moving next.
OnMovePreviousClicked	This event is fired when the user clicks on the Previous button.	SurveyPageArgs: This argument contains information about the page that was the source of moving previous.
OnResponseReceived	This event is fired when a user has finished the survey and his response is ready to	SurveyResponseArgs: Contains the user response XML representation and the response ID.

	be saved.	
OnSaveSurvey	This event is fired when the user click on the save button.	SurveyArgs: Contains the new saved survey data object and survey Xml

NOTE: It's not required to use XML physical file to display the survey in the web control. You can simply set the XML definition of the survey direct to the control without providing any file paths.

This is useful in case that you prefer to store the XML file in other storage medium such as databases.

You don't have also to store the response XML in a physical path. You can make use of *OnResponseRecieved* event to get the response XML and store it in your database.

To have a complete example on using the web controls without physical files, please check the "Sample 2 – Dynamic Data Loading" web project in the "Samples" folder in your package.

In the survey viewer web control, you need to add the following lines in the Page\_Load event of the web page:

Example:

```
if(!IsPostBack)
{
    this.ucViewer.DataBind();
}
```

This is required by the web control to start binding the data to the web control based on the set properties.

You can also set the properties in the server side and then bind the data for the web control.

Example:

```
if(!IsPostBack)
{
    this.ucViewer.XmlFilePath = "surveys/sample.xml";
    this.ucViewer.OutputFolder = "responses";
    this.ucViewer.DataBind();
}
```

### 1.3.2 Survey Builder Web Control

To provide a survey designer to your end users, add the survey builder user control to your web page. The following is a sample definition for the web control:

```
<Zodiac:SurveyBuilderWebControl
    ID="ucSurveyBuilderWebControl"
    runat="server"
    XmlFilePath="<<XML_File_Path>>"
    CssClass="bluesky"
    PreviewButtonVisible="false"
    CancelButtonVisible="true"
    CancelButtonUrl="http://www.mentor-logic.com"
    CancelButtonText="Cancel" />
```

The following is a list of the available web control properties:

Property	Description
ID	The web control unique ID
XmlFilePath	Gets or sets a physical XML file path of the survey.
Xml	Gets or sets the survey XML definition direct to the control without specifying a physical file path.
CssClass	The CSS class of the web control.
PreviewButtonUrl	The URL of the survey preview page. This page will have the survey viewer web control.
PreviewButtonText	The text to be displayed in the Preview button.
PreviewButtonVisible	Specify if the preview button should be displayed or not.
CancelButtonUrl	The URL of the custom cancel page. This can be a relative URL or a full URL.
CancelButtonText	The text to be displayed in the Cancel button.
CancelButtonVisible	Specify if the Cancel button should be displayed or not.
ImagesExtensions	Gets or sets the allowed image files extensions. Ex: png jpeg bmp
AudioExtensions	Gets or sets the allowed audio file extensions. Ex. mp3 wav 3gp
VideoExtensions	Gets or sets the allowed video extensions. Ex. avi mpeg mov
EnableMedia	Gets or sets if uploading media is enabled in the web control or not.
EnableBranching	Gets or sets if defining branching rules is enabled in the web control or not.
MaxChoicesCount	Gets or sets the maximum allowed number of choices to be added per question.
MaxPagesCount	Gets or sets the maximum allowed number of pages to be added per survey.
MediaUploadPath	Gets or sets the path where that media files will be uploaded.

The following is a list of the web control events:

Event	Description	Arguments
OnSaveSurvey	This event is fired when the survey is saved.	SurveyEventArgs: Survey event argument which contains the Survey object and the XML representation of the survey.

NOTE: It's not required to assign XML physical file path to the web control. You can simply set the XML definition of the survey direct to the control without providing any file paths.

This is useful in case that you prefer to store the XML file in a database or external storage medium.

To have a complete example on using the web controls without physical files, please check the "Sample 2" web project in the "Samples" folder in your package.

### 1.3.3 Survey Response List Web Control

To view the survey responses list web control, add the survey response list web control to your page. The following is a sample definition for the web control:

```
<Zodiac:SurveyResponseListWebControl
    ID="ucSurveyResponseListWebControl"
    runat="server"
    CssClass="bluesky"
    SurveyXmlFilePath="surveys\\sample.xml"
    ResponsesFolderPath="Responses"
    PageSize="10" />
```

The following is a list of the control properties:

Property	Description
ID	The web control unique ID
CssClass	The CSS class name to be applied to the web control.
ResponsesFolderPath	The folder path to retrieve the responses XML files.
Responses	Gets or sets the Survey Responses. Use this property if you don't want to read the responses from XML file and instead you need to handle it manually.
ResponseLink	Gets or sets the response link format. The format should contain one parameter to replace it with response ID. Example: response.aspx?id={0} Use this property if you want to provide custom page to show the response details of each response.
SurveyXmlFilePath	Gets or sets the survey XML file path.
PageSize	Gets or sets the number of records per page. Ignore this property to disable the paging in the web control.

### 1.3.4 Survey Report Web Control

To view the survey statistics, add the survey report web control to your web page. The following is a sample definition for the web control:

```
<Zodiac:ReportSurveyStatisticsServerControl
    ID="ucReport"
    runat="server"
    CssClass="bluesky"
    SurveyXmlFilePath="surveys\\sample.xml"
```

```
ResponsesFolderPath="Responses"  
QuestionID="2" />
```

The following is a list of the control properties:

Property	Description
ID	The web control unique ID.
CssClass	The CSS class name to be applied to the web control.
SurveyXml FilePath	Gets or sets the survey XML definition file path.
ResponsesFolderPath	Gets or sets the survey responses folder path.
QuestionID	Gets or sets the question ID. If set, the web control will show the statistics only for this question.

## About MentorLogic

MentorLogic is a leading vendor of collaborative components for Microsoft .NET technologies. MentorLogic helps customers build applications with unparalleled richness, responsiveness and interactivity. Created with passion, MentorLogic products help thousands of developers every day to be more productive and deliver reliable applications under budget and on time.

The vision is to enhance the reusability in the software development industry and provide collaborative, integrated and productive solutions and components for software professionals. We believe that the reusability and the flexibility in integration is one of the critical challenges today. MentorLogic target is to provide the best state-of-the-art solutions which can help achieving better software and more productive solutions.



Contact:

[support@mentor-logic.com](mailto:support@mentor-logic.com)

[sales@mentor-logic.com](mailto:sales@mentor-logic.com)

Tel. (+202)(010-3786460)

<http://www.mentor-logic.com>