OWL C# Software Generation and Maintenance Document

Author Chris Barrieau

**Contents**

[Introduction](#h.30j0zll)

[Setup](#h.1fob9te)

[Building OWLWebsite](#h.3znysh7)

[Getting files](#h.2et92p0)

[Building in Visual Studio](#h.tyjcwt)

[Confirming URL for OWLTest](#h.3dy6vkm)

[Confirming Database connection](#h.1t3h5sf)

[Building OWLTest](#h.4d34og8)

[Testing](#h.2s8eyo1)

# Introduction

This document outlines the process to create and generate the OWLWebsite component and OWLTest test component as they currently exist as of 10/07/2015. If there are issues, please contact author Chris Barrieau through email [cbarrieau@gmail.com](mailto:cbarrieau@gmail.com), or through Slack on the Team 4 group.

# Setup

If the software doesn’t build, the following changes may be required. The software may require two changes in order to be able to run properly – Powershell version 3 and JSON.net. Powershell is necessary for the OWLTest application to work properly (due to a bug in Visual Studio 2015), and JSON.net is necessary for handling JSON objects in OWLWebsite.

Powershell can be installed by following the instructions here:

<https://technet.microsoft.com/en-us/library/hh847837.aspx#BKMK_InstallingOnWindows7andWindowsServer2008R2>

JSON.net can be installed by following the instructions here:

<https://www.nuget.org/packages/newtonsoft.json/>

# Building OWLWebsite

## Getting files

1. Connect to the Github repository and position it at c:\OWL
2. There should be two folders in c:\OWL\swegroup-4\ : OWLTest and OWLWebsite

## Building in Visual Studio

1. In the Visual Studio main page, go to File and select open->website.
2. Navigate to c:\OWL\swegroup-4\OWLwebsite
3. Select View in the main page and select Output. This will put the Output in the lower window.
4. Go to Build in the menu bar and select Rebuild Solution
5. In the ErrorList you should see 0 errors.
6. Output should say Rebuild All: 1 succeeded, 0 failed, 0 skipped.
7. Press the green arrow in the toolbar to launch. You may get a warning about debug, just select continue. The browser should launch with a window displaying the options to search in the database in JSON.

### Confirming URL for OWLTest

1. Select OWLWebsite in the Solution explorer window (May need to go to View->Solution Explorer if it is not visible)
2. Click on OWLWebsite(x)
3. Check the URL. The OWLTest application will need to be updated to that value to connect.

### Confirming Database connection

1. Go to App\_Code and select “Service.cs”
2. Update string Connect to be the value for your database, if necessary (you can try the default to start, it should be available in SQL Server).
3. Do a build.

# Building OWLTest

OWLTest is the client which tests the OWLWebsite.

1. Go to File…Open Project and navigate to c:\OWL\swegroup-4\TestOWL\TestOWL.sln
2. Click on Program.cs and update string url in GenerateGetRequest()
3. Click on Build…Rebuild
4. You should see Build: 1 succeeded or up-to-date, 0 failed, 0 skipped in the Console window
5. If the green arrow it is grayed out, you will need to click on TestOWL in solution explorer, select properties, and go to profile. Make sure Launch
6. Click the Green Arrow
7. The Console app should launch. It will error out as there is no connection.

# Testing

To test, you will need to launch two visual Studio 2015 environments. NOTE: you need to open the OWLTest project in Visual Studio 2015 first or the program will not launch. This appears to be a bug in Visual Studio 2015.

1. Launch OWLWebsite following the instructions above. Wait until the web page is up.
2. Launch OWLTest following the instructions above
3. When OWLTest launches, You should see an output window showing the tests. You should see a message showing errors and successful tests. There should be 0 errors.

For further testing of the full system, see the Test document.

# IIS

For setup of the IIS portion, front-end and etc, see the IIS document.