

Documentation

Project Description

The [EZ Wild Apricot Web Designer](#) (aka WATM) is a JavaScript library that uses the [jQuery](#) library included with every [WildApricot](#) website. EZ Wild Apricot Web Designer will help any administrator manage and replace nearly every piece of text in Wild Apricot. You can make a variety of changes such as global search and replace of text, modifying form labels and buttons, changing properties on any Cascading Style Sheets (CSS) class or ID including hiding text, labels or buttons. You can also use EZ Wild Apricot Web Designer to change specific styles of your WildApricot theme. EZ Wild Apricot Web Designer uses a CSV-formatted configuration file to store all the changes in a well documented, easy to maintain way.

The configuration file must be named `wildapricot-textmanager-config.csv` in order to be used by EZ Wild Apricot Web Designer.

Browser Based CSS Inspector

After installation just add `?dev` after the public URL to go into "development" mode. You can then click on any element on the page to find the CSS class and/or ID that can be used to identify the element.

Demo: Click <https://newpathconsulting.wildapricot.org/?dev> to go into dev mode. Then click on any element to inspect the CSS class or ID.

Using Dev Tools to inspect

[Using Copy Selector and Copy Styles to quickly find a CSS Selector](#)

[Learning more about about Cascading Style Sheets \(CSS\).](#)

[How to use Chrome Developer Tools](#)

[How to use Chrome Developer Tools to find CSS classes or IDs](#)

Change History

0.1 - Initial Release 11/19/18

0.2 - Added support for many more system flags, added French translation into CSV configuration file, changed testing button toggle label 11/27/18

0.3 - Added support for adding a style, added several more system labels, added support for limiting the language toggle display "flicker" 12/5/18

0.4 - Added support for effective primary menu design changes, added support to override any CSS class or ID 12/13/18

0.5 - Added support for mouseover, mouseout (hover) and :before and :after pseudo-elements 1/15/19

0.6 - Added support for SCSS-style variables 1/24/19

0.7 - Now managed in GitHub, fixed issues related to Internet Explorer 11 support in the CSV parser and in the WATM library 2/23/19

0.71 - The Wild Apricot Text Manger config file is not cached while in testing mode 2/6/20

0.8 - WATM will now automatically enable when it goes into public view in Wild Apricot, new attribute function has been added 6/15/20

0.81 - fixed code typo in wildapricot-textmanager.js due to reformatting 6/29/20

0.82 - added display of WATM version on console log 9/8/20

0.9 - added CSS class/id inspector invoked with ?dev in URL, added more error/information logging, strip leading/trailing space in "function" field of configuration file 11/20/20

0.91 - Fixed replace_delay function that works for form dropdowns, added "-n" suffix to the replace-delay function. For example, replace-delay-3 will delay for 3 seconds 11/23/20

0.93 - support for switching languages using embedded WildApricot "widgets" in 3rd party content management system, added a standard French translation file 2/15/22

Installation

Script Setup

1. In Admin View of Web Apricot, browse to Settings -> Site -> Global JavaScript. More details on inserting JavaScript into Wild Apricot are available in [Wild Apricot Documentation](#).
2. Copy and paste the following lines to the Global JavaScript.

NOTE: this code snippet assumes you have uploaded all files into the folder /resources/Theme/WildApricotTextManager

```
<!-- jQuery-CSV Source: https://github.com/evanplaice/jquery-csv -->
<script src="/resources/Theme/WildApricotTextManager/jquery.csv-0.8.9-mod.js">
</script>
<script src="/resources/Theme/WildApricotTextManager/wildapricot-textmanager.js">
</script>
<script>
var textManagerMultilingualMode = true;
var primaryLanguageButtonName = "English";
var alterativeLanguageButtonName = "Français";
var languageButtonHtmlID = "languageButton";
var alterativeLanguageClassName = ".french";
var primaryLanguageClassName = ".english";

$(window).bind("load", function() {$('#textmanager_overlay').css('display',
'none'); }); // Fail-safe to remove white overlay
</script>
```

3. To deactivate the multilingual mode edit this JavaScript variable declaration as follows:

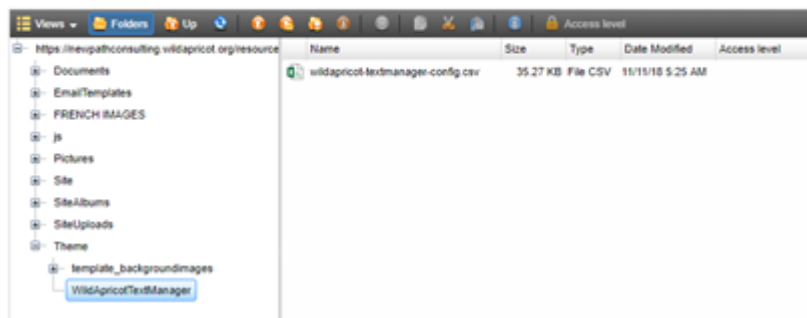
```
var textManagerMultilingualMode = false;
```

Files Setup

1. In Admin view of Wild Apricot, browse to [Website -> Files](#).
2. Create a new folder named `WildApricotTextManager` under the Theme folder.

Files

You are currently using 1 MB (0%).
100 MB limit for each file upload.

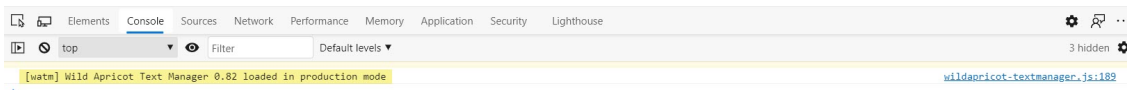


3. Upload [jquery.csv-0.8.9-mod.js](#) to the new folder.
4. Upload [wildapricot-textmanager.js](#) to the new folder.
5. Upload [wildapricot-textmanager-config.csv](#) to the new folder.

Testing Installation

To test whether EZ Wild Apricot Web Designer has been installed properly:

1. Click Public View to browse the Wild Apricot website.
2. Append `?dev` and hit return to reload the page. You should see the CSS class/id inspector.
3. Right click on the page and select 'Inspect'. Invoking Developer Tools may be different depending on the browser you are using.
4. Navigate to Console and look for the message in yellow below. The version number should appear. If you do not see this message, or you see errors EZ Wild Apricot Web Designer may not have been properly installed.



Overlay Setup (Optional)

Normally, when the EZ Wild Apricot Web Designer is enabled, the original text is displayed on the page load and then replaced within one second. This "flicker" can be hidden with the HTML and CSS below. EZ Wild Apricot Web Designer will automatically remove the overlay once the script has been completed, reducing the flicker.

Open the each Wild Apricot Page Template in use, add a [Custom HTML](#) gadget anywhere in the template. Edit the code of the Custom HTML gadget and include this HTML snippet:

```
<div id="textmanager_overlay"></div>
```

You must also add these lines to the Wild Apricot global CSS in the [Website -> CSS](#) menu:

```
#textmanager_overlay {  
    position: fixed;  
    width: 100%;  
    height: 100%;  
    top: 0;  
    left: 0;  
    right: 0;  
    bottom: 0;  
    background-color: white;  
    z-index: 2;  
}
```

Multilingual Setup (Optional)

Standard WildApricot translation file

EZ Wild Apricot Web Designer can also be used to enable 2 languages for WildApricot websites. EZ Wild Apricot Web Designer does *not* require you to create any additional site pages or page templates. Translations can be applied for any Wild Apricot module without the need to create multiple versions of a page, event, donation form etc. A French translation file is included as sample CSV configuration file, named `FR-wildapricot-textmanager-config.csv`. Remember to rename this file to `wildapricot-textmanager-config.csv` before using it. This file can be edited in Microsoft Excel or another CSV editor. When saving the file, make sure to save with the UTF-8 encoding format to preserve any accented or multi-byte characters.

Moving the language switcher button

To move the Language button from the default location:

1. In Admin View, browse to each page template in use.
2. Click Edit, then Gadgets, then drag and drop Custom HTML gadget to where you would like to place the button.
3. Click Edit code and delete the default "Insert your HTML code here".
4. Add a single space into the gadget. It will show empty but there will be a "placeholder" space.
5. Click Settings button for the new gadget. On the left panel, show Advanced and then change the HTML ID to `languageButton`.
6. Save the template.
7. Repeat this for each page template.
8. (Optional) Customize the look of the button by applying custom CSS to the ID `#languageButton` in the configuration file on the Custom CSS section.

Add the translated content

The below is an example of adding translated content to a Wild Apricot "Contact" page.

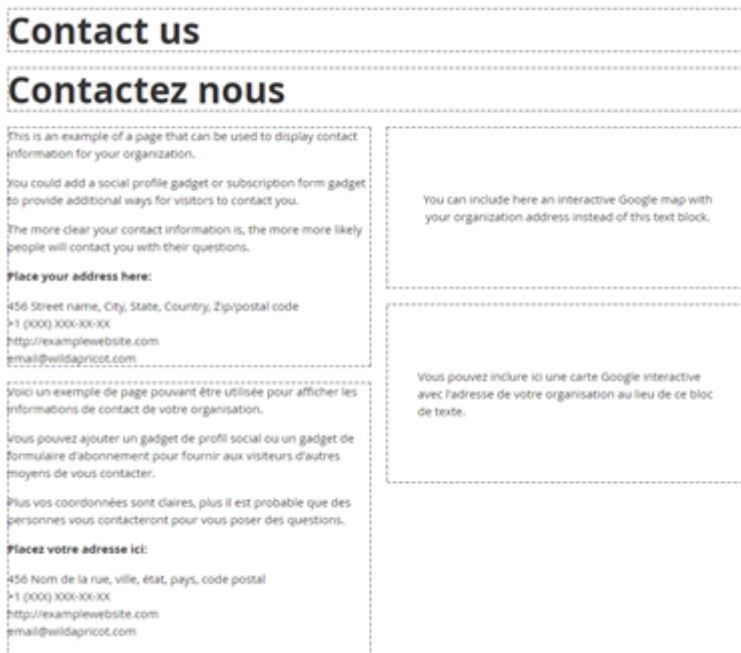
1. Start with your Contact page. In Admin Mode, browse to Website -> Site pages -> Contact and click Edit. You may have a different page on your site, and you can start with any page.
2. Click on Gadgets and drag a new Content gadget below the existing Contact us heading. If your second language is French, then type in `Contactez nous` into this new space. Adjust font sizes and padding to match the original content.



3. Show [Advanced settings section](#) on the left settings panel and add `french` to the CSS class text box.
4. Click on the original English Contact us heading and add `english` to the CSS class text box.



5. Repeat for other content blocks you wish to localize and Save page. The result should look similar to below:



Note that if EZ Wild Apricot Web Designer is enabled, one of the two languages will be hidden based on the state of the language, even in the Edit mode. Please make sure `textManagerProductionMode = false` is set in Global JavaScript and then click the button to disable EZ Wild Apricot Web Designer to see both sets of languages at once.

Support for embedded WildApricot widgets on 3rd party Content Management Systems

If you are using the WildApricot widget code using iframes, use the `?secondLanguage` argument at the end of widget URL to trigger translations to be loaded.

1. Alter the WildApricot widget code to support the `?secondLanguage` widget.
2. Embed the widget on the translated page on the 3rd party CMS. This works with systems like Drupal, Squarespace and WordPress.

Example code:

For ENGLISH pages use the default language widget:

```
<iframe width='750px' height='400px' frameborder='no'
src='https://mysite.wildapricot.org/widget/join-us'
onload='tryToEnableWACookies("https://mysite.wildapricot.org");' ></iframe>
<script type="text/javascript" language="javascript"
src="https://mysite.wildapricot.org/Common/EnableCookies.js" ></script>
```

For FRENCH pages add the `?secondLanguage` widget:

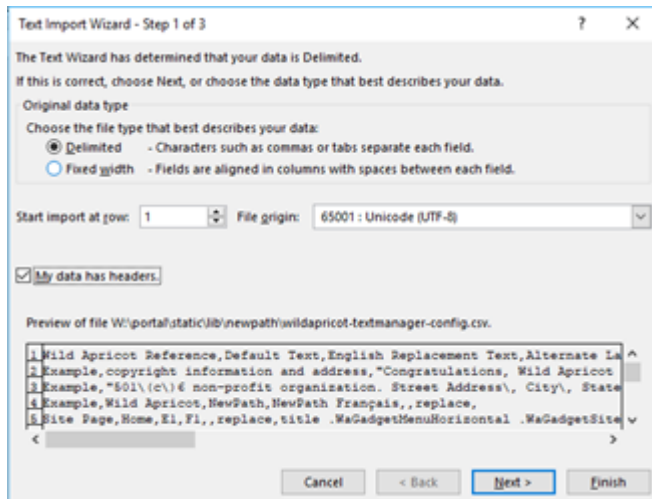
```
<iframe width='750px' height='400px' frameborder='no'
src='https://mysite.wildapricot.org/widget/join-us/?secondLanguage'
onload='tryToEnableWACookies("https://mysite.wildapricot.org");' ></iframe>
<script type="text/javascript" language="javascript"
src="https://mysite.wildapricot.org/Common/EnableCookies.js" ></script>
```

EZ Wild Apricot Web Designer Configuration File

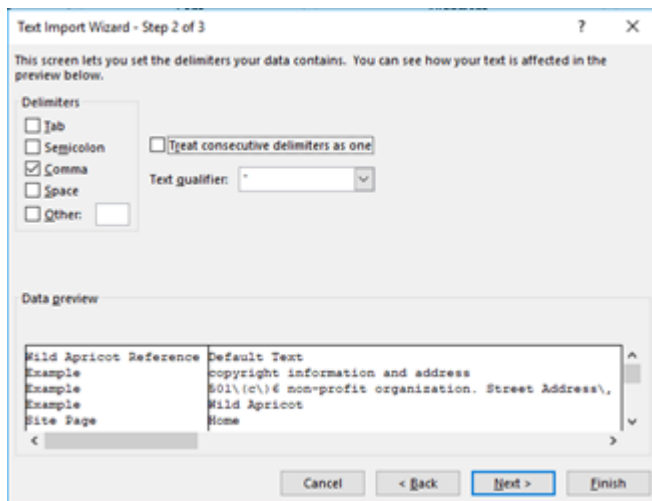
To maintain all the EZ Wild Apricot Web Designer text changes, translations and styling changes, you must edit and maintain the CSV configuration file `wildapricot-textmanager-config.csv` using Excel or another commas separated file (CSV) editor.

Instructions for editing the file in Excel

1. Open the `wildapricot-textmanager-config.csv` file Choose Delimited, check My data has headers and click Next



2. Check Comma for the Delimiters option, Click Finish



Saving the configuration file

When saving the file, use the CSV UTF-8 (Comma delimited) (.csv). **Other formats will break the configuration file!*

The configuration file may be cached in the web browser between changes while in production mode. If you don't see your configuration file changes reflected on the

website, you will need to [force-refresh your browser](#) to reload the latest configuration file.

Fields in the configuration file

Note: Inserting or moving columns will break the configuration file!

Wild Apricot Reference : The field can be used to group configuration lines together in the configuration file. It is for reference use only.

Default Text : The text that will be searched for replacement in a Wild Apricot gadget or label. This is a case-sensitive and space sensitive field.

English Replacement Text : If you wish to replace the text with another value, place the new text in this field. This is an optional field.

Alternative Language Text : If you wish to translate text with another language, place the translated language in this field. This field is an optional field.

Notes : Notes can be added here for reference.

Function : This field is used to manage text. If a function is used the **Default Text**, **English Replacement Text** and **Alternative Language Text** fields will be used. It is an optional field.

Query : This is the targeted CSS ID, CSS Path or CSS Style to target for changes. It is a required field.

Style : Contains the CSS style properties and value to apply to the **Query** element

Note: You can leave the **Function** field empty if you wish to apply CSS using the **Query** and **Style** fields.

Text Management Functions for **Function** field:

The functions below change the text with the contents of **English Replacement Text** or the **Alternative Language Text** column. Any text in the **Default Text** column is ignored.

- **text** - sets text of the element selected by the **Query** column.
- **hide** - hides the element selected by the **Query** column.
- **button** - sets value of the button selected by the **Query** column.
- **placeholder** - sets placeholder attribute. Only used for search boxes.
- **delay** - page will pause one second before replacing text. This can be used on text elements that are written with JavaScript after page load (eg the member directory gadget).

The following functions expect to use **Default Text** column as the search criteria for replacement.

- **replace** - Searches for **Default Text** column and replaces this sub-string in any element. If **Query** column is blank, the entire page is searched.
- **replace_element** - Searches text in **Default Text** column and replaces the text of the entire element. If **Query** column is blank, the entire page is searched.
- **replace_delay** - Replaces string after one second delay. Add a **_n** suffix after **replace_delay** to increase delay beyond the default 1 second. **replace_delay_3** will delay 3 seconds.
- **attribute** - Replaces string contained in any HTML **attribute** tag

There is a subtle difference between the `text`, `replace` / `replace_delay` and `replace_element` functions.

Function `text` ignores the `Default Text` value and simply replaces the text inside the element that matches the `Query`. It is *not* a search/replace function.

Example:

If the ID is `#myelement` and you wish to replace the content with "Wild Apricot is awesome!" create the following configuration row:

- Function is `text`
- Query is `#myelement` and
- English Replacement Text is `Wild Apricot is awesome!`

Functions `replace`, `replace_delay` and `replace_element` do a search & replace based on the `Default Text` field.

Example:

If the ID is `#myelement` and you wish to find the text `Personify` anywhere in an element and replace it with `Wild Apricot` create the following configuration row:

- Function is `replace`
- Query is `#myelement` and
- Default Text is `Personify`. Make sure case is correct as the search is case-sensitive.
- English Replacement Text is `Wild Apricot`

If the original text is `Personify is awesome!` the configuration above will replace the string with `Wild Apricot is awesome!`.

Example:

If the ID is `#myelement` and you wish to find the text `Personify` anywhere inside an element and replace *the whole element* with `Wild Apricot is great!` create the following configuration row:

- Function is `replace_element`
- Query is `#myelement` and
- Default Text is `Personify`. Make sure case is correct as the search is case-sensitive.
- English Replacement Text is `Wild Apricot is great!`

If the original text is `Personify is awesome!` the configuration above will replace the element with `Wild Apricot is awesome!`. It simply searches for the `Default Text` and when found uses the `English Replacement Text`.

The `replace_delay` function waits a preset number of seconds before making a replacement to allow any other Javascript to execute before making element changes.

In the above examples, the `English Replacement Text` can be replaced with `Alternative Language Text` if multilingual mode is turned on.

Utility Functions

- `inactive` - disables the configuration row. This function can be used to save a configuration, but not activate it for use.

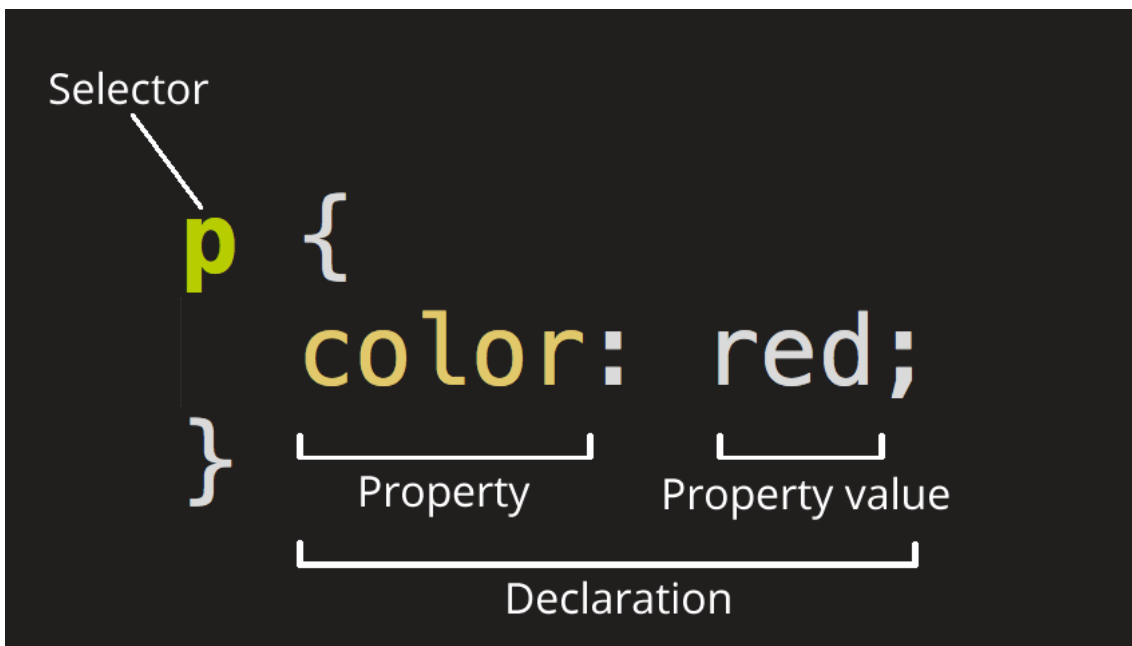
CSS-only functions:

- `mouseover` - Sets CSS inside an event handler when hovering over an element.
- `mouseout` - Sets CSS inside an event handler when no longer hovering over element.

Query: The CSS selector to target for the change. A selector can include HTML elements, classes and IDs as well as a CSS path (eg `div.class > td > #myID`). [Using Copy Selector and Copy Styles to quickly find a CSS Selector](#)

Style: The CSS ruleset (properties & values) to apply to selector in the **Query** column. Optional.

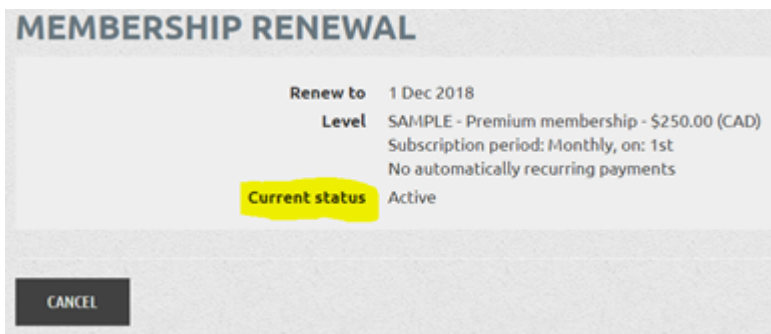
Each property and value should be enclosed with a double quote (") (eg `"color":"red";`)



Example:

Change the label "Current status" on Membership Renewal system gadget:

Here is what the standard Membership Renewal system gadget looks like:



- ☐ Open wildapricot-textmanager-config.csv in your text editor or Excel.
- ☐ Search the file for "Current status". Verify **Wild Apricot Reference** column matches the name of the gadget we're looking at.
- ☐ Type in new text `My New Text!` in **English Replacement Text** column.

116	WaGadgetContactProfile	Social network login			text
117	WaGadgetMembershipRenewal	Renew to			text
118	WaGadgetMembershipRenewal	Level			text
119	WaGadgetMembershipRenewal	Current status	My New Text!		text
120	WaGadgetMembershipRenewal	Cancel			button
121	WaGadgetMembershipRenewal	Update and next			button
122	WaGadgetMembershipRenewal	Your profile			text

- ☐ (Optional) Type translated language text in **Alternative Language Text**.
- ☐ Save as CSV UTF-8 (Comma delimited) (*.csv)

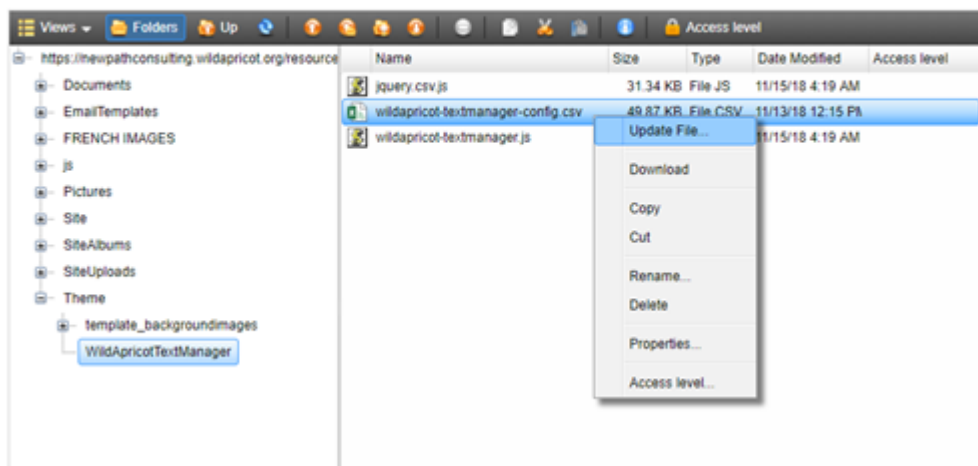
File name:

Save as type:

- ☐ Upload/Update .csv file - Note: [WebDAV is available to update](#) the WATM configuration file to speed up the process of updating the file between changes.

Files

You are currently using 1 MB (0%).
100 MB limit for each file upload.



- ☐ Load the membership renewal page and use the toggle buttons to test.

Here is what the new Membership Renewal system gadget should look like:

MEMBERSHIP RENEWAL

Renew to 1 Dec 2018

Level SAMPLE - Premium membership - \$250.00 (CAD)

Subscription period: Monthly, on: 1st

No automatically recurring payments

My New Text! Active

System Requirements

EZ Wild Apricot Web Designer must be used with a free or paid WildApricot account. EZ Wild Apricot Web Designer is supported on the latest versions of Chrome, Safari, Firefox and Edge. Older browsers like Internet Explorer on Windows are supported "best effort," without formal testing or 100% compatibility.

Further Examples

Hide an element:

Default Text: Type the 6 characters you see in the picture

Function: hide

Query: .captchaGuiding

Replace title of page on all menus:

Default Text: Directory

English Replacement Text: Members Directory

Function: replace

Query: title .WaGadgetMenuHorizontal .WaGadgetMenuVertical .WaGadgetSiteMap

.WaGadgetNavigationLinks .WaGadgetBreadcrumbs

Replace title of page on only main menu:

Default Text: Directory

English Replacement Text: Members Directory

Function: replace

Query: .WaGadgetMenuHorizontal

Replace text of any element of any page:

Default Text: 501\(\c\)6 non-profit organization

English Replacement Text: My Organization Name

Function: replace_element

(Note: Use backslash to escape special characters such as commas, parentheses, brackets.)

Set CSS of the H2 CSS selector:

```
Query: h2
Style: { "color" : "red", "font-style" : " italic" }
```

Set CSS of :before or :after (Note the use of standard CSS formatting):

```
Query: .WaGadgetCustomMenu.menuStyleNone .orientationVertical li:before
Style: { content: "|"; color: #5e5e5e }
```

Set CSS :hover of a button (Note, :hover is not needed in Query):

```
Query: .functionalButton
Style: { "color": "white", " background": "#a80532" }
```

Replace text "Purchase Organ Music" in the HTML title attribute

```
Default text: Purchase Organ Music
English Replacement Text: test
Function: attribute
Query: title
```