Documentation

Project Description

[EZ WildApricot Web Designer 2](#) (aka WATM), has been redesigned from scratch. The product has been rewritten in JavaScript (ECMAScript 6) and no longer needs jQuery to operate.

EZ WildApricot Web Designer will help any administrator change, replace or translate e nearly every piece of hard-coded and configurable text in WildApricot. It also contains several "macros" that can add new content gadgets and content styling features for your WildApricot website.

You can make a variety of single page, whole site, and WildApricot content changes such as:

- search and replace of a particular text string
- modify field labels on forms, button labels
- change hard-coded text, errors, and warnings
- hide text on any user interface items
- allow your site to be offered in an unlimited number of languages

EZ WildApricot Web Designer module can be used to make WildApricot sites available in two or more languages by dynamically replacing content gadgets and hard-coded strings with a translations. The translations are all stored in CSV files in WildApricot File Managers and can be edited from a visual inspector and editor as well.
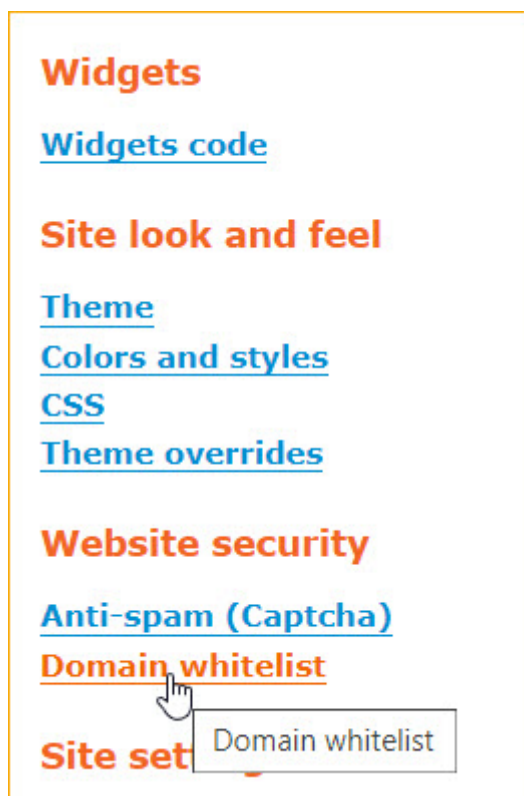

Installation

Files Setup

1. Follow these [instructions to connect to your WildApricot file folder](#) or go to the Website -> Files manager.
2. Unzip the downloaded zip file and upload the folder `WildApricotTextManager` into /Theme/WildApricotTextManager.
3. If you are setting up a multilingual site:

- Add a Content Gadget to your page template in the location you would like the language toggle to appear. Set the ID of this gadget to `language_switch`
- Create a separate CSV for each language you need in the `translations` folder.

The configuration and translation files must be saved as a CSV file format in UTF-8 format.

Domain whitelisting

WildApricot has implemented a [Content Security Policy (CSP)](#) to provide additional protection against data theft, site defacement, malware, and more. CSP works by detecting if custom code is being called from an external URL and then comparing those URLs against a whitelist of verified and approved websites. If the external URL being used is not on the whitelist, the code is then blocked from running. EZ WildApricot Web Designer requires access to certain external URLs in order to function correctly, so these URLs will need to be added to the whitelist. To view and manage your

whitelist, navigate to the "Settings" screen from the Website section of the WildApricot backend.

Adding EZ WildApricot Web Designer URLs to Custom Whitelisted Domains

Under the section "Custom whitelisted domains", you must add manually URLs used by EZ WildApricot Webs Designer. This is what the custom domain whitelist will look like once you add all core and add-on required domains.

The following is a list of all the external URLs currently being implemented by EZ WildApricot Web Designer and can be added manually:

Core Plugin

- hook.us1.make.com

EZ Add-Ons

- *.fontawesome.com

Script Setup

1. In WildApricot Admin, navigate to Settings -> Site -> Global JavaScript. More details on inserting JavaScript into WildApricot are available in the [WildApricot 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`

```html
<script src="/resources/Theme/WildApricotTextManager/wildapricot-textmanager.js">
</script>
<script>
  // Enter your license key if you have one - omit to use a trial that uses the first
10 lines of a config file
  license_key = "";
</script>
```

3. Enter your license key in the quotes `""` to apply your license.

NOTE: to use the development license key database add this to the `<script>` tag:

```
checkCode = "4suuck1up58qja9qfcqyosyhni63jwsn";
```

Multilingual Script Setup

If you want to configure multi-lingual support use this install script instead:

```html
<script src="/resources/Theme/WildApricotTextManager/wildapricot-textmanager.js">
</script>
<script>
  // Enter your license key if you have one - omit to use a trial that uses the first
10 lines of a config file
  license_key = "";

  /*
   Array of available languages
   Can omit if site is not multilingual
   Use "watm_language_name" to assign the language name displayed to the user
   Use "watm_language_className" to assign the language slug used on your site
  */
  watm_language_name[0] = "English"; // Default language
  watm_language_className[0] = "english"; // watm_language_csv_file[0] will default to
english.csv
  watm_language_name[1] = "Français";
  watm_language_className[1] = "french"; // watm_language_csv_file[1] will default to
french.csv
  watm_language_name[2] = "Español";
  watm_language_className[2] = "spanish"; // watm_language_csv_file[2] will default to
```

```
spanish.csv
  watm_language_name[3] = "□□□";
  watm_language_className[3] = "japanese"; // watm_language_csv_file[3] will default
to japanese.csv
</script>
```

Recommended Language Class Names

This is the list of recommended class names for each language. If you use these class names in the watm_language_className assignments in the script above, the browser's default language will automatically switch the website into the language selected by the user.

```
arabic
bengali
bulgarian
chinese
croatian
czech
danish
dutch
english
estonian
filipino
finnish
french
german
hebrew
hindi
hungarian
indonesian
italian
japanese
korean
latvian
lithuanian
macedonian
malay
norwegian
persian
polish
portuguese
romanian
russian
serbian
slovak
slovenian
spanish
swedish
thai
turkish
ukrainian
vietnamese
```

Extra Install Script Options

Enablic the Inspector & Editor without Being Logged in as an Admin

By default, the Inspector & Editor can be used only when logged in as an administrator to the WildApricot Admin view. If you'd like to be able to view the Inspector & Editor without being logged in to the Admin view, add this line in the install script

```
enable_public_editor = true;
```

To save any changes you make while in this mode, you will need to provide an administrative username/password during the first save.

Hiding the EZ WildApricot Web Designer Icon

By default the EZ WildApricot Web Desginer icon appears in the bottom left corner of all pages even if you are not logged in as an admin. In this mode, clicking on the icon will *not* bring up the Inspector.

To hide the EZ WildApricot Web Designer icon add this line to the install script:

```
hideWATMIcon = true;
```

Installing EZ Add-ons

To install an EZ add-on, place the add-on file into the `ez-addons` folder. Next, add the following code to your JavaScript configuration:

```
ez_addons = [];
```

Within the square brackets, add type the name of the add-ons within quotes. Separate multiple add-ons with a comma, with each add-on name within their own quotation marks.

Switching and Maintaining Languages

Using the Language Toggle

EZ WildApricot Web Designer provides an easy to install toggle for switching between the various languages on your site. To add it to your site, simply add a Content Gadget to your page template in the location you would like the toggle to appear. Set the ID of this gadget to `language_switch`. This gadget will now be replaced with the language toggle when viewed from the frontend.

Adding translations to content gadgets

There are 2 categories of text that can be translated in WildApricot:

- text and images placed on pages or page templates using WildApricot's Content Gadget
- hard coded text that appears in other WildApricot dynamic gadgets such as the name of pages in a menu bar, membership applications and membership directories.

EZ WildApricot Web Designer can translate both types of content.

To translate a content gadget you simple must add the language class name (eg "english" or "french") to the class settings of the gadget. The assigned class(es) are found in the "Advanced" settings on every content gadget added to a page or page template.

Translating the hard coded text in WildApricot dynamic gadgets requires identifying the CSS ID or CSS Path and targeting each element with an appropriate translation. There are hundreds of pieces of hard coded text throughout a WildApricot site. Thankfully NewPath has identified the hard coded elements and offers a translation file for hard coded elements in multiple languages with [WildApricot Translation Support Service & Training](#).

Using the Language Switch Hook

You are not limited to using the EZ WildApricot Web Designer Language Toggle. You can create your own buttons, links, or menu items for switching the language. To switch language, you would link to `?watm-<language>` where `<language>` is the class name for the language you wish to use from the script setup. Note, there is no slash in this link - you are linking to the current page, and appending the language hook to it.

EXAMPLE:

```
<a href="?watm-french">Cette page est également disponible en français</a>
```

EZ WildApricot Inspector & Editor

Using the Inspector & Editor, you can use to view the styling and targeting information of the various elements on your website, as well as make modifications to your EZ WildApricot Web Designer configuration files. This information is necessary when updating your configuration and translation files.

When logged in as a WildApricot administrator, an icon will appear at the bottom-left of the browser window. You can move the location of the icon by right clicking on it to move it to the bottom-right of the browser window, and right clicking again will move it back to the bottom-left.

To launch the EZ WildApricot inspector and editor click on the icon. While in Editor mode, you will not be able to follow any links on the web page. Instead you will be able to hover over every element on the web page to inspect it for modification.

When you hover over an element a red outline will appear around the element ready for inspection. Click on the element to open the inspector.

Click the `Exit Editor` button in the inspector to return to normal browsing.

Inspector

The top half of the panel that appears is the Inspector. Clicking on the element will display the element's CSS Class (if a class has been set), as well as the CSS path to the element.

The `Copy Element Class` , and `Copy CSS Path` buttons will copy the related information to the clipboard. You can use either one for the Query part of the configuration. CSS Path is always more specific and usually the best Query to use. The CSS Element Class can have unintended consequences on other elements that share the same CSS Class, but sometimes using a CSS Class is a good fallback if the CSS Path is not working to change the site.

The CSS Path or CSS Element Class must be placed into the Query column in the Editor to "target" the element you wish to modify with one of the [functions](#) available in EZ WildApricot Web Designer.

COMPATIBILITY NOTE: The Copy Element Class and Copy CSS Path buttons only work on secure (https) websites. If your website is running insecurely, we recommend adding a [free SSL certificate to your custom domain](#).

The `View Properties` button will display the details about that element, such as the text it contains, styling, and any resource links.

The `View Error Log` button will open a new window and display any errors that have been encountered in the currently loaded configuration file (config.csv, english.csv or language.csv).

Editor

The bottom half of the panel displays the configuration file Editor. In the editor you can select the file you wish to update from the dropdown. Selecting the file will automatically load the configuration file editor. You can resize the columns as needed by dragging on the divider on the right hand side of the panel. You can also drag the rows to rearrange them. Right clicking on a row will give you the options to create and delete rows, and download the currently saved configuration file

Once you are finished making your changes, save the file back to your website by clicking on `Save changes to site` . You will need to authenticate once with your administrative WildApricot account.

If you choose to download the configuration file and edit it outside of the Inspector, the changes will be applied only when you manually upload the configuration file. The configuration file is a comma-separated file that can be edited in any spreadsheet. If

you decide to edit in a spreadsheet, ensure you save in UTF-8 format to preserve any international characters.

When you use the Inspector to save the configuration file, the current page will automatically refresh, and your changes should be immediately visible. If you don't see your changes, you may need to do a hard refresh of your site to see the changes. To do this, exit the inspector/editor and press `Ctrl` + `Shift` + `R` on PC or `Option` + `Shift` + `R` on Mac. You may need to do this for each language enabled on your site.

The Error Log

As EZ WildApricot Web Designer parses each configuration file it may encounter an error while processing a configuration row. If it does encounter an error, the error will be logged into the Error Log, and EZ WildApricot Web Designer willproceed to process the next configuration row in the configuration line. The configuration row with the error will not be proceessed until the error has been resolved.

The `View Error Log` button in the inspector opens the error log and displays the error message, timestamp and the ability to delete the error from the log. The error message will contain information about which line created an error and the error message. Currently correctly-formatted CSS Paths or CSS Element IDs that are not found on the page will *not* be logged as errors, as they will be simply skipped.

Note that blank lines in the configuration files are not counted and may throw off the line counter in the error message. It is recommended that configuration files have no empty lines to ensure any error messages point to the correct line number.

You can clear the error log by clicking `Clear All Errors` button.

Debugging the configuration or translation files by "stepping through"

You can debug the processing of the config.csv or the language.csv files by enabling a step-though debugger in EZ WildApricot Web Designer.

To enable the debugger add these lines to the EZ WildApricot Web Designer install JavaScript:

```
// Enable step-through debugging of config.csv:
stepThroughConfig = true;

// Enable step-though debugging of selected language:
stepThrough = true;
```

When step-through debugging is enabled, you can open the development console to view the processing of the config.csv or language.csv files. The console will display what file and line number it is processing, and the EZ WildApricot Web Designer function being processed on that line. It will then pause for 5 seconds and then more to the next line. You can watch the web page to see how the changes are occuring, and see what line in the file may be causing errors.

To change the delay time, add this to the EZ WildApricot Web Designer install JavaScript:

```
// Step-through delay in seconds:
stepThroughSpeed = <seconds>;
```

If you want the processing display to start after a certain line in the csv, add this to the EZ WildApricot Web Designer install JavaScript:

```
// Start config.csv step-through at a certain line:
stepThroughConfigFrom = <linenumber>;

// Start current language step-through at a certain line:
stepThroughFrom = <linenumber>;
```

Using EZ WildApricot Web Designer with WildApricot iframe Widgets

The EZ WildApricot Web Designer affects any page loaded from a WildApricot website, including the widgets published through the Website Settings. These are implemented as iframes which load the Javascript libary just like any web page. It's best to test the content on a regular WildApricot website before implementing the iframe gadgets to make sure all changes are rendering outside the iframe first.

To reference the widgets in an iframe in a different langauge when mutli-lingual mode is turned on use the ?watm- query parameter on the iframe URL in the `src` parameter of the iframe tag.

Example:

```
<iframe
  width="800px"
  height="4500px"
  frameborder="no"
  src="https://mysite.wildapricot.org/widget/Sys/Profile/?watm-french">
</iframe>
```

The iframe code above Will load the WildApricot profile widget in an iframe and trigger the French version of the profile.

If you omit the ?watm- in the `src` paramater the widget will load in the current language set previously by the language dropdown menu or the previously loaded langauge.

EZ WildApricot Web Designer Functions

When editing the EZ WildApricot Web Designer configuration and translation files, the following functions are available: Each function operates in a specific way to modify, style, hide or translate parts of your website.

text

> *Changes the text in targeted element(s), note that any links are eliminated in the targeted element(s). Note that the Default Text is empty (and ignored if filled) when using the text function.*

EXAMPLE:

```
Function: text
Replacement Text: Change my password!
Query: .loginBoxChangePassword
```

replace

> *Finds and replaces specific text in targeted element(s). Note that the Default Text must match the text that is being replaced. The search is done without case sensitivity.*

EXAMPLE:

```
Default Text: Home
Function: replace
Replacement Text: Accueil
Query: .menuInner, .WaGadgetBreadcrumbs > div > ul > li
```

replace_element

> *Finds the element(s) containing the specific text and replaces all the text in that element with the Replacement Text. The search is done without case sensitivity.*

EXAMPLE:

```
Default Text: Our Address
Function: replace_element
Replacement Text: Our offices are currently closed for renovations purposes.
Query: .menuInner, .WaGadgetBreadcrumbs > div > ul > li
```

shortdelay

> *Changes the text in targeted element(s) after **n** second - used for dynamically generated elements, such as a member directory. Note that the Default Text must match the text that is being replaced. The search is done without case sensitivity. Default delay time is one second, and you can customize this by adding the following to the JavaScript configuration:*

```
short_delay = 1;
```

`shortdelay` function will trigger the `replace` function after a **n** second delay if "Default Text" is present (otherwise it triggers `text` function).

EXAMPLE:

```
Replacement Text: Coordonnées
Function: shortdelay
```

```
Query: #membersTable > thead > tr > th:nth-child(1)
```

longdelay

> *Changes the text in targeted element(s) after **n** second - used for dynamically generated elements, such as a member directory. Note that the Default Text must match the text that is being replaced. The search is done without case sensitivity. Default delay time is three seconds, and you can customize this by adding the following to the JavaScript configuration:*

```
long_delay = 3;
```

`longdelay` function will trigger the `replace` function after a **n** second delay if "Default Text" is present (otherwise it triggers `text` function).

EXAMPLE:

```
Replacement Text: Coordonnées
Function: longdelay
Query: #membersTable > thead > tr > th:nth-child(1)
```

button

> *Changes the text on targeted button(s). The search is done without case sensitivity.*

EXAMPLE:

```
Replacement Text:  Click here to login
Function: button
Query: .loginButton
```

delaybutton

> *Changes the text on targeted button(s), after a 1 second delay, useful for widgets that have a JavaScript rendering delay*

EXAMPLE:

```
Replacement Text:  Click here to login
Function: delaybutton
Query: .loginButton
```

inactive

> *Ignores the configuration row. This is useful when troubleshooting an item or you wish to save details in the configuration file, but not activate it.*

The style column can be used in conjunction with a function, or on its own.

```
Query: h3
Style: color: blue;font-style: italic
```

hide

> *Hides the targeted element(s). This is useful when you wish to hide an element altogether. Neither Default Text or Replacement Text fields are necessary or used.*

EXAMPLE:

```
Function: hide
Query: #idFooterPoweredByWA
```

placeholder

> *Changes the placeholder text of targeted form element(s)*

EXAMPLE:

```
Replacement Text: Enter your search keywords here
Function: placeholder
Query: .searchBoxField
```

attribute

> *Sets/changes attribute of targeted element(s). Enter attribute name in "Default Text" column, and the attribute value in "Replacement Text"*

EXAMPLE:

```
Default Text: target
Replacement Text: _blank
Function: attribute
Query: #id_Footer1 a
```

@media

> *Sets a @media CSS Rule. Enter rule name in "Replacement Text" column*

EXAMPLE:

```
Replacement Text: screen and (max-width: 900px)
Function: @media
Query: #id_Header2
Stye: display: none;
```

link

> *Changes the target link of an existing link. Specifically it changes the `href` URL of targeted element(s). Note that almost always the query must contain the `a`*

> *target after the CSS Element Class or CSS Element Path.*

EXAMPLES:

```
Replacement Text: /donate
Function: link
Query: .headerDonateBtn a
```

```
Replacement Text: https://newpathconsulting.com/watm
Function: link
Query: #idFooterPoweredByWA a
```

createlink

> *Adds a link to an existing non-linked piece of text*

EXAMPLE:

```
Default Text: Some Text String that Needs a Link
Replacement Text: https://newpathconsuling.com
Function: createlink
Query: #someID
```

source

> *Changes the* `src` *URL of targeted element(s). This is useful when changing the
> iframe src or img src URL to load an alternative image or iframe.*

tooltip

> *Creates a text popup when mouse is hovered over element*

EXAMPLE:

```
Replacement Text: View/Edit Your Profile
Function: tooltip
Query: .loginBoxProfileLink
```

googlefont

> *Installs a Google font that can be used throughout other CSS property
> declarations. Place the font family name to use in the "Replacement Text" field.
> The font family name can be obtained from the [Google Fonts](#) website*

EXAMPLE:

```
Replacement Text: Lobster
Function: googlefont
```

EZ Add-Ons

EZ WildApricot Web Designer includes a number of add-ons that can be used to extend the capabilities of the WildApricot website module. To install an add-on, place the add-on file into the `ez-addons` folder. Next, add the following code to your JavaScript configuration:

```
ez_addons = [];
```

Within the square brackets, add type the name of the add-ons within quotes. Separate multiple add-ons with a comma, with each add-on name within their own quotation marks.

FontAwesome Add-On

EZ WildApricot Web Designer comes with a free Add-on for FontAwesome, making it easy to insert FontAwesome icons by using macros. To start, [sign up for FontAwesome](#) and add the provided JavaScript code snippet to your WildApricot Global JavaScript settings.

Here's an example of the code you may add. Make sure the `YOUR_FONTAWESEOME_KIT_ID` is replaced with the actual ID below.

```
<script src="https://kit.fontawesome.com/YOUR_FONTAWESEOME_KIT_ID.js" crossorigin="anonymous"></script>
```

To enable the addon, add this line to your EZ WildApricot Web Designer JavaScript installation script:

```
ez_addons = ["fontawesome"];
```

To insert an icon, first [find the name of the icon you wish to use](#). Add

```
[ez-fa]icon name[/ez-fa]
```

to any WildApricot content or headline gadget.

You can copy the name of icon by clicking on the icon and copying the text above the icon:

EXAMPLE:

```
Macro Text: [ez-fa]home[/[ez-fa]
Result: 
```

By default, the FontAwesome macro uses the "Solid" icon styles. To use any of the other styles, add a "style" parameter to the macro:

```
Macro Text: [ez-fa style="regular"]home[/[ez-fa]
```

These styles are currently available in FontAwesome, and some icon sets require a paid subscription:

- **Regular:** [Free Icons](#) | [All Icons](#)
- **Solid:** [Free Icons](#) | [All Icons](#)
- **Light:** [All Icons](#)
- **Thin:** [All Icons](#)
- **Duotone:** [All Icons](#)
- **Brands:** [Free Icons](#)

EZ-Notice Add-On

The EZ-Notice Add-On enables displaying important, colourful messages on your website. Using this macro, any text is be displayed in a colorful box on a WildApricot web

page, spanning the width of the layout that contains the macro. It can be used on any WildApricot content page, page template, or system template.

EZ-Notice

This the default notice!

This is a notice with color "red"

This is a notice with color "#155724"

To use the macro, insert your notice text inside the `[ez-notice][/ez-notice]` tags. An optional `color` parameter can decorate the notice. You can use a [color name](#) or [HTML color](#) code. EZ-Notice will use your selected color as the text color for the the notice, and automatically set the notice's border and background color to a matching lighter color.

EXAMPLES:

```
Macro Text: [ez-notice]Guild renewal fess are due by January 15th.[/ez-notice]
Macro Text: [ez-notice color="red"]Saturday's potter classes have been cancelled.[/ez-notice]
```

EZ-Toggle Add-On

The EZ-Toggle Add-On enables expandable and collapsible content sections (also known as "accordions"). Website visitors can click on the section header to open that section and automatically close the other sections on the page. This macro can be used to shorten long content such as a Frequently Asked Questions section.

EZ-Toggle

THIS IS SECTION ONE ⊖

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam nisi tortor, egestas sed mauris vitae, dignissim fringilla velit. Maecenas lobortis lacinia sodales. Nam tempor augue dolor, id maximus ligula blandit quis. Nulla egestas dui risus. Curabitur eu malesuada urna. Curabitur sed ultricies nulla. Duis laoreet elementum ante, placerat auctor lorem scelerisque in. Phasellus sed ornare ipsum. Etiam tempor quam maximus urna dictum, sed fermentum magna dapibus. Sed blandit velit quis blandit semper. Etiam dictum faucibus ligula. Fusce eu massa sed velit laoreet auctor. Integer at nibh facilisis, sagittis libero et, rhoncus ex. Phasellus aliquam faucibus sapien vitae aliquam. Suspendisse gravida maximus risus, non vulputate est condimentum ut.

THIS IS SECTION TWO ⊕

THIS IS SECTION THREE ⊕

The macro has two required components. The first is the `title` parameter - use this to specify what the section title should be. Next, place the content for that section

within the `[ez-toggle][/ez-toggle]` macro tags. Add a new macro for each section you wish to have. Each section is not required to be placed next to each other and can be located anywhere on the page.

EXAMPLE:

```
Macro Text: [ez-toggle title="How do I join?"]Visit Membership to find and purchase
the membership that's right for you.[/ez-toggle]
Macro Text: [ez-toggle title="Who can be a member?"]Anyone with an interest in our
industry may join.[/ez-toggle]
Macro Text: [ez-toggle title="What are the benefits for students?"]We have
scholarships students can qualify for.[/ez-toggle]
```

EZ-Tabs Add-on

The EZ-Tabs Add-On creates a tabbed content area on a page. Website visitors can click on the tabs to switch between different content sections.

The macro has two required components. The first is the `title` parameter - use this to specify what the label on the tab should be. Next, place the content for that tab within the `[ez-tabs][/ez-tabs]` macro tags. Add a new macro for each tab you wish to include.

EXAMPLE:

```
Macro Text: [ez-tabs title="2022 Scholarship Winners"]Khloe Blanchard, Deacon Chang,
Aine Kerr[/ez-tabs]
Macro Text: [ez-tabs title="2021 Scholarship Winners"]Ella-Mai Kramer, Tea Mcfarlane,
Kaci Rankin[/ez-tabs]
Macro Text: [ez-tabs title="2020 Scholarship Winners"]Jennie Miranda, Carina Myers,
Ruby-May Waters[/ez-tabs]
```

EZ-Library Add-On

The EZ-Library Addon makes it easy to insert a document library into a WildApricot website. The add-on takes care of formatting the presentation of the selected documents, sorting them by title or date, and makes them available for one-click download.

Below are our recent board minutes - click on the file name to download.

Document	Date
January 2018 Minutes	1/12/2018
August 2017 Minutes	8/25/2017
May 2017 Minutes	5/12/2017
November 2017 Minutes	4/10/2017
November 2016 Minutes	11/4/2016
February 2016 Minutes	4/19/2016
November 2015 Minutes	11/6/2015
September 2015 Minutes	9/11/2015
April 2015 Minutes	4/10/2015
February 2015 Minutes	2/20/2015
January 2015 Minutes	1/9/2015
November 2014 Minutes	10/31/2014

DO I NEED ANY SPECIAL SOFTWARE TO VIEW DOCUMENTS? ⊕

WHAT ARE THE TERMS FOR DOWNLOADING? ⊕

HOW DO I REPORT INACCURACIES IN THE DOCUMENTS? ⊖

Curabitur est enim, tincidunt nec tincidunt vitae, pulvinar luctus lacus. Curabitur et elementum nulla, vel convallis nisl. Praesent vel nunc ullamcorper, imperdiet justo in, porttitor arcu. Vestibulum fermentum tempus tempus. Donec id porta tortor, sit amet cursus enim. Duis consequat turpis id aliquet sollicitudin. Donec aliquet congue dolor. Proin lobortis id lacus eu pulvinar.

President's Letter | Fall Notice

Sed culpa, or nostrum, yet ullamco, yet dolorem, voluptatem. Natus eius ullamco modi laboris. Aperiam commodi. Qui enim, yet aliqua modi aspernatur, for sint. Nihil laboriosam, yet nulla qui illum id laboris. Ipsum architecto, and ipsa, nor voluptate. Vitae excepteur, for autem ad. Quaerat nisi. Nequeporro culpa, nor cillum, but natus. Aut voluptatem.

USAGE:

This add-on comes with two macros that must be used together. The first macro is `[ez-library][/ez-library]` used to configure the library. This macro has three mandatory parameters:

- `folder`: the path to the WildApricot file manager folder containing all the files to be included in the library. This path is relative to the standard "Resources" folder, and the path should not include "Resources". If your documents are located in the "Documents" folder, set the path to "documents". For example, if they are located in a subfolder called "Minutes" inside of the documents folder, set this to "documents/minutes".

- `sort`: specifies how to sort the files. Set this option to "name" to sort by the display name you assign to the documents in ascending order. Set it to "date" to sort by the date you assign to each document in descending order.

- `view`: specifies the layout of the document library. Currently, only one view is available: "list".

  - The "list" view displays the files in a three-column table. The first column contains an image representing the document type, the second

column contains the file's display name, and the third column displays the assigned date. Click on the display name to download the file.

The second macro is `[document][/document]` - this is used for each document you wish to include in the library, and must be located within the `[ez-library][/ez-library]` macro. This macro requires two parameters:

- `date` : This is the date you wish to use for this document. The date is used for sorting and is displayed in the library. The date must be formatted as "YYYY-MM-DD".

- `filename` : This is the filename that the file is saved as.

Within the `[document][/document]` macro, enter the display name to use for the document link.

EXAMPLE:

```
Macro Text:

    [ez-library folder="documents/minutes" sort="date" view="list"]
      [document date="2014-11-01" filename="minutes Nov. 1 2014.docx"]November 2014
Minutes[/document]
      [document date="2017-04-11" filename="Minutes 11 4 17.pdf"]November 2017
Minutes[/document]
      [document date="2015-11-07" filename="meeting 7 nov 2015.doc"]November 2015
Minutes[/document]
    [/ez-library]
```

Browser Requirements

EZ WildApricot 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.

Release 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 WildApricot 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 WildApricot, 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

0.92 - Fixed bug that prevented text containing HTML and links to be changed with replace or replace_element 3/5/2021

0.93 - support for switching languages using embedded WildApricot "widgets" in 3rd party content management system, added a standard French translation file 2/16/22

0.94 - added support for entering/exiting Inspector panel, copy/paste of element ID, class and CSS path in Inspector panel, more options added install script 4/19/2022

0.95 - added support for Inspector Hover mode which shows the outlines of each element on a page when using the Inspector 5/11/2022

0.951 - fixed translations to be rendered for multi-page widgets embedded with a primary and second language 5/17/2022

0.952 - fixed bug on Copy ID button in inspector that adds a # to the id on the clipboard 5/25/2022

0.96 - added showing the selected text of an element, ability to provide alternative text or translation and a drop down for the function, added license checking, trial mode 6/4/2022

2.0b1 - new inspector and configuration file editor, a view properties window including image color picker, support for unlimited languages, easy configuration file upload and preview of changes, rewritten in ECMAScript 6 (no more jQuery!) 8/13/2022

2.0.1 - added a new function for adding a Google font (`googlefont`) and adding a link to a piece of text (`createlink`), added support for cookies to work better 9/8/2022

2.0.2 - bug fixes, minimized french.csv file 9/20/2022

2.0.3 - allow moving of WATM icon to the right or left 10/4/2022

2.0.4 - add [ez-fa] macro for adding FontAwesome glyphs into a WildApricot page 10/6/2022

2.0.5 - added "EZ Macro" add-on architecture, and added [ez-tabs], [ez-notice], [ez-toggle], [ez-library] macros 10/13/2022

2.0.6 - added `delaybutton` function and changed the `delay` function to use `replace` when default text is present 12/06/2023

2.0.7 - changed `delay` to `shortdelay` and `longdelay`, the delays are configurable in the documented Javascript variables 01/04/2023

2.0.8 - NO code changes - README.md updated installation instructions (custom domain before script installation) 02/15/2023

2.0.9 - removed the CSS from install script and added it into the code to ensure no "white screen of death" scenarios if Javascript fails to install 02/18/2023

2.1.0 - added the `replace_element` function for backwards compatibility to pre-v2 WATM/EZ Website Designer versions 04/09/2023

- the EZ license is now checked once per day, rather than on every page load
- if the [web browser's language is set](#) EZ WildApricot Web Designer will automatically switch to the language, if available.
- added an error log and a "View Error Log" button to the inspector
- when an error is encountered while parsing the CSV file the error is logged, and parsing continues with the next line
- third-party script files are now included, removing requirement for domain whitelisting
- added instructions on adding translated content gadgets
- added "step through" debugging mechanism to process config and language files line by line with delay