ezSearch
Documentation

Contents

1 Introduction

ezSearch is an Umbraco package designed to provide a general purpose search for your sites that is easy to install, easy to customise and easy to use. Inspired by XSLTSearch, **ezSearch** brings the same simplicity to your razor projects and is powered by the built in Umbraco examine layer, so is blazingly fast. If you want a search plugin that just works, then **ezSearch** is for you.

2 Installation

2.1 System Requirements

Before you get started, there are a number of things you will need:

1. .NET 4.0+
2. Umbraco 6.0.4+
3. MVC Rendering Engine Enabled

2.2 Install ezSearch

To install **ezSearch**, simply download the package file from the project page on our.umbraco.org at the following location:

http://our.umbraco.org/projects/website-utilities/ezSearch

Once downloading, install as you would any other Umbraco package via the package installer in the developer section of the back office.

Once installed, you should have an **ezSearch** macro available in the Macro's list, as well as a *"Search Page"* doctype and template for convenience.

2.3 Adding ezSearch to a Template

To add **ezSearch** to your site, you'll want to drop the **ezSearch** macro created during installation to a template. Depending on your skill level, this can be done in one of two ways.

2.3.1 Via the UI

To add **ezSearch** to your template via the Umbraco UI, whilst in the template editor interface, place the cursor in your template where you would like to insert the **ezSearch** macro then click the *"insert macro"* button in the toolbar, and select *"ezSearch"*.
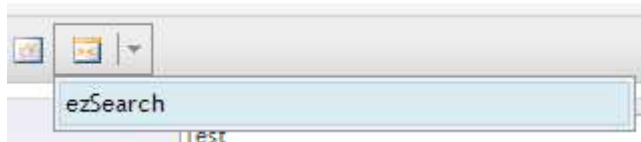
This will then launch the *"macro configuration"* dialog where you can configure **ezSearch** (See the Configuration section for details on each of the configuration options).

Once you are happy with the configuration of **ezSearch**, click the *"ok"* button to have Umbraco insert the macro for you.

NB: Be sure to save your template before continuing.

2.3.2 Via Razor

To add **ezSearch** to your template via razor, you should use the Umbraco helper available in your views and call the RenderMacro method like so:

Example:

```
1.  @Umbraco.RenderMacro("ezSearch", new { ... })
```

NB: For more information on the options available for the RenderMacro method, please see the documentation on our.umbraco.org at the following location:

http://our.umbraco.org/documentation/Reference/Templating/Macros/Razor/

To configure **ezSearch**, simply pass the configuration fields in via an anonymous object to the RenderMacro method (See the Configuration section for details on each of the configuration options).

3 Configuration

ezSearch comes with a number of configuration options so that you can alter its behaviour to your needs. Details of these options and how they affect **ezSearch** are as follows:

3.1 RootContentNodeId

Type	Possible Values	Description
Int	A valid content node id	Sets the start node from where content based searches should start. The default value for **RootContentNodeId** is -1 and will search all pages on the current site. **NB**: If you wish to have the search to search all content across all sites, please comment out the if statement at line 47 in ezSearch.cshtml.

3.2 RootMediaNodeId

Type	Possible Values	Description
Int	A valid media node id	Sets the start node from where media based searches should start. The default value for **RootMediaNodeId** is -1 and will search all media on the current site.

3.3 IndexType

Type	Possible Values	Description
String	**CONTENT**, **MEDIA** or **BOTH**	Sets which indexes **ezSearch** should search, either content or media or both. The default value for **IndexType** is "BOTH".

3.4 SearchFields

Type	Possible Values	Description
String	A coma separated list of field aliases	Lists the field aliases **ezSearch** should search against. The order of the fields dictates the priority of that field, running from most important to least important. For a page to be returned, it **must** contain all the keywords supplied in the search term across any of the fields, however its order in search results will depend upon which field contains those keywords. The default value for **SearchFields** is "nodeName, metaTitle, metaDescription, metaKeywords, bodyText"

3.5 PreviewFields

Type	Possible Values	Description
String	A coma separated list of field aliases	Lists the field aliases **ezSearch** should use as source for the preview in the search results. **ezSearch** will use the first field from the list that has a value on the returned node, so fields should be supplied in order of importance.

3.6 PreviewLength

Type	Possible Values	Description
Int	A positive integer	Sets the maximum length of the preview text before it is concatenated. Concatenated preview text will be cropped at the last complete word and suffixed with ellipsis. The default value for **PreviewLength** is 250.

3.7 PageSize

Type	Possible Values	Description
Int	A positive integer	Sets the number of results **ezSearch** should display per page. The default value for **PageSize** is 10.

3.8 HideFromSearchField

Type	Possible Values	Description
String	A valid field alias	Sets the alias of the property that should determine whether the page should appear in the results or not. The field should be a true / false field with its checked value indicated that the page should be omitted. The default value for **HideFromSearchField** is "umbracoNaviHide".

3.9 SearchFormLocation

Type	Possible Values	Description
String	**TOP, BOTTOM, BOTH** or **NONE**.	Determines where the search form should be located, either the top, bottom, both or none at all. The default value for **SearchFormLocation** is "BOTTOM".

4 Customization

Once you have **ezSearch** installed, you will probably want to be able to change the look of what is output. To aid with this, **ezSearch** uses a number of styles for you to target, and uses dictionary items to allow you to localize the output.

4.1 Styling

What follows are code snippets of the various components of the **ezSearch** output.

4.1.1 .ezsearch-form

Example:

```
1.  <form action="" method="GET" class="ezsearch-form">
2.      <input type="text" name="q" />
3.      <input type="submit" value="Search" />
4.  </form>
```

4.1.2 .ezsearch-summary

Example:

```
1.  <div class="ezsearch-summary">
2.      <p>...</p>
3.  </div>
```

4.1.3 .ezsearch-results-range

Example:

```
1.  <div class="ezsearch-results-range">
2.      <p>...</p>
3.  </div>
```

4.1.4 .ezsearch-results

Example:

```
1.  <div class="ezsearch-results">
2.      ...
3.  </div>
```

4.1.5 .ezsearch-result

Example:

```
1.  <div class="ezsearch-result">
2.      <h2><a href="...">...</a></h2>
3.      <p>...<strong>...</strong>...</p>
4.  </div>
```

4.1.6 .ezsearch-pager

Example:

```
1.  <div class="ezsearch-page">
2.      <p>
3.          <a|span class="prev" href="...">Previous</a|span>
4.          <a|span class="page" href="...">1</a|span>
5.          ...
6.          <a|span class="page" href="...">10</a|span>
7.          <a|span class="prev" href="...">Next</a|span>
8.      </p>
9.  </div>
```

NB: Where a|span is displayed, **ezSearch** will either render an a tag or a span tag depending on whether the link is active or not.

4.1.7 .ezsearch-no-results

Example:

```
10. <div class="ezsearch-no-results">
11.     <p>...</p>
12. </div>
```

4.2 Localization

To change the text strings present in ezSearch, these are controlled via Umbraco dictionary items:

4.2.1 [ezSearch] Next

The label for the *"Next"* link in the search pager.

4.2.2 [ezSearch] Previous

The label for the *"Previous"* link the search pager.

4.2.3 [ezSearch] Search

The label for the *"Search"* button and placeholder for the *"Search"* field.

4.2.4 [ezSearch] Summary

The format string for the search summary view. Enter {0} to display the entered search term and {1} for the number of results found.

The default value for the summary string is *"Your search for "{0}" matched {1} page(s).".*

4.2.5 [ezSearch] Results Range

The format string for the results range view. Enter {0} to display the start result position and {1} to display the end result position for the current page.

The default value for the results range string is *"Showing results {0} to {1}.".*

4.2.6 [ezSearch] No Results

The format string for the no results view. Enter {0} to display the entered search term.

The default value for the no results string is *"No results found for search term {0}."*.

5 Integration

The **ezSearch** macro's job is to find and display content that matches the search term passed in to it. **ezSearch** looks for the search term on the current pages querystring in a variable named *"q"*.

To perform a search, you can enable the search form in the options for the **ezSearch** macro and navigate to the search page to perform a search there, however, it is likely that you would want to trigger the search from other locations too (such as the header or footer of your site). To do this, simply create a form with its action attribute set to the URL of the search page, it's method attribute set to *"GET"* and with an input field with its name attribute set to *"q"* as well as some form a submit button, and **ezSearch** will take care of the rest.

Example:

```
1.  <form action="/search" method="GET">
2.      <input type="text" name="q" />
3.      <input type="submit" value="Search" />
4.  </form>
```

TIP: Once you have a form submitting from any location to the search page, you can simply turn off the **ezSearch** form via the macro options.