# UMB8 Simple Reviews

# Introduction

This is a simple package for Umbraco 8 that adds review functionality to your website. UMB8SR can relate reviews to your website or relate reviews to individual content items. This is useful if you want relate reviews to products, blog articles, services or anything like that. A review can be easily related to any content node.

# Fundamentals

This package is designed for developers and to speed up the development process of adding end to end reviews functionality to your website. Full documentation is provided but a basic level of programming / Umbraco experience is required.

# Features

**Capture Reviews form**

There is a built-in form for you to capture reviews. This is possible through adding a simple partial view to your page. You also have full access to the HTML and CSS to give you full control over the form style.

**Review Partial Views**

We have built you a few partial views to make it easy for you to display reviews on your website.

**Basic API**

We have built you a basic API exposing some of the most used methods. With that said though, reviews are simply Umbraco content nodes so if required you have the ability to write to own code to add, update, view reviews.

**Relate reviews to your site as a whole**

You can relate a review to your website. Most of the time you may just want to relate reviews to your website rather than to specific products or content. This feature allows for you to do that.

**Relate review to specific content items**

You can relate reviews directly to Umbraco content nodes. In some cases, you may want to relate reviews to Umbraco content items. Some examples might be products, articles, services etc. This feature allows you to do this.

**Relate reviews to a logged in member**

You can relate reviews to a logged in member. In some cases, you may only want signed up members to have the ability to leave reviews. This featured allows you to do this.

**Choose where to store reviews**

When adding reviews, you can pass in a parentId which will determine where to store the reviews. Out of the box we create a review folder, and this is the default location were reviews will get stored. In some cases, you may want to store a review under a product or service. This feature makes that possible.

**Import Reviews**

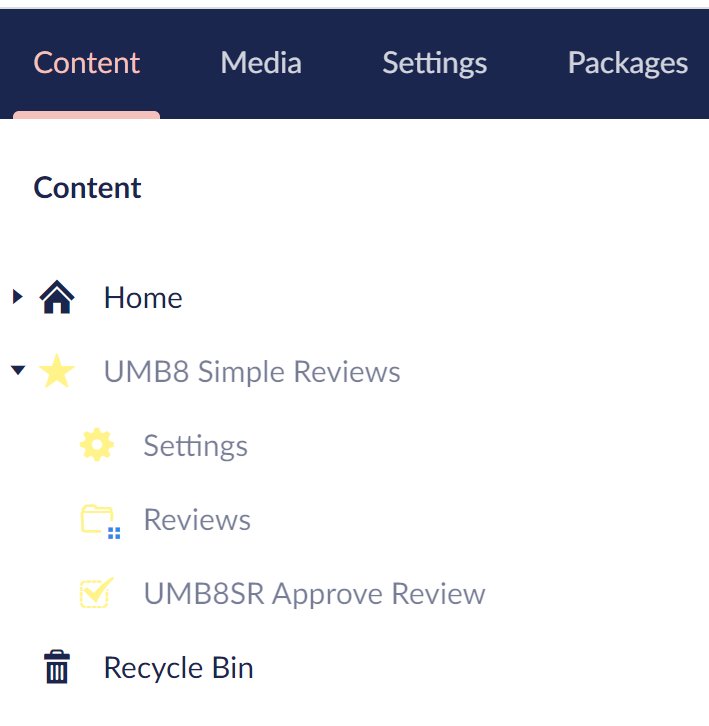
We have provided a review spreadsheet template. You can add all your reviews to the spreadsheet and easily import batch reviews.

**Approve Reviews**

We have built a review approval process which can easily be enabled and disabled. If enabled, you will be notified via email when a review has been made. This is a one click process to approve or disapprove reviews.

# Installation & Configuration Guide

1. Install the package through Umbraco content repository or manually using the zip.
2. Once complete go to the content section. You will now see some new content nodes in the root. By default, these will be unpublished.



1. Click on each UMB8 Simple Reviews, Reviews, and UMB8SR Approve Review content nodes and save and publish.
2. Now click on the settings note and start populating the required / desired settings. A break-down of each setting is detailed below. Once populated Save and publish.
   * **Site Url (required)**

This is the root domain name for your website. We use this for various reasons but the main one is for the approval process emails and automatic approval urls. This should be your full website url EXCLUDING a '/' on the end. Eg. So for example <https://www.yourwebsite.com>

* + **Add Review Complete Page (required)**

Select one of your website pages as a review complete page. This will be a thank you page which the user will get re-directed to once they have submitted a review.

* + **Enable Google ReCaptcha (optional)**

We have built a review capture form and this form has Google Re-Captcha capabilities. If you want to add Google Re-Captcha to this form, then please check his setting. If you enable this setting you must also enter a value public and private key.

* + **Google ReCaptcha PublicKey (optional)**

If you have enabled Google Re-Captcha then this field is required. You can get your own public and private keys here. <https://www.google.com/recaptcha/intro/v3.html>

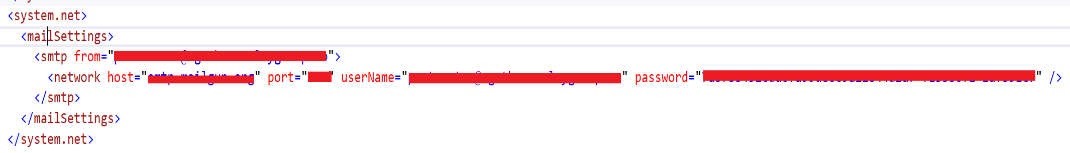
* + **Google ReCaptcha PrivateKey (optional)**

If you have enabled Google Re-Captcha then this field is required. You can get your own public and private keys here. <https://www.google.com/recaptcha/intro/v3.html>

* + **Enable Approvals (optional)**

If you want reviews to run through an approval process before they are displayed on your website, then check this setting. When the user submits a review if this is setting is checked the administrator will be notified via email. There will be a one click link in the email to approve the review.

**\*please note - do not forget to set up your SMTP settings within your web.config**



* + **Approval To Name (required)**

This field is required but only used if enable approvals is checked. This is the name of the person who will be in charge of approving reviews.

* + **Approval To Email (required)**

This field is required but only used if enable approvals is checked. This is the email address of the person who will be in charge of approving reviews.

* + **Approval From Email (required)**

This field is required but only used if enable approvals is checked. This is the email from address where the approval emails will be coming from.

* + **Review Approval Complete Page (optional)**

This is the link to the page where the administrator will be redirected to after they have successfully approved a review. By default, we will provide some un-styled on-screen messages if nothing is set here.

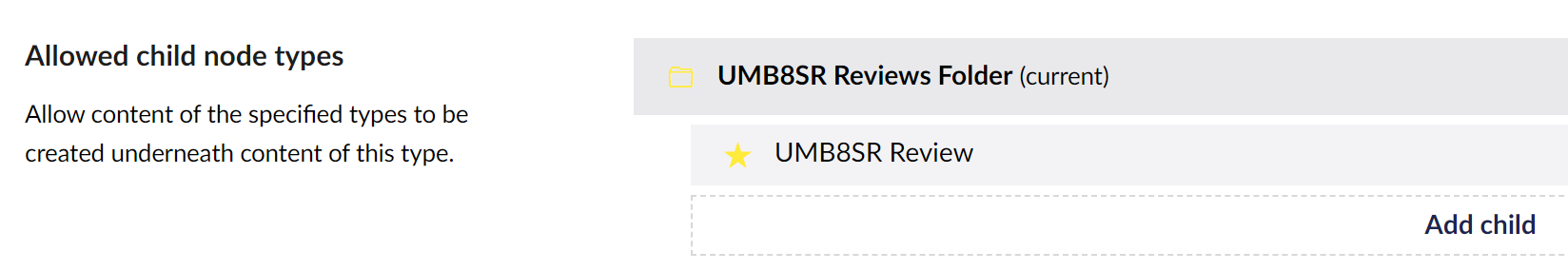
* + **Import CSV File (optional)**

If this a feature so you can upload your own reviews in bulk. If you want to import reviews have a look in your project for this file. /App\_Plugins/UMB8SR/ImportTemplate.csv. This is a review template. You can create a spreadsheet with your data matching this format. You can then upload this to the umbraco media section. Then populate this field. When you save and publish the settings the reviews will be imported.

* + **Import Review FolderId (optional)**

By default, reviews will be stored in the ‘UMB8 Simple Reviews/Reviews’ folder which is a simple Umbraco content node with List View. You can override this by setting the Id of the content name where you want the reviews to be stored.

**\*please note – Do not forget to allow umb8srReview document types to be added as children of the content node you have enter in here.**



# Adding Review Form

We have built a form already for you and can be found here.

/Views/Partials/UMB8SR/reviewForm

To add this to your page you can add the following code into your view.

**Add a review form relating to your Website**

This will add a form that will relate the review to your website rather than an Umbraco content node such as a product, service, for article. By default, the review will get stored in the ‘UMB8 Simple Reviews/Reviews’ folder unless you have set the ‘Import Review FolderId’ field.

**@Html.Action("UMB8SRReviewForm", "UMB8SRReviewFormSurface", new { siteReview = true })**

**Add a review form relating to an Umbraco content node**

This will add a form that will relate the review to an Umbraco content node such as a product, service, for article. The way this will work is it will relate the review to the content node the form has been added to. With that said, if you want the review to be related to a product then the form should be directly added to that product page and siteReview set to false. By default, the review will get stored in the ‘UMB8 Simple Reviews/Reviews’ folder unless you have set the ‘Import Review FolderId’ field.

**@Html.Action("UMB8SRReviewForm", "UMB8SRReviewFormSurface", new { siteReview = false })**

**Add a review form relating to an Umbraco content node and store the review as a child of that content node.**

So this will be the same as above ‘Add a review form relating to an Umbraco content node’but instead of storing the review in the ‘UMB8 Simple Reviews/Reviews’ folder or in your custom folder by setting the ‘Import Review FolderId’ field. Instead this can be overridden so you can stored that review as a child of the content code the form is on (Or anywhere for that matter).

**@Html.Action("UMB8SRReviewForm", "UMB8SRReviewFormSurface", new { siteReview = false, parentId = 1111 })**

**\*please note - do not forget to set parentId to the id where you want to store the review. Also please remember to make sure this documentType allows umb8srReview document types to be added as children**

# Showing Reviews on your site

We have created some partial views to help you with this. Since reviews are simply Umbraco content nodes then there is no reason why you can also write your own code to handle this. Our partial views will serve as a great starting point.

**/Views/Partials/UMB8SR/getAllApprovedReviews.cshtml**

**/Views/Partials/UMB8SR/getApprovedReviewsByRelatedId.cshtml**

**/Views/Partials/UMB8SR/getApprovedSiteReviews.cshtml**

# Basic API Endpoints

As you have seen we have added a few API methods to get at the review data. These methods can be used within your views and partial views. You can see how these work by looking at the partial views provided above.

@using Current = Umbraco.Web.Composing.Current

@using UMB8SR.Models

@using UMB8SR.Helpers

@{

IUmbHelper umbHelper = Current.Factory.GetInstance<IUmbHelper>();

List<UMB8SRReviewObject> reviews = umbHelper.GetAllApprovedReviews(6);

}

**Here is a list of our helper methods.**

UMB8SRSettingsObject GetSettings();

List<UMB8SRReviewObject> GetAllApprovedReviews(int take = int.MaxValue);

List<UMB8SRReviewObject> GetApprovedSiteReviews(int take = int.MaxValue);

List<UMB8SRReviewObject> GetApprovedDescendantReviews(IPublishedContent content, int take = int.MaxValue);

List<UMB8SRReviewObject> GetApprovedReviewsByRelatedId(int relatedId, int take = int.MaxValue);

**Here is a list of our service methods**

UMB8SRReviewObject InsertReview(UMB8SRReviewObject reviewObject);

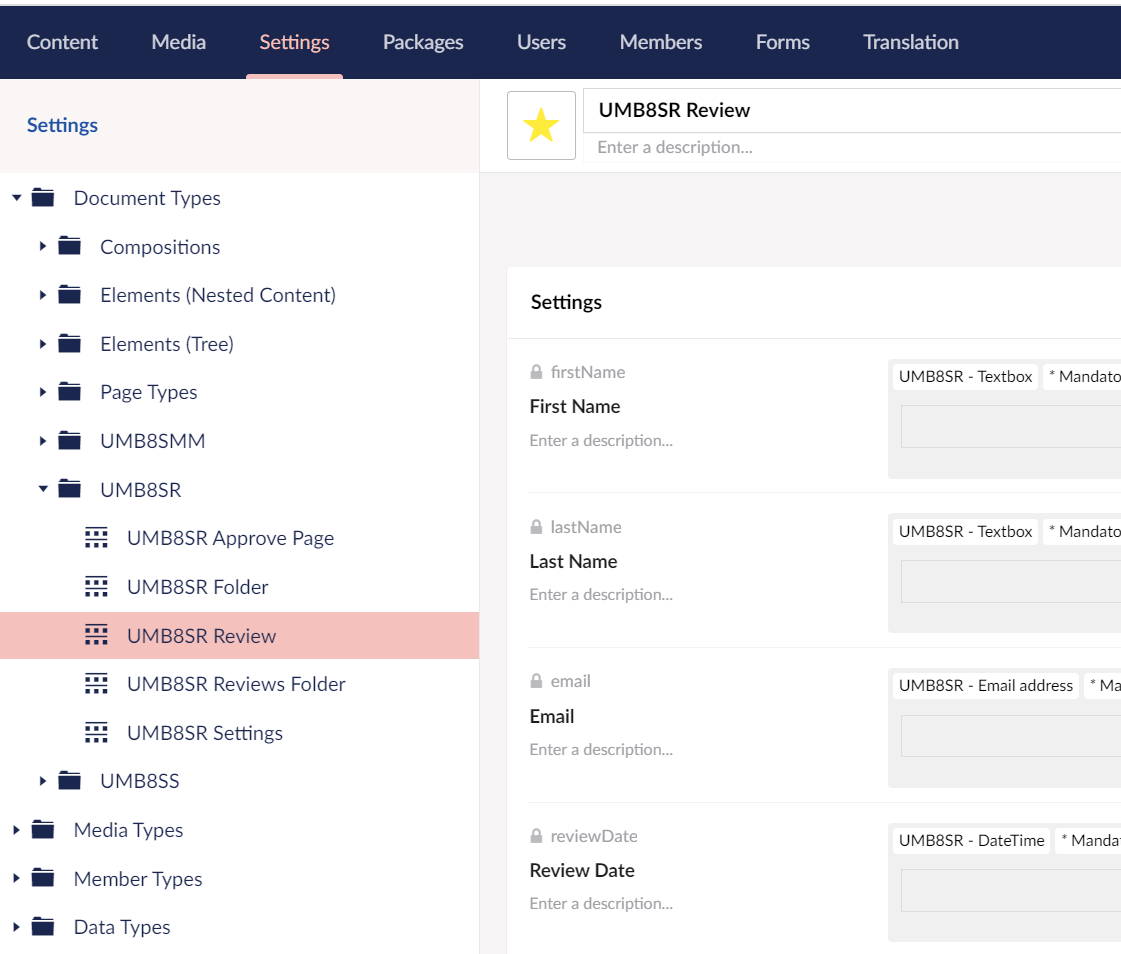
bool ApproveReview(int reviewId);

void ImportReviews(int reviewFolderId, string importCSVFileUrl);

# Extras & Session Variable

When inserting reviews there is a couple of session variables that can be used to override the default functionality. Basically, if these session variables have been set before the review form loads them the values will be used when inserting the review.

Please get familiar with the UMB8SR Review document type and this will all make more sense.



**Here is a list of session variables which can be used.**

**UMB8SR\_ReviewsFolderId**

Used to override the ReviewsFolderId field

**UMB8SR\_RelatedId**

Used to override the Used to override the ReviewsFolderId field

**UMB8SR\_MemberId**

Used to override the MemberId field

**UMB8SR\_PhotoUrl**

Used to set the PhotoUrl field

# Styling

You can style the forms and partial views however you want. You have access to the cshtml files to add your own HTML and CSS.

We have however got some default CSS which can be found here.

App\_Plugins\UMB8SR\css\style.css