

SEO Checker

User manual



1	INTRODUCTION	4
2	INSTALLATION.....	5
2.1	Install a license	5
2.2	Give a user access to SEO Checker	6
3	SEO CHECKER FOR CONTENT EDITORS.....	7
4	VALIDATE PAGES	8
4.1	Manual validation.....	8
4.2	Automatic validation	8
4.3	Scheduled validation	9
4.4	Validation queue	10
5	ISSUE OVERVIEW.....	11
5.1	Validation issues.....	11
5.2	Inbound link errors.....	12
5.3	Configuration errors.....	13
5.4	Delete and ignore validation issues.....	15
5.5	Bulk Actions	15
6	REDIRECT MANAGER	16
6.1	Create a new redirect	16
6.2	Redirect data type	18
6.3	Redirect Import/Export.....	18
6.4	Automatically fix broken links after url change	20
7	SEO CHECKER CONFIGURATION	21
7.1	Configuration	21
7.2	Redirect settings	22
7.3	Domain settings.....	23
7.4	Document type settings	23
7.5	Ignore list	25
7.6	Email settings	26
7.7	Configure the SEO Checker Data Type	27
7.8	Additional data types	27
7.9	Notifications	27
8	LIBRARY	29
8.1	MVC.....	29
8.2	Razor	30
8.3	XSLT.....	30
8.4	.Net Usercontrol.....	31
8.5	MVC.....	32
8.6	Special properties	32
9	EXTEND SEO CHECKER	33

9.1	URL Rewrite provider	33
10	MANUAL INSTALLATION	34
10.1	Manual installation of files	34
10.2	Alternative Configuration file location	36
10.3	Install database scripts	36
10.4	Additional Applications.config configuration	41
10.5	Additional Trees.config configuration	42
10.6	Additional Dashboard.config configuration	42
10.7	Add the XSLT Extension	42
10.8	Add the notfound handler	42
10.9	Additional web.config configuration	42
10.10	Add the Delete Action to config Extension	42
10.11	Add the license file	43
11	VALIDATION RULES	44
11.1	Page validation rules	44
11.2	Configuration validation rules	45
12	TROUBLESHOOTING	46
12.1	I don't see the SEO checker section	46
12.2	I don't see the SEO checker dashboard.....	46
12.3	I see the following message on the redirect overview.....	46
12.4	I see the following message on the validation queue.....	46
12.5	Validation is reporting errors on the document.....	46
12.6	Some pages don't get validated	46
12.7	I don't receive email notifications.....	46
12.8	I found a bug in the software	46
12.9	I have another question	46

1 Introduction

SEO checker helps you find common SEO issues on your website such as missing meta tags, broken links etc. SEO checker allows you to fix issues on the page before publishing the page.

Every page that has a template assigned can be validated. All validations are based on the "Search Engine Optimization Starter Guide" document from Google <http://tinyurl.com/yjm24yq> check chapter validation rules for a full list of all validations.

SEO checker allows you to preview the Google search result snippet when editing content and will give feedback about keyword usage in content and meta information.

SEO checker will report any inbound link that is broken and will automatically redirect to the correct page when a link pointing to your website uses the wrong extension, php instead of aspx for example and will automatically redirect to the correct page when an editor renames or moves a document.

SEO checker allows you to generate meta data based on existing content on the page, can automatically generate robot.txt and sitemap.xml files for you.

SEO checker will also automatically solve canonical issues using its own URL Rewrite module.

SEO Checker can schedule the validation process and allows users to receive notifications about issues on the site.

IMPORTANT:

SEO checker requires **IIS7** and is compatible with Umbraco version **4.8 and up to 7.x**. SEO checker is compatible with SQL Server (Express), SQL CE 4 and Mysql databases.

SEO Checker uses HTML Agility pack which is licensed under MS-PL License.

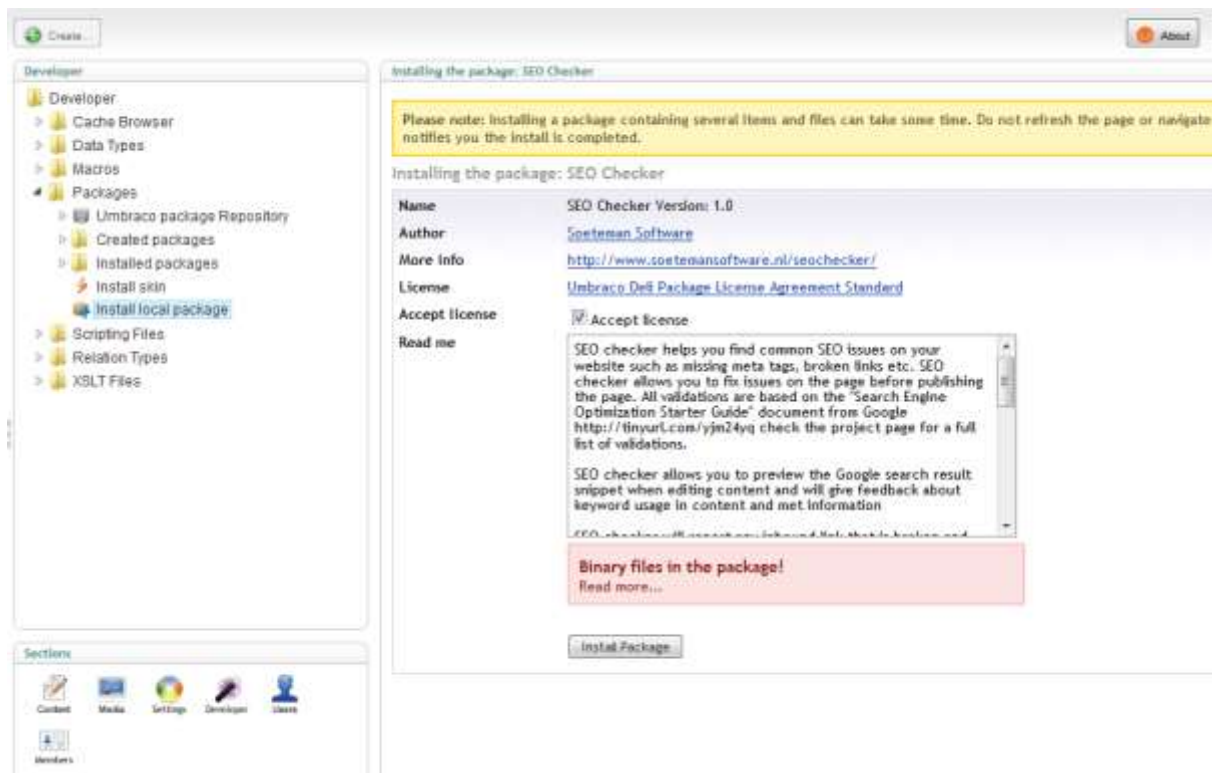
<http://htmlagilitypack.codeplex.com/license>

SEO Checker uses Lumenworks Framework IO which is licensed under MIT license

<http://www.codeproject.com/Articles/9258/A-Fast-CSV-Reader>

2 Installation

Install SEO Checker using the Umbraco package installer.



Make sure that the installer has modify rights on the following folders:

- /bin
- /App_Data
- /Umbraco
- /XSLT

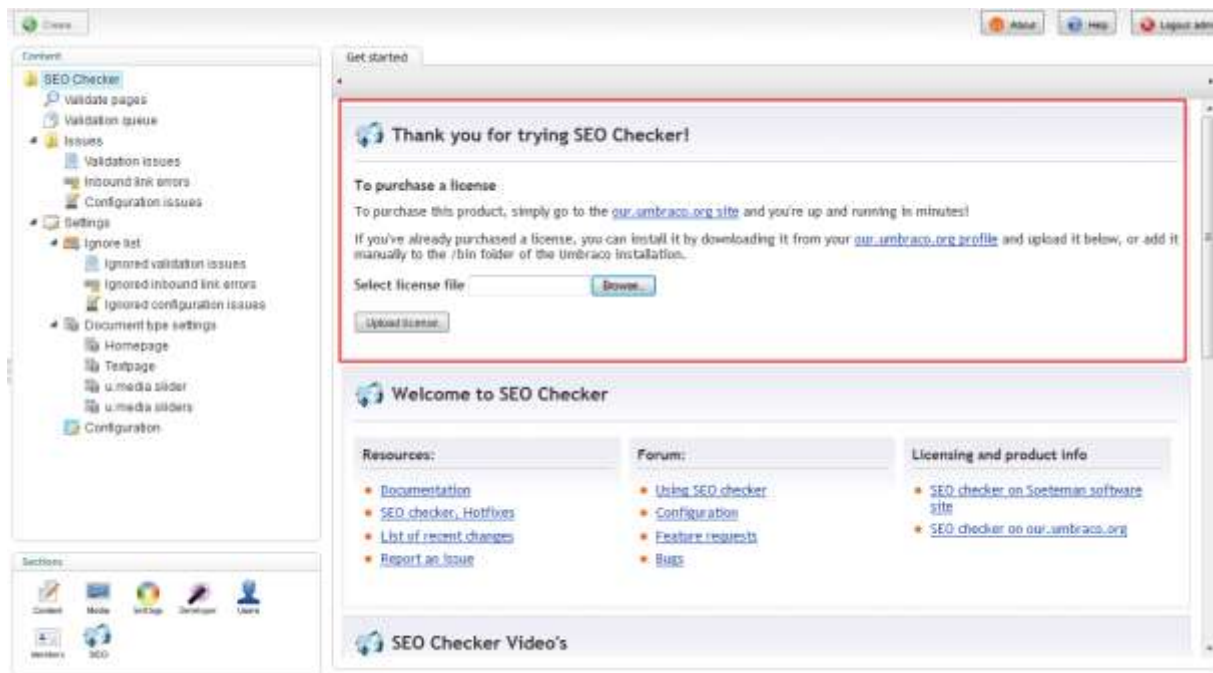
The installer also needs rights to create tables in the database.

If for some reason you can't give the installer these rights, you can install the package manually. Read the chapter manual installation how to do this.

Once the installer is finished you will see the new SEO Checker section.

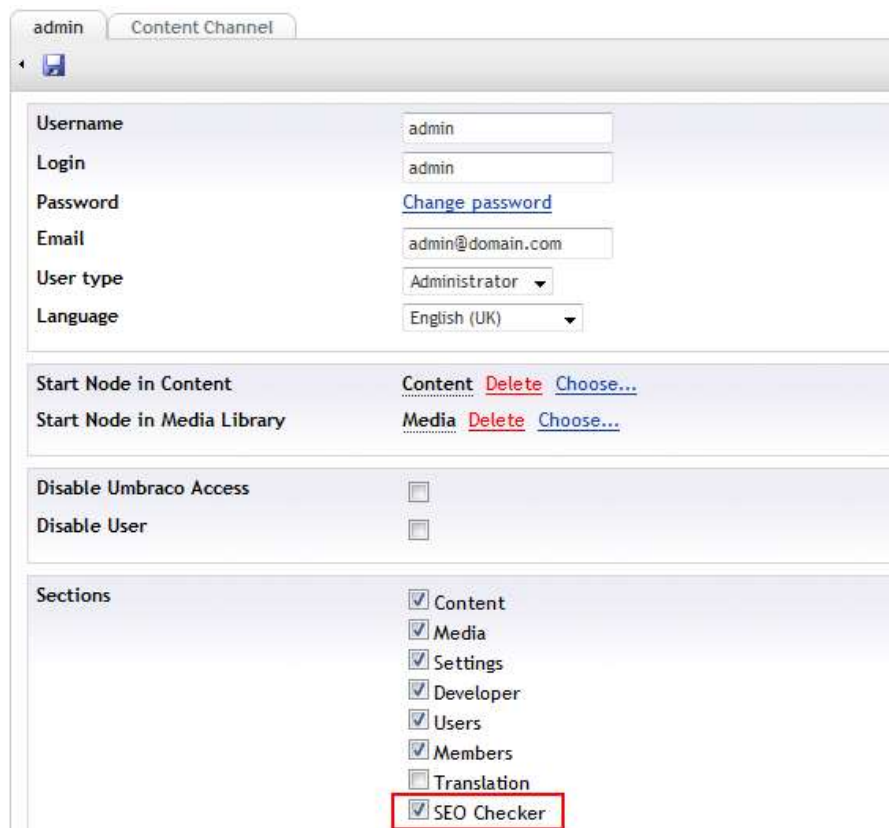
2.1 Install a license

To install a license download the license from your profile on our.umbraco.org and go to the SEO Checker section in Umbraco. When running in trial mode, upload instructions for a license are shown. Upload the license file to remove the trial restrictions



2.2 Give a user access to SEO Checker

When installing SEO Checker, the user installing the package will be given rights to the SEO Checker section automatically. If you need to give more users access to the SEO Checker section, edit the user and enable the SEO Checker section.



3 SEO Checker for content editors

SEO Checker comes with a Data Type that can be added to document types. The SEO Checker Data Type will show a Snippet preview how the page will be shown in the Google search results¹.

Snippet preview	Installing simple Umbraco modules © 2013 demo.com/installing-modules/ Installing simple Umbraco modules encapsulate specific bits of advanced functionality that are easily added to your website.
Focus keyword The keyword you want to rank this page for, example: Umbraco editor	Umbraco modules
SEO title When left empty the configured SEO title property for the document type will be used	Installing simple Umbraco modules
SEO description When left empty the configured SEO description property for the document type will be used	Installing simple Umbraco modules encapsulate specific bits of advanced functionality that are easily added to your website.
Validation result	The focus keyword is used in the SEO title. The focus keyword is used in the SEO description. The focus keyword is used in the body text. The focus keyword is not found in the page title (H1) The document contains "Lorem ipsum" test content.

It's possible to specify a focus keyword, this is the keyword you want to rank the page for, use the Keyword tool(1) to get help selecting the keywords you want to rank the page for. SEO Checker will validate if the keyword can be found in the most important elements of the page:

- Page Title (H1)
- Url
- SEO Title (<title>)
- SEO Description (<meta name="description".....)
- At least once more in the text of the page

You can also specify the SEO Title, SEO Description and SEO keywords (when configured). When you already have fields for those properties defined in your document type you can configure the Data Type to use those properties. It's also possible to use default properties to generate the meta data fields, check chapter configuration for various configuration options per document type of configuration of the Data Type.

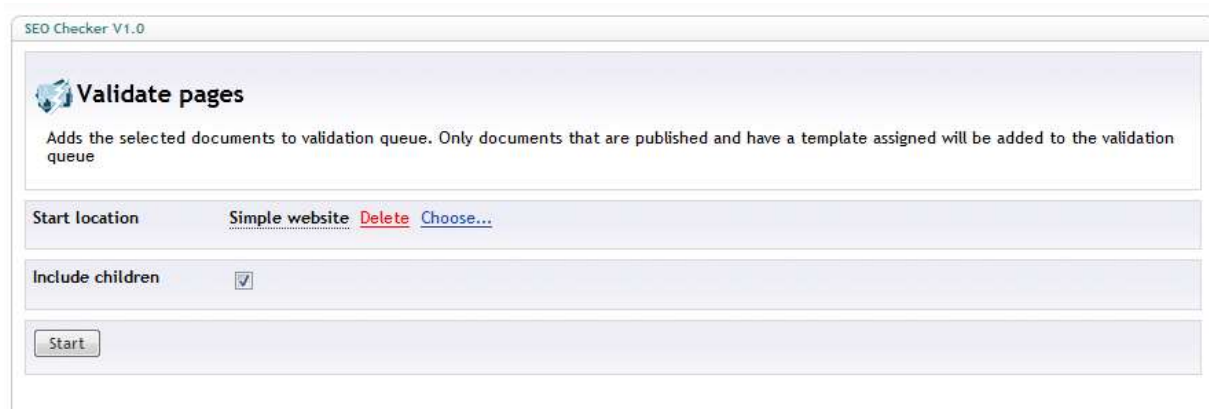
The page will be checked for other SEO Issues on the page also. See Validation rules for a list of all validations. **The unpublished version of the page will be checked, this allows you to solve issues before publishing the page.**

¹ Currently only normal snippets are supported, rich snippets will be supported in a future release. The preview requires <title> and < meta name="description" .. tags available in the template that renders the page.

4 Validate Pages

4.1 Manual validation

To validate a site, or a portion of a site open SEO Checker and select the **Validate Pages** option from the menu. This will show the following screen. By default the root is selected, if you want to validate a portion of a site specify the start location. When the include children all published children will be validated also. When you hit start the pages will be added to the validation queue.



SEO Checker V1.0

Validate pages

Adds the selected documents to validation queue. Only documents that are published and have a template assigned will be added to the validation queue

Start location: Simple website [Delete](#) [Choose...](#)

Include children: ☒

[Start](#)

Only published pages having a template assigned will be validated

4.2 Automatic validation

When a page gets published and have a template assigned it will be added to the validation queue automatically. Check chapter configuration if you want to disable automatic validation.

Only published pages having a template assigned will be validated

4.3 Scheduled validation

It's possible to schedule the validation of pages for a certain day and time. Open SEO Checker, right click on the validate pages item and select **Add scheduled validation**. This will show the following screen.

SEO Checker V1.1

Scheduled validation
Add documents to the validation queue on a certain day and time. Only documents that are published and have a template assigned will be validated!

Validation options 1

Start location: Simple website [Delete](#) [Choose...](#)

Include children: ☒

Schedule options 2

Scheduled task name:

Notify emailaddress:
To specify more than one emailaddress use a comma seperated list of email addresses.

Validate every: ☒ Day ☐ Week

Time: :

1. Validation options. Specify the start location and whether to validate children of that page also
2. Schedule options. Specify a name, the email address to receive a notification² and schedule options.

It's possible to have more than one scheduled tasks. This allows you to schedule different parts of the site on a weekly basis. The saved scheduled task will appear in the tree and when saved the scheduled task will give information when it did run, or when it will be executed again.

Schedule info

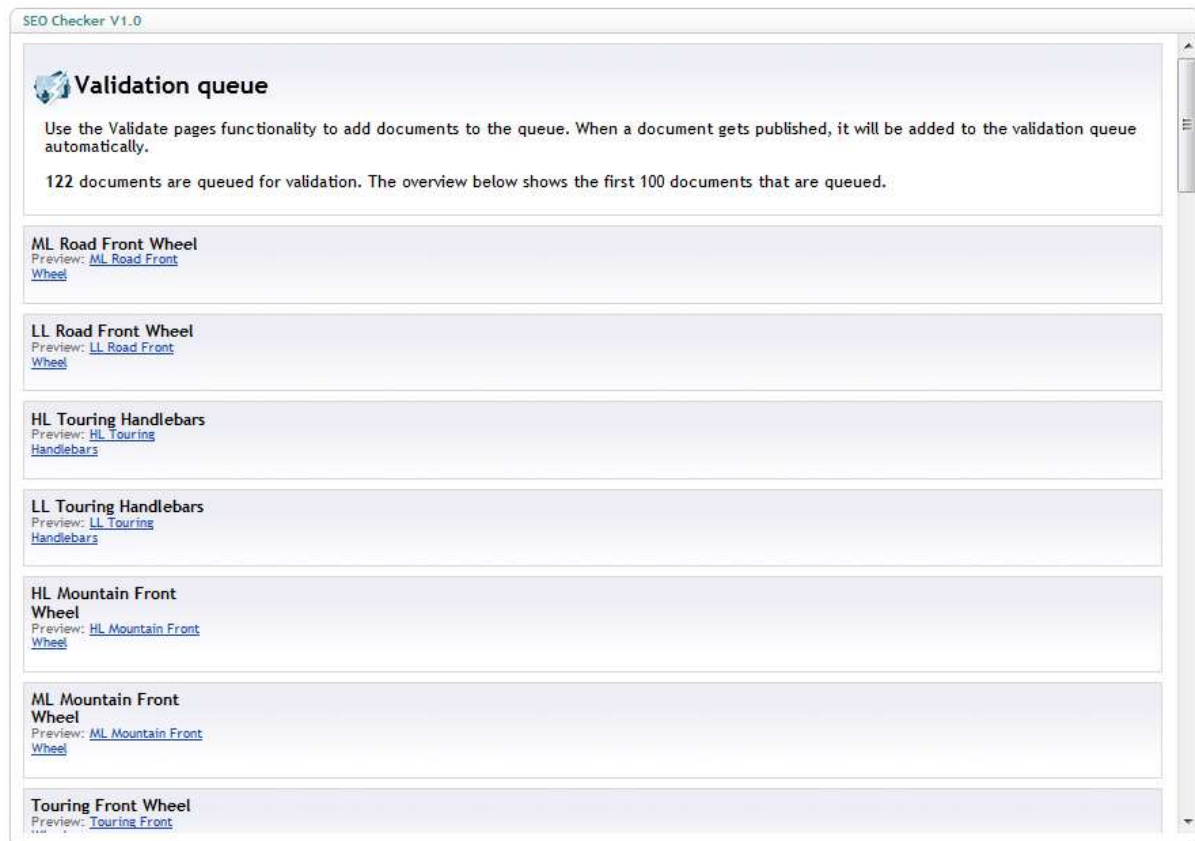
Next run time	13 Jun 2013 03:00
Last time executed	10 Jun 2013 15:18

Only published pages having a template assigned will be validated

² See chapter configuration how to specify email settings and how to change the lay-out of the email.

4.4 Validation queue

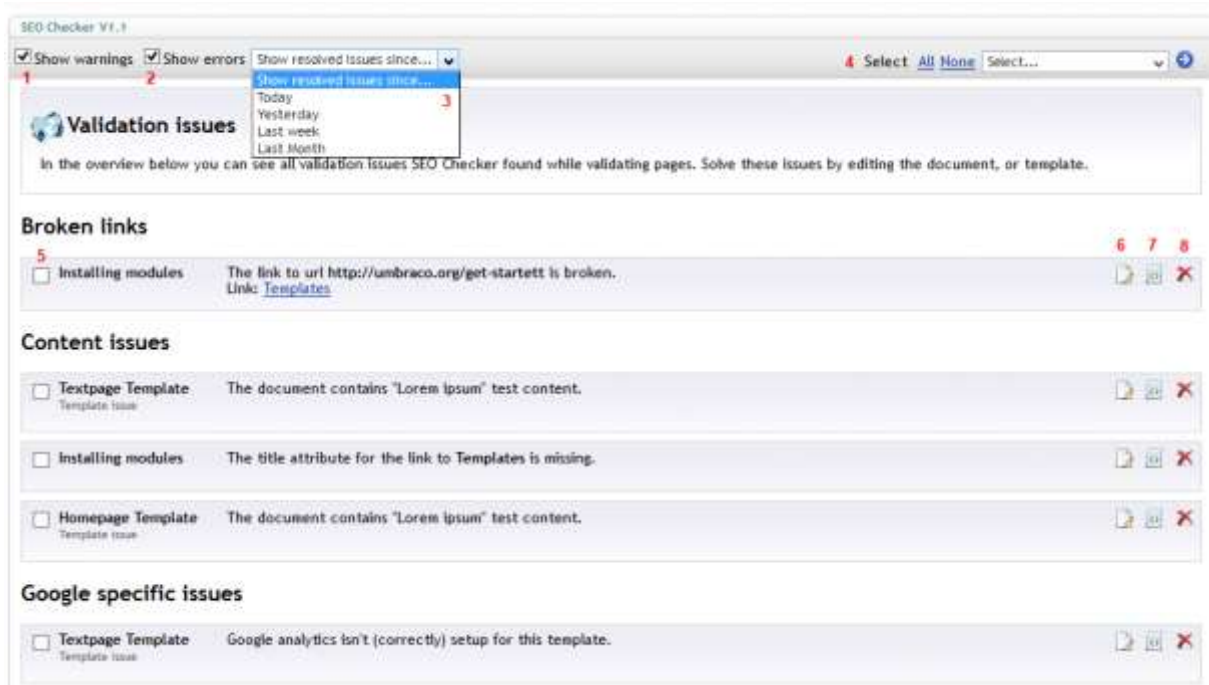
Pages will not be validated immediately, instead they will be added to the validation queue. Validation occurs on the background so you will not have to wait until the validation process is finished. When all pages are validated the result will be available in the Validation issues overview.



5 Issue overview

5.1 Validation issues

This overview will show all issues found when validating the page.



Options:

1. When selected warnings will included in the overview.
2. When selected errors will included in the overview.
3. When this option is selected solved items since the selected period are also displayed in the overview. Solved issues are displayed in strikethrough. In the example above the metadata issues are solved.
4. Allows you to (de)select all issues and select a bulk action. A bulk action is a single action you can apply to multiple items. The following bulk actions are available:
 - a. Revalidate selected items.
 - b. Delete selected items
 - c. Ignore selected items
5. Check items that you want to use in a bulk operation.
6. Opens the document
7. Opens the template assigned to the document. This option will only be available for Users that have access
8. Deletes the issue from the overview, or adds the item to the ignore list. See the delete and ignore validation issues for more details.

5.2 Inbound link errors

SEO checker comes with a module that will try to fix links automatically. The following situations can be fixed by the module automatically:

- Links to old pages that are renamed will be redirected to the new page. For example if you had a page **Modules** and you renamed it to **Umbraco Modules**, the url <http://mydomain.com/modules.aspx> will be redirected to <http://mydomain.com/umbraco-modules.aspx>. This will also work with extension less url's.
- Links to pages with the wrong extension will be redirected to the correct url. For example if people are linking to a page using <http://mydomain.com/umbraco-modules.php> the request will be redirected automatically to <http://mydomain.com/umbraco-modules.aspx> (or a directory url when you use extension less url's). This can also be handy when a site is migrated from a php, or other non .net platform to automatically redirect to the correct page. The following extensions are supported:
 - o .asp
 - o .cfm
 - o .htm
 - o .html
 - o .jsp
 - o .php
 - o .xhtml

When SEO Checker can't fix the link automatically it will be logged. The inbound link errors overview will show you all logged inbound link errors.

The screenshot shows the 'Inbound link errors' section of the SEO Checker V1.6 interface. It includes a search bar for URLs, a table of broken links, and a summary of the number of errors.

Inbound link errors

In the overview below you can fix url's pointing to your site that throw a page not found error (404). Simply fix the broken links by pointing them to a new page or delete them from the overview.

Search url's: Search 4

6	Url	Redirect to	Last Referer	Accessed	Last time accessed	2	3
☐	The link to getting-started is broken.	Getting started Delete Choose...		1 time(s)	03/07/2014 09:28:00		
☐	The link to modules is broken.	Installing modules Delete Choose...		1 time(s)	03/07/2014 09:29:00		

This overview gives you the following options:

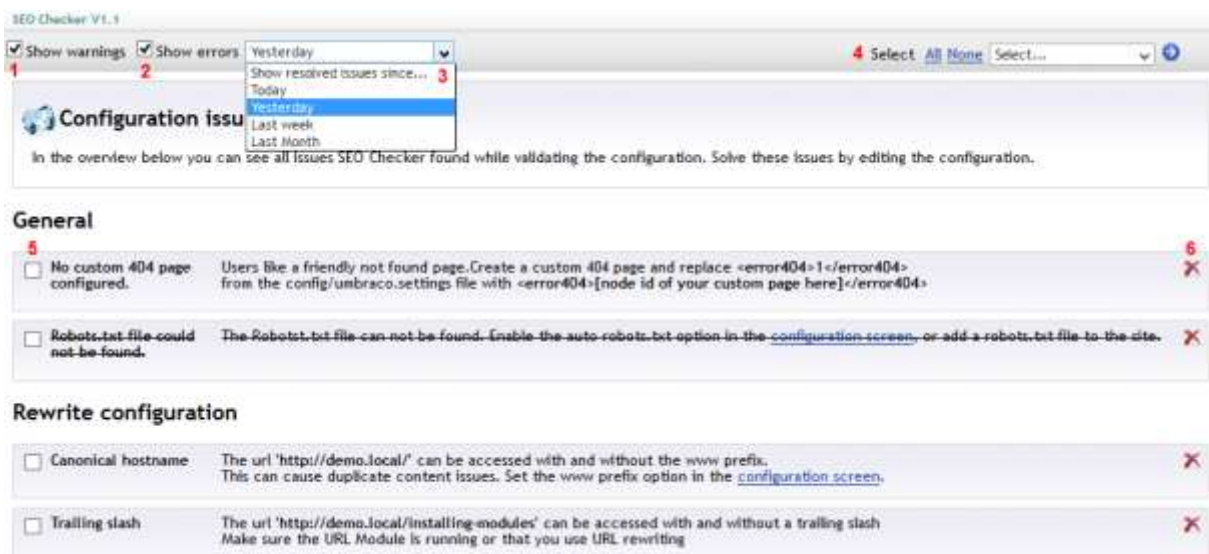
1. Pick a node where the page should redirect to
2. Select a redirect via advanced options (see also chapter redirects)
3. Deletes the issue from the overview, or adds the item to the ignore list. See the delete and ignore validation issues for more details.
4. Search for a specific redirect
5. Select options to show in the overview
6. Check items that you want to use in a bulk operation.
7. Allows you to (de)select all issues and select a bulk action. A bulk action is a single action you can apply to multiple items. The following bulk actions are available:
 - a. Assign selected items to a single node.
 - b. Delete selected items
 - c. Ignore selected items
8. Save selected options.

When you picked a page the next time a visitor requests that page the request will be redirected to the new selected page.

Redirects can be edited later using the Redirect manager.

5.3 Configuration errors

This overview will show all issues found in the configuration of the website.

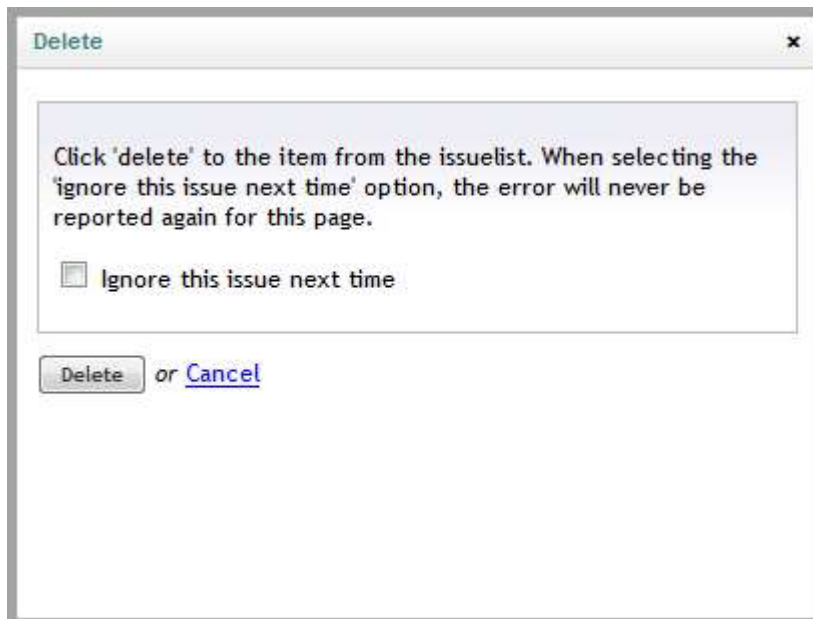


1. When selected warnings will included in the overview.
2. When selected errors will included in the overview.
3. When this option is selected solved items since the selected period are also displayed in the overview. Solved issues are displayed in strikethrough. In the example above the robots.txt issue is solved.
4. Allows you to (de)select all issues and Ignore all selected items
5. Check items that you want to use in a bulk operation.

6. Deletes the issue from the overview, or adds the item to the ignore list. See the delete and ignore validation issues for more details.

5.4 Delete and ignore validation issues

Using the delete button in the various overviews you can delete an issue from the overview. This will show the following dialog.



When you hit “delete” directly it will just remove the item from the overview. Next time validation occurs the issue will be reported again. When you check the “Ignore this issue next time” checkbox the item will be added to the ignore list and the issue will not be reported again next time validation occurs. Use the Ignore list overviews to see which items are ignored.

5.5 Bulk Actions

Every selected bulk action will open a Dialog, in the example below we have selected the redirect selected items option from the inbound link errors overview. When we pick a single node and hit the redirect button all selected items will be redirected to this document in the future.



6 Redirect manager

Using the redirect manager you can manage all redirects on your website. The manager will open with an overview of all the redirects on your website.

SEO Checker V1.4

☒ Show content ☒ Show media

Select [All](#) [None](#)

Manage redirects

Find redirects to certain pages and edit or delete them.

Search redirects

	Url	Redirect to node	Last Referrer	Accessed	Last time accessed	
☐	/go-further	/go-further/		2 time(s)	07/10/2013 14:37:00	
☐	/media/1001/duug_logo_orange_rgb_xxl.jpg	/media/1001/duug_logo_orange_rgb_xxs.jpg		2 time(s)	07/10/2013 14:35:00	
☐	/media/1004/protected.pdf	/media/1018/public.pdf		2 time(s)	09/10/2013 09:43:00	
☐	/media/1009/protected.pdf	/media/1019/logintemplateformasterpage.txt		1 time(s)	09/10/2013 11:37:00	
☐	/installing-modulesz	/installing-umbraco-modules/		1 time(s)	02/10/2013 10:30:00	
☐	/installing-modules	/installing-umbraco-modules/		1 time(s)	07/10/2013 15:51:00	
☐	/installing-umbraco-modules.php	/installing-umbraco-modules/		1 time(s)	08/10/2013 11:17:00	

The overview will show you the source url and where to redirect to. When the url you redirect from has domain, or querystring information then the url will only redirect when the domain, or querystring is used. This allows you to redirect to different pages when you have a single page coming from a different domain or containing a different querystring. When available it will show you the last referrer also.

The overview will show how many times the page has been accessed and what the last date/time was when the redirect was used. This allows you to delete redirects over time.

Options are:

- Edit
- Delete

It's also possible to select multiple items and perform a bulk action. Available bulk actions are:

- Redirect selected items
- Delete selected items
- Ignore selected items

6.1 Create a new redirect

To create a new redirect right click on the Redirect manager node and select the create action. This will open a default Create dialog. Specify the url you want to redirect from and select Create. This will show the following edit screen. SEO Checker will determine if the redirect is a content, or media redirect based on the file extension.

Basic settings

Old url ¹
The old redirect where you want to redirect from.

Redirect to node ² [Installing modules](#) [Delete](#) [Choose...](#)
Specify a node you want to redirect to. If the page you want to redirect to can't be selected via a node use the Redirect to Url textbox.

Or redirect to url
Specify the url you want to redirect to. Only use this when the page you want to redirect to can't be selected via the node picker!

Advanced settings

Wildcard url ³ ☐
When selected all underlying pages of the old url will also map to the redirected node. If url was /news then oews/newsitem will also redirect to the selected node.

Domain ⁴
Only redirect when the url comes from the specified domain

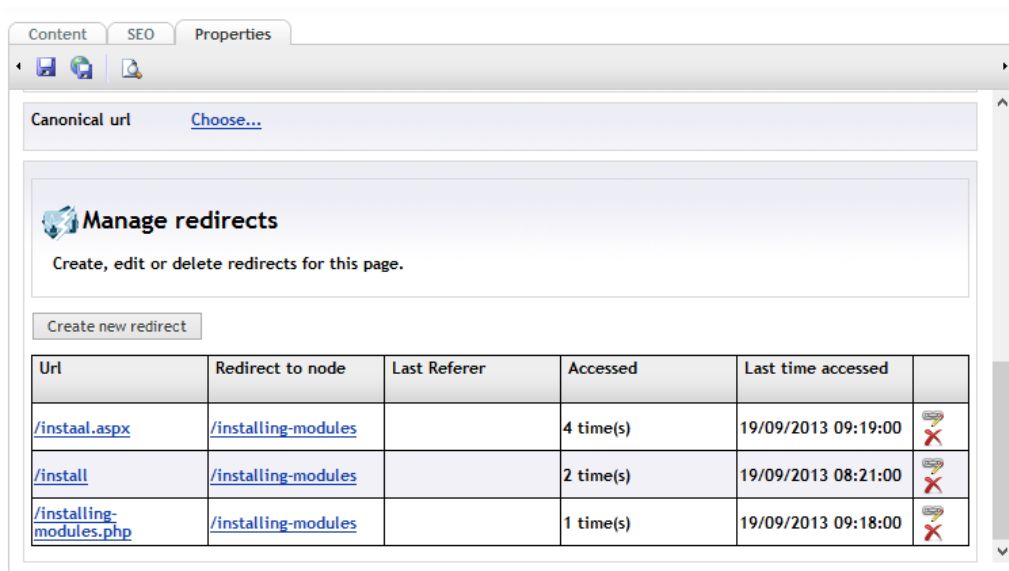
Querystring ⁵
Only redirect when the querystring from the url matches the specified querystring parameter(s)

1. The old url you want to redirect from.
2. The node or url you want to redirect to³.
3. Check wildcard url checkbox When you want all items underneath this path to redirect to this page. For this example modules/seochecker will also redirect to installing modules when wildcard url is checked. The wildcard url wil be displayed as modules/* in the overview
4. Only redirect when the request is for the specified domain
5. Only redirect when the request contains the specified querystring.

³ For content redirects the picker will open as content picker, otherwise the media picker will be shown.

6.2 Redirect data type

The complete manager is also available as a data type. When used as data type it will only show the redirects for the page the data type is used on!



Url	Redirect to node	Last Referer	Accessed	Last time accessed	
/install.aspx	/installing-modules		4 time(s)	19/09/2013 09:19:00	
/install	/installing-modules		2 time(s)	19/09/2013 08:21:00	
/installing-modules.php	/installing-modules		1 time(s)	19/09/2013 09:18:00	

6.3 Redirect Import/Export

Users that have access to the settings options of SEO Checker can use the redirect import/export functionality.

6.3.1 Redirect import

Right click on the redirect manager node and select import redirects. Select the type of file you want to import (Excel or CSV) and select upload. The following screen will be shown.

SEO Checker V1.4



Import redirects

Import redirects from a Csv or Excel file into this Umbraco environment.

Datasource options 1

Select the file type
Select the type of file you want to import the redirects from
Excel file

Select a file to import
Browse...
No file selected.
Selected file: SEOChecker-redirect-export-7.xlsx

Select Excel worksheet
Redirects

Field mapping 2

Source url column
Choose the column from the datasource you want to use as the url you want to redirect from.
Source

Target url column
Choose the column from the datasource you want to use as the url you want to redirect to.
Target

	A	B
1	Source	Target
2	/media/123/duug_logo.jpg	/media/1003/duug_logo_orange_rgb_xxs.jpg
3	/ga-verder	/go-further/
4	/test1	/kan/niet/vinden/

Mapping options

Try to map target url against a node ☒ 3
When checked the import process will try to map the target url against a node.

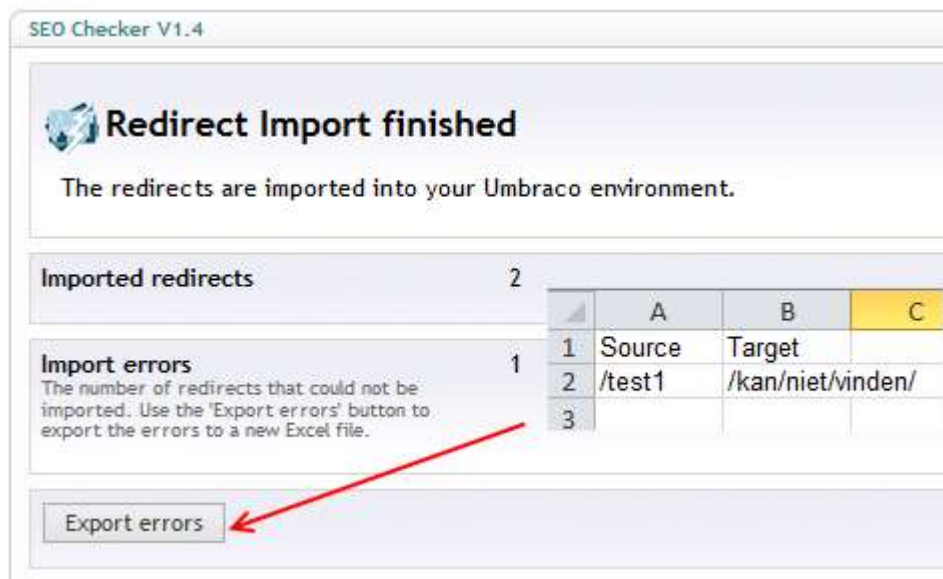
Only import when target url matches node ☒ 4
When checked the node will only be imported when it can be mapped against a node.

Import

1. Set the data source options. Select the worksheet in case of excel, or a different file.
2. Select the column names you want to map against the source and target redirect columns.
3. Source and target columns need to be specified as url's. During import SEO Checker will try to map the target column against a node when "try map target url against a node" option is enabled.
4. When enabled SEO Checker will only import redirects that can be mapped against a node. When the redirect cannot be mapped it will be reported after the import.

Append /* to the url in the source column when you want to import a wildcard url

The import button will start the import process and will report how many redirects got imported and will report the errors. When errors occurred during import you can export the redirects to an excel file to fix those and import them again.



6.3.2 Export redirects

Right click on the Redirect manager and select Export redirects to export redirects from the Umbraco environment to an Excel file, or select an xml sitemap file to create an Excel export file.

When selecting Redirects as export option all logged redirects and/or broken links will be exported. When selecting XML-Sitemap you can specify the url of an XML-Sitemap. This doesn't have to be the xml sitemap of the Umbraco installation but can also be an XML Sitemap of another site. This makes it ideal to use the xml Sitemap export for Site migrations.

Once you select export an Excel file is created with a Source and Target column.

Wildcard url's will be exported with /* as prefix

6.4 Automatically fix broken links after url change

When a content or media item name changes this normally results in a 404 error when the url to that content, or media item is requested. SEO Checker keeps track of the changes when you change the names and will automatically create a redirect on the first request after the change is made.

7 SEO Checker configuration

7.1 Configuration

To configure the default settings for SEO Checker go to the SEO Checker section into Umbraco, open the settings tree and select configuration. This page allows you to set the default configuration options

7.1.1 Triggers

When the “Validate a page when published” option is selected. Pages will be validated after they are published.

7.1.2 XML Sitemap

SEO Checker can create a dynamic XML Sitemap for you that you can submit to Google, Bing etc to make it easier for search engines to crawl your website. When enabled you can view the XML Sitemap in your browser using <http://domain.com/sitemap.xml> or use the Preview link on the configuration page. When the exclude **umbracoNaviHide** option is selected all pages containing the **umbracoNaviHide** property set to true will be excluded from the sitemap.

Specific sitemap option per document type can be set on the document type configuration. It's also possible to add a property with the alias **seoXmlSiteMapHide** to the page. This needs to be a true/false datatype. When the value is true the node will be excluded from the sitemap.

7.1.3 Robots.txt

SEO Checker can create a dynamic robots.txt file for you. Robots.txt files give search engines information what to crawl or what to exclude when crawling your website. When the XML Sitemap option is enabled, the location of the sitemap will be added to the robots.txt file. Below the XML Sitemap that will be generated by default:

```
# SEO Checker for Umbraco
Sitemap: http://www.soetemansoftware.nl/sitemap.xml
User-Agent: *
Disallow: /aspnet_client/
Disallow: /bin/
Disallow: /config/
Disallow: /data/
Disallow: /macroScripts/
Disallow: /umbraco/
Disallow: /umbraco_client/
Disallow: /usercontrols/
Disallow: /xslt/
```

When you already have a robots.txt file this file will be picked up automatically. When you want to link to the sitemap xml use:

```
Sitemap: http:// {HTTP_HOST}/sitemap.xml
```

{HTTP_HOST} will be replaced with the domain name.

You can change the robots.txt file per site on the domain settings.

7.1.4 URL Rewriting

By enabling Url rewriting SEO Checker will make sure multiple versions of a single url gets redirected to the correct url. For example <http://www.soetemansoftware.nl/products/> can be redirected to <http://soetemansoftware.nl/products/> or vice versa.

URL Rewriting is enabled by default and comes with two options:

- WWW prefix. Specify if you want to use www prefix or not
- Trailing slash. Specify if the url needs to be with or without a trailing slash⁴

The following situations will cause a redirect to the unique url:

Situation	Requested url	Redirected url
Site is configured to use www prefix	http://mysite.com/	http://www.mysite.com/
Site is configured to not use www prefix	http://www.mysite.com/	http://mysite.com/
Site is configured to use directory url's and trailing slash	http://mysite.com/contact.aspx	http://mysite.com/contact/
Site is configured to use directory url's and not trailing slash	http://mysite.com/contact/	http://mysite.com/contact
Site is configured to not use directory url's	http://mysite.com/contact/	http://mysite.com/contact.aspx
Request to the homepage node (/home/)	http://mysite.com/home/	http://mysite.com/
Requested url contains upper and lowercase characters	http://mysite.com/CONTACT/	http://mysite.com/contact/

It's recommended to use URL Rewriting. Only disable this option if you configured URL rewriting yourself using a URL Rewriting module.

7.1.5 General settings

Using the general settings you can configure the following items:

- Keyword selection tool. By default the Google adwords url is configured to use as the keyword selection tool. You can change the url here
- When show Template errors for editors is enabled, issues found in templates will be reported to content editors also.

7.2 Redirect settings

Specify the redirect and op

- Store domain for not found url's. When enabled the domain will be stored in the not found information. This allows you to specify multiple targets for a url coming from different

⁴ Only available when the site uses directory url's

domains such as www.soetemansoftware.nl/products and support.soetemansoftware.nl/products

- Store querystring for not found urls. When enabled the querystring information from the request will be stored in the not found information. This allows you to specify multiple targets for a url containing different querystring such as www.soetemansoftware.nl/page.php?id=4 and www.soetemansoftware.nl/page.php?id=5
- Forward querystring. When enabled the querystring information will be forwarded to the page where the request is redirected to.
- Redirect when node exists. When enabled the page will always be redirected to the configured redirect page. Even when the original node exists in Umbraco.

7.3 Domain settings

Using the domain settings you can specify the canonical url, a site and language specific not found page and customize the robots.txt file per site/rootnode. For editing the robots.txt file. Right click the rootnode and select edit robots.txt file. Canonical domain

With canonical domain you can set the preferred domain when a site can be found on multiple domains. Make sure to enter the production domain here, even when the site is not live yet.



The screenshot shows a form titled "Canonical domain". Below the title is a text box containing "seochecker-demo.com". To the left of the text box, there is explanatory text: "This domain will be used to generate the Canonical meta tag. Make sure to enter the production domain of the website." Below this text is a note: "Leave empty if you don't want to use Canonical domains."

The given domain will be used in the metadata library to generate the metadata of the page.

7.3.1 Not found page

The not found page settings allows you to specify a not found page without modifying config files. You can also specify a not found page per site and per language. The not found page will be used for .aspx as non .aspx extensions such as .php etc.



The screenshot shows a form titled "English (United States)". Below the title is a text box containing "Page not found". To the right of the text box are two links: "Delete" and "Choose...". Below the text box is a note: "Specify a not found page the English (United States) language."

7.4 Document type settings

7.4.1 Default SEO Properties

Using default SEO Properties you can specify default values for SEO Title and SEO description. When the page is missing the SEO title or description info these mapped properties will be used. The title will be copied. The description will take the first paragraph from the mapped property.

The screenshot shows two configuration panels. The top panel, 'Default title properties', explains that when the title is empty, the checker uses selected properties to render the title tag, with 'Node Name' being a recommended property. It features a list box with 'Body text' and 'Site Description', and another list box with 'Site Name' and '@nodeName'. Between them are '>>' and '<<' buttons. To the right are up and down arrow buttons for sorting. The bottom panel, 'Default description tag properties', explains that when the description tag is empty, the checker uses selected properties to render the meta description tag, noting that only the first paragraph will be used. It features a list box with '@nodeName' and 'Site Name', and another list box with 'Site Description' and 'Body text'. Similar '>>', '<<', and sort buttons are present.

Using the sort buttons it's possible to change the order of the default properties. The first property containing data will be used.

7.4.1.1 Title template

It's also possible to specify a template for the SEO Title. This template will be used to render the <title> tag in snippet preview and on the page⁵

The screenshot shows the 'Title template' configuration. On the left, a text area contains a sample template: '[#@seoTitle] | SEO Checker' and '[#@seoTitle] | [SiteName] © 2012'. Below this, 'Placeholder syntax' is explained: '[# propertyAlias]' and '[\$ propertyAlias] (recursive)'. 'Fixed properties' are listed: '@nodeName (Name of page)' and '@seoTitle (the seo title property)'. On the right, a preview box shows the rendered output: '[#@seoTitle] | [SiteName] © 2013'.

The template can contain fixed text and placeholders, placeholders have the same syntax as you would use in Umbraco Macro's:

- [#Property alias] will get the property from the current document
- [\$property alias] recursive look up the property

The following fixed properties can be used in the template:

- @nodeName contains the name of the page
- @seoTitle contains the value of the SEO title property

When using the above template configuration the <title> tag will be rendered as:

```
<title>Simple website | Simple website © 2013</title>
```

⁵ This requires you use the SEOChecker library methods to render the Meta data

7.4.2 Robot settings

Besides blocking specific areas of a site for search engines using the robots.txt file you can specify search engines settings on document type level also. Per document type you can specify the following settings:

- Robots Index. Helps search engines indicate if documents based on this document type need to be in the index.
 - o index will include the document,
 - o noindex will not include the document.
- Robots Follow. Helps search engines indicate if it needs to follow links for documents based on this document type:
 - o follow will follow the links
 - o nofollow will not follow the links

7.4.3 XML Sitemap settings

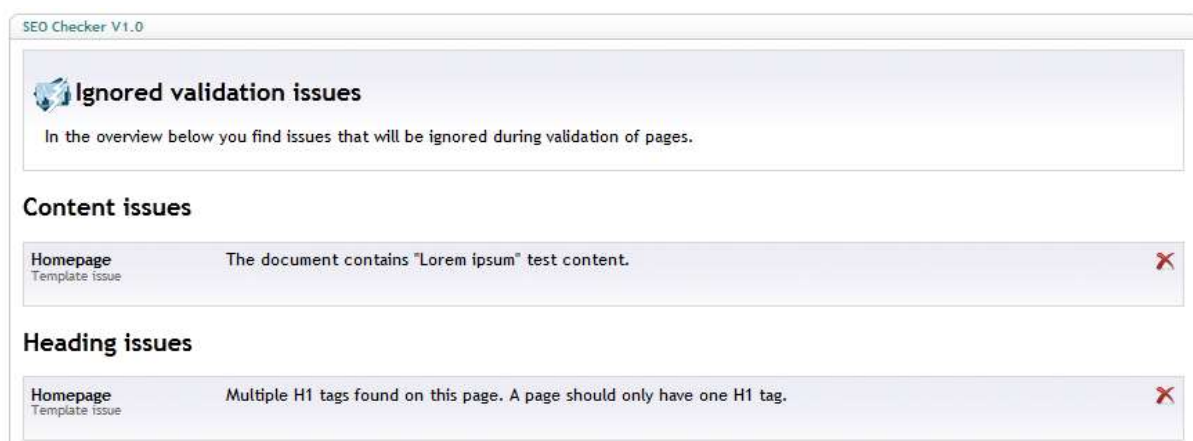
Per document type you can specify the following XML Sitemap settings:

- Exclude in XML Sitemap. When checked pages based on this document type will not be rendered in the XML Sitemap.
- Sitemap priority. Specify the priority for the sitemap from 0.1 (lowest priority) to 1.0 (highest priority)
- Change frequency. Specify how frequently pages based on this document type will change.

7.5 Ignore list

Use the following ignore lists to show the items that will be ignored during validation:

- Ignored validation issues
- Ignored inbound link errors
- Ignored configuration issues



Use the delete option to remove an item from the ignore list. When you remove an item it will be validated again during the next validation process.

7.6 Email settings

7.6.1 Scheduled task emails

Specify the from address, from name, subject for the scheduled task email. You can also specify an XSLT that renders the body of the email. You can modify the standard email XSLT, or create a new one and assign it here. The following XML structure will be used to render the template.

```
<?xml version="1.0" encoding="utf-8"?>
<seoCheckerScheduledValidationResult>
  <name>
    <!--The name of the scheduled task-->
  </name>
  <rootNodeId>
    <!--The id of the selected root node-->
  </rootNodeId>
  <rootNodeName>
    <!--The name of the selected root node-->
  </rootNodeName>
  <validateChildren>
    <!--'true' when validating children also-->
  </validateChildren>
  <date>
    <!--The current date-->
  </date>
  <nextDate>
    <!--Datetime, of next validation-->
  </nextDate>
  <validationQueueCount>
    <!--The number of items in the validation queue when the task was executed-->
  </validationQueueCount>
</seoCheckerScheduledValidationResult>
```

It's possible to use XSLT extensions, all XSLT extensions that requires a page context will not work.

7.6.2 User notification emails

Specify the from address, from name, subject for the notification email. You can also specify an XSLT that renders the body of the email. You can modify the standard email XSLT, or create a new one and assign it here. The following XML structure will be used to render the template.

```
<?xml version="1.0" encoding="utf-8"?>
<seoCheckerNotificationResult>
  <user>
    <!--The name of the user to send the notification to-->
  </user>
  <validationIssuesCount>
    <!--The number of open validation issues when the task was executed-->
  </validationIssuesCount>
  <inboundLinkIssuesCount>
    <!--The number of open inbound link issues when the task was executed-->
  </inboundLinkIssuesCount>
  <configurationIssuesCount>
    <!--The number of open configuration issues when the task was executed-->
  </configurationIssuesCount>
  <validationQueueCount>
    <!--The number of items in the validation queue when the task was executed-->
  </validationQueueCount>
  <date>
    <!--The current date-->
  </date>
</seoCheckerNotificationResult>
```

It's possible to use XSLT extensions, all XSLT extensions that requires a page context will not work.

7.7 Configure the SEO Checker Data Type

By default the SEO Checker data type will show a field for SEO title and description. If you already have those properties on your document type you can use configuration of the document type to map those .

Edit datatype: SEO Checker

Name: SEO Checker

Render control: SEO Checker

Data Editor GUID: ee3b7251-b701-4643-86ce-02bf32f96d5c

Settings

1. Map title against property
When you already using a property for the title tag you can select the property here. SEO Checker needs this information to give good feedback!

2. Map keywords tag against property
When you already using a property for the keywords tag you can select the property here. SEO Checker needs this information to give good feedback!

3. Map description tag against property
When you already using a property for the description tag you can select the property here. SEO Checker needs this information to give good feedback!

4. Use keyword tag
Google does not use the keywords meta tag for web ranking, so only enable this if you want to optimize for other search engines that use the keyword tag.

1. Map SEO title against the existing SEO title property
2. Map SEO keywords against the existing SEO keywords property
3. Map SEO description against the existing SEO description property
4. Use the keywords meta tag (*Google and all major search engines ignore this meta tag*).

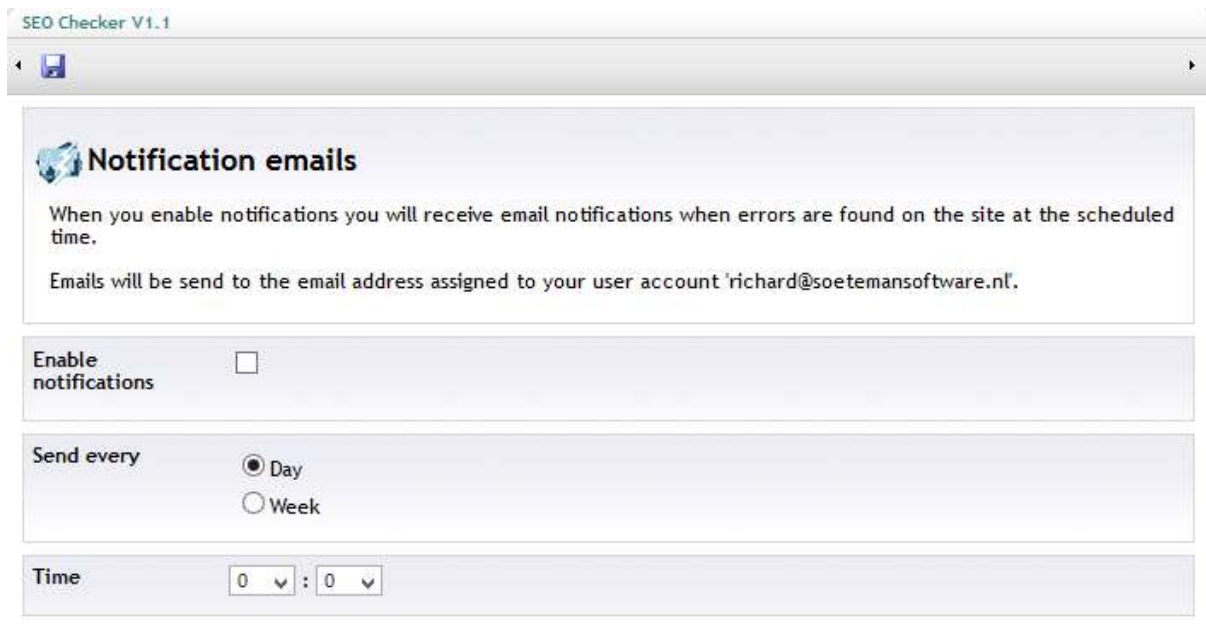
7.8 Additional data types

SEO Checker comes with two extra datatypes you can use to modify robots and XML sitemap settings on page level. Those are:

- SEOChecker robots, allows you to override configured Meta data for robots on a single page.
- SEOChecker XML Sitemap options, allows you to override configured XML sitemap settings on a single page.
- SEO Checker redirects manager. A data type to manage redirects to a specific node.

7.9 Notifications

SEO Checker allows users to receive notifications of logged issues on the site. Every user can setup notifications when the user has access to the SEO Checker section. The user can specify the day/time of the notification email to be sent.



Notification emails

When you enable notifications you will receive email notifications when errors are found on the site at the scheduled time.

Emails will be send to the email address assigned to your user account 'richard@soetemansoftware.nl'.

Enable notifications ☐

Send every ☒ Day ☐ Week

Time 0 : 0

By default the notification email⁶ will contain the following items:

- The number of validation issues
- The number of configuration issues
- The number of Inbound link errors
- The number of items in the validation queue

7.10 Web.config settings

In web.config you can specify a few additional Appsettings

7.10.1 Alternative license file location

Using the key SEOChecker.LicenseFilePath you can specify an alternative path where SEOChecker can find the license file.

Example:

```
<add key="SEOChecker.LicenseFilePath" value="~/config/seo-checker.lic" />
```

7.10.2 SEOChecker.ShowDomainNameInDomainSettings

When all root nodes have the same language codes it's impossible to see which item you are editing in domain settings. Apply this setting and the domain assigned to the node will be displayed in domain settings.

```
<add key="SEOChecker.ShowDomainNameInDomainSettings" value="false" />
```

⁶ See the configurations chapter how to modify the default email.

8 Library

SEO Checker comes with a small library to render Meta Tags in HTML. This is only necessary when you use the SEO Checker data type.

8.1 MVC

8.1.1 Render title

Renders the <title> tag based on SEO title value from the SEO Checker data type and configured template.

```
@{
var meta = Model.Content.GetPropertyValue<SEOChecker.MVC.MetaData>("seoChecker");
}
<title>@meta.Title</title>
```

8.1.2 Render description meta tag

Renders the <meta name="description"..... tag value based on SEO Description value from the SEO Checker data type, or the configured default property.

```
@{
var meta = Model.Content.GetPropertyValue<SEOChecker.MVC.MetaData>("seoChecker");
}
<meta name="description" content="@meta.Description" />
```

8.1.3 Render keywords meta tag

Renders the <meta name="keywords"..... tag based on SEO Description value from the SEO Checker data type, or the configured default property. This tag will only be rendered when SEO Checker data type is configured to use keywords.

```
@{
var meta = Model.Content.GetPropertyValue<SEOChecker.MVC.MetaData>("seoChecker");
}
<meta name="keywords" content="@meta.Keywords" />
```

8.1.4 Render robots meta tag

Renders the <meta name="robots"..... tag based on the configured robot settings.

```
@{
var meta = Model.Content.GetPropertyValue<SEOChecker.MVC.MetaData>("seoChecker");
}
<meta name="robots" content="@meta.Robots" />
```

8.1.5 Render all tags

Renders all above settings using a single statement

```
@{
var meta = Model.Content.GetPropertyValue<SEOChecker.MVC.MetaData>("seoChecker");
}
@meta.AllTags
```

8.2 Razor

8.2.1 Render title

Renders the <title> tag based on SEO title value from the SEO Checker data type and configured template.

```
@Html.Raw (SEOChecker.Library.Metadata.Current.Title)
```

8.2.2 Render description meta tag

Renders the <meta name="description"..... tag based on SEO Description value from the SEO Checker data type, or the configured default property.

```
@Html.Raw (SEOChecker.Library.Metadata.Current.Description)
```

8.2.3 Render keywords meta tag

Renders the <meta name="keywords"..... tag based on SEO Description value from the SEO Checker data type, or the configured default property. This tag will only be rendered when SEO Checker data type is configured to use keywords.

```
@Html.Raw (SEOChecker.Library.Metadata.Current.Keywords)
```

8.2.4 Render robots meta tag

Renders the <meta name="robots"..... tag based on the configured robot settings.

```
@Html.Raw (SEOChecker.Library.Metadata.Current.Robots)
```

8.2.5 Render all tags

Renders all above settings using a single statement

```
@Html.Raw (SEOChecker.Library.Metadata.Current.RenderTags ())
```

8.3 XSLT

SEO checker comes with an XSLT extension to access the library methods. This Extension can be found in seoChecker.Metadata.

8.3.1 Render title

Renders the <title> tag based on SEO title value from the SEO Checker data type and configured template.

```
<xsl:value-of select="seoChecker.Metadata:Title()" disable-output-escaping="yes"/>
```

8.3.2 Render description meta tag

Renders the <meta name="description"..... tag based on SEO Description value from the SEO Checker data type, or the configured default property.

```
<xsl:value-of select="seoChecker.MetaData:Description()" " disable-output-escaping="yes"/>
```

8.3.3 Render keywords meta tag

Renders the **<meta name="keywords".....** tag based on SEO Description value from the SEO Checker data type, or the configured default property. This tag will only be rendered when SEO Checker data type is configured to use keywords.

```
<xsl:value-of select="seoChecker.MetaData:Keywords()" " disable-output-escaping="yes"/>
```

8.3.4 Render robots meta tag

Renders the **<meta name="robots".....** tag based on the configured robot settings.

```
<xsl:value-of select="seoChecker.MetaData:Robots()" " disable-output-escaping="yes"/>
```

8.3.5 Render all tags

Renders all above settings using a single statement

```
<xsl:value-of select="seoChecker.MetaData:RenderTags()" " disable-output-escaping="yes"/>
```

8.4 .Net Usercontrol

8.4.1 Render title

Renders the **<title>** tag based on SEO title value from the SEO Checker data type and configured template.

```
<%=SEOChecker.Library.MetaData.Current.Title %>
```

8.4.2 Render description meta tag

Renders the **<meta name="description".....** tag based on SEO Description value from the SEO Checker data type, or the configured default property.

```
<%=SEOChecker.Library.MetaData.Current.Description %>
```

8.4.3 Render keywords meta tag

Renders the **<meta name="keywords".....** tag based on SEO Description value from the SEO Checker data type, or the configured default property. This tag will only be rendered when SEO Checker data type is configured to use keywords.

```
<%=SEOChecker.Library.MetaData.Current.Keywords %>
```

8.4.4 Render robots meta tag

Renders the **<meta name="robots".....** tag based on the configured robot settings.

```
<%=SEOChecker.Library.Metadata.Current.Robots %>
```

8.4.5 Render all tags

Renders all above settings using a single statement

```
<%=SEOChecker.Library.Metadata.Current.RenderTags () %>
```

8.5 MVC

If you are using MVC you can get the meta data also using the `GetPropertyValue` method. This requires an extra reference to `SEOChecker.Mvc.dll`.

`@model.Content.GetPropertyValue<Metadata>` will give you the typed Metadata object for the given property alias.

```
@{
    var metaData =
Model.Content.GetPropertyValue<Metadata>("seoChecker");
}
<title>@metaData.Title</title>
<meta name="keywords" content="@metaData.Keywords" />
<meta name="description" content="@metaData.Description" />

<link rel="canonical" href="@metaData.CanonicalUrl"/>
<meta name="robots" content="@metaData.Robots" />
```

8.6 Special properties

SEO Checker comes with a few special property aliases you can use on your document type to exclude from Sitemap specify canonical urls' etc.

Alias	Type	Description
seoCanonicalUrl	Content picker	Allows you to specify a canonical url. This url will be used to render the canonical tag using the library.
seoXmlSiteMapHide	True/false	When true the node will not be shown in the XML Sitemap.
seoExcludeValidation	True/false	When true SEO Checker will not validate the page using the Validate pages option in the SEO Checker section. Validation on the SEO Checker data type will still occur.
seoForceHTTPS	True/False	Forces SEO Checker to render the HTTPS variant in XML Sitemap

9 Extend SEO Checker

It's possible to extend the functionality of SEOChecker by referencing the SEOChecker.Extensions project.

9.1 URL Rewrite provider

The url rewriter provider is used to rewrite all canonical issues for a website request to a single canonical url. Check the url rewriting chapter how to use the functionality. It's also possible to extend the functionality by implementing your own logic.

To write your own URLRewrite provider add a reference to the SEOChecker.Extensions dll and create a class that derives from the SEOChecker.Extensions.Providers.UrlRewriteProvider.

UrlRewriteProviderBase class. Override the RewriteUrl method to method to implement your rewriting logic. Modify the UrlBuilder object that gets inserted to rewrite the url.

In the sample below we redirect every website request for http to https.

```
/// <summary>
/// Forces Redirect to HTTPS
/// </summary>
public class HTTPSRedirect : UrlRewriteProviderBase
{
    public override void RewriteUrl(UrlBuilder builder, UrlRewriteConfiguration config)
    {
        if (builder.Scheme.Equals("http", StringComparison.InvariantCultureIgnoreCase))
        {
            builder.Scheme = "https";
        }
    }
}
```

10 Manual Installation

If you renamed the Umbraco folder or for some reason can't give the installer sufficient rights to create tables in the database, or the sufficient rights to modify the following folders /bin, /umbraco you need to install do a manual install of SEO Checker.

10.1 Manual installation of files

- Open the folder in the zip file.
- Copy **.dll** files to the folder **/bin**
- Create folder **/umbraco/plugins /SEOChecker**
- Create folder **/umbraco/plugins /SEOChecker /Config**
- Create folder **/umbraco/plugins /SEOChecker /css**
- Create folder **/umbraco/plugins /SEOChecker /Images**
- Create folder **/umbraco/plugins /SEOChecker /Pages**
- Create folder **/umbraco/plugins /SEOChecker /scripts**
- Create folder **/umbraco/plugins /SEOChecker /Usercontrols**
- Create folder **/umbraco/plugins /SEOChecker /Images/Actions**
- Create folder **/umbraco/plugins /SEOChecker /Images/Content**
- Create folder **/umbraco/plugins /SEOChecker /Images/Tray**
- Create folder **/umbraco/plugins /SEOChecker /Images/Tree**
- Create folder **/umbraco/plugins /SEOChecker /Pages/Dialogs**
- Create folder **/umbraco/plugins /SEOChecker /Pages/IgnoreList**
- Create folder **/umbraco/plugins /SEOChecker /Pages/Issues**
- Create folder **/umbraco/plugins /SEOChecker /Pages/Settings**
- Copy **SEOChecker.config** from zip file to the **/umbraco/plugins /SEOChecker /Config** folder
- Copy **seochecker-notification.xslt** from zip file to the **/XSLT** folder
- Copy **seochecker-scheduledtask.xslt** from zip file to the **/XSLT** folder
- Copy **.css** files from zip file to the **/umbraco/plugins /SEOChecker /css** folder
- Copy **navigation.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Actions** folder
- Copy **bug.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Actions** folder
- Copy **delete.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Actions** folder
- Copy **document-pencil.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Actions** folder
- Copy **chain-edit.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Actions** folder
- Copy **script-code.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Actions** folder
- Copy **hammer-screwdriver.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Content** folder
- Copy **seochecker.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Content** folder
- Copy **application-pencil.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **blue-document-text.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **calendar.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **calendar-pencil.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **chain-exclamation.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder

- Copy **documents-stack.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **inbox-arrow.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **light-bulb.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **magnifier-zoom.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **mail.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **mail-pencil.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **mails.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **script-codeTree.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **server-pencil.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **chain-pencil.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **table-pencil.png** from zip file to the **/umbraco/plugins /SEOChecker /Images/Tree** folder
- Copy **bulkaction.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages/Dialogs** folder
- Copy **ConfirmDelete.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages/Dialogs** folder
- Copy **ConfirmDeleteIgnorelist.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages/Dialogs** folder
- Copy **IgnoredConfigurationIssues.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages/IgnoreList** folder
- Copy **IgnoredInboundLinks.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages/IgnoreList** folder
- Copy **ConfigurationIssues.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages/Issues** folder
- Copy **InboundLinks.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages/Issues** folder
- Copy **ValidationIssues.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages/Issues** folder
- Copy **Config.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages/Settings** folder
- Copy **DocumentTypeSettings.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages/Settings** folder
- Copy **emailsettings.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages/Settings** folder
- Copy **LicenseError.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages** folder
- Copy **Validate.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages** folder
- Copy **ValidationQueue.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages** folder
- Copy **ScheduleValidation.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages** folder
- Copy **notifications.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages** folder
- Copy **Redirects.aspx** from zip file to the **/umbraco/plugins /SEOChecker /Pages/Redirects** folder

- Copy **seochecker_advancedsettingspanel.js** from zip file to the **/umbraco/plugins /SEOChecker /Scripts** folder
- Copy ***.ascx** from zip file to the **/umbraco/plugins /SEOChecker / Usercontrols** folder

10.2 Alternative Configuration file location

By default the SEOChecker.config file is stored in the location /Umbraco/plugins/config/ In case you want to change this you can add the **seoChecker.ConfigFile** appSettings key to the web.config file, where the value needs to contain the relative path to the configuration file. When this key is found this location will be used.

```
<add key="seoChecker.ConfigFile" value="~/App_Data/SEOChecker.config" />
```

10.3 Install database scripts

10.3.1 SQL Server

Install the tables for SQL Server executing the following script.

```
CREATE TABLE [dbo].[SEOChecker_ValidationIssues] (
    [IssueId] [int] IDENTITY(1,1) NOT NULL,
    [DocumentID] [int],
    [ErrorCode] [nvarchar](50) NOT NULL,
    [ValidatorAlias] [nvarchar](50) NOT NULL,
    [ErrorDescription] [nvarchar](1000) NULL,
    [Error] [nvarchar](250) NULL,
    [Template] [nvarchar](50) NULL,
    [Category] [nvarchar](50) NOT NULL,
    [ValidationType] [nvarchar](50) NOT NULL,
    [ErrorType] [nvarchar](20) NOT NULL,
    [PropertyAlias] [nvarchar](150) NULL,
    [DataTypeId] [uniqueidentifier] NULL,
    [CrawlDate] [SmallDateTime] NOT NULL,
    [ResolvedDate] [SmallDateTime] NULL,
    [Ignore] [bit] NOT NULL,
    CONSTRAINT [PK_SEOChecker_ValidationIssues] PRIMARY KEY CLUSTERED
(
    [IssueId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

CREATE TABLE [dbo].[SEOChecker_ConfigurationIssues] (
    [IssueId] [int] IDENTITY(1,1) NOT NULL,
    [ErrorCode] [nvarchar](50) NOT NULL,
    [ValidatorAlias] [nvarchar](50) NOT NULL,
    [ErrorDescription] [nvarchar](1000) NULL,
    [Error] [nvarchar](250) NULL,
    [ErrorType] [nvarchar](20) NOT NULL,
    [Category] [nvarchar](50) NOT NULL,
    [CrawlDate] [SmallDateTime] NOT NULL,
    [ResolvedDate] [SmallDateTime] NULL,
    [Ignore] [bit] NOT NULL,
    CONSTRAINT [PK_[SEOChecker_ConfigurationIssues] PRIMARY KEY CLUSTERED
(
    [IssueId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
```

```
CREATE TABLE [dbo].[SEOChecker_Queue] (
    [DocumentID] [int] NOT NULL,
    [QueuedDateTime] [smalldatetime] NOT NULL,
    CONSTRAINT [PK_SEOChecker_Queue] PRIMARY KEY CLUSTERED
(
    [DocumentID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

CREATE TABLE [dbo].[SEOChecker_PageNotFound] (
    [NotFoundId] [int] IDENTITY(1,1) NOT NULL,
    [Url] [nvarchar](500) NOT NULL,
    [DocumentID] [int],
    [Ignore] [bit] NOT NULL,
    CONSTRAINT [PK_SEOChecker_PageNotFound] PRIMARY KEY CLUSTERED
(
    [NotFoundId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

CREATE TABLE [dbo].[SEOChecker_Robots] (
    [NodeId] [int] NOT NULL,
    [UserAgent] [nvarchar](100) NOT NULL,
    [RequestDateTime] [smalldatetime] NOT NULL
) ON [PRIMARY]

CREATE TABLE [dbo].[SEOChecker_URLHistory] (
    [UrlHistoryId] [int] IDENTITY(1,1) NOT NULL,
    [NodeId] [int] NOT NULL,
    [ParentId] [int] NOT NULL,
    [UrlPart] [nvarchar](100) NOT NULL,
    CONSTRAINT [PK_SEOChecker_URLHistory] PRIMARY KEY CLUSTERED
(
    [UrlHistoryId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

CREATE TABLE [dbo].[SEOChecker_ScheduledValidation] (
    [Id] [int] IDENTITY(1,1) NOT NULL,
    [TaskName] [nvarchar](250) NOT NULL,
    [NotificationEmailAddress] [nvarchar](250) NOT NULL,
    [ExecuteEvery] [nvarchar](50) NOT NULL,
    [ExecuteDays] [nvarchar](50) NULL,
    [ExecuteHour] [int] NOT NULL,
    [ExecuteMinute] [int] NOT NULL,
    [ValidationRoot] [int] NOT NULL,
    [IncludeChildren] [bit] NOT NULL,
    [NextRun] [smalldatetime] NOT NULL,
    [LastTimeExecuted] [smalldatetime] NULL,
    CONSTRAINT [PK_SEOChecker_ScheduledValidation] PRIMARY KEY CLUSTERED
(
    [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

ALTER TABLE [dbo].[SEOChecker_Queue]
ADD [Priority] int NOT NULL

CREATE TABLE [dbo].[SEOChecker_Notifications] (
    [Id] [int] IDENTITY(1,1) NOT NULL,
    [NotificationUser] [nvarchar](250) NOT NULL,
```

```

[ExecuteEvery] [nvarchar](50) NOT NULL,
[ExecuteDays] [nvarchar](50) NULL,
[ExecuteHour] [int] NOT NULL,
[ExecuteMinute] [int] NOT NULL,
[NextRun] [smalldatetime] NOT NULL,
CONSTRAINT [PK_SEOChecker_Notifications] PRIMARY KEY CLUSTERED
(
    [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

ALTER TABLE [dbo].[SEOChecker_PageNotFound]
ADD
    [TimesAccessed] [int] NULL,
    [LastTimeAccessed] [smalldatetime] NULL,
    [RedirectUrl] nvarchar (500) NULL,
    [Referer] [nvarchar](500) NULL,
    [ContentType] [nvarchar](50) NULL,
    [Domain] [nvarchar](250) NULL,
    [QueryString] [nvarchar](250) NULL

Update SEOChecker_PageNotFound set Domain = '' where Domain is null
Update SEOChecker_PageNotFound set TimesAccessed = 0 where TimesAccessed is null
Update SEOChecker_PageNotFound set QueryString = '' where QueryString is null
Update SEOChecker_PageNotFound set ContentType = 'content' where ContentType is
null
Update SEOChecker_PageNotFound set RedirectUrl = '' where RedirectUrl is null

```

10.3.2 MySQL

Install the tables for MYSQL executing the following script.

```

CREATE TABLE seochecker_configurationissues (
    IssueId int(11) NOT NULL AUTO_INCREMENT,
    ErrorCode nvarchar(50) NOT NULL,
    ValidatorAlias nvarchar(50) NOT NULL,
    ErrorDescription nvarchar(1000) DEFAULT NULL,
    Error nvarchar(50) DEFAULT NULL,
    ErrorType nvarchar(20) NOT NULL,
    Category nvarchar(59) NOT NULL,
    CrawlDate datetime NOT NULL,
    ResolvedDate datetime DEFAULT NULL,
    Ignore bit(1) NOT NULL,
    PRIMARY KEY (IssueId)
)

CREATE TABLE seochecker_pagenotfound (
    NotFoundId int(11) NOT NULL AUTO_INCREMENT,
    URL nvarchar(500) NOT NULL,
    DocumentID int(11) DEFAULT NULL,
    Ignore bit(1) NOT NULL,
    PRIMARY KEY (NotFoundId)
)

CREATE TABLE seochecker_queue (
    DocumentID int(11) NOT NULL,
    QueuedDateTime datetime NOT NULL,
    PRIMARY KEY (DocumentID)
)

CREATE TABLE seochecker_robots (
    NodeId int(11) NOT NULL AUTO_INCREMENT,
    UserAgent nvarchar(100) NOT NULL,
    RequestDateTime datetime NOT NULL,
    PRIMARY KEY (NodeId)
)

```

```
)
CREATE TABLE seochecker_urlhistory (
  UrlHistoryId int(11) NOT NULL AUTO_INCREMENT,
  NodeId int(11) NOT NULL,
  ParentId int(11) NOT NULL,
  UrlPart nvarchar(100) NOT NULL,
  PRIMARY KEY (UrlHistoryId)
)

CREATE TABLE seochecker_validationissues (
  IssueId int(11) NOT NULL AUTO_INCREMENT,
  DocumentID int(11) DEFAULT NULL,
  ErrorCode nvarchar(50) NOT NULL,
  ValidatorAlias nvarchar(50) NOT NULL,
  ErrorDescription nvarchar(1000) DEFAULT NULL,
  Error nvarchar(250) DEFAULT NULL,
  Template nvarchar(50) DEFAULT NULL,
  Category nvarchar(50) NOT NULL,
  ValidationType nvarchar(50) NOT NULL,
  ErrorType nvarchar(20) NOT NULL,
  PropertyAlias nvarchar(150) DEFAULT NULL,
  DataTypeId char(36) DEFAULT NULL,
  CrawlDate datetime NOT NULL,
  ResolvedDate datetime DEFAULT NULL,
  Ignore bit(1) NOT NULL,
  PRIMARY KEY (IssueId)
)

CREATE TABLE SEOCHECKER_SCHEDULEDVALIDATION(
  Id int(11) NOT NULL AUTO INCREMENT,
  TaskName nvarchar(250) NOT NULL,
  NotificationEmailAddress nvarchar(250) NOT NULL,
  ExecuteEvery nvarchar(50) NOT NULL,
  ExecuteDays nvarchar(50) DEFAULT NULL,
  ExecuteHour int(11) NOT NULL,
  ExecuteMinute int(11) NOT NULL,
  ValidationRoot int(11) NOT NULL,
  IncludeChildren tinyint(1) NOT NULL,
  NextRun datetime NOT NULL,
  LastTimeExecuted datetime DEFAULT NULL,
  PRIMARY KEY (Id)
)

ALTER TABLE SEOChecker_QUEUE
ADD Priority int(11) NOT NULL

CREATE TABLE SEOCHECKER_NOTIFICATIONS(
  Id int(11) NOT NULL AUTO_INCREMENT,
  NotificationUser nvarchar(250) NOT NULL,
  ExecuteEvery nvarchar(50) NOT NULL,
  ExecuteDays nvarchar(50) DEFAULT NULL,
  ExecuteHour int(11) NOT NULL,
  ExecuteMinute int(11) NOT NULL,
  NextRun datetime NOT NULL,
  PRIMARY KEY (Id))

ALTER TABLE SEOChecker_QUEUE
ADD
  TimesAccessed int(11) NULL,
  LastTimeAccessed datetime NULL,
  Referer nvarchar (500) NULL,
  RedirectUrl nvarchar (500) NULL,
  ContentType nvarchar (50) NULL,
  Domain nvarchar (250) NULL,
  QueryString nvarchar (250) NULL
Update SEOChecker_PageNotFound set Domain = '' where Domain is null
Update SEOChecker_PageNotFound set TimesAccessed = 0 where TimesAccessed is null
Update SEOChecker_PageNotFound set QueryString = '' where QueryString is null
```

```
Update SEOChecker_PageNotFound set ContentType = 'content' where ContentType is null
Update SEOChecker_PageNotFound set RedirectUrl = '' where RedirectUrl is null
```

10.3.3 SQL CE

Install the tables for SQL CE executing the following scripts. You need to execute them one by one.

```
CREATE TABLE seochecker_configurationissues (
  IssueId int NOT NULL IDENTITY(1,1),
  ErrorCode nvarchar(50) NOT NULL,
  ValidatorAlias nvarchar(50) NOT NULL,
  ErrorDescription nvarchar(1000) DEFAULT NULL,
  Error nvarchar(50) DEFAULT NULL,
  ErrorType nvarchar(20) NOT NULL,
  Category nvarchar(50) NOT NULL,
  CrawlDate datetime NOT NULL,
  ResolvedDate datetime DEFAULT NULL,
  Ignore bit NOT NULL,
  PRIMARY KEY (IssueId)
)
```

```
CREATE TABLE seochecker_pagenotfound (
  NotFoundId int NOT NULL IDENTITY(1,1),
  URL nvarchar(500) NOT NULL,
  DocumentID int DEFAULT NULL,
  Ignore bit NOT NULL,
  PRIMARY KEY (NotFoundId)
)
```

```
CREATE TABLE seochecker_queue (
  DocumentID int NOT NULL,
  QueuedDateTime datetime NOT NULL,
  PRIMARY KEY (DocumentID)
)
```

```
CREATE TABLE seochecker_queue (
  DocumentID int NOT NULL,
  QueuedDateTime datetime NOT NULL,
  PRIMARY KEY (DocumentID)
)
```

```
CREATE TABLE seochecker_robots (
  NodeId int NOT NULL IDENTITY(1,1),
  UserAgent nvarchar(100) NOT NULL,
  RequestDateTime datetime NOT NULL,
  PRIMARY KEY (NodeId)
)
```

```
CREATE TABLE seochecker_urlhistory (
  UrlHistoryId int NOT NULL IDENTITY(1,1),
  NodeId int NOT NULL,
  ParentId int NOT NULL,
  UrlPart nvarchar(100) NOT NULL,
  PRIMARY KEY (UrlHistoryId)
)
```

```
CREATE TABLE seochecker_validationissues (
```

```
IssueId int NOT NULL IDENTITY(1,1),
DocumentID int DEFAULT NULL,
ErrorCode nvarchar(50) NOT NULL,
ValidatorAlias nvarchar(50) NOT NULL,
ErrorDescription nvarchar(1000) DEFAULT NULL,
Error nvarchar(250) DEFAULT NULL,
Template nvarchar(50) DEFAULT NULL,
Category nvarchar(50) NOT NULL,
ValidationType nvarchar(50) NOT NULL,
ErrorType nvarchar(20) NOT NULL,
PropertyAlias nvarchar(150) DEFAULT NULL,
DataTypeId uniqueidentifier DEFAULT NULL,
CrawlDate datetime NOT NULL,
ResolvedDate datetime DEFAULT NULL,
Ignore bit NOT NULL,
PRIMARY KEY (IssueId)
)
CREATE TABLE SEOCHECKER_SCHEDULEDVALIDATION(
    Id int(11) NOT NULL AUTO_INCREMENT,
    TaskName nvarchar(250) NOT NULL,
    NotificationEmailAddress nvarchar(250) NOT NULL,
    ExecuteEvery nvarchar(50) NOT NULL,
    ExecuteDays nvarchar(50) DEFAULT NULL,
    ExecuteHour int(11) NOT NULL,
    ExecuteMinute int(11) NOT NULL,
    ValidationRoot int(11) NOT NULL,
    IncludeChildren tinyint(1) NOT NULL,
    NextRun datetime NOT NULL,
    LastTimeExecuted datetime DEFAULT NULL,
    PRIMARY KEY (Id)
)

ALTER TABLE SEOChecker_QUEUE
ADD Priority int(11) NOT NULL

CREATE TABLE SEOCHECKER_NOTIFICATIONS(
    Id int(11) NOT NULL AUTO_INCREMENT,
    NotificationUser nvarchar(250) NOT NULL,
    ExecuteEvery nvarchar(50) NOT NULL,
    ExecuteDays nvarchar(50) DEFAULT NULL,
    ExecuteHour int(11) NOT NULL,
    ExecuteMinute int(11) NOT NULL,
    NextRun datetime NOT NULL,
    PRIMARY KEY (Id))
ALTER TABLE SEOChecker_PageNotFound
ADD
    TimesAccessed int NULL,
    LastTimeAccessed datetime NULL,
    Referer nvarchar (500) NULL,
    RedirectUrl nvarchar (500) NULL,
    ContentType nvarchar (50) NULL,
    Domain nvarchar (250) NULL,
    QueryString nvarchar (250) NULL

Update SEOChecker_PageNotFound set Domain = '' where Domain is null
Update SEOChecker_PageNotFound set TimesAccessed = 0 where TimesAccessed is null
Update SEOChecker_PageNotFound set QueryString = '' where QueryString is null
Update SEOChecker_PageNotFound set ContentType = 'content' where ContentType is
null
Update SEOChecker_PageNotFound set RedirectUrl = '' where RedirectUrl is null
```

10.4 Additional Applications.config configuration

Add the following line to the `/config/ applications.config` file

```
<add alias="seochecker" name="SEO checker" icon="seochecker-tray.png"
sortOrder="700" />
```

10.5 Additional Trees.config configuration

Add the following line to the **/config/ trees.config** file

```
<add silent="false" initialize="true" sortOrder="9" alias="seochecker"
application="seochecker" title="Seo Checker" iconClosed="folder.gif"
iconOpen="folder_o.gif" assembly="SEOChecker" type="Trees.SEOCheckerTree" action=""
/>
```

10.6 Additional Dashboard.config configuration

Add the following line to the **/config/ Dashboard.config** file

```
<section alias="SEOCheckerDashboardSection">
  <areas>
    <area>seochecker</area>
  </areas>
  <tab caption="Get started">

<control>/umbraco/plugins/seochecker/usercontrols/SeoCheckerDashboard.ascx</control
>
  </tab>
</section>
```

10.7 Add the XSLT Extension

Add the following line to the **/config/xsltExtensions.config** file

```
<ext assembly="SEOChecker.Library"
type="SEOChecker.Library.XsltExtensions.Metadata" alias="seoChecker.Metadata">
```

10.8 Add the notfound handler

Add the following line to the **/config/404handlers.config** file, just before the **handle404** entry of Umbraco

```
<notFound assembly="SEOChecker"
type="Handlers.NotFoundHandlers.Inbound404Handler" />
```

10.9 Additional web.config configuration

Install the following HTTP Modules to the to the bottom of the **system.webServer/modules** section in your web.config file.

```
<remove name="SEOCheckerValidationqueueModule" />
<add name="SEOCheckerValidationqueueModule"
type="SEOChecker.HttpModules.ValidationqueueModule, SEOChecker" />

<remove name="SEOCheckerUrlModule" />
<add name="SEOCheckerUrlModule" type="SEOChecker.HttpModules.UrlModule, SEOChecker"
/>
```

10.10 Add the Delete Action to config Extension

Add the following line to the **/Umbraco/config/UI.xml** file

```
<nodeType alias="seochecker">
  <header>Scheduled task</header>
  <usercontrol>/create/simple.ascx</usercontrol>
  <tasks>
    <delete assembly="SEOChecker.Core" type="Umbraco.Tasks.DeleteTask" />
  </tasks>
</nodeType>
```

10.11 Add the license file

If you've purchased a license for SEO Checker you can download the license from your our.umbraco.org profile and add it to the **/bin** folder of your Umbraco installation

11 Validation rules

11.1 Page validation rules

Validation error	Content	Template
Document contains broken links	x	X
Document contains broken media	x	X
Missing link title Attribute on a link	X	
Missing alt attribute on an Image tag	X	
SEO Description is empty	X	X
SEO Title is empty	X	X
SEO Title length longer than 65 characters	X	
SEO Description is less than 50 characters	X	
SEO Description is longer than 160 characters	X	
SEO Title is missing		X
SEO Description missing		X
Document contains multiple SEO Titles		X
SEO Description contains <umbraco:item		X
SEO Keywords contains <umbraco:item		X
H1 tag missing		X
Multiple H1 tags found		X
H1 tag is empty	X	X
Google Analytics code is missing on the page		X
Content of rendered page is empty		X
Document contains Lorem Ipsum test content	x	X
Document contains a lot of inline CSS code		X
Document contains a lot of inline Javascript code		X
Document contains a lot of HTML comments		X
Document contains a large initial viewstate		X
Broken CSS Stylesheet references		X
Broken Javascript references		X

11.2 Configuration validation rules

Validation error	Type
Homepage can be accessed by /default.aspx	Canonical issue
Homepage can be accessed by /rootnode url	Canonical issue
Page can be accessed using uppercase and lowercase characters	Canonical issue
Page with aspx extension is not redirecting to the page without file extension when the site is using directory url's	Canonical issue
Page without extension is not redirecting to aspx page when not using directory url's	Canonical issue
Page can be accessed with and without trailing slash when using directory url's	Canonical issue
Website can be access using www and non www url's	Canonical issue
Website is not setup for www, or non www hostname	IIS configuration issue
aspx	
Robots.txt file is missing	Search engine robot issue
Robots.txt is blocking crawling of the site	Search engine robot issue
Robot meta tag on homepage is blocking crawling of the site	Search engine robot issue
Friendly 404 error page is missing	General issue
Latest update is more than 1 month old	General issue
SEO Checker Data type is missing from the document type	General issue
No SEO checker configuration found for any document type	General issue

12 Troubleshooting

12.1 I don't see the SEO checker section

Make sure the package is installed correctly and you have access to the SEO Checker section in Umbraco.

12.2 I don't see the SEO checker dashboard

Make sure the package is installed correctly. Check manual installation of the package

12.3 I see the following message on the redirect overview

"The Url module is not running, this means url rewriting and redirects will not happen. Check the user manual how you can configure the Url module manually."

Make sure that the following item is added to the modules section of your web.config file.

```
<add name="SEOCheckerUrlModule" type="SEOChecker.HttpModules.UrlModule, SEOChecker" />
```

12.4 I see the following message on the validation queue

"The validation queue HTTP module is not running, this means validation tasks will not execute. Check the user manual how you can configure validation queue HTTP module manually."

Make sure that the following item is added to the modules section of your web.config file.

```
<add name="SEOCheckerValidationqueueModule"
type="SEOChecker.HttpModules.ValidationqueueModule, SEOChecker" />
```

12.5 Validation is reporting errors on the document.

Make sure the page doesn't contain errors in macro's and template .

12.6 Some pages don't get validated

Make sure the document has a template assigned.

12.7 I don't receive email notifications

Make sure the mail server is correctly setup in web.config and make sure the email address of the user that should receive the notification is correct.

12.8 I found a bug in the software

If you found a bug in the software, please take a look at our download location

<http://soetemansoftware.nl/seo-checker/downloads> to see if there is a new version available. If not please send us an email support@soetemansoftware.nl so we can fix the bug and create a hotfix for you.

12.9 I have another question

If you have a question please email us at support@soetemansoftware.nl we love to help you.