

DIGIMAKER 6.0



Digimaker 6.0 Using Friendly URLs

Copyright and Proprietary Information

© **Digimaker™ 2008**. This user guide constitutes proprietary information of Digimaker. No part of this user guide may be reproduced, transmitted, transcribed, stored in a retrieval system or translated into any language in any form, by any means, without the written permission of Digimaker.

Digimaker reserves the right to make changes in this user guide at any time and without notice.

Digimaker makes no warranties, express or implied, in this user guide. In no event shall Digimaker be liable for any indirect, special, incidental or consequential damages arising out of purchase or use of this user guide or the information contained herein.

Using friendly URLs in Digimaker 6

USING FRIENDLY URLs IN DIGIMAKER 6	1
INTRODUCTION	3
BEFORE AND NOW	3
CONTROLLING THE NAMING OF YOUR FRIENDLY URL	3
FOR THE DEVELOPER	4
ADDITIONAL QUERYPARAMETERS	4
ACTIVATION / SETUP	4

Introduction

Dynamic CMS driven websites tend to generate the links to different pages based on IDs found in a database. Digimaker does the same thing. From version 6 of Digimaker the dynamically generated URLs can be as we call it “Friendly”, that is: easier to read for the end user and more acceptable for search engines such as Google.

Before and now

Before a typical link on your Digimaker website would be:

<http://www.myweb.com/mytemplate.aspx?m=354&amid=6632>

With Friendly URLs this could translate into: http://www.myweb.com/news/_latestItem.html

Controlling the naming of your friendly URL

Let’s start with a breadcrumb example reflecting the menu item structure inside Digimaker, and say it looks like this:

Home > News > Sports > Football

The Friendly URL could with the digimaker.config setting above look like this:

<http://www.myweb.com/football.html> or <http://www.myweb.com/20.html>

If a text (like “football”) or number should be used is based on the “Alias” entered for the menu item, unless an alias is provided the id will be used.

Note: The URL is still search engine friendly even though the URL uses numbers.

If the setting “ShortPath” is changed to “false” the friendly URL could be like this:

<http://www.myweb.com/home/5/17/football.html> having the entire bread crumb path in the URL.

For the developer

Additional Querystring parameters

You as the developer decides if your extra querystring parameters should be supplied to the functions that support it or not, if not you simply append "?<parameter>=<value>" to the functions return value. Passing as a parameter is the recommended, in case the format of friendly URL changes in the future. A friendly URL with additional query string parameter(s) will look like this:

http://www.myweb.com/news/_latestItem.html?newParam=here

For helper functions on creating links the same as before are used, found in: SiteBuilder.WebControls.ContentToolkit.Url

Activation / Setup

In your digimaker.config file there is a collection named UrlRewrite, you will find this in the Web section.

Here is what it typically looks like:

```
<collection name="UrlRewrite">
  <add key="Enabled" value="true" />
  <add key="ShortPath" value="true" />
  <add key="ServerAddresses" value="www.domain.com" />
</collection>
```

To enable or disable the functionality, simply change the value for "Enabled" to either false or true. A value true will enable friendly URLs on your site.

Making your templates ready

All links that is not absolute should start with a "/", you might already have this then you don't need to modify anything.

What to validate start with a "/":

- Anchors e.g. Another menu item
- Images e.g.
- Stylesheet e.g. <link rel="stylesheet" type="text/css" href="/main.css" />
- Javascript e.g. <script type="text/javascript" src="/functions.js"></script>
- Other embedded things like flash and video players.
- If anything is missing then you should have gotten the idea by now 😊

Why?

It's all about understanding how browsers interpret the relative URLs on the web page.

When the URL is relative / does not start with e.g. http:// it can e.g. look like this "images/pic.gif" for an image inside the "images" folder. The browser will have to prefix relative URLs with part of the address in the browser in order to get a complete URL for where the picture exists.

If the URL in the browser is <http://www.domain.com/> or <http://www.domain.com/myTemplate.aspx?m=30> or (friendly) <http://www.domain.com/30.html> the "path" to prefix relative URLs will be "<http://www.domain.com/>" making the path to the picture "<http://www.domain.com/images/pic.gif>", and in this case it will be correct regardless of friendly URL or not.

Consider an example where you have your templates inside a folder called "web2" on the same domain. A link here could be <http://www.domain.com/web2/myTemplate.aspx?m=30>. In the case where you want to re-use the pictures from the website's root folder you would have to make the images path's like this "../images/pic.gif" or (here comes the same thing as mentioned before) you could make the path's like this "/images/pic.gif" then it doesn't matter how many levels of folders you have, the path will always start on the "root".

The same "problem" is what you will get with friendly URL's where **e.g.:**

http://www.domain.com/news/_latestItem.html looks like "_latestItem.html" exist in a folder "news".

Customizing the alias validation

Not all characters are legal in URLs and Digimaker validates this for menu item and article aliases. In case you want to customize the various characters that should be (not) allowed you can do that by modifying digimaker.config.

Here is an example that shows the default value:

```
<section name="custom">
  <appSettings>
    <add key="AliasIllegalCharactersRegExp" value="/[^\&gt;%&amp;&quot;&lt;*&\\|:\.\\?\\$@\\#]+/gi" />
    .....
    .....
  </appSettings>
</section>
```

The value here is and must be encoded both to make the xml (digimaker.config is an xml file) valid and when output on the website it must be a valid JavaScript regular expression. The backslashes are escaping the next character. That is required for those characters that have special meaning in regular expressions.

How this expression work: all characters inside the square brackets are illegal, if you want to e.g. make hyphen "-" illegal you can add it after the "#" just before the "]". If we don't escape and encode the above it will look like: `/[^\&>*&\\|:]"/+gi`

How it's all maintained

For every request that is "Friendly" the URL is parsed and rewritten on the context to be "normal" while the page is processed. As the friendly URL will hold the alias or the id of menu item to access we need to look up the id and/or the template for specified menu item. Instead of querying the database each time we keep these values in lists updated upon changes within Digimaker. They are updated by having Digimaker backend request a URL on the website (handled by the link-handler), that is why the UrlRewrite section in digimaker.config holds the key [ServerAddresses](#), that way all the front-end servers (there will be more than one in load balanced environments) can be updated with new data when a menu item or article's info is changed.

Such update will happen when the user changes a menu item's alias or template, a template's filename or an article's alias.