

Kentico 8.2 - Initial site install, setup and configuration

MINNESOTA KENTICO USER GROUP – 1/21/2015

JEFF STEIL, DAHLIN DEVELOPMENT

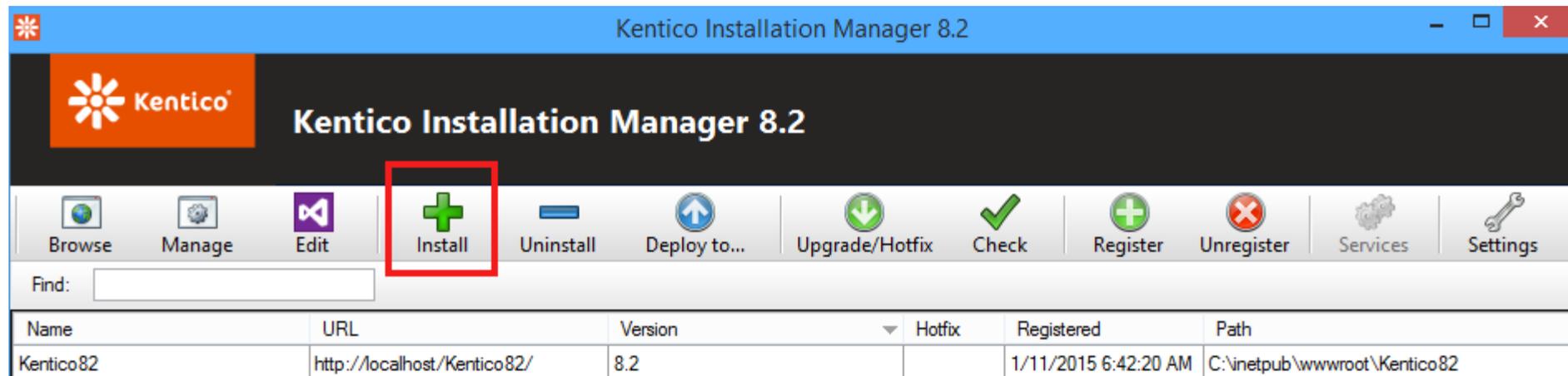


Goals

- Walk through a custom installation of Kentico 8.2 as a website project, selecting options that will optimize the installed website
- Configure User Accounts
- Naming Conventions
- Creating Page Templates – 404 ,and sitemap pages.
- Core Site Files
- Media Libraries
- Site Configuration – Files, SMTP, redirects, etc
- Javascript and CSS minification and compression.
- Delete/Disable default objects and page templates
- Stop/Modify scheduled tasks that don't apply to the site

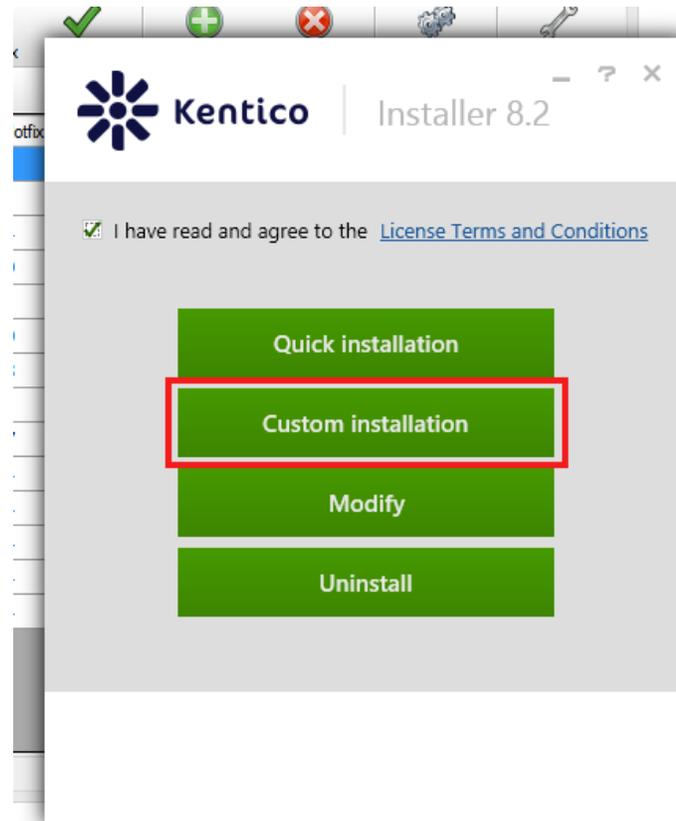
KIM 8.2

- Click the Install button to begin the installation.



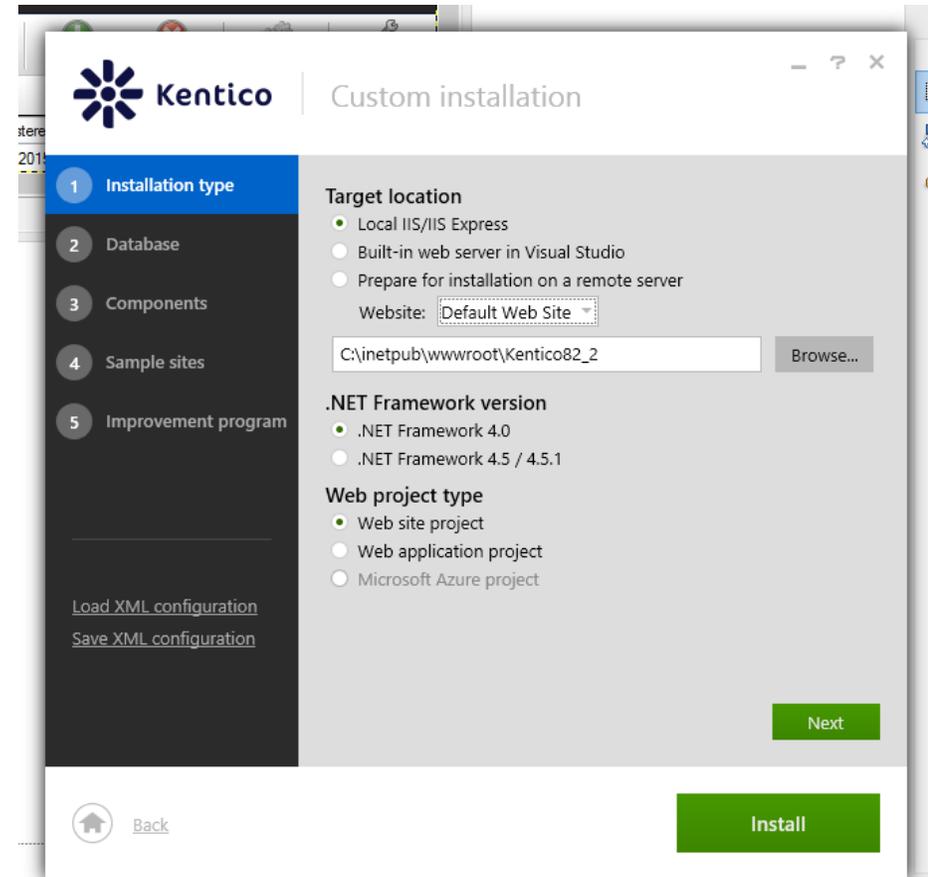
KIM 8.2 – Custom Installation

- Choose Custom Installation



Custom Installation– Installation Type

- Choose a location to install the website.
- The rest of the options can remain as the defaults, you may want to use .Net 4.5 depending on your preferences.



Custom Installation - Database

- Create the database

The screenshot shows the 'Custom installation' wizard for Kentico, specifically the 'Database' step. The interface is divided into a left sidebar and a main content area. The sidebar contains a progress indicator with five steps: 1. Installation type, 2. Database (highlighted in blue), 3. Components, 4. Sample sites, and 5. Improvement program. Below the sidebar are links for 'Load XML configuration' and 'Save XML configuration'. The main content area displays the following configuration options:

- Installation type:** Installation with database, I have access to SQL server
- SQL server:** A dropdown menu showing 'JEFF-PC'.
- Authentication:** Integrated Windows authentication, SQL Server account. Below this is an 'Establish connection' button and a 'Connected' status indicator with a green checkmark.
- Database name:** A dropdown menu showing 'Kentico82_2'. Below this is an unchecked checkbox for 'Overwrite existing database'.

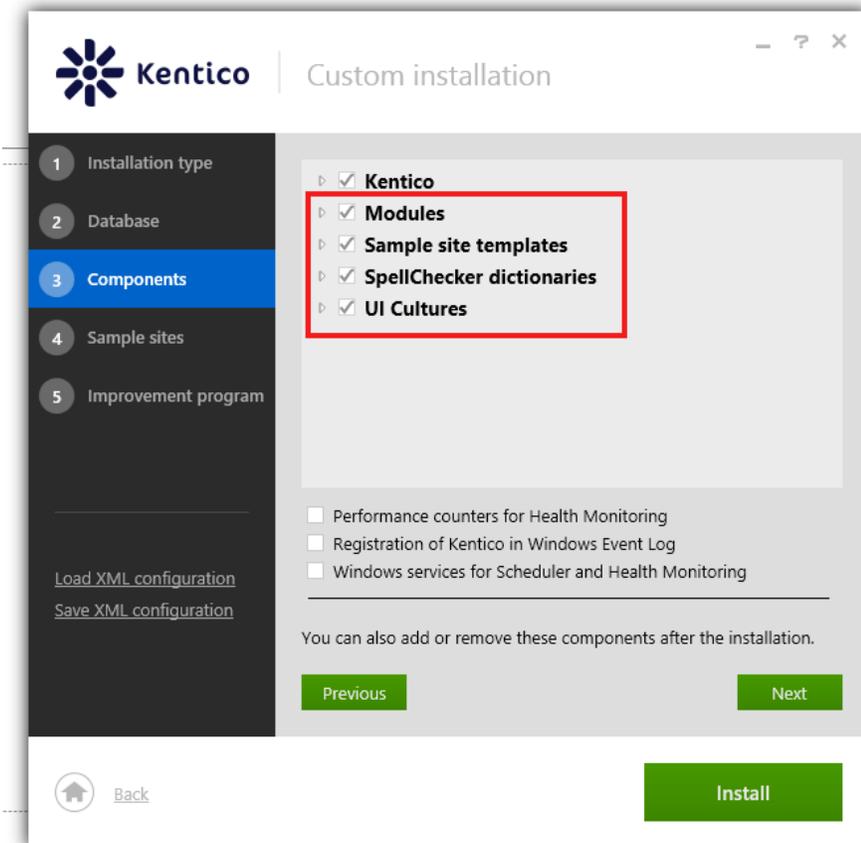
At the bottom of the main content area are 'Previous' and 'Next' buttons. At the very bottom of the window are a 'Back' button with a home icon and a large green 'Install' button.

Custom Installation - Components

Selecting only the components needed in the site will save disk space and result in faster build times.

A site built with only the Forms module installed will have 6,705 files vs. 9,174 files for a full install.

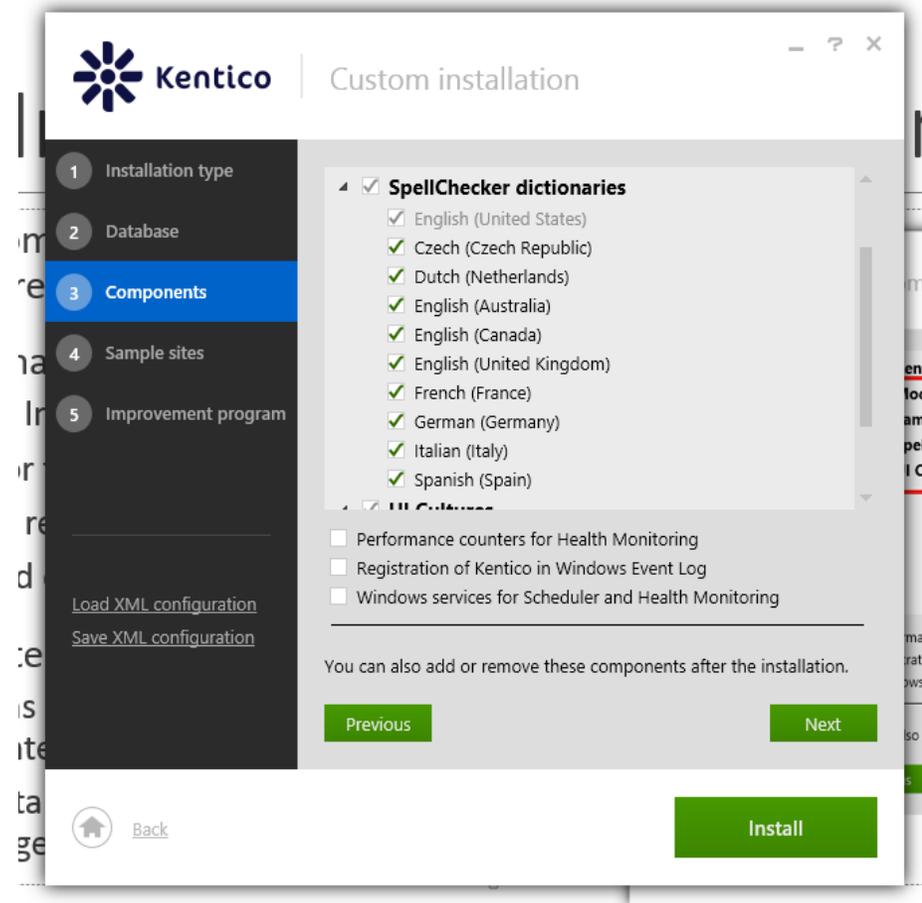
- Modules (some examples)
 - WebDav – only for Intranet sites
 - WIF Integrator – for third party integrations
 - API Examples – for reference only
 - E-commerce – used on e-commerce sites
- Sample site templates
 - In previous versions of KIM you could select only the templates you wanted to install.
 - In Kentico 8+ it installs all of them, but you can still choose which site gets imported.
 - This seems like a bug in KIM 8+ but actually was by design.



Custom Installation – Components (cont)

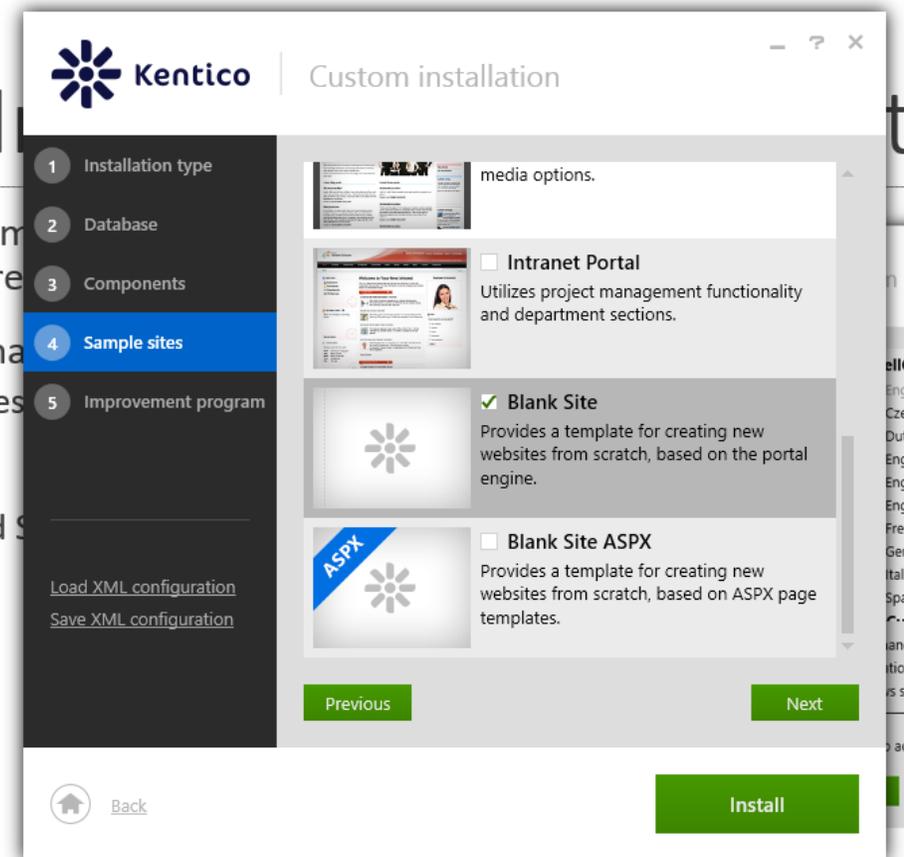
Selecting only the components needed in the site will save disk space and result in faster build times.

- SpellChecker dictionaries
 - Only keep languages that are relevant to your site.
- UI Cultures
 - Uncheck Czech and Slovak
- Health Monitoring
 - Adds overhead to the website, can be added later.
- Windows Event Log
 - Events get logged to the Kentico Event Log, shouldn't need to check this.



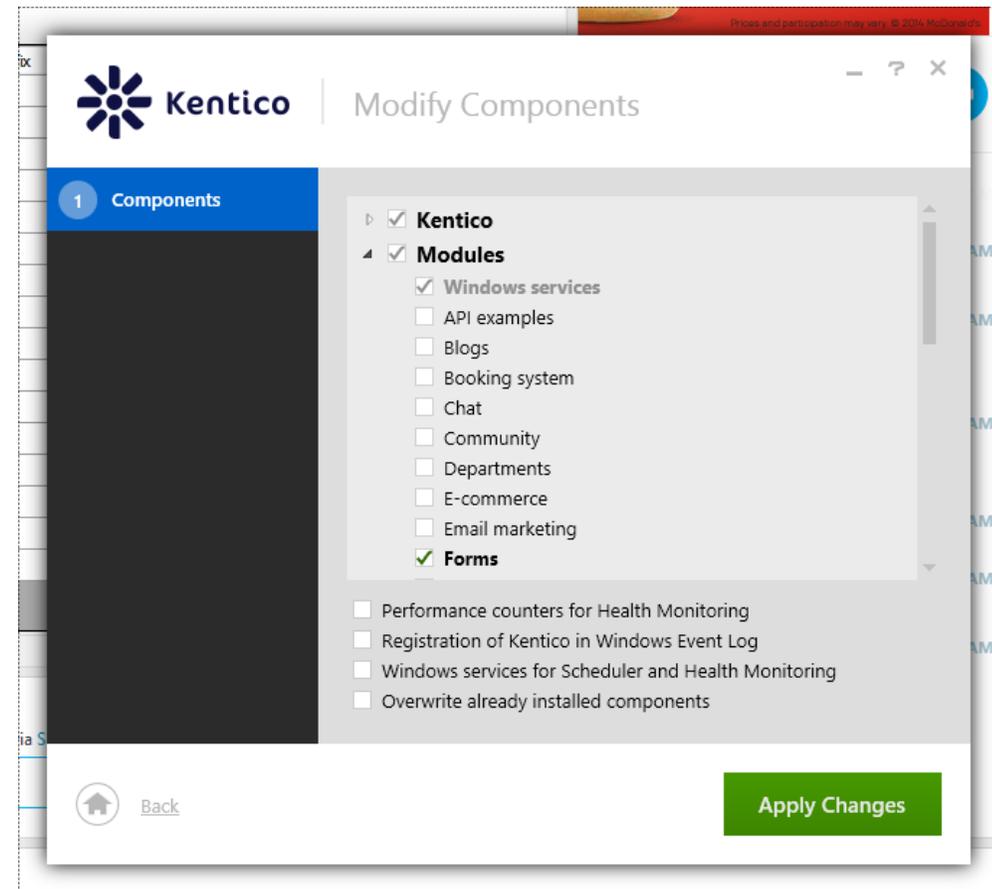
Custom Installation – Sample Sites

- For a development site choose **Blank Site** as the sample site to import, this uses the preferred portal engine.
- Choosing **Blank Site ASPX** will make future upgrades much more difficult as changes to the Kentico API will require code changes to your site.



Adding Components to a Site

- If you didn't add a component that is needed you can always add it to the site using KIM.
- Select Install > Modify and then the site you want to add a component to.



Upgrade to the Latest Hotfix

- After installing the website, make sure it has been upgraded to the latest hotfix.

The screenshot shows the Kentico Installation Manager 8.2 interface. The top toolbar contains several buttons: Browse, Manage, Edit, Install, Uninstall, Deploy to..., Upgrade/Hotfix (highlighted with a red box), Check, Register, Unregister, Services, and Settings. Below the toolbar is a table listing installed projects:

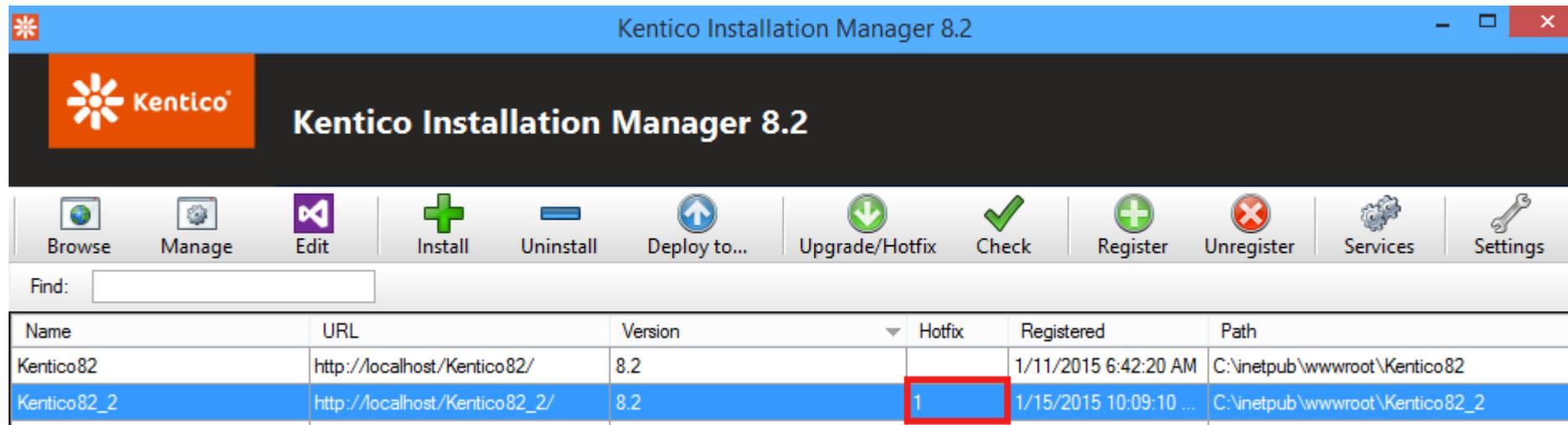
Name	URL	Version	Hotfix	Registered	Path
Kentico82	http://localhost/Kentico82/	8.2		1/11/2015 6:42:20 AM	C:\inetpub\wwwroot\Kentico82
Kentico82_2	http://localhost/Kentico82_2/	8.2		1/15/2015 10:09:10 ...	C:\inetpub\wwwroot\Kentico82_2

The second screenshot shows the 'Hotfix/upgrade' dialog. It has two main sections: 'Available updates' and 'Projects to be updated'. Under 'Available updates', there is a tree view showing 'Hotfixes' > '8.2' > '8.2.1'. Under 'Projects to be updated', there is a table:

	Name	URL	Version	Downloaded
<input checked="" type="checkbox"/>	Kentico82_2	http://localhost/Kentico82_2/	8.2	No

Upgrade to the Latest Hotfix (cont)

- The site has now been upgraded to the latest hotfix.

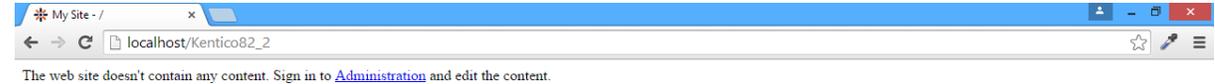


The screenshot shows the Kentico Installation Manager 8.2 application window. The interface includes a toolbar with various actions like Browse, Manage, Edit, Install, Uninstall, Deploy to..., Upgrade/Hotfix, Check, Register, Unregister, Services, and Settings. Below the toolbar is a search bar and a table listing installed sites. The table has columns for Name, URL, Version, Hotfix, Registered, and Path. The 'Hotfix' column for 'Kentico82_2' is highlighted with a red box and contains the value '1'.

Name	URL	Version	Hotfix	Registered	Path
Kentico82	http://localhost/Kentico82/	8.2		1/11/2015 6:42:20 AM	C:\inetpub\wwwroot\Kentico82
Kentico82_2	http://localhost/Kentico82_2/	8.2	1	1/15/2015 10:09:10 ...	C:\inetpub\wwwroot\Kentico82_2

Empty Kentico 8.2 Website

- With a Blank site, there is no content.
- Log into the Administration portion of Kentico (previously CMSDesk and Site Manager)
- User name is **administrator** with a blank password.



Change Site Name

- Go to Sites and click on the pencil (edit) link next to Blank Site.
- Change the Site display name to **My Site**.
- Change the Site code name to **MySite**.

Change Administrative Password

- Go to Configurations > Users and add a password for administrator.
- Delete the Andy user.

Users

Mass email

On-line users

New user

Advanced search

Reset Search

Actions	User name	Full name	Email	Nickname	Created	Enabled
	administrator	Global Administrator	administrator@localhost.local			Yes
	Andy	Andrew Jones	andy@localhost.local		1/15/2015 10:40:49 AM	Yes
	public	Public Anonymous User				Yes

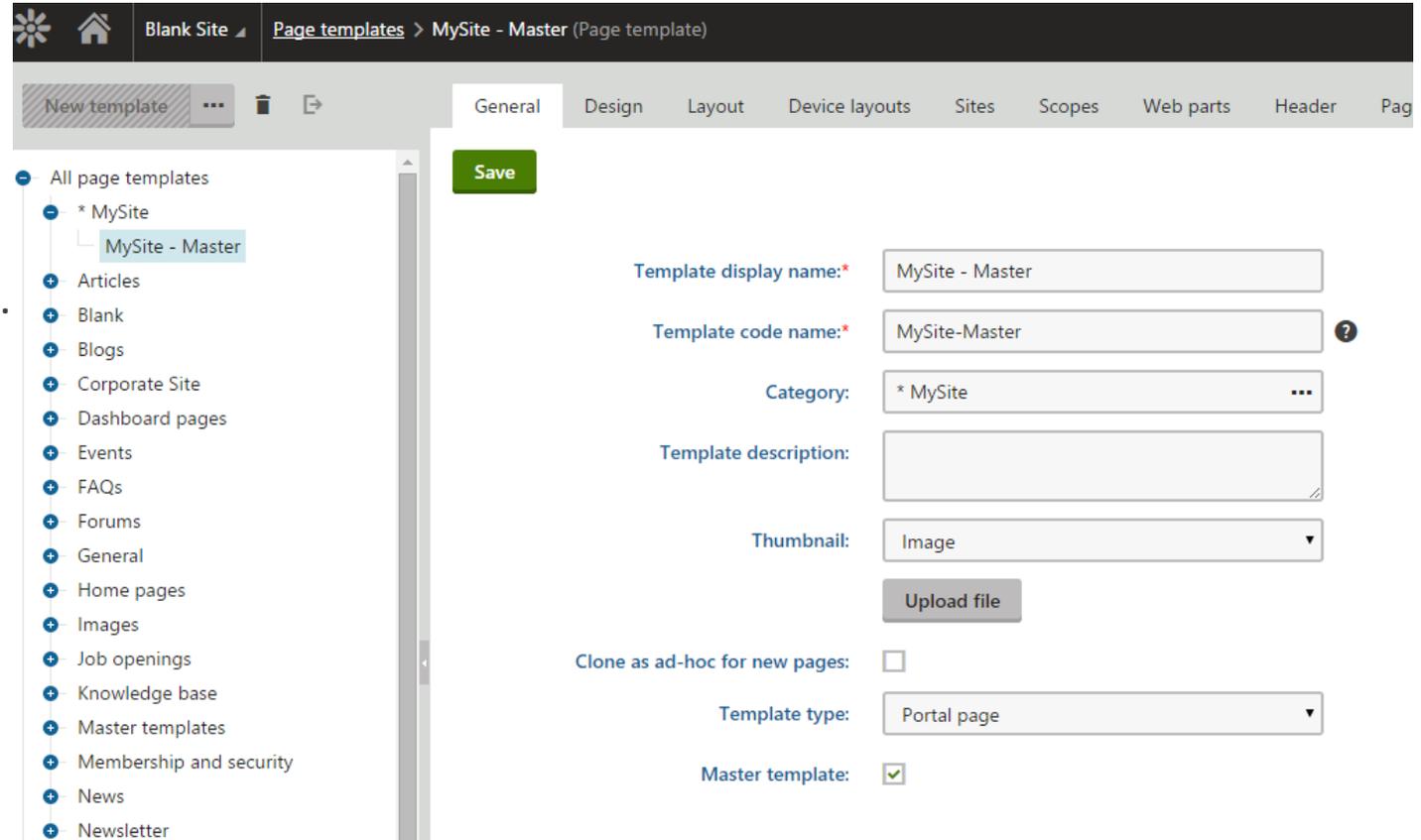
Items per page 25

Naming Conventions

- A naming convention of MySite.* will be used throughout the site.
- Custom Controls, Modules, and Services will go under a MySite folder.
- Custom classes will be placed under App_Code/MySite
- Stylesheets, Fonts, and Images will be placed under:
 - /App_themes/MySite/css
 - /App_themes/MySite/fonts
 - /App_themes/MySite/images
- Javascript will be placed under:
 - /CMSScripts/Custom/MySite

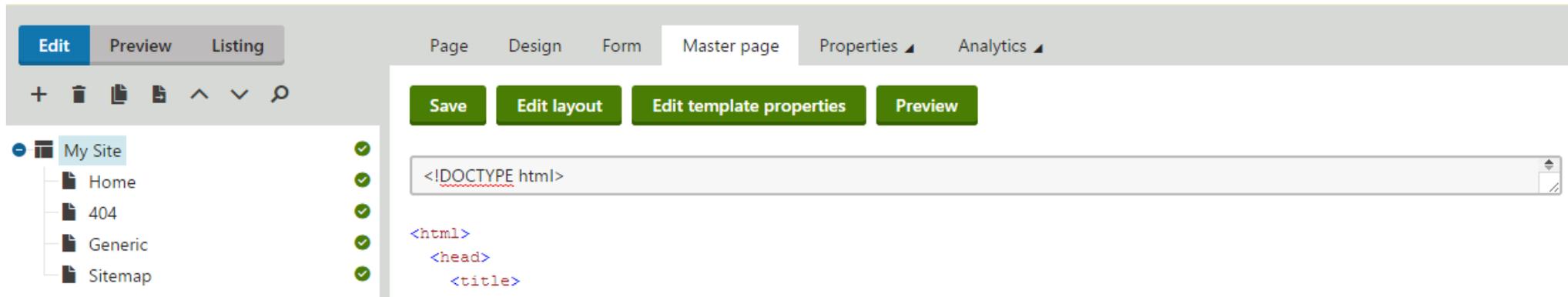
Page Templates

- Go to Development > Page Templates
- Create a new category – * MySite
 - The asterisk (*) at the front of the category ensures that it shows up at the top of the Page Template categories.
- All templates should have a type of **Portal Page**.
- Prefixing templates with MySite will make them easier to find when adding new pages.



Page Templates – Master Page

- Change the doc type to HTML5.



Page Templates – Generic/404

- Add a new template for a Generic Page
 - Create a new template with a name of **MySite – Generic**.
 - This template will have a single editable text web part on it.
- Go to Content Management > Pages
- Click the plus (+) to add a new page.
- Click **Page (menu item)**.
- Enter **404** as the page name.
- For the template, search for **Generic**.
- Select the **MySite – Generic** template.
- Because we don't want the 404 page shown in navigation or the sitemap, go to the General > Navigation tab and uncheck the **Show in navigation** and **Show in sitemap** checkboxes.

Page Templates - Sitemap

- Kentico allows you to automatically generate sitemaps for your websites according to the Google Sitemap Protocol. They help search engines correctly index the content of websites and can have a significant effect on the resulting search ranking.

- Add a new template with a name of **MySite - Google Sitemap** and place it under the * **MySite** category.

- On the layout tab enter:

```
<cms:CMSWebPartZone ZoneID="ZoneA" runat="server" />
```

- On the design tab, add a Google Sitemap (XML Sitemap) web part to ZoneA.
 - Set the path to */%*.
 - Set the page types to CMS.Menuitem and any other page types on the site.
 - The order by expression should be nodelevel, nodeorder.
 - Check **Disable the view state** under performance.

- Go to Content Management > Pages

- Click the plus (+) to add a new page.

- Click **Page (menu item)**.

- Enter **Sitemap** as the page name.

- For the template, search for **Sitemap**.

- Select the **MySite – Google Sitemap** template.

- Because we don't want the Sitemap page shown in navigation, go to the General > Navigation tab and uncheck the **Show in navigation** checkbox.

Media Library

- Go to Content Management > Media libraries
 - Media libraries are useful for storing media:
 - Get absolute url to an asset, does not use getattachment like attaching assets to a document does.
 - Media libraries allow bulk uploads.
- Media libraries are used to store images and documents added by content editors. However, the core site files (css, fonts, images, and javascript) are not stored in the media library.
 - Create an images folder.
 - Create a documents folder.

Configuration (Content)

- Go to Configuration > Settings
- Switch to the My Site site. Most of the settings should be changed under the specific site and not Global.
- Settings > Content:
 - Select **/Home** as the default alias path.
 - Enter **/404** as the Page not found URL. You will also need to add **runAllManagedModulesForAllRequests="true"** to the modules element in the web.config.
 - Enter **My Site** as the page title prefix.
- Settings > Content > Content Management
 - Uncheck Enable On-site editing.
 - Uncheck Enable On-site editing button.
 - These setting interfere with page previews and we have found they are rarely used by actual content editors.

Configuration (URLs and SEO)

- Settings > URLs and SEO
 - Remove the .aspx from **Friendly URL extensions**. This will make the friendly url the same as the alias path.
 - Check **Move ViewState to the end of the page**.
 - Set the **Google sitemap URL** to sitemap.xml
 - Set the **Google sitemap path** to /Sitemap.
 - Change **Redirect invalid case URLs...** to Redirect all requests to lower case URLs.
 - Remove the .aspx from Files friendly URL extension.
 - Change **Use URLs with trailing slash** to Always.
 - Change **Process domain prefix** to Always use domain without www prefix.
 - Change **Default Path** to Use domain root. This will link your home page to “/” instead of “/home”.

Configuration (System)

- Settings > System
 - Change Site time zone to the correct time zone for the site.
- System > Emails
 - Set the SMTP server values. Consider using an external SMTP service such as sendgrid to relieve the server hosting the website from sending emails as well.
- System > Files
 - Change the settings to store files in the file system, not the database. This will increase the size of the site but will keep the size of the database lower, which may be helpful if backing up and restoring the site at some point.
 - Keeps database storage costs down.
 - Will be faster as Kentico does not have to perform a query to retrieve the file.

Configuration (System)

- System > Performance
 - Check **Enable output compression** to compress HTML output. This setting is under Global. Browsers indicate whether they can accept compressed output through the Accept-Encoding token in the HTTP request. Browsers that don't accept compressed output will receive uncompressed page output.
- System > Debug
 - The settings are under Global.
 - Check **Disable debugging**. This disables all debugging on the site regardless of individual debug settings.

Remove favicon.ico

- Delete favicon.ico from the root directory of the site.
- Upload it to media library. This will allow it to be changed from within Kentico.
- Add reference to it in master page.
- To upload a .ico file you need to add it to Configuration > Settings > Content > Media
 - In the Security section, add ;ico to **Media file allowed extensions**.

PersonalSite	1/15/2015 10:02 AM	File folder	
Properties	1/15/2015 10:02 AM	File folder	
Views	1/15/2015 10:02 AM	File folder	
App_Offline.htm_	9/15/2011 11:48 AM	HTM_ File	1 KB
clientaccesspolicy.xml	8/18/2014 5:00 AM	XML File	1 KB
CMSApp_MVC.csproj	12/19/2014 6:06 PM	CSPROJ File	9 KB
Default.aspx	1/15/2015 10:04 AM	ASPX File	1 KB
Default.aspx.cs	8/18/2014 5:00 AM	CS File	1 KB
DocumentRESTService.svc	8/18/2014 5:00 AM	SVC File	1 KB
favicon.ico	8/18/2014 5:00 AM	ICO File	3 KB
Global.asax	1/15/2015 10:05 AM	ASAX File	1 KB
Kentico82_2	1/15/2015 10:05 AM	Internet Shortcut	1 KB
ObjectTranslationRESTService.svc	8/18/2014 5:00 AM	SVC File	1 KB
parameters.config	12/19/2014 5:29 PM	CONFIG File	2 KB
RESTService.svc	8/18/2014 5:00 AM	SVC File	1 KB
web.config	1/15/2015 10:27 AM	CONFIG File	14 KB
wse3policy.config	12/19/2014 5:29 PM	CONFIG File	2 KB

CSS, Images, and Fonts

- Add custom CSS to Development > CSS Stylesheets
 - Remove the two corporate style sheets.
 - Add a new stylesheet for MySite.
 - Click the theme tab and add a new folder, css.
 - When style sheets are added under a theme their location within the site structure will be in /App_Themes/MySite/css
 - Create a new folder for vendor css and add any vendor (third-party) css files to MySite > css > vendor
 - For responsive stylesheets we generally add them as separate files here as well. For example:
 - mysite.css
 - mysite_tablet.css
 - mysite_mobile.css
- Store Site Images under Development > CSS Stylesheets > MySite in an images folder
- Store fonts under Development > CSS Stylesheets > MySite in a fonts folder

Javascript

- Add custom Javascript to Development > javascript
 - Create a folder for MySite
 - Location within site structure will be in /CMSScripts/Custom/MySite
 - Create a new folder for vendor javascript and add any vendor (third-party) javascript files to MySite > javascript > vendor

Javascript and CSS minification and compression

- Go to Configuration > Settings > System > Performance
- Minifying javascript and css removes all unnecessary characters from the code that are not required by the browser to correctly process the resource. This is turned on by default for both resource types.
- Compression further reduces the size of resources by encoding their data. To apply compression, you must also enable minification for resources of the given type. To compress Javascript and CSS, check the **Allow resource compression** checkbox.
 - May not want to do this until right before launch to aid with debugging.
- Always request CSS through GetResource.ashx to take advantage of minification and compression.
- Always request javascript through GetResource.ashx as well.

Custom Code

- In the website directory, create a **MySite** folder at the root level.
 - Store custom Controls and Services here.
- For custom classes, create a **MySite** folder in App_Code
- Do as much as you can within the site vs. creating external assemblies.
- Don't overwrite Kentico files.
- Use the built-in Kentico web parts whenever possible, the more custom code you have, the more difficult upgrades become.

Web.config

- Because extensionless URLs have been set up in Kentico, ensure that the trailing slash will only be applied if the URL doesn't have any extension:
 - `<add key="CMSUseTrailingSlashOnlyForExtensionLess" value="true" />`
- Configure postbacks to also use extensionless urls, otherwise by default any postback will add a .aspx suffix to the request:
 - `<add key="CMSUseExtensionOnPostback" value="false" />`
- To configure your site so that the URL of the home page only contains the domain in all circumstances:
 - `<add key="CMSDefaultPageURL" value="~/ " />`

Caching

- Generally leave caching alone as Kentico has good defaults to caching out of the box.
- An exception might be for an extremely static site with minimal updates.
 - You can turn on output caching on the master page to cache full pages.

Output cache

Use output cache: Yes No

Cache minutes:

Clear output cache

Allow file system cache: Yes No

Redirects

- If you are replacing an existing site, you may want to redirect pages from the old site to the new site.
- If you have access to IIS, you could install the URL Rewrite module and configure it in web.config. You can use URL Rewrite to add rules to redirect based on url pattern.
- If the old site is small or you don't have access to IIS on a shared host, you can set up page aliases within Kentico to do the redirecting.
- For example, to redirect requests to oldpage.chm to a new page on your site, add a page alias to handle that.

Page aliases

Add new alias

Actions	<u>URL path</u>	<u>URL extensions</u>
 	/oldpage	.chm

Scheduled Tasks

- Go to Configuration > Scheduled Tasks
- When all modules are installed during installation, a lot of tasks are scheduled by default as well, even if the component is never used in the website.
- Global lists a lot more
- The default frequency on some tasks is higher than it should be.
 - **Remove expired sessions** is scheduled to run every minute, it can probably be a lot less frequent.
 - **Delete old temporary attachments** is scheduled to run every hour.
 - **Delete old system cache files** is scheduled to run every hour.
 - **Report subscription sender** is scheduled to run every minute, if you aren't using reporting then the frequency can be turned down or the task can be stopped altogether.
 - **Synchronize web farm changes** is also scheduled to run every minute and can be turned down or the task can be stopped altogether if you aren't using a web farm for your site.
- Just keep in mind that if you turn off a task for a module you will need to remember to turn it on if you start using the module at a later time.

Deleting Web Templates

- Go to Development > Web Templates.
 - By default, all web templates are included in the installation.
 - In previous versions (7 or lower) of Kentico you could choose not to install templates. In 8+ you must install all templates, even in a custom installation.
 - After talking to Kentico support, it is recommended not to delete web templates. They replied that deleting the templates could cause issues when pre-compiling the project (publishing in VS) or when deploying to Azure. They also recommended backing up the templates if you do delete them!

Remove Assigned Objects

- Go to site > assigned objects
- Page Templates
 - If there are any templates you won't be using, remove them.
- Classes
 - You can clean up the classes used by the system by deleting those you won't be using, possibly SharePoint – Transformations, etc.
- Modules
 - Modules that were installed but won't be used for the site can be removed. An example might be E-Commerce

Questions?

Jeff Steil

jeff@dahlindev.com | 320-492-9451

Dahlin Development | dahlindevelopment.com

- * Follow our blog - Slides from this presentation, Kentico tips, advice, and more! *