Website Framework

It wont be a drop in replacement for the current site. Theres too much that will be changing. From the database technologies right down to the removal of “publications” part. I should be able to write an “article importer” tho. None of the current pages will be re-usable.

Im debating how the “publications” part will be changed. Im kinda looking at each domain gets its own “content” (articles) but that’s gonna be a total mess up with the whole zoutnet / zoutpansberger / mirror thing.

Will be possible for instance on zoutnet to list the top articles from zoutie and mirror via the api. But it wont be as “seemless”.

LiN was about unifying each publication into a bigger organization. Im looking at splitting each domain to its own again.

Overall the website framework should we go ahead will be a huge undertaking again. Will definitely make “micro” sites easier. As you could use the framework for all micro projects like this elections site. If I get the site to the state as outlined here, everything on the elections / directory / pieter’s new guest house will be able to be built in it.

# Admin

* Dashboard
  + Can edit the blocks. Modules can have “dashboard” blocks which allows the user to add module’s blocks to the dashboard. Drag and drop style similar to pages. The stats form part of these blocks. They will probably be “dashboard only” modules.
  + Update checker. If updates are available it will show it as such and allow the user to update the framework. For the most part I will use webhooks to automatically update the sites so shouldn’t ever really have the need to update. Just useful for anyone else other than us using the framework.
* Content
  + By default takes you to the last content type you captured with an option to change it to other content types
  + Going back to the list, you select the type and it shows those “articles” / “content”
  + The content forms are built similar to the pages part where you drag and drop “elements” onto the page. This will allow for some really interesting content abilities
* Pages
  + Will work sort of similar to the current pages. “group” now will be a dropdown with all the groups. So you can filter the pages based on the group the page falls under. So like: core, streets, adverts, internals, info etc (users define the groups). Makes it easier working on a certain part of the site.
  + Each page will be able to redefine its own masterhead / footer / css / js overwriting the global options.
  + Restrictions will have a few more options
  + Meta will allow you to “edit” the default html or just fill in the fields to use the default.
  + Content blocks:
    - No more set styles. Instead the user can save a “snippet” and name it for reuse.
    - Admin for “article list” etc will have to change. Need to figure out how still. But guessing you choose a type and then it gives you filtering options based on the form created for the content type.
    - These will still be “modular” and each module will be able to add tabs to the admin toolbar etc. so not using comments? Then it wont have the comments toolbar in admin.
    - Content blocks can be saved viewable in the list for the module for this page only, this page group only and all. Currently it’s this page only and all in the existing framework. The list will show the restrictions

\* Pages gets a drop down arrow button next to it with a dropdown menu (global options)

* + - Masterhead
    - Footer
    - Css
    - Javascript
    - Navigation
* Users
  + List of users. Same as current site
* File Browser
  + See note at the end about it becoming its own project.
* Settings
  + Domain Settings
  + Content Types
    - Each content type allows you to build the form for this content type
    - Flags: Create “flags” for filtering
  + Restrictions / Permissions

# Front End

The front end will be totally blank. All code to be added via pages using twig templating language. <http://twig.sensiolabs.org/> (there is a ton of documentation on it and it’s a very mature project)

The backend will make “data” available to the frontend like the current site does with the ?showdata url parameter. (ie [www.zoutpansberger.co.za?showdata](http://www.zoutpansberger.co.za?showdata) – make sure you have JSONview extention installed)

Libraries included by default:

* jQuery
* jQuery UI
* Bootstrap
* Font-awesome
* Modernizr
* jQote2 (javascript templating)
* Plupload (uploading)
* Select2 (selectbox replacement)
* Moment (javascript date manipulation)
* BA-BBQ (hash stuff – page#***tab=tab1*** etc)
* jQuery Mousewheel
* jQuery cookies
* jQuery hotkeys
* toastr (toaster messages, will be using them in admin mainly but they can be used for the front end)

any other libraries etc needed can be uploaded to a certain folder in the file browser and they will be concat’ed with all other files of the type there and added. So for instance the source code will be as follows:

* <link rel="stylesheet" href="main-libraries-styles.css">
* <link rel="stylesheet" href="global-css-and-concat-of-all-css-files-in-the-css-folder-for-domain.css">
* <link rel="stylesheet" href="pages-css.css">

The site will be clever enough to read when the files were modified / database updated for each and create a cache busting new file name. keeps the transfers down as css / js can be cached. (the same 3 “files” for javascript)

Css / javascript should be minified and where possible obfuscation’ated

All images will run through the internal image optimizer. So all url’s for images should be something like */thumbnail/100/90/……* for the cropped version and */image/1000/1000/……* for the resized version

# FileBrowser

Will need a re-write. I want to create a new project just for this. As it can be re-used in multiple projects. Will allow me to keep it updated independently of any project.

Allow drag and dropping of files / organising them etc.

# Ckeditor

I want to give the option where I can to either use ckeditor / bootstrap wysiwyg / plain textarea. Like code blocks have no need to have ckeditor blocks (ckeditor just screws up the code)