

April 14, 2017

University of North Dakota

Technical Recommendations

mStoner

1. HTML Development
2. Integration of External Tools
3. OU Campus

Sections

Overview

As part of the University of North Dakota (UND) website redevelopment initiative, mStoner and UND worked together to review the current web environment including the OU Campus content management system (CMS) and other third-party tools.

This document outlines the core technology components in OU Campus and surrounding technologies for the upcoming UND website.

HTML Development

Standards Based HTML

mStoner builds and tests HTML/CSS with the following in mind:

- **Accessibility.** During design and development of HTML/CSS, we review deliverables to meet Section 508 and WCAG 2.0 AA standards for accessibility.
- **Browser compatibility.** We test our HTML/CSS on different devices and browsers as part of our standard QA process. This includes:
 - Windows 8 and 10 operating systems: Internet Explorer 11; EDGE; and Chrome (latest).
 - Mac OSX operating system (latest): Firefox (latest); Chrome (latest); and Safari (latest).
 - Android 5, 6, and 7: Android Browser (latest), Chrome (latest).
 - iOS 8, 9, and 10: Safari (latest), Chrome (latest).
- **Valid HTML.** We produce valid HTML5 markup, based on W3C standards.

HTML Build Process

mStoner creates HTML markup, JavaScript, and CSS using a front-end development build process using modern tools. This process separates the management of front-end code from the CMS itself. While HTML markup will become integrated with CMS templates, we recommend continuing management of HTML, JavaScript, and CSS in a separate build process.

- mStoner uses **SASS** markup to create CSS.
- HTML will be build on a **lightweight grid system** without the use of a larger framework. Individual components (jQuery, sliders, etc.) will be included as required. Key elements include:
 - Gulp task runner to execute tasks and utilities in the build process
 - Node/NPM libraries and package management
 - Nunjucks templating for HTML
- mStoner will manage source code in source control using **Git** for the duration of the project.
- **Knowledge transfer to the UND team for the HTML project** will take place as part of the training that will occur shortly after the site launch.

Integration of External Tools

Website Search

UND uses a Google Search Appliance (GSA) to index the website and provide website search results. UND has customized multiple front-end outputs from the GSA to deliver search results.

Google has ended sales of new appliances and will end support agreements for appliances in 2018. UND has support for their search appliance through 2018.

The new website will deploy the existing GSA for website search. UND will need to:

- Update styles in the existing front-ends to match the new website design.
- Re-index websites as they relaunch with new content and information architecture and to remove old content.

By the end of 2018 UND will need to plan for the replacement of the Google Search Appliance with a new tool.

Possible Google Search Appliance Replacements

UND does not need to replace its Google Search Appliance until 2018. While the replacement will take place after the relaunch of the website, possible alternative search solutions include:

- OU Search
- Thunderstone (search appliance) - <https://www.thunderstone.com/>
- Apache Solr (self-hosted) - <http://lucene.apache.org/solr/>
- Algolia (SaaS) - <https://www.algolia.com/>
- Cludo (SaaS) - <https://www.cludo.com>
- SwiftType (SaaS) - <https://swifttype.com>
- Funnelback (SaaS or self-hosted) - <https://www.funnelback.com>

With the recent announcement of Google discontinuing their Site Search product as well, it is possible the landscape of search options will evolve quickly over the next 12 months, thus presenting additional options.

Web Event Calendar

UND plans to select a new web event calendar solution to replace the outdated custom-built solution on the current website. The need for an improved calendar was raised by a number of stakeholders during discovery. mStoner and UND will collaborate on the selection criteria and process, and look for a calendar that will:

- Allow events to be tagged with a **controlled taxonomy vocabulary**.
- **Enable visitors to the event calendar to filter** their views based on the taxonomy terms relevant to them.
- Feed event content into various UND webpages.
- Allow event owners to **submit events** through an approval process and ultimately published.
- Use styling consistent with the new UND website design.

As part of the website redevelopment project, mStoner will provide UND with a shortlist of calendar systems for consideration.

Forms

Editors can create web forms within the UND site and use processing scripts to manage processing of forms. Editing forms is problematic, and requires use of an older WYSIWYG editor. There are three primary ways forward for UND in managing forms:

- Choose a **third-party forms tool such as Formstack, FormAssembly, or Wufoo** for editors to use to manage forms and process them. Editors would use a separate tool to create their forms and embed them into CMS pages. These tools have fairly sophisticated form creation tools and integrations for processing, but some of the simplicity of the current mechanism would be lost.
- **Develop a table transformation or use forthcoming OU Campus “components”** functionality to allow editors to continue to create simple forms in the CMS. This would allow UND to use the newer WYSIWYG, and continue to leverage existing processing options.
- **Continue to use the existing form creation and processing tools.** This keeps editing within the CMS but doesn't address issues with the editing process. A future version of OU forms that includes file attachments could eventually help but there isn't a known timeframe for this.

We think the best solution is to **pursue use of a table transformation or OU Campus “component”** to allow the WYSIWYG to be updated but continue to give editors access to build forms in OU Campus. This would also leverage the existing infrastructure for processing.

Directories and Faculty Finder

UND uses custom code and a MySQL database to pull faculty and staff directory information from PeopleSoft into pages within the CMS.

A faculty finder also exists as a separate web application. The new website needs to include these functions which are managed outside of the CMS and included on pages.

mStoner and UND should collaborate on the display template for this functionality, and on the necessary styles to match the output to the new design. The code components that perform data retrieval will also need to be moved into the new OU Campus sites and should be part of an inventory of existing code that should be included in the new site.

Emergency Alerts

UND manages emergency alerts on the website as content items in OU Campus that can be turned on and off in emergency situations. Separately, UND uses Sungard Assurance Notification Manager to allow the campus community to subscribe to emergency text message alerts. In order to handle emergency notifications on the new website, UND and mStoner should:

- Explore the **Emergency Alerts module in OU Campus** for implementing alerts in the new implementation.
- **Evaluate the existing emergency alert code** to determine if it could be retained.
- **Determine if integration between Assurance NM and the website is feasible**, and if these systems can share alert functionality and content.

mStoner will design alert functionality for the page templates, and the functional specifications defined later in the project will detail how alerts should function on the site.

OU Campus

Site Architecture

UND manages the website across several subdomains and sites with OU Campus. This structure will continue, with “sites” within OU Campus for the following:

- www.und.edu
- aero.und.edu
- arts-sciences.und.edu
- business.und.edu
- education.und.edu
- engineering.und.edu
- graduateschool.und.edu
- law.und.edu
- med.und.edu
- nursing.und.edu
- library.und.edu

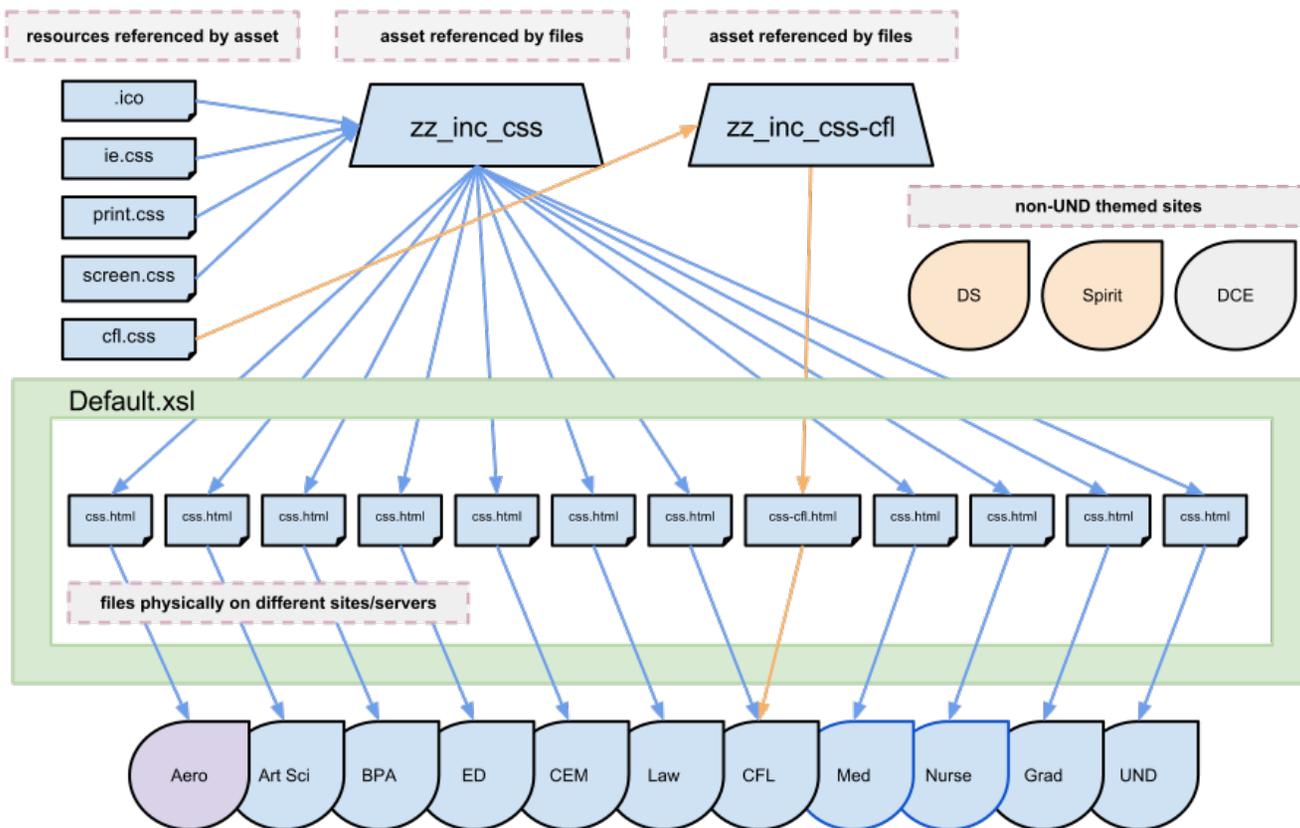
Template development will need to start in a new instance of the www.und.edu site and share components with new instances of the other sites. Code from the existing sites will need to be included in the new sites to retain key functionality.

Publishing Shared Assets Across Sites

UND uses a global structure of OU Campus Assets and include files to allow publishing of changes to files across multiple OU Campus sites on multiple servers. This diagram reflects the current workflow.

The new site will retain this structure and new template development should adopt this pattern.

The future state will include the Aerospace site. We also anticipate consolidating CSS into a single file to share across sites.



Approach to CMS Template Build

When HTML is complete and functional specifications are created, the UND, mStoner, and OmniUpdate teams will collaborate on the approach to building CMS templates. This includes the following considerations:

- Creation of **separate new sites within OU Campus** for the main UND site and subdomains.
- **Management of new assets and snippets.**
- Establishment of **staging URLs and publishing targets** which mirror the production configuration for include files, server side processing, and content sharing.
- Explore use of **OU Campus components and asset templates**, pending their scheduled availability this summer. OmniUpdate announced these new features at the User Conference in early April.
- **Inheritance of navigation** implemented in a similar fashion to the current site. Inheritance may apply to site headers and footers as well.
- **Integration of existing Cold Fusion and PHP code** for desired components. UND has developed include files and assets with code that may be preferred for certain functionality (e.g. breadcrumbs) or required for certain content (e.g. directories).

OU Campus Environment Improvements

Several opportunities to improve the development environment and management of UND websites:

- **Establish a test instance of OU Campus.** mStoner and UND will work with the OU team to establish a test instance of OU Campus in conjunction with the implementation of new templates. The new template build would still take place in a new site within the test instance.
- **Improve the final check configuration prior to publishing within OUCampus,** which has been problematic for UND. We recommend revisiting this configuration after launch, to avoid impacting migration. With new information architecture, content, and code, we can work together to establish settings to make the check valuable. This configuration needs to take into account the governance approach for content.
- The UND team would like to **explore Git workflow** for code going into the CMS. mStoner has a build process for HTML/CSS/JavaScript which should be considered.
- **Evaluate the need for a content delivery network (CDN) for static assets.** mStoner plans to use SVG sprites and CSS and JavaScript compilation / minification. This approach will minimize the number of http requests, which may alleviate the need for a CDN. mStoner and UND should revisit this topic when HTML is complete.

OU Campus Environment Improvements (continued)

- **Resolve issues with Google Analytics integration in OU Campus.** Individual sites have Google analytics integration turned on, but do not always tie in exclusively to the data for the respective site. Configuring views per site with only the relevant analytics will make this information more useful to editors consuming analytics within OU Campus.
- UND should **upgrade the WYSIWYG editor** used in the CMS in conjunction with a solution for form building.

Migration

Migration of content from existing sites into the new page templates is a significant effort. **mStoner will be responsible for several areas of content migration:**

- Base 30-page content migration (30 pages)
- Academic programs content migration (225 pages)
- Manual content migration (300 pages)
- Scripted migration to build site IA using new templates mapped from existing content

Significant collaboration and planning is required to execute the migration. The UND team can aid this process by:

- Establishing an approach for URL redirects for pages that change locations in the IA.
- Creating an inventory of existing code components that need to function in the new templates. These include the faculty finder, faculty/staff directory, A-Z index, form processing scripts, and other points of integration.