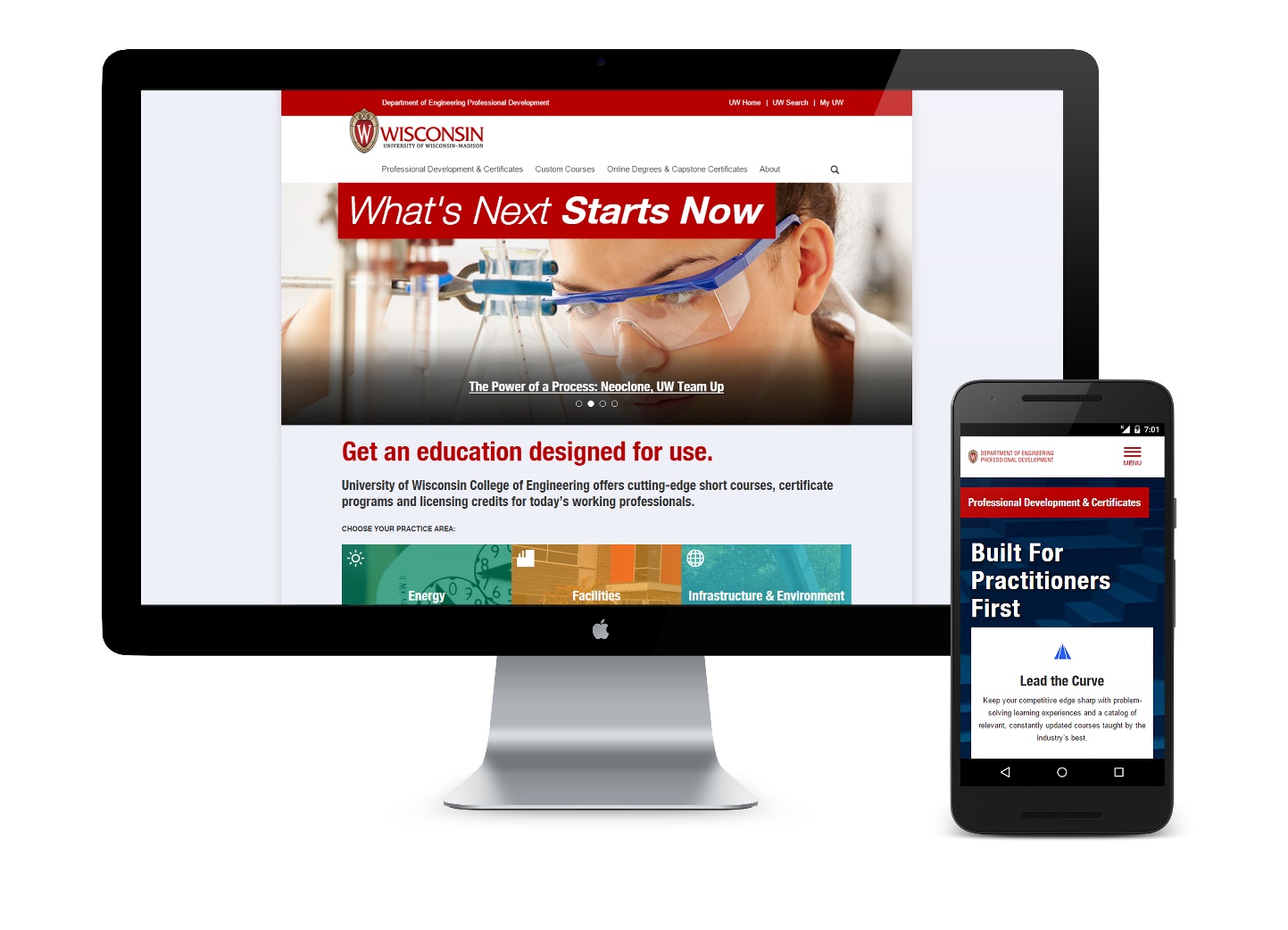
Engineering Professional Development

<https://epd.wisc.edu>

Background

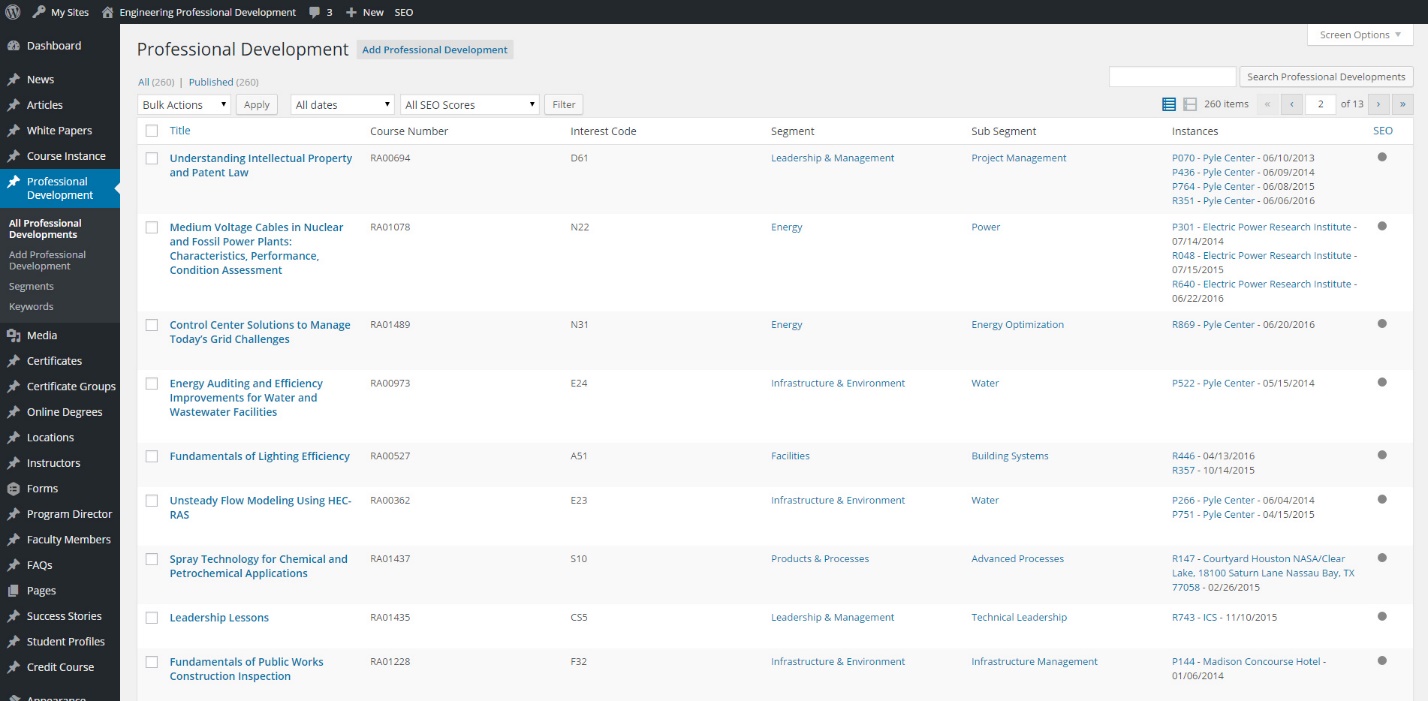
Engineering Professional Development, EPD, has a number of websites that represent its’ products. Several hundred course pages are dynamically fed into the site including data on Instructors, Program Directors, Faculty Members, Locations, Accommodations and Success Stories.



Application Framework

Our solution to fit EPD’s needs was to develop a highly customized WordPress MultiSite installation which allows them to manage user access, marketing materials, form content, SEO and the wide range of features WordPress offers. The theme was built using the [Sage Starter Theme](https://roots.io/sage/) and heavily customized using custom post\_types and [Advanced Custom Fields](http://www.advancedcustomfields.com/).

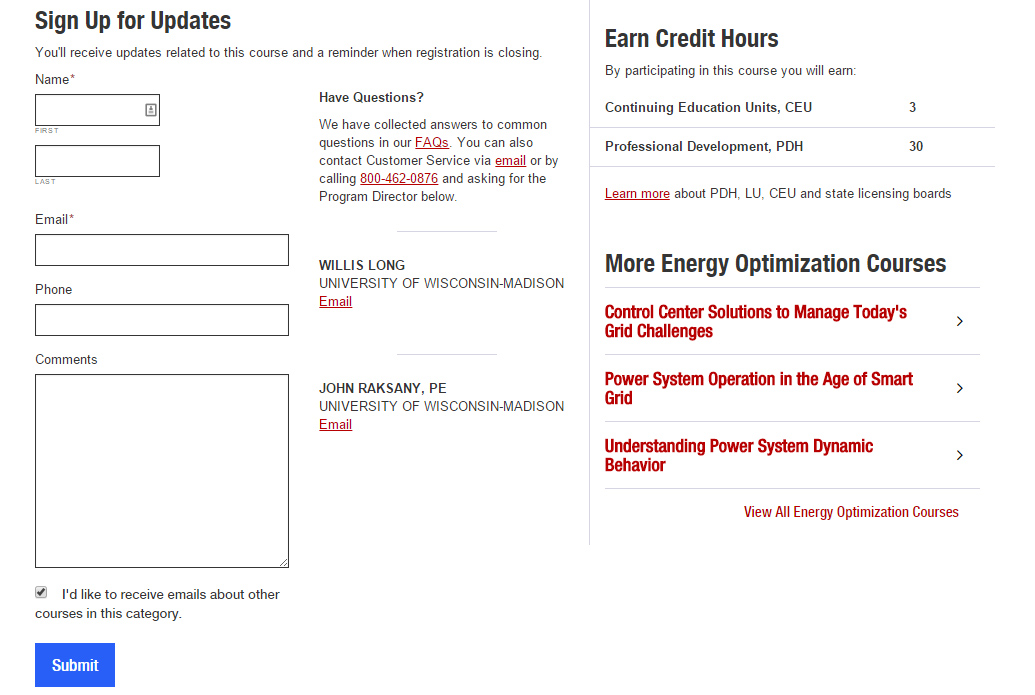
The Courses in the site are Data Driven from Continuing Studies Information System (CSIS). EPD needed a solution that would sync data with CSIS on a nightly basis so the only content entry they have to worry about entering manually is marketing content. We structured the Data Model within WordPress to fit their needs and customized the columns so they could easily verify at a quick glance the Data contained within the system.



The Sage Starter Theme allows for a modern development workflow utilizing the following build tools:

* [gulp](http://gulpjs.com/) build script that compiles both Sass and Less, checks for JavaScript errors, optimizes images, and concatenates and minifies files
* [BrowserSync](http://www.browsersync.io/) for keeping multiple browsers and devices synchronized while testing, along with injecting updated CSS and JS into your browser while you're developing
* [Bower](http://bower.io/) for front-end package management
* [asset-builder](https://github.com/austinpray/asset-builder) for the JSON file based asset pipeline
* [Bootstrap](http://getbootstrap.com/)
* [Theme wrapper](https://roots.io/sage/docs/theme-wrapper/)
* ARIA roles and microformats

[Gravity Forms](http://www.gravityforms.com/) was utilized to generate course interest and manage the course Waitlist. Previously EPD was using iContact for their Form information but Gravity Forms allowed us to extend it precisely to match EPD’s needs so they have no need for iContact anymore. Forms can be configured to utilize custom notifications so when a user requests more information on a specific course, that course will automatically email the associated Program Director who can then get in touch with the user directly.

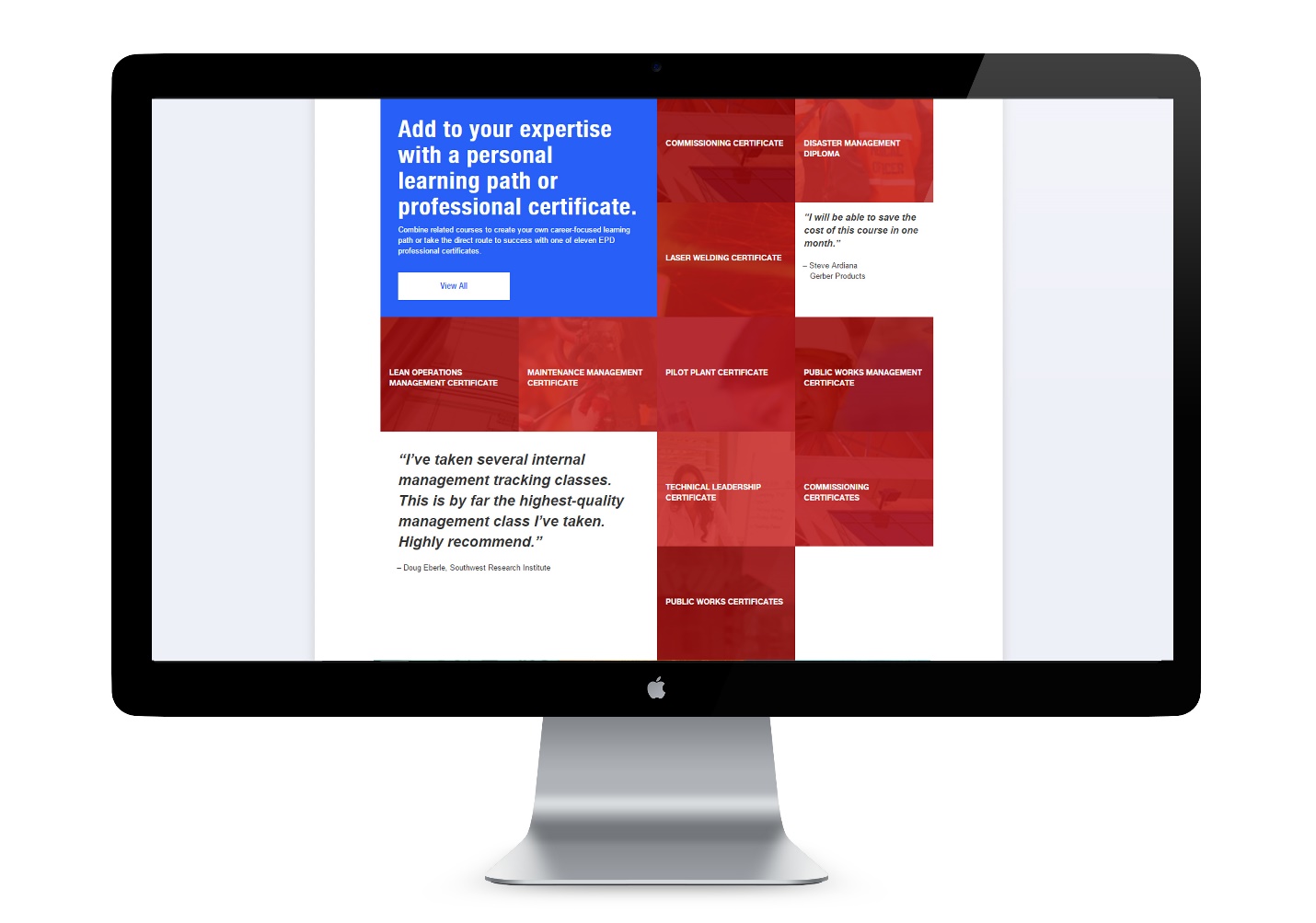


Front End

The redesign was intended to be aesthetically pleasing with a focus on consistency and readability. Each section has a Landing Page Template which is heavily customizable from within the CMS. We utilized [Flexible Fields](http://www.advancedcustomfields.com/resources/flexible-content/) to achieve this functionality which allows the client to customize the content exactly how they need and not have to come ask for a development update.

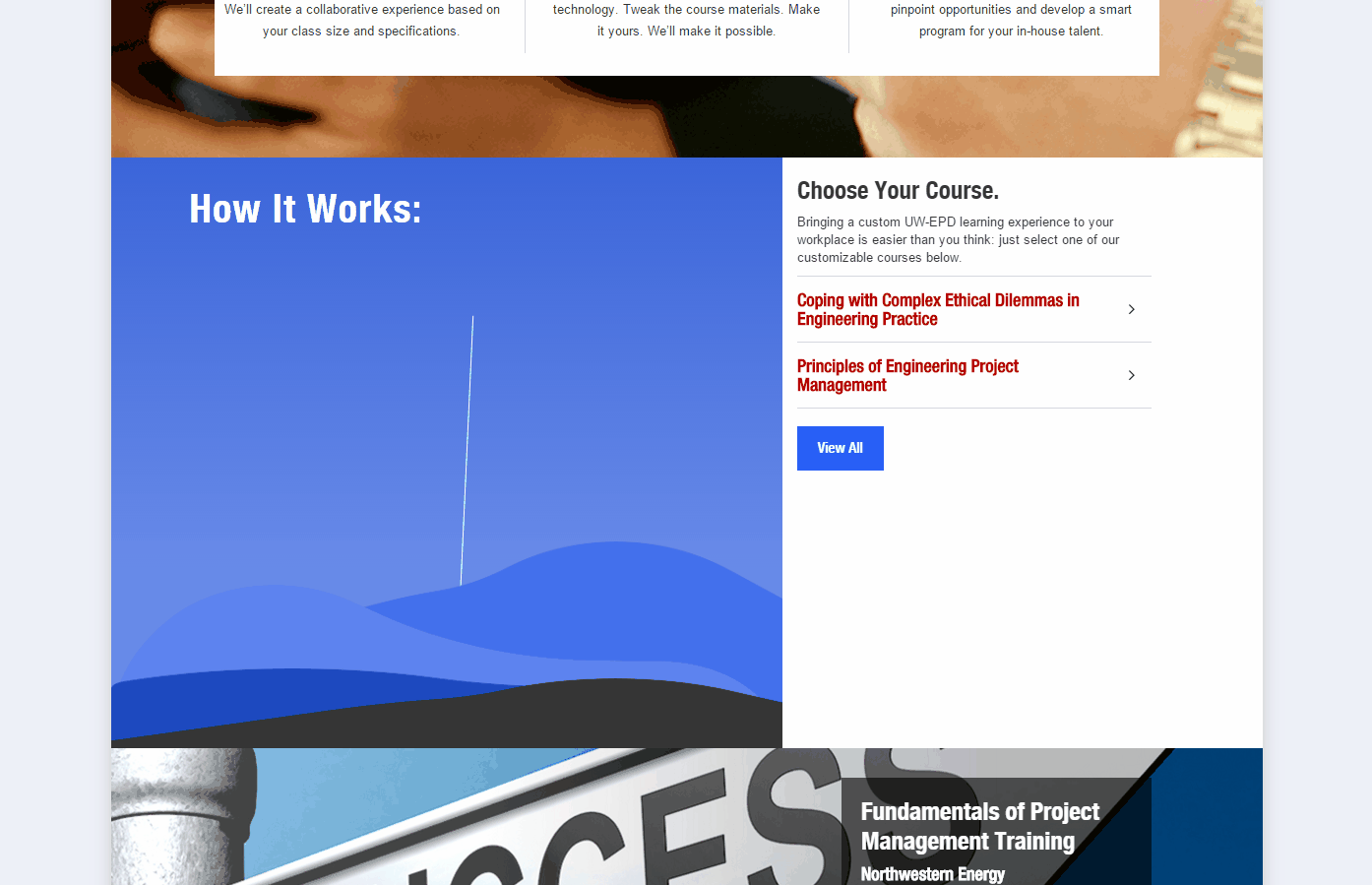


The Online Degrees Landing Page



The Certificates Display.

This is one example of a Flexible Field being used. The client can reorder each boxes positioning as well as swapping out content and making the boxes larger or smaller.

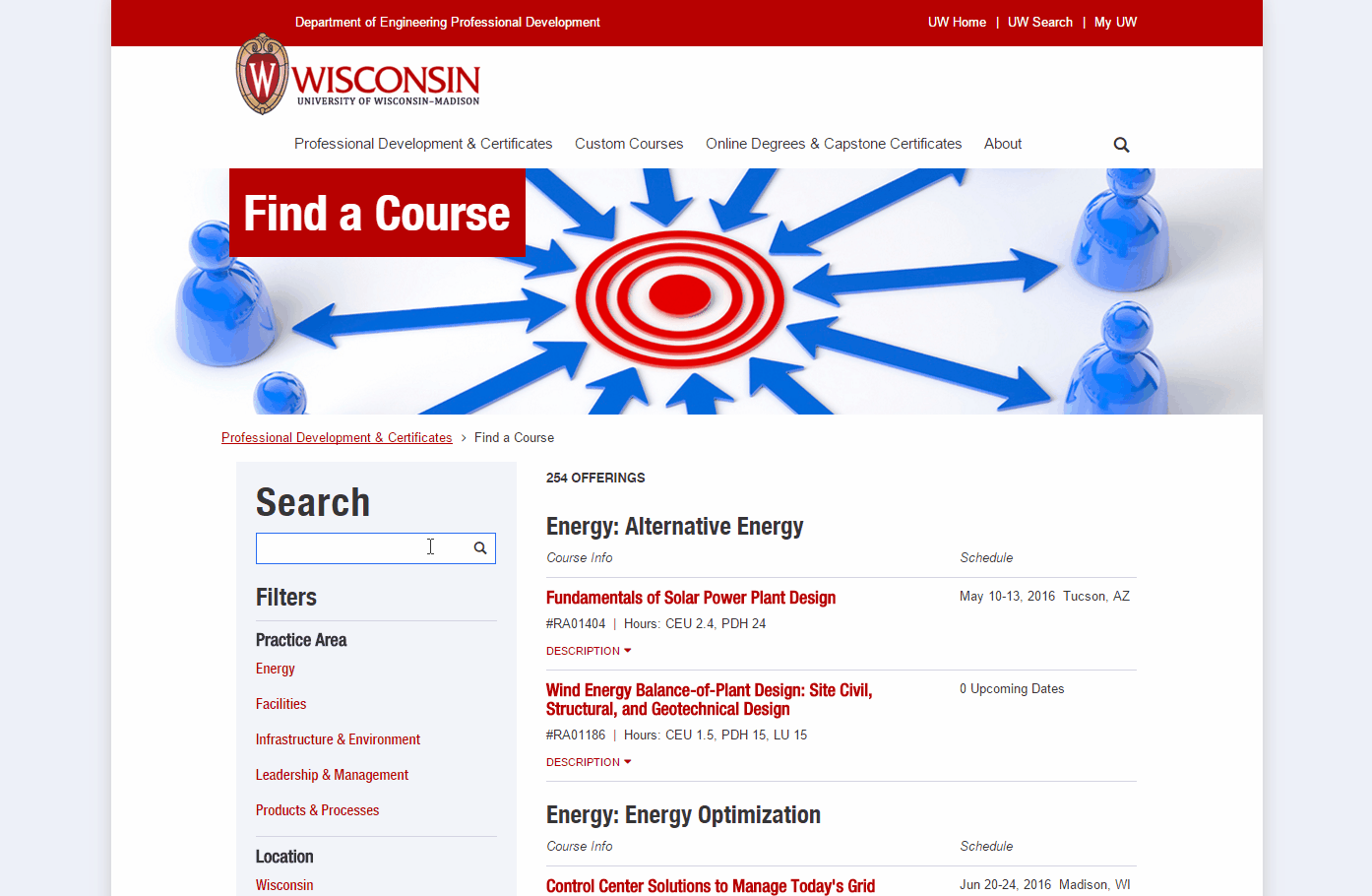


An Animation which is data driven.

Customized Search

The technologies used for the advanced search feature are the [JSON REST API](http://wp-api.org/) on the backend to expose all custom data that we want to filter on. This plugin is very extensible and when configured correctly, very powerful. It allows us to heavily customize the data grouping before it is handled by javascript to reduce the client-side calculations necessary.

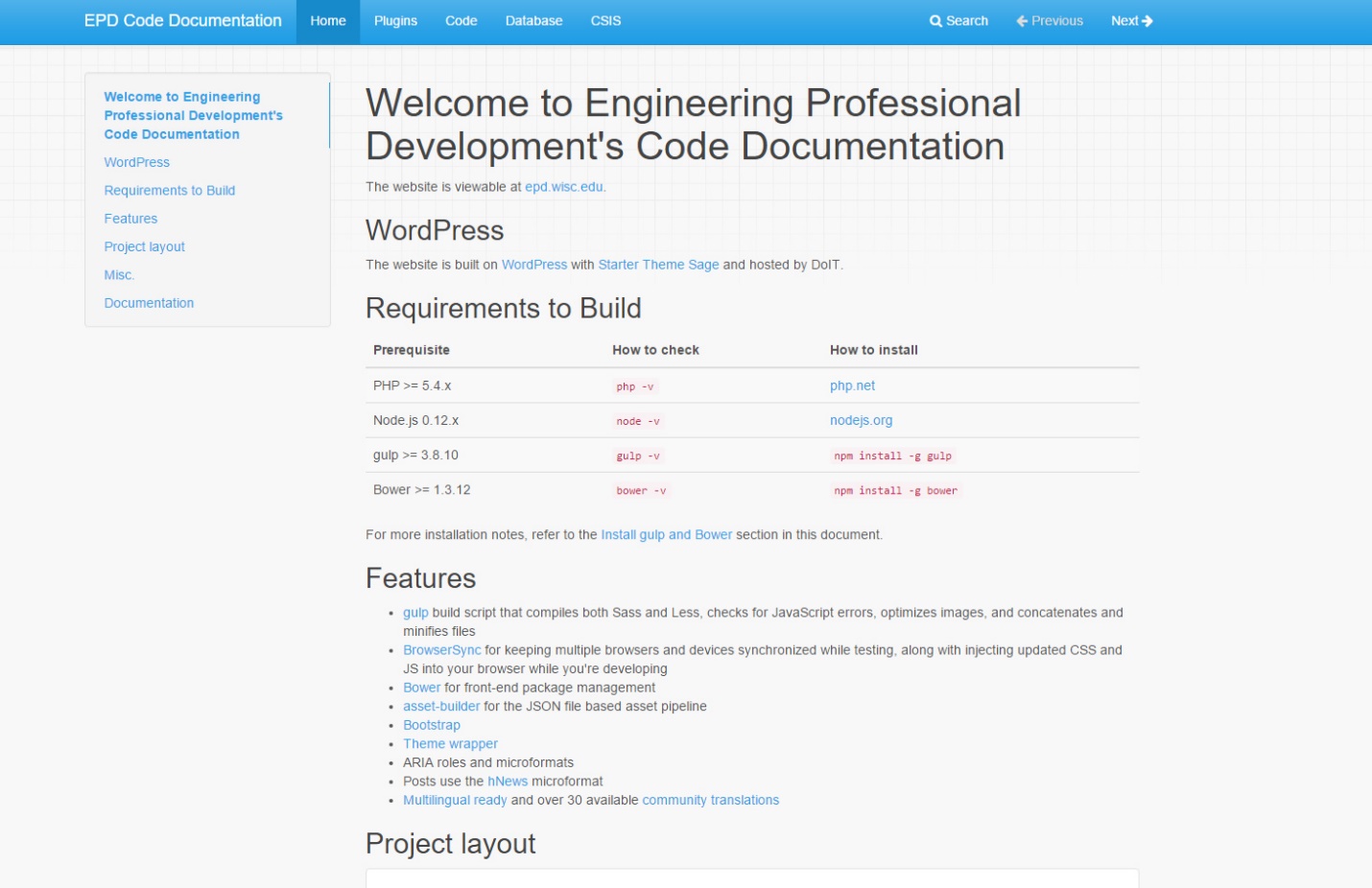
We then utilized [AngularJS](https://angularjs.org/) on the Front-End to load all the custom data and developed custom directives to allow for advanced filtering on more than just a text field. For example, "Date Range", "Location" and "Degree" filtering for courses.



The data is fully loaded client-side which allows for searching and filtering to happen lightning fast rather than having to execute a post-back upon every filter.

Code Documentation

Code Documentation was provided using [MkDocs](http://www.mkdocs.org/) so any future developers would be provided all the information they need to jump right in.



Screenshot of the Code Documentation