

Interactive Web Design through Survey and Adoption of Modern Web-Technologies
Yusof Hussein Al-Wadei

ABSTRACT

Technologies used in modern web development are continuously being created and improved by innovators. These technologies allow a dynamic, reactive, user-customized experience in ways that were impossible only a few years ago. Content management systems (CMS) are an integrative technology which allows developers to utilize different functions to deploy projects much faster. The Drupal 7 content management offers many modules and tools that simplify creating complex functions and user interfaces; in addition, it provides the ability to integrate some of the newest and most powerful web technologies available. In this project, over 15 web-based technologies were surveyed, tested, or implemented within the Drupal installation. Several were implemented to create a live web portal using the Drupal 7 CMS. These included LiveConnect for Facebook integration with social media which provides the capability to restrict access to content through permissions on previously existing social media accounts. JQueryUI was utilized which provides many interactive widgets useful in extending the functionality of a web page. KnockoutJS helps create rich and responsive layouts by using dependency tracking and declarative binding properties. Furthermore, other web-based technologies like Twitter Bootstrap, SignalR and many others can be selectively combined to create a fluid and interactive web design. The final web portal resulting from this research into the integration of multiple modern web technologies is publicly accessible at <http://autotune.roofcalc.com> and has been used for communicating ORNL research and development projects with the public.

INTRODUCTION

Current building energy models (BEMs), using EnergyPlus or other simulations, are unreliable because they have to constantly be calibrated to match actual energy usage data. Currently this is a very costly method because it requires considerable manual effort by experienced skilled professionals which can also be imperfect, non-repeatable, and non transferable. The Autotune project is developing a methodology for automated BEM tuning methodology creates reproducible measured data by selecting best-match input parameters. An Autotuning methodology would significantly reduce the costs of building energy retrofit projects, extend their reach into smaller buildings, and improve the energy savings measurement and verification for performance contracting and other purposes.²

An important aspect of the Autotune Project is providing a public interface to share the simulations data with the world. The web portal for the project's results must include safety and access restriction features as well as being compatible with the latest web technologies to provide an interactive and dynamic website. To complete this task, Drupal 7 Content Management System (CMS) was chosen because of its reputable compatibility with web-based technologies and design fluidity. Drupal 7 CMS is one of the most widely used online content management systems out there today. Drupal is an open source CMS platform that is completely powered by users and community developers. With a community size of over 600,000 this translates into forming one of the leading online applications that supports the latest technologies that the web has to offer. Another great feature about Drupal is the fact that its community is so diverse, including over 228 nations using 181 languages. Therefore, using this CMS allows the Autotune Project to be shared with the widest range of people and possibly using the latest technologies.¹

There are many web-based technologies that are being developed. However, only a small amount of the latest technologies were tested in this experiment. LiveConnect is an application that allows visitors to register on to the website using social media accounts like Facebook.⁶ Once registered, users can then use the same feature to login to the website and gain access to the data. KineticJS is an HTML5 canvas that extends the 2-D functionality of any web page. It's most notable functions are the high performance path and pixel detection for both desktop and mobile applications.⁴ jQueryUI provides low level interactions with the UI providing both extended functionality to the webpage as well providing aesthetic improvements to the website.³ Finally, one of the last large scale technologies that was tested was KnockoutJS, which simplifies dynamic JavaScript user interfaces by applying the Model-View-View-Model pattern. This JavaScript library adds key concepts such as declarative binding, automatic refresh and dependency tracking to provide high-level Document Object Model

(DOM) manipulation of UI elements. In KnockoutJS, complexity can be scaled up without having to introduce inconsistencies.⁶

DRUPAL 7 CMS

In any software development framework, a development phase must be created in which tests and experiments can be carried out before choosing what will be included in the release candidates. During this research an agile software development methodology was followed. Because many web-technologies were experimented with during the development of the website which were then implemented at later phases, it was then necessary to follow this methodology. The Agile software development philosophy is built on the foundation that requirements and solutions must be implemented into the website through adaptive yet structured planning.

Because one of the aspects of this research was to also survey the newest web technologies along with creating a dynamic website, it was important to setup a development cycle that included a local server, a development live server, and a final live server for deployment. Drupal's adaptive CMS proved to be an important feature during this aspect of the research because it was very simple to transfer the website cleanly from one development stage to the next. In order to deploy Drupal 7 on a local server we must also install Apache HTTP Server, MySQL database, and the latest PHP hypertext processor. This would form the local development server where all of the experiments begin. To setup a Drupal website, it is necessary to create the database and a user with access to the database. The database would then be used to store all of the websites data and the created user would serve as the administrator of both MySQL and the Drupal website. A new installation of Drupal would include the default home page that also serves as the node base for all of the articles created in the default theme.

Some of the requirements for this website installation including creating a fluid and interactive design that was also mobile browser friendly, print friendly, included an email server, included a user interface to access the simulations data, and most importantly include access restrictions. After searching through many options available through Drupal, it was narrowed down to two themes that provided the most options and had the most flexibility in design. These were the Fusion and Zen themes. Both themes are very customizable and come with a starter theme kit that allows developers to build an entirely new sub-theme based on the framework of the original theme. However, Fusion was chosen because it provides a graphic user interface that can be used to put final touches on the website, that isn't available in Zen, which provides for a more fluid and interactive design. Using Fusion's starter theme, a

completely new and custom theme was created while also utilizing some of Fusion's properties such as design flexibility to support of and tablet browsing.

Two steps were taken to make the web pages print friendly. The first step was through the design, by grouping certain blocks together and limiting the use of floating images, and without any additional code changes, the web pages were relatively print friendly in any view. The second step was adding some CSS code to control the print output by controlling what is displayed in the print view. After creating this CSS code and adding it to the CSS folder of the theme it was then important to link it to the theme using a print style sheet reference code, "stylesheets[print][]= css/print.css,".

There are several options to creating an email server in a Drupal's website. However, by far the most straight forward method is setting up an SMTP server. First, the SMTP Authentication Support module must be downloaded and installed in Drupal's module's directory. Second, the latest version of PHPMailer must also be installed; however, this folder must be placed within the SMTP Authentication Support directory. Finally, the module must be enabled and configured using the administrator's menu. This module can be configured using a custom SMTP server or it can also be configured to use existing server like Gmail's SMTP server.

One of the most important features is to create a secure user-interface to have public access to the simulations data. To complete this task several options were explored. First, Drupal's content creator was used to create a form to serve as the UI for the database. However, it only had limited options and didn't have the functionalities and flexibility needed. Then, a pure PHP form was created and tested within Drupal. Unfortunately, this did not work because the PHP form didn't integrate well within the page because Drupal limits the amount of PHP code used in pages so that it won't affect the website's security. Finally, after some research, a better option was discovered in Drupal's Form API. In order to employ this option a module had to be created first. To create a module two files are needed, the information (info) file and the module file. The info file only has three lines of text that include the title of the module, a small description, and the version of Drupal it runs on. The module file is where the code for the user interface is written; this code is a mix of Drupal's own syntax in the API and PHP. The API syntax includes many features including access restriction, multi-step forms, creating pages, and form verification functions.

Another important feature of the website are access restrictions. To complete this feature, another module named Open Access, was used. This module is very useful because user permissions can be created with respect to a certain group. Then users can be given a set of permissions by simply adding them to the group. This eliminates the need to recreate user permissions for every user. Another great feature of this module is that it can be used with any

part of the website whether it was pages, blocks, module, or nodes. And if a user does not have the correct permissions not only will they be denied access to the content, but the content will not be visible to them within the page.

WEB BASED TECHNOLOGIES

There were several independent web based technologies that were tested in this research along with those that are extensions of Drupal. One of the first that was tested and implemented in the website is LiveConnect. LiveConnect is a web browser feature that allows Java JavaScript software to intercommunicate within the same webpage or different web pages. In this instance LiveConnect was used to allow to register and login using their Facebook accounts. This was utilized using a Drupal for Facebook module that integrates Facebook user information with the website user. After this technology was configured in the Drupal website, visitors could quickly register using their already established social media accounts, mainly Facebook, to register. Once registered, users can then log into the website by visiting the website while logged in to their social media site. Also, users who registered using the traditional registration form can connect their accounts with their social media accounts to gain the same features.⁶

Another technology that was tested was KinectJS. KineticJS is an HTML5 canvas that extends the 2-D functionality of any web page. It's most notable functions are the high performance path and pixel detection for both desktop and mobile applications. While currently it has no use in the current release of the website, it definitely has some potential use in the future as a demonstration of building energy modules and their different parameters. To add KineticJS to a Drupal website the JavaScript Library file must be added to the libraries directory along with the canvas background picture. Then the canvas code can be added to any page as source code in full HTML mode.⁴

jQueryUI is a great web technology that has elements that can add functionality and order to any page. jQueryUI is a library of JavaScript elements that has low level document object model (DOM) manipulation of UI elements and provides object interactions, widgets, and utilities. Some examples of the UI elements include date picker widget, progress bar, slider, accordion, tabs, and adding properties like draggable, sortable, resizable, etc. The jQueryUI elements library is already included in Drupal with a default grey theme. In order to add the elements, a small PHP script must be added to reference the UI elements from the system and can then be followed by the code for the element and the content all in PHP text mode.³

Another similar technology is KnockoutJS. KnockoutJS simplifies dynamic JavaScript Uis by applying the Model-View-View-Model framework. This JS library adds key concepts such as declarative binding, automatic refresh and dependency tracking to provide high-level DOM

manipulation of UI elements. In KnockoutJS complexity can be scaled up without having to introduce inconsistencies. It provides an alternative to jQueryUI to complete difficult functions with simple codes. Because of this unique framework and separation between the model view and the view of the model, Adding KnockoutJS to Drupal is somewhat more difficult than jQueryUI. First, KnockoutJS requires two different files of JavaScript. One is the KnockoutJS library and the other is the view-model JavaScript that defines the data and the behavior of your UI. Both of these files are placed in Drupal's libraries directory. They are then referenced in the HTML code of the UI, which includes its own syntax for dependency tracking.⁵

CONCLUSION

There were also some other technologies that were tested, but due to time restrictions they could not be given the same attention as the previously listed technologies. Some of these technologies are SignalR, an asynchronous library that aids in the development of real-time interactive web applications;⁷ GUI Twitter Bootstrap, which provides a library of CSS, HTML, and JS files that help to create simple and flexible pages;⁹ and finally Subversion, which is a version control software that allows developers to track the progress of the projects and release candidates.⁸

In conclusion, there are numerous web technologies in existence today; each providing their own features and properties. In today's worlds it is important to keep track of the advancement of technologies as it allows developers to improve their web-design. The goal of this research was to create an interactive and fluid website to serve as the user interface for the Autotune Project. During this process several technologies were explored and employed into the website to reach the goal. While not all of the technologies explored were implemented into the website, simply knowing about them and how to use them creates numerous amounts of possibilities for the future that did not exist before.

ACKNOWLEDGMENTS

I, Yusof Al-Wadei, would like to express my extreme gratitude to all those who me the opportunity to complete this internship and report.

I am deeply indebted to my mentor Dr. Joshua New, R&D staff and Primary Investigator, whose help, mentorship, experienced guidance, gratifying encouragements, and leadership helped greatly in the success of my work.

I would also like to express my appreciation to Dr. Jibonananda Sanyal, Research Associate Postdoc, for his assistance, encouragement, his unwavering confidence in my work, and daily doses of advice which I will carry with me for a long time.

Dr. New and Dr. Sanyal, have made me grow both as a student, as professional, and as a developer. Their practical guidance in the field of Computer Science and in professionalism is very valuable to me.

I would also like to thank the administrators of ORAU, ORISE and the SULI program for their assistance throughout every step of this process.

Also, to The U.S. Department of Energy (DOE) Office of Science which funds the SULI, CCI and VF programs, and giving me this golden opportunity to work in this lab.

10 August, 2012

Yusof Al-Wadei

REFERENCES

1. Drupal, Come for the software stay for the community. <http://drupal.org/>.
2. J. New, J. Sanyal, M. Bhandari and S. Shrestha, *In Proceedings of the 5th National SimBuild of IBPSA-USA*, International Building Performance Simulation Association (IBPSA), "Autotune E+ Building Energy Models." Aug. 1-3, 2012.
3. jQuery User Interface. <http://jqueryui.com/>.
4. KineticJS, Enterprise Class Interactive Web Graphics. <http://www.kineticjs.com/>.
5. Knockout, Simplify dynamic JavaScript UIs by applying the Model-View-View Model (MVVM). <http://knockoutjs.com/>.
6. LiveConnect. <https://developer.mozilla.org/en-US/docs/LiveConnect/>.
7. SignalR. <https://github.com/SignalR/SignalR/>.
8. TortoiseSVN the coolest interface to (Sub)version control. <http://tortoisesvn.net/>.
9. Twitter Bootstrap, Designed for everyone, everywhere. <http://twitter.github.com/bootstrap/>.