Untitled Developers

Herbie Duah

John Loudon

Michael Ortega

Luke Sanchez

Capstone Team Project: MSI Web 2.0

Final Report

CS 486c - Spring 2016

5/4/16

Table of Contents

Introduction

The Martin-Springer Institute has come to us requesting an overhaul to their current exhibit website. The system that we designed made extensive use of the WordPress content management system which is widely used on the World Wide Web. The key goal of the developers of this system were to make a highly robust, modern website that is subject to modification by the client. We accomplished this by pursuing the following features:

- Multi-browser compatibility (Firefox, Chrome, Safari, Opera, Internet Explorer)
- Responsive styling
- Maintainability

The largest risk we faced with this project was ensuring that content was faithfully transferred from the old website. Because

Process Overview

The process that we followed throughout the semester was an *agile* mode of operation. Weekly meetings were held to discuss what the current week's "sprint goals" were, what the status was on the previous week's sprint goals, and to reorient ourselves as developers on the progress that had been made so far. In addition to weekly developer meetings, we met with our mentor (Prof. Jacobs) weekly as well to collect feedback on our progress and elicit any concerns we had as junior developers. The team also dedicated bi-weekly meetings with our client to focus our design on our end-users wants and needs. This allowed us to push iterative prototypes to the client and make changes throughout the entire life cycle. Each team member played a critical role to the project's success:

Herbie - The primary style and graphic artist of the team. Provided numerous iterations of prototypes of the new 'look and feel' for the site before committing source code changes. Herbie used a variety of prototyping tools to simulate site behavior as the design evolved in complexity.

John - The WordPress architect and web programming lead. John maintained the GitHub repository and all other development environments that used source code directly. Worked closely with Herbie to ensure style choices were integrated correctly within the WordPress environment.

Michael - Research and development lead on new interactive maps technology. Crafted a new content module for the website making use of the GoogleMaps API and ensured integration within the final WordPress environment.

Luke - As the team leader, Luke mapped out all required content modules and ensured that the project moved forward. Luke was responsible as the main point of contact between the

development team and the client and also played a large role in the creation of accompanying project documentation related, but not tied to the source code of the deployed system.

All team members dedicated an equal amount of time per week to development and maintained constant contact/communication via a Slack channel set up at the beginning of Fall 2015.

Requirements

In this section we present two tables which summarize our functional and non-functional requirements as outlined by our requirements document created Fall 2015 along with the status of their completion:

Requirement Spec	Completed (Y/N)
The website will include all stories, terms and definitions currently found on the old site	Y
Users will be able to access Exhibit, History, and Resource website directories	Y
Users will be able to view an interactive timeline of World War II	N
Users will be able to view three unique, interactive maps detailing events surrounding the town of Bedzin	Y
Users will be able to access educational materials related to the exhibit	Y
Users will be able to access Martin-Springer Institute contact information	Y
Users will be able to access and website through modern web browsers (Chrome, Firefox, Safari, Internet Explorer and Microsoft Edge)	Y
The website interface will be responsive to mobile devices and reformat itself accordingly	Y

Table 1. Functional Requirements

Requirement Spec	Completed (Y/N)
Pages will load quickly on both desktop, mobile and tablet	Y
Images will be optimized throughout the entire site for web delivery in terms of both speed and quality	Y
Videos will be hosted externally by free services such as YouTube for efficiency in web delivery that we cannot provide ourselves	Y
Scripts, Styles, and HTML will be minified (remove whitespace and a few other tricks to minimize the amount of data to be transferred)	Y
Documentation will be available as both hard copy and digital documents for ease of use	Y

Table 2. Non-functional Requirements

Architecture

Here we present the highest level view of our system, based on the classic client/server architectural pattern:

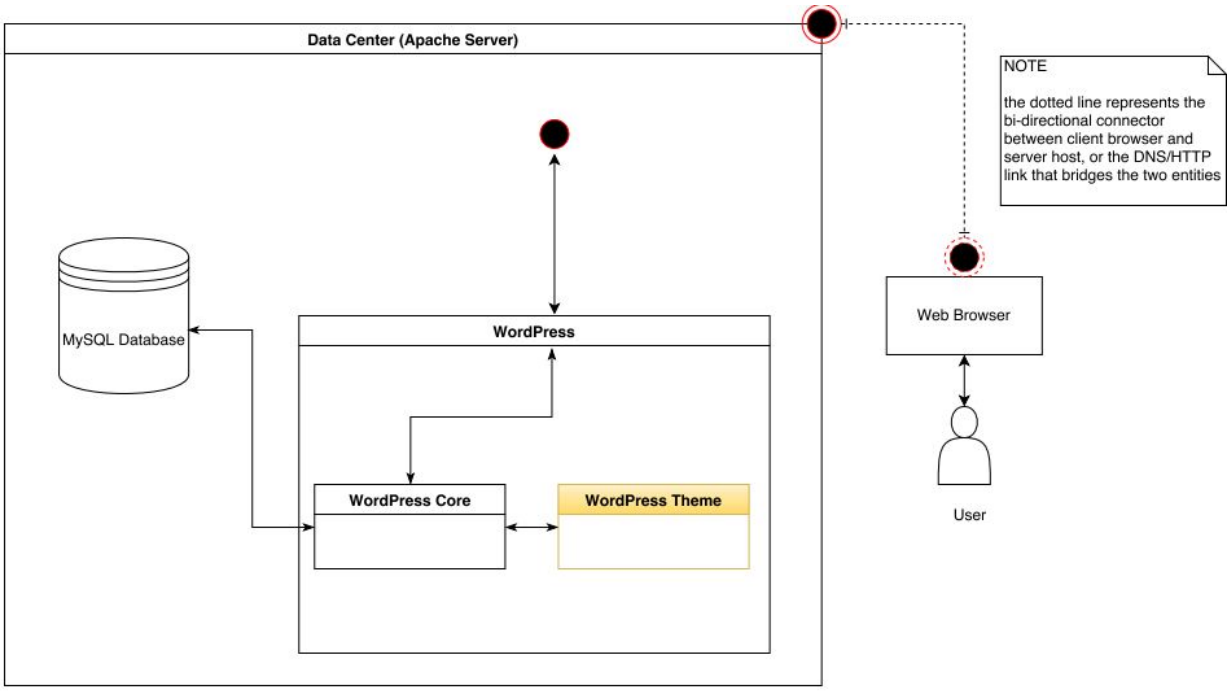

Fig 1. Client/Server architecture

In this view we are able to see the various high-level components that comprise our system; the user's **web browser**, an **apache web server**, the **WordPress** package and the **MySQL database** that is accessed by WordPress.

Component Discussion

Web Browser - Our system is a website which is comprised of HTML, CSS and JavaScript which is processed and rendered by any particular web browser our users happen to be using. We focused development around Firefox, Google Chrome, Internet Explorer and Safari as these are the most popular web browsers that we have come across.

Apache Server - All files that compose the system must be located on some physical hardware and we decided to use the NAU servers that are reserved for faculty and students.

WordPress - The WordPress core is what allows us to create highly customized template files for the client, so that any new content that gets pushed to the site will format itself accordingly to various screen sizes and dimensions.

SQL Database - The SQL database manages all content on the site and is updated automatically through the WordPress core.

HTTP requests are the connector mechanism that allows distributed clients access to the sites content, while custom PHP calls are used server-side to pull the appropriate content into correctly formatted HTML documents.

Prescriptive to Descriptive

Not much changed from our original architectural conception of the website, aside from our final hosting platform. This can be attributed to the very general architecture design that we did up front. Due to time constraints, we decided to use the third-party hosting service that was used by the previous iteration of the site instead of setting up a server with NAU.

Testing

Testing occurred throughout the entire lifecycle of development. Early prototypes layouts were tested against the client expectations and were then integrated into the main code base. Testing of new UI components were tested on local instances of WordPress before being deployed to the production server. The testing that we employed to the fullest extent during this project was capability demonstration with our end-client. Further testing details and results can be found in the *Untitled Developers* separate *Test Plan* document.

Future Work

The client may now add new content to the website, in order to provide further educational resources to any new site visitors. Future web developers that may take over maintenance of the site could potentially optimize the content for search engine exposure. This was discussed early on in development but this feature was scrapped due to time constraints.