

TJHSST Senior Research Project: TJHSST Website Redesign 2007-2008

Martin Elthon

June 11, 2008

1 Abstract

The purpose of this project is a redesign of the TJHSST website backend. Through the use of PHP and MySQL databases, this project will result in a redesigned administrative interface for the TJHSST website. The current state of the TJHSST website is in a state of disrepair, and web pages have to be edited manually. To resolve this, and help with the general overhaul of the current site, this project will form the foundation of the future web site. Although the website is now stable, very little of the functionality has been restored. This project restores the functionality that the site once had, but in a cleaner, more stable interface.

2 Background

The goal of this project is to form the backbone for the new TJHSST website. This replaces a series of two website systems that have been used in the interim of when this project was developed. The two different systems were a temporary PHP system, the most recent system, and an old system based on a xml templating engine called Krysalis. This old system was the standard for a CMS system for quite some time, before PHP became the dominant language. In this system, the website is made using XML documents, creating custom tags for common features, essentially extending XHTML. Unfortunately, the Krysalis system was quirky and unstable, and

at one point completely collapsed. This system was replaced by a temporary PHP system, which does not restore the functionality of the site.

3 Introduction

The current state of the TJHSST website is decrepit. It was written a long time ago in a language that does not exist anymore. After a collapse of the system the previous year, a "hackish" job was done to bring the site back online. However, at a sharp cost—the whole of the core site is not dynamic. This means that the administration has to manually edit the page. Late last year, a team was formed to redo the current site, and this project is a large part of that effort. This project's goal is to provide a new management interface for the administration to manage news posts, and the various dynamic content that the TJHSST site provides. Written using PHP, XHTML, and CSS, using LigHTTPD and MySQL 5 for the web and database server, this new site is using the latest web-development technology to create a lasting site. Hopefully, through careful documentation and good coding practices, the nightmare of the current site can hopefully be avoided in the future.

One of the main goals of this project is to make it very easy to set up, modify, and extend its functionality. With every new year comes new demands for the web site, and new ideas to make it better. Unfortunately with the old system, this was very difficult, since the only people who really understood how the old site worked were those who had been involved in writing it. This new system is very easy to understand and follow.

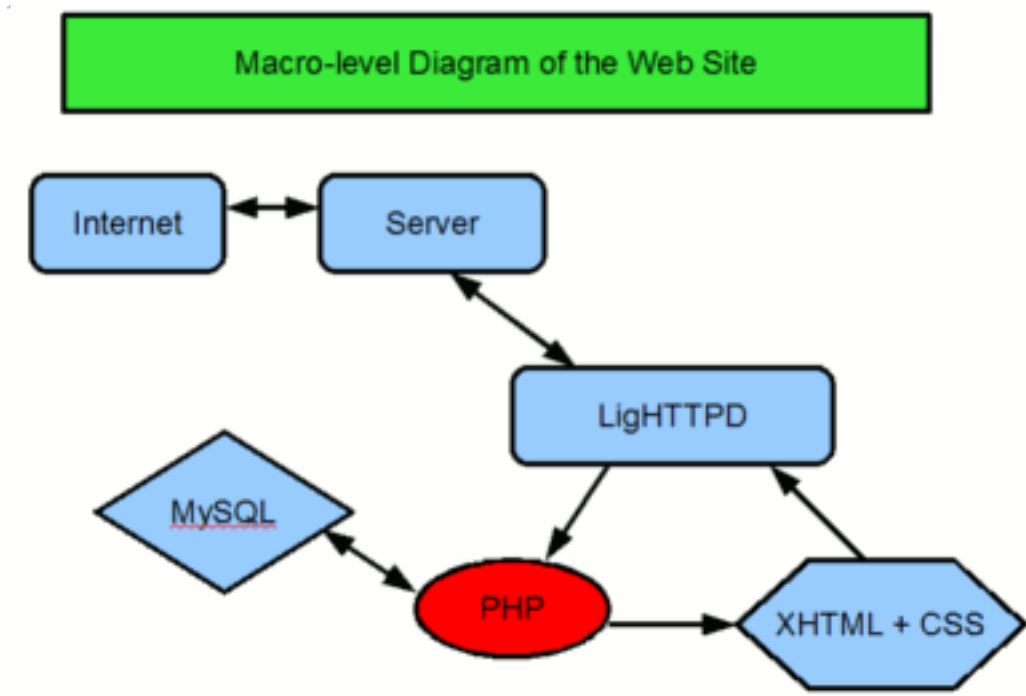
4 Preliminary Testing

As of right now, there is only one user of this backend. He is the main user of the curent TJHSST backend, and can therefore provide useful suggestions as to what features need to be implemented. Basic testing has been done with all the features, such as testing authentication when a user is not authenticated. Meanwhile during these tests, the database was monitored for changes, to see whether a function actually made a change. The almost all of the features have been implemented. Secure login and authentication, account management, staff email management, news management, and spell checking for news submission. The system is flawless and has had no errors af-

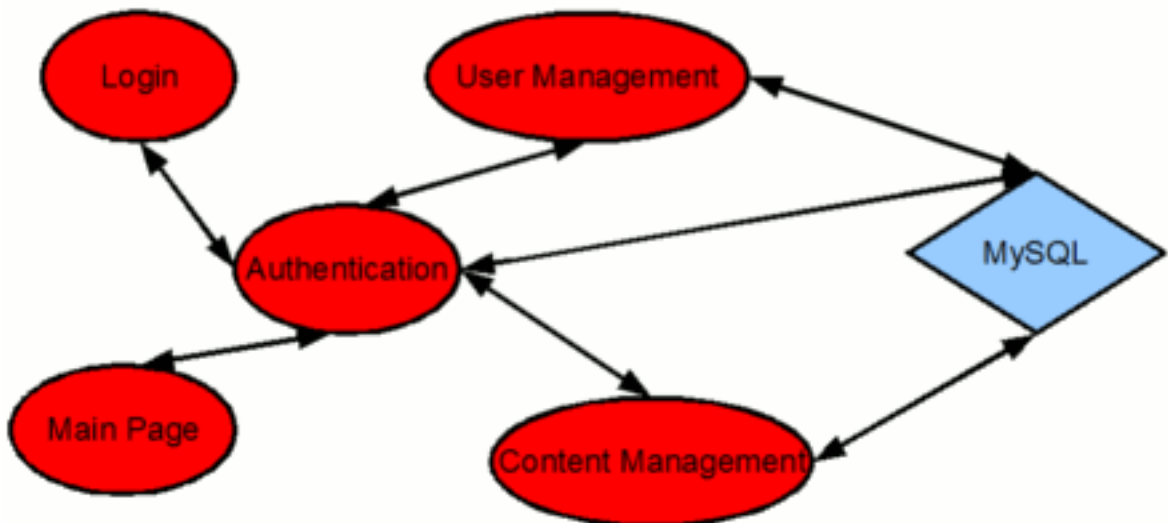
ter the most recent fixes. The front end has been started, and I have handed it off to other students to finish.

One strong advantage to this new design is not only the new features, but also the modular nature of the site. Authentication is used in all pages, but none of the actual authentication is done on a page. Instead, the `auth.php` file handles all authentication requirements, including both checking for a special cookie and redirecting to the login page if the check failed.

5 Development Diagrams



Micro-level Diagram of the Web Site showing dependence



6 Analysis and Results

One of the main goals of this project would be to have an organized layout and structure. This is the most important thing a web site can have for a backend. This allows new features to be implemented quickly and cleanly. Below is a diagram of how the website works as a whole, showing interaction between the internet, the server, and processes on the server. The second diagram shows the modular nature within the program itself.

The results so far have shown the software used to be reliable and speedy. Since the site is under a significant amount of work every day, it is hard to say the true reliability of the software itself. I have never had the system crash or become unusable. So far, all results have been positive. The coding is easy to read, and is very readable.

There are a few concerns with this project, however. When the old website was written, the standard for advanced web design was these XML engines like Krysalis. Nowadays, the standard is PHP. One possible concern is that something will replace PHP as the standard and this new website will have the same problems that the old website had. The hope is that PHP remains the language of choice, or at least is still mainstream until the website is replaced again.

From this project I have learned much about what are the various advantages that each system of web design has. The first comparison is that between Ruby and PHP. The most common misconception is that Ruby can do more than PHP. This is absolutely not true. Not only can PHP do just as much as Ruby, but PHP is much easier to code and has much more community support. The second misconception that people have had about web development is that one has to use Apache 2, since everything else is not as good. I have found that LighHTTPD is actually much nicer to use. It is relatively new, but it already has a significant amount of documentation. The configuration is much nicer, since it is all in one file, and many features are already integrated or easily configurable. Rewrite rules, for example, allow for redirection of web addresses to make them look nicer. This takes a link to `http://management.usa.acme.com/pageload.php?name=sales` to `http://www.acme.com/management/sales`. In Apache, this requires a detailed process of installing and configuring a module, but in LighHTTPD, it is simply a one-line change.