

No Work Like Rework: Issues In The Design Of A Math Test Sign-Up Application

Ghassan Alkadi, Southeastern Louisiana University, USA
Theresa Beaubouef, Southeastern Louisiana University, USA

ABSTRACT

This paper introduces a test sign-up application developed for a math department at a university. The requirements, design, and final software product are presented, along with one very important unexpected problem that arose after completion of the work - the system to be implemented and maintained by the client was not compatible with the platform supported by the client. In spite of the ensuing frustration and challenges, the end result was a success in many ways. In addition, it was a great educational experience for both the students and faculty involved in this project.

Keywords: Open Source Vs Microsoft; Operating Environment; Development Environment; Portability

INTRODUCTION

The Test Sign-Up system is a computerized environment for both math faculty and students enrolled in math courses. Faculty members utilize the system for scheduling the administration of math tests for various courses, and students use it to schedule times for taking their required tests. The system allows for convenience for all parties involved, as well as providing a computerized record system for all activities.

This work began as a student project assignment in a junior level computer science course. The students needed a real world project, and the math department needed a better automated system for allowing students to sign up for taking tests. A faculty member from the math department, who administered the prior test sign-up system, became the client and groups of students in the course formed teams to develop the needed system.

A System Requirements Specifications (SRS) document was produced by the development team and accepted by the client to lay the framework for the production of the Test Sign Up software. The client accepted the specification, and the software was developed and implemented on a Windows server as a web based software to act as an aid in administering specific test sign-ups. The website consists of a series of web pages where students can easily sign up for tests and coordinators can effectively administer these tests. Each of these two basic types of users interacts with the system in different ways. The coordinators initially interact with the system. They provide important data to the system such as test dates and class information. Later on, students sign up for the posted tests. The developers' main goal was to produce software that satisfied these two most important requirements.

The remainder of this paper provides a full description of the system and its functionality designed for the client. A problem arose, however, when the administration rejected approval of the system because it was Windows based, whereas they would only support a LAMP system: "**Linux, Apache, MySQL and PHP.**" So this paper presents the development of the system and the subsequent problems encountered during the conversion from the Windows server environment to LAMP. Comparisons between the implementations on the two platforms are also discussed. Most importantly, this paper discusses the advantages and disadvantages of using the .Net Framework vs. the LAMP programming environment, and briefly describes the students' perspectives on why it was important to learn developing software on both platforms.

BACKGROUND

The Math Test Sign-Up is a new system, designed to be a replacement for the inefficient manual system previously in use by the Math Department (Figure 1). The system uses the university's existing **L**ightweight **D**irectory **A**ccess **P**rotocol (LDAP) server to verify users' credentials and to retrieve users' names and email addresses. The system must rely on a remote web server while being run from a client browser. The web server is the main interaction point for both students and coordinator users.

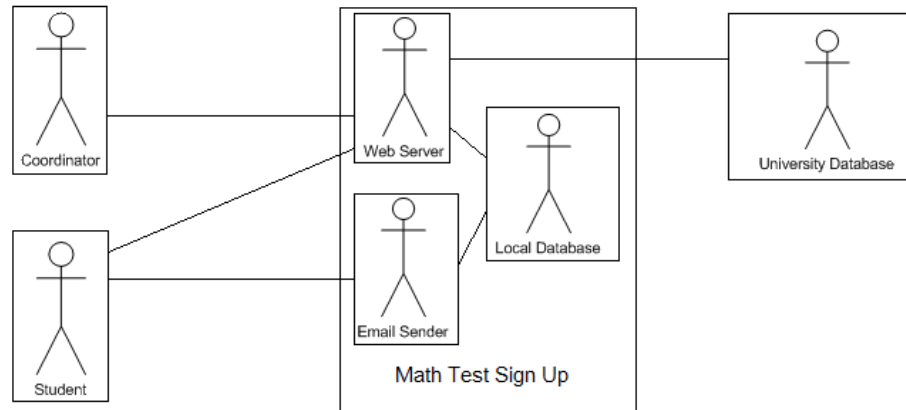


Figure 1: Math Test Sign-Up System

System Functions

Each of the two types of users for this system, coordinators and students, has its own set of functions as listed below.

Coordinator Functions: The system shall allow coordinators to:

- **login** to the system
- **add, edit, view, and delete:** semesters, classes in a semester, or sections in a class
- **add, view, and delete:** students within a section
- **add, edit, view, and delete:** tests within a semester; **delete** test times within a test
- **add, view, and delete:** students within a test time
- **view the number of:** people signed up at a particular testing time, no-shows at testing times that have already passed, students in individual classes signed up
- **view** a list of students by section who have and who have not signed up for a test
- **view** the number of students who did and did not show up for each time slot
- **batch import:** classes, sections, and students for a semester

Student Functions: The system shall allow students to:

- **login** to the system
- **view** available testing times for a test
- **sign up** for one testing time per test, prior to the cut off time (30 min)
- **remove** oneself from a testing time
- **receive emails** from: the system once testing times are first available, and again twenty-four hours prior to the testing time as a reminder, giving the date, time, and location

OPERATING ENVIRONMENT

Originally, the system was to be run on a Windows server running ASP.NET and SQL Server 2008 with users accessing the system using any modern web browser such as Internet Explorer 7 or later versions, or Mozilla Firefox. Additionally, user documentation, made accessible online through a page in the website, must be available to guide the user. The only link to an external system is the link to the university directory for verifying the user credentials. The university directory may also provide student names and email addresses. LDAP is used to access the directory. The functions of the system used by the coordinators include the following Use Cases:

- Use Case: User **Login & Validate** User
- Use Case: **Add/Edit/Delete/List** Semester(s)
- Use Case: Confirmation
- Use Case: **Add/Edit/Delete/List**: Classes, Sections, Test, Test Times
- Use Case: **View** Report
- Use Case: **Batch** Import

Use cases describing the functions of the system used by the student users include:

- Use Case: **View** Available Testing Times
- Use Case: **Sign Up** for Test Time
- Use Case: **Remove** from Test Time
- Use Case: **Send Email** Reminder
- Use Case: **Send** Available Test Email

Sample screen shots of the system are shown below. Figure 2 shows the student side and Figure 3 shows the coordinator, or administrative, side of the application. Because the application is in operation with real student data, the students' names were obscured in the figure to protect their privacy.

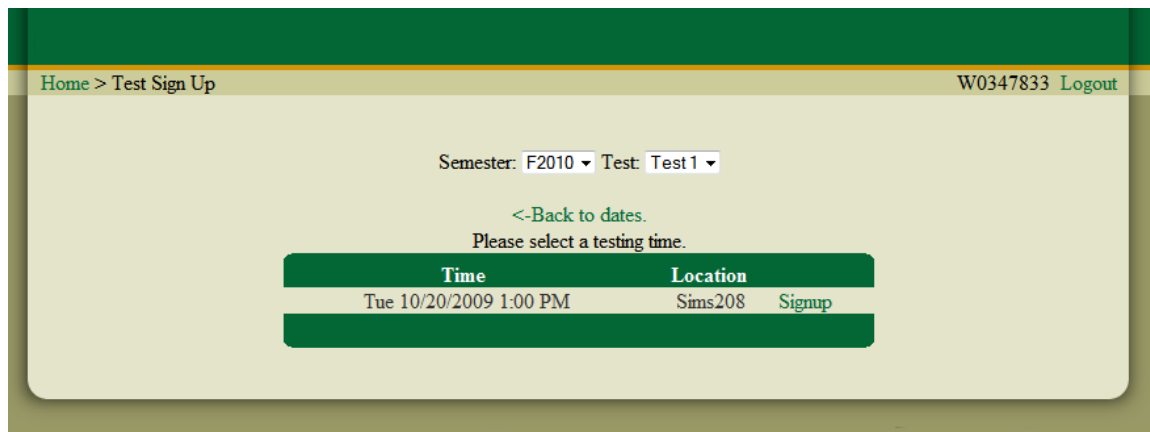


Figure 2: Student View of Test Sign-up System

Home > Semesters > Tests > Test Times > Students admin1 Logout

Semester: F2010 Test: Test 1 Test Time: 10/20/2009 1:00:00 PM Sims208

Quick Attendance Scan W#:

W_Num	First_Name	Last_Name	Attended	
W0266044			<input type="checkbox"/>	Edit Delete
W0339102			<input type="checkbox"/>	Edit Delete
W0310799			<input checked="" type="checkbox"/>	Edit Delete
W0347833			<input checked="" type="checkbox"/>	Edit Delete

[Add Student to Test Time](#) [Search For Student](#)

W# W#
 Add First Name
 Add Last Name
 Search

[View Report](#)

Figure 3: Administrator's View of the Student Test Sign-up System

DEVELOPMENT ENVIRONMENT

One of the main differences between ASP.NET and PHP is their development environments. ASP.NET's development environment is built into Microsoft Visual Studio (VS). VS includes everything a programmer needs to get started in ASP.NET, including a built-in development server, SQL server explorer, *intellisense* (for ASP.NET controls, C# / VB, even JavaScript), CSS stylesheets builder, and most importantly, a powerful debugger (even for JavaScript). When developing in ASP.NET all of the development is done inside of VS.

There are several development environments for PHP, such as Eclipse PHP and Zend Studio. However, they do not provide the same functionality as VS does by default. To obtain the same functionality in PHP, a programmer needs to install several plug-ins and other programs. For the Test Sign-up application, only Notepad++ was required. Its only feature for PHP is basic syntax highlighting.

From the developers' perspective with this project, ASP.NET won "hands down" on development environment, since VS provides many features not found in other environments. Various aspects of the two environments are discussed next.

Database

For the ASP.NET version of the application, MS SQL Server 2008 Express for database was used. LINQ to SQL ORM is used to communicate with the database without the need for writing any SQL queries. "LINQ is a set of extensions to the .NET Framework that encompass language-integrated query, set, and transform operations." "LINQ to SQL is an object-relational mapping (ORM) framework that allows the direct 1-1 mapping of a Microsoft SQL Server database to .NET classes, and query of the resulting objects using LINQ". To set up and manage the database, SQL Management Studio was used. SQL server combined with SQL management studio and LINQ to SQL provided a platform for rapid development. Writing LINQ statements with *intellisense* was immensely faster than writing complex SQL statements. Moreover, SQL Management Studio was much more intuitive than the PHP version, and it offered a complete database management environment.

MySQL was used for the database in the PHP version. PHP Data Objects (PDO) was used to provide an abstraction layer between the databases. PDO allows the developer to completely change the database engine by only changing the connection string. SQL queries were hard coded to communicate with the database, and the software tool phpMyAdmin was used for database management.

Portability

VS, ASP.NET, and MS SQL server can only run on a windows operating system. ASP.NET must run on a windows server with MS IIS (Internet Information Services) web server. This limits the use of these tools to windows platforms.

PHP and MySQL, however, can run on a wide variety of operating systems. In addition, they run on a variety of web server applications. Obviously, PHP is more portable and potentially requires cheaper software to set up than the MS systems do.

Ease of Use

Developing in ASP.NET is an easy process with ASP.NET controls. Instead of writing HTML, a user can simply drag controls on to the webpage to produce the desired HTML output. Partial page updates using AJAX are as simple as dragging two controls on the page: a script manager and an update panel. “When these technologies are combined in the AJAX model, web applications are able to make quick, incremental updates to the user interface without reloading the entire browser page. This makes the application faster and more responsive to user actions”. ASP.NET also requires a backend language, either C# or MS VB, which provides a separation from design and backend code.

Alternatively, developing in PHP requires a programmer to write all the code in order to produce the desired HTML output. However, there are third party plug-ins that can ease the amount of coding. Implementing AJAX requires that all the JavaScript be coded manually. In the Test Sign-up project, jQuery, a JavaScript library, along with web services to implement the AJAX functions, were used.

CONCLUSION

In this paper we discussed the issues associated with the development of a Test Sign-up system for a math department. In general, the Windows based application was easier to use for the development team and it allowed for faster implementation of their design than did the PHP environment. The PHP environment, however, was cheaper in terms of software costs, hence its support by the administration, and it allowed for greater portability. In particular, we presented the issues associated with the decision of development environment and platform for operation and the problems faced when a software design team is forced to switch from one application environment to another after the project has been developed.

Overall, the development team was not pleased that the project work needed to be re-done for a different system after having completed the work in a manner that not only met, but exceeded the client’s expectation. A great educational experience was gained because the students were forced to change their development environments while satisfying the client’s requirements. In retrospect, however, the team members realized that in doing the project in two different ways, they learned a great deal more about systems and languages - experience that will help to guide their future decisions as software engineers.

AUTHOR INFORMATION

Ghassan Alkadi earned the Ph.D. in Computer Science from Louisiana State University in 1999. Since that time Dr. Alkadi has taught a variety of computer science courses at all levels, and is currently an Associate Professor at Southeastern Louisiana University. Dr. Alkadi’s research interests include Software Engineering, Information Systems, Object Oriented Design, and computer science education. E-mail: galkadi@selu.edu. Corresponding author.

Theresa Beaubouef earned the Ph.D. in Computer Science from Tulane University in 1992. Since that time she has taught a variety of computer science courses at all levels, and is currently a Professor at Southeastern Louisiana University. Dr. Beaubouef’s research interests include uncertainty in databases, data mining, spatial databases, and computer science education.

REFERENCES

1. “Ajax - MDC,” <https://developer.mozilla.org/En/AJAX>
2. Flasko, Elisa, “Introducing LINQ to Relational Data,” Microsoft Data Platform Development Technical Article, January, 2008, <http://msdn.microsoft.com/en-us/library/cc161164.aspx>
3. jQuery: The Write Less, Do More, JavaScript Library, <http://jquery.com>, Martin, Robert C., *Clean Code, A Handbook of Agile Software Craftsmanship*, Prentice Hall, 2009, Pearson Education, Inc.
4. “Lightweight Directory Access Protocol (LDAP),” <http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/ref-guide/ch-ldap.html>
5. Microsoft Developer Network, “LINQ,” <http://msdn.microsoft.com/en-us/netframework/aa904594.aspx>
6. Microsoft, “.NET Framework Developer Center,” <http://msdn.microsoft.com/en-us/netframework/default.aspx>
7. Microsoft, “SQL Server 2008 Express,” <http://www.microsoft.com/express/sql/default.aspx>
8. “Microsoft Visual Studios 2008,” <http://www.microsoft.com/visualstudio/en-us/default.mspix>
9. Murray, G. and Ball, J., “Including Ajax Functionality in a Custom JavaServer Faces Component,” Sun Developer Network, <http://java.sun.com/javaee/javaserverfaces/ajax/tutorial.jsp>
10. “MySQL,” http://mysql.com/?bydis_dis_index=1
11. “phpMyAdmin,” http://www.phpmyadmin.net/home_page/index.php
12. PHP Group, “PHP Data Objects,” December 2009, <http://www.php.net/manual/en/intro.pdo.php>
13. Pressman, Roger, *Software Engineering, A Practitioner’s Approach*, McGraw-Hill, Seventh Edition 2010.
14. Timberlake, B., “Building a LAMP Server,” <http://lamphowto.com/>
15. “VS 2008 Web Designer and CSS Support,” <http://weblogs.asp.net/scottgu/archive/2007/07/25/vs-2008-web-designer-and-css-support.aspx>
16. Zend, “PHP Web Application Server,” <http://www.zend.com/en/>