DATABASE-DRIVEN WEBSITES: A WORKING COURSE MODEL

Paul J. Kovacs, Robert Morris University, kovacs@rmu.edu

ABSTRACT

As a historical study of the Internet would reveal, the original rationale of the World Wide Web was to find and display information, and Web sites were static or informational in nature. However as the Internet grew, the demand for more useful Web sites has driven the merger between Web sites and data base technologies. An E-commerce site and other Internet applications provide a Web interface to access information stored in database systems. Because these applications are data-intensive, information systems consisting of Web pages coupled with database management systems is the future for Web system applications. This paper discusses a working course model for creating a data base-driven Web site.

Keywords: Database Web Sites, Dynamic Web Sites, Web Database, Database Web Interaction, E-commerce

INTRODUCTION

As a historical study of the Internet would reveal, the original rationale of the World Wide Web was to find and display information, and Web sites were static or informational in nature. However, after the Internet grew users began to be aware that greater interactivity would make the World Wide Web more useful. As a result, the demand for dynamic, interactive, data-intensive Web sites drove the merger between Web sites and data base technologies. An E-commerce Web site making use of a database for billing information and online orders or a site that displays current stock market information are example of database driven Web sites. Because these applications are data-intensive, information systems consisting Web pages coupled with database management systems is the future for Web system applications.

A number of mature technologies are widely used for Web access. The common Application Program Interfaces for data access strategy consist of Object Linking and Embedding Database (OLE DB), Open Data base Connectivity (ODBC), Java Database Connectivity (JDBC) and ActiveX Data Objects (ADO). Microsoft's ASP.NET, Sun's Java Server Pages (JSP) and Allaire's Cold Fusion are commonly used web access and programming tools. Additionally, the three-tier architecture has replaced the two tiers model for thin client and security reasons. Extensible Markup Language (XML) is the next generation standard technique for data transferring and communication within heterogeneous systems and among enterprises [9].

If one of the objectives of an information technology program is to teach students how to develop systems using mature technologies, then it is necessary for information technology educators to develop relevant courses of study. Because technology areas such as web-based applications are being increasingly used by organizations to support their work, students will also need these skills to compete in the job market. [10]. Keeping Information Technology students current with Database Technology should be a priority for those teaching Database Management [2]. This paper details the features of a Database Driven Web Design course of study as developed for Computer Information Systems majors.

LEARNING ENVIROMENT

The purpose of this course is to design and deploy a database driven Web site in a team environment during one academic semester. To accomplish this, it is necessary to have a Web development environment consisting of Web server software, Web server based programs, database software, and web authoring/site management software. Additionally, the appropriate classroom and hardware are necessary.

Software

The most important concern for such a course is the selection of the Web server software. This can determine the database selection, the web authoring and site management software, and the web access and programming tools. A review of numerous Web hosting services indicated that the two major web servers are Apache and Microsoft Internet Information Services (IIS). Additionally, according to Netcraft's September 2005 survey, Apache gained 703K sites (69.91 percent of the market) and Microsoft Internet Information Service (IIS) added 217K sites (22.71 percent the market).

Microsoft IIS is the choice for technology programs using the Windows environment. IIS is generally a point and click affair and interfaces with Microsoft web site management and development tools. If the school is currently using Visual Studio.NET, than IIS must be the choice. Visual Studio.NET supports ASP.NET, which does not interface with Apache. ASP.NET also allows for the programming of the web site in Visual Basic, C# and Java and offer database connections via ADO.NET. Microsoft Access can be used to create data-driven Web sites. With Access, it is relatively quick and easy to create a Web based database. Access also supports XML, and Access can be used with both classic Active Server Pages (ASP) and the newer .NET languages. However, because Access does not scale as well as SQL Server, Visual Studio.Net supplies the MS SQL Server2000 Desktop Engine (MSDE).

Apache is far more robust and flexible and scales better than IIS. However, this flexibility comes with a slight drawback because it is a little more complex to learn than IIS. Apache works much better on a Linux/Unix platform then on a Windows platform. Apache interfaces well with web programming software such as PHP and as well as MySQL database software often referred to as a LAMP Platform.

Additional software includes Web authoring/site management tools such as MS Front Page or Macromedia Dreamweaver as well as Graphic design software such as Adobe Photoshop or Macromedia Fireworks.

Classroom/Hardware

A computer laboratory is the recommended environment in which to teach this course. Specifically, the lab should include a permanently mounted overhead projection system, instructor's station equipped with a microprocessor, VCR, document camera, and student workstations configured as LANs. Each student workstation or at least one workstation should have Web server software installed and running, whether that software is Apache or Microsoft IIS. In addition to the workstations, a department or school Web server is desirable with port 80 and 21 open to allow on-line Web page development and file transfer operations. Additional hardware within the computer lab can include a document printer, document scanner, and access to a digital camera.

COURSE PREREQUISITES

In order to develop data-driven Web applications, students must use multiple tools to integrate Web pages, style sheets, Web servers and databases. This course model assumes that students are exposed to many of these tools in previous courses. The intent of this model is not to teach HTML, ASP.NET, PHP, or database design. Rather, the purpose is to allow the student to make use of these tools to design and deploy database driven Web site during one academic semester.

Introductory Web Design Course

Many introductory Web design courses teach web-authoring software such as Microsoft FrontPage or Macromedia Dreamweaver. Although these editors allow for Rapid Application Development, they also generate the HTML and cascading style sheets (CSS) for the Web designer. Students with knowledge of only a Web authoring software package may have difficulty when required to embed JavaScript, Visual Basic Script, or PHP code into HTML. Students need a working knowledge of HTML and CSS. Furthermore, students should not only know how to position HTML and CSS elements but also how to organize web page designs, control the visibility and presentation of HTML layers, and write client-side scripts in JavaScript.

Object-Oriented Programming Language

HTML is a markup language and alone will not allow interactivity to a database. JavaScript is an object-oriented language used to create interactive features within a Web page. However, JavaScript only works within a Web browser and does not take advantage of the server side of the Web. To develop a fully interactive Web site that request special processing on a Web server such as database queries or handling form data, a program running on a Web server must accept and process data. Therefore, it is necessary for the student to have prior instruction in a course that focuses on object-oriented programming and design such as Java, C#, or Visual Basic.

Database Management Systems

Since the use of relational database technology is crucial for this course model, students need more than a limited exposure to database systems as taught in many introductory IT courses. Students need instruction in a course that provides experience with database design and implementation based on a thorough analysis of requirements and information modeling. The course should include hands-on experience with the design of and interaction with a database management system. In such a course, students would learn to create data models of user's needs, gain foundation skills in database design, and learn to use standard Structured Query Language (SQL) to interact with databases.

COURSE ORGANIZATION

The phases for the course are listed in Table 1, followed by a detailed discussion of specific issues regarding each phase.

Table 1. Course Phases

1. Select Application	4. Design Web Site
2. Establish Workgroups/ Time Constraints	5. Test/Deploy Web Site
3. Conduct Front/Back End Analysis	6. Student/Site Evaluation

Select Application

Researchers have investigated project-based learning in a wide selection of settings and disciplines and have generally found it to be effective in increasing student motivation and improving student problem solving, providing students with an integrated learning situation, and addressing different learning styles. Project-based learning motivates the students to do additional work, illustrates to the students the value of the material covered, and most importantly, provides practical experiences that enriches the student's academic experiences. [1, 3, 5]. As a result, the selection of an appropriate application or project is extremely important. This project can be simulated or real-world.

The instructor initiates simulated projects and acts as the user. A simulated project allows for a great deal of control over the system requirements and incomplete projects are not an issue. The main thing to avoid in the selection of a simulated project is to not select an application whose subject material is beyond the scope and interest of the students. A topic that is of interest to all students and easily compartmentalized will permit a focus on the objectives

The instructor, the student, the university, or a business can initiate a real-world project. Since the requirements come from an actual end-user, real-world projects can have authentic meaning to the students. Additionally, real-world projects can promote industry/university relationships and can be a potential employee source for a student.

However, real-world projects involve a bit of risk taking in a non-controlled environment and an incomplete project can be a major issue for the end-user as well as the instructor. Selecting a real-world project means paying considerable attention to scope and time constraint issues because the students have only a semester to complete the project. Additionally, given the level of student expertise, it may be impractical to implement some requirements of real-world projects, such as credit card processing.

No matter if a real-world or simulated project is selected, the application needs to incorporate some use of client-side scripting (for example, form validation) and cascading style sheets (CSS) to give the pages a consistent look. The application needs to have at least two views: the visitor's view and the administrator's view, which is a password-protected back end for accessing and updating the database. Optionally, the application could make use of many other features such as server side include files, file transfers, email generation, a secure socket layer, and possibility video and animation.

This course has made use of both simulated and real-world projects. Each had no more than six components. One workgroup designed a real-world web-based auto parts application in which users could search for auto parts. Another group developed a stimulated project that involved a Pizza/Movie rental store that took orders on-line. Each project had no more than four related database tables.

Establish Workgroups/Time Constraints

The construction of a Web site is accomplished as a group as part of a Web development team and group size is determined by the class size and the number of project components of the Web site. However, a recommendation is to avoid groups that are too small or too large. Experience suggests that no more than eight per work group is an ideal number.

The instructor and each workgroup must decide on time constraints. This can be accomplished by (1) having the students manage their own time with a final due date or (2) requiring that a specific component of the Web site be completed throughout the semester. The first approach gives the students the responsibility for completion of the project and the second approach gives the instructor control. The author's experience suggests that the second approach works best.

Conduct Front/Back-End Analysis

Following the formation of work groups and the establishment of time management constraints, each work group conducts a front- and back-end analysis. This phase can closely follow the Analysis phase in the traditional systems development life cycle. Data is collected from the end-user (real or instructor) pertaining to the scope of the Web site, which includes the purpose and target users as well as the Web server needs in relation to expected site traffic and growth, database requirements, and graphics.

Since information is at the heart of the Web application, the database needs to be accurate and dependable. A complete data and work flow analysis should take place with the end-user (real or instructor) to understand how the information needs to be accessed and presented. The students can then normalize the data ready to create a model for usability testing prior to application development.

During this phase, the instructor's role is that of a project leader and each group expresses the system requirements as objectives delivered to the instructor. The deliverable can describe the appearance of each page, navigational links among pages, and details of the database including tables and fields. The document should also include an implementation plan indicating how tasks such as the database programming or graphic design are distributed among team members. The entire process should not take more than four weeks.

Design Web Site

Once each group has analyzed content and the instructor approves that content, the process of Web site designing can begin. This phase can follow the systems design phase in the traditional systems development life cycle (SDLC) and the instructor role again is that of a project leader. How the design is accomplished is up to each group. For example, since each site will have a home page, all group members can be responsible for its development. For content pages, graphics, and database design**,** members of each group may want to work individually or collaborate with another person.

It is necessary for the instructor to consider several important issues relating to the design of the Web site pages. For example, to what extent will the instructor permit individual or team creativity? Generally, good Web design requires consistency. One negative side to consistency is the elimination of techniques from the learning process. One way to manage this is to promote consistency via the home page that will carry common elements on all pages created by using frames or tables. Beyond this, each group could be permitted to be creative but maintaining consistency within their group as long as the creativity meets the requirements of the system. Each group is required to provide deliverables to the instructor marking progress and the design process should not take more than three weeks.

Test/Deploy the Web Site

A deadline should be set for the completion of the Web site approximately two weeks before the end of the semester. At this time, testing is done on both the client-side as well as the server-side. Following the testing, all groups deploy their respective Web sites to the Web server. In the case of a simulated project, the Web server may be the schools' or the instructors'. Conversely, with the real-world project, the Web server may be the end user's own server or a commercial hosting service. Each group then will document the web site.

Student/Site Evaluation

The completed database Web site and the contribution of individuals to each group comprises the basis for grading. The author has found the following breakdown to be most practical:

Individual contributions to the group:
Each student's individual contribution to the group is the basis for 30 percent of the final grade. Rather than having the instructor make this determination, individual group members will evaluate each other as per an anonymous evaluation sheet constructed by the instructor.

Completed Project:
The completed project will consist of 70 percent of the final grade. The deliverables the instructor has collected during the analysis when acting as project leader comprises one part. Another part is the degree of completion of the project based upon the overall requirements.

CONCLUSIONS

As the *World Wide Web* has grown from displaying static, informational pages to using databases and other mature technologies for interactive Web access, so too must the objectives of academic technology courses grow to reflect these changes and remain relevant. The purpose of this course model is to develop and deploy a database driven Web site in a team environment during one academic semester. Hopefully, the described course is a realistic guide for students. A learning experience that allows students to deliver a system related to a technology in high demand, demonstrates the difficulty and importance of working with a time schedule, and enables them to participate in an actual group project.

REFERENCES

1. Albanese, M. & Mitchell, S. (1993). Problem-based learning: A review of literature on its outcomes and implementation issues. *Academic Medicine, 68*(1), 52-81.
2. Barker, D. I. (1994). A technological revolution in higher education, *Journal of Educational Technology Systems, 23*(2), 155-168.
3. Buck Institute (1999). *Project based learning*. [Online]. http://www.bie.org/pbl [8 May 2000].
4. Date, C. J. & Darwen, H. (2000). *Foundation for Future Database Systems: The Third Manifesto.* Upper Saddle River, NJ: Addison-Wesley.
5. Hutchings, P. & Wutzdorff, A. (1988). *Experimental learning across the curriculum: Assumptions and principals*. *New Directions for Teaching and Learning, 35*, 5-19.
6. *IS2002, Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems*, (2002). Curriculum effort of Association for Computing Machinery (ACM), Association for Information Systems (AIS), Association of Information Technology Professionals (AITP)
7. Mannio, V. M. (2004). *Database: Design, Application, Development, & Administration*. Burr Ridge, IL: Irwin, McGraw Hill.
8. Morrison, M. & Morrison, J. (2002). *Database-Driven Web Sites*. Boston, MA: Course Technology.
9. Riccardi, G. (2003). *Database Management with Web Site Development Applications*. Upper Saddle River, NJ: Addison Wesley.
10. Seyed-Abbassi B. (2002). The Evolution of an advanced database course in an information systems curriculum. *ISECON 2002 19th Annual Conference on Information Systems Education.*
11. Tretten, R. & Zachariou, P. (1995). “Learning about project based learning.” Paper prepared for the Autodesk Foundation, San Rafael, CA.