Communications of the Association for Information Systems

Volume 17 Article 13

February 2006

Renewable Student Projects: A Learning Strategy

Chris Bennett
University of Georgia

Richard T. Watson
Terry College of Business, University of Georgia, rwatson@uga.edu

Recommended Citation
Bennett, Chris and Watson, Richard T. (2006) "Renewable Student Projects: A Learning Strategy," *Communications of the Association for Information Systems*: Vol. 17 , Article 13.
DOI: 10.17705/1CAIS.01713
Available at: https://aisel.aisnet.org/cais/vol17/iss1/13

Communications of the Association for Information Systems

RENEWABLE STUDENT PROJECTS: A LEARNING STRATEGY

Chris Bennett
Department of Computer Science
University of Georgia
benoit@uga.edu

Richard T Watson
Terry College of Business
University of Georgia
rwatson@terry.uga.edu

ABSTRACT

Renewable course projects (i.e. projects whose lives extend beyond the end of an academic semester) provide an effective alternative to class assignments that require repetition of artificial tasks typical in many information systems and technology-related courses. Though these "throwaway" projects are certainly meritorious, renewable projects enable students to engage in real-world development paradigms, such as modularization and life cycles, to develop lasting systems that solve relevant problems. By embracing the full project life cycle, modularization of work, and open source software, renewable projects expose students to traditional aspects of classroom development and also to the more real world features of the project cycle. Two case studies provide experiential evidence of the success of renewable projects.

Keywords: renewable projects, learning strategy, open source, information systems

I. INTRODUCTION

Traditional projects and assignments in technology-related courses typically require students to develop systems with certain functionality designed to concretize abstract classroom topics into something more tangible and therefore more easily understood. Often these projects are discarded at the end of the semester because they create little or no real world value. That is, systems are produced in a closed loop where no regard is given to what came before or what will come after. In introductory courses where the abstractions are elementary, this outcome is often unavoidable and indeed acceptable. In more advanced courses, however, assignments whose lives extend beyond the end of an academic semester are much more viable and desirable. Renewable projects extend the traditional course project model to expose students to realistic development and project life cycles and also to provide a more practical product that reflects established skills.

Several problems in traditional information systems courses have been identified, including the incredibly rapid changes in the field, the frequent exclusion of industry in the process, and the failure to design material and resources for the broadest possible cross-section of students [Tucker, 1996]. Since hands-on assignments weigh so heavily in students' understanding of

material in computer and other technology courses, projects become a powerful means to enhance and improve curricula. We present a model for renewable projects with three key components. Each component reinforces the idea of iteratively solving real world problems over multiple semesters by focusing on:

- The entire life cycle

A successful renewable assignment exposes participants to system development life cycles both within and across semesters. Even well designed traditional projects may only concentrate on the life cycle of the project within the semester. For renewable projects, the development loop opens beyond a single semester so that students consider inherited work, as well as the work that is being left to the next group. Continuity applies regardless of the development paradigm (for example, an agile methodology) used within a semester. The focus is on the processes within that semester and on the importance of thinking beyond the here and now.

- Modularization and teams

A renewable project should embrace modularization of work into teams. This sort of division of labor is common in real world development projects. Specialization allows students to concentrate on specific aspects of a project and avoid spreading themselves too thinly across all roles. Exposure to the realities of interoperability between roles in projects is extremely valuable for any student. Interoperability provides experience with interfacing both at the system and the interpersonal levels that is critical to the success of larger development undertakings.

- The open source paradigm

The final component of our model is the extensive use of open source software throughout the process. The open source software paradigm embraces open development and support for sustainable projects benefiting the community at large.

In Section IV we present findings from two case studies in renewable projects. In the first project, a course on XML was developed that uses Wiki technology to teach students about XML and to provide adaptable materials for a field that changes almost daily. In the second project, students developed a content management system for the Open Tourism Consortium [Watson et al., 2004] using the open source technology Apache Cocoon [Cocoon, 2005]. This renewable project allows students to learn about current technologies, where textbooks and other resources often fail to keep pace, and also about the full system development life cycle.

II. RENEWABLE PROJECTS

Traditional projects in information systems and technology-related courses often limit student exposure to a subset of real-world system development such as certain phases of the development life cycles. Many of these assignments simply concretize certain abstract concepts for students without exposing them to the cyclical nature of the development process. For example, a typical project in a database management course might require individual or groups of students to develop a data model for a hypothetical company. While such an exercise can certainly provide students with some knowledge of the fundamental concepts of the relational model and the data modeling in general, it still lacks key components that students might expect to face in more realistic project situations. In defining renewable projects, we initially propose three components that differentiate them from traditional assignments, including a focus on:

- entire life cycles,

- modularization and teamwork, and
- open source strategies for aspects of system development and processes.

LIFE CYCLES

The traditional approach in course assignments typically omits the input and output aspects of the life cycle of a project. That is, real projects typically inherit work (e.g., requirements, models, and code) from previous iterations of the development process, whereas academic projects typically close the loop to concentrate on the processing in the middle. In addition, workers are responsible for the legacy they leave for the next generation of project members. Embracing the entire life cycle familiarizes students with the concepts of the particular course in which they are enrolled, as well as with the issues of an iterative process where output becomes input.

Figures 1 and 2 illustrate how renewable projects open up the life cycles of classroom projects. While typical course projects omit the real-world iterative nature of most development projects, renewable projects embrace the entire development loop in Figure 1 exactly because they expose students to other integral processes of the software development cycle shown in Figure 2. Creating and maintaining real systems requires that developers and managers focus on more than just the current iteration of development by considering both previous work and the work that will be left behind for others. If the development cycle in Figure 1 is seen as the process in Figure 2, the inputs and outputs of a renewable project are the inheritance and legacy of each semester.

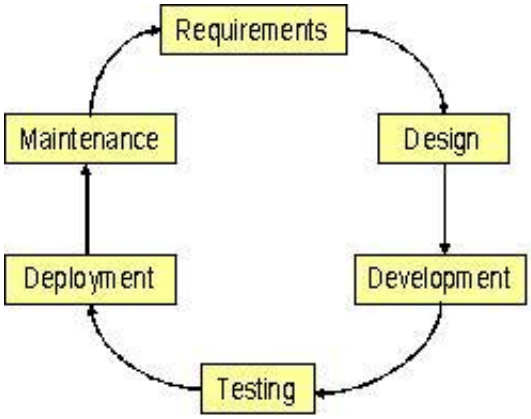

Figure 1. Embracing the loop

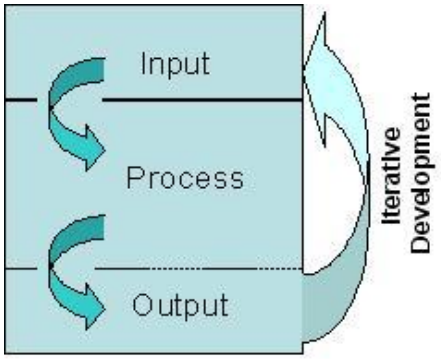

Figure 2. Development life cycle

WORK MODULARIZATION / DIVISION OF RESPONSIBILITY

Traditional projects also vary widely in their success at the division of work. Modularizing project responsibilities for system development across several roles is another incarnation of division of labor with its same advantages. In addition to the economies of scale afforded by specialization, the contracts between parties also provide clearly defined expectations and responsibilities. Renewable projects provide an opportunity to take on more challenging (and often more rewarding) systems because they are no longer limited to a single semester. By emphasizing the separation of responsibilities into roles and teams, projects can exploit the strengths of division of labor, exposing students to the realities of interfacing with agreements and contracts between the parties involved in the project.

Figure 3 shows a possible organizational structure within a class for a large project. From the beginning a renewable project should clearly define teams and their roles in development of a system. By encapsulating the functionality of a system into modules and the teams responsible for them, expectations and responsibilites are clearly defined for students. Of course, an academic setting includes participants other than students. Instructors and outside parties, such as industry, must play an important part in the project. In addition to instruction, the professor of the course should also mediate interaction between teams and monitor progress within and across semesters. He or she should also bring in outside resources such as industry, research groups, or other organizations. We discuss the division of responsibilities for our two case studies in later sections.

OPEN SOURCE SOFTWARE

We believe renewable projects should incorporate open source software (OSS) for any technology components of an assignment. In recent years, OSS has become a popular paradigm for developing tools for public use

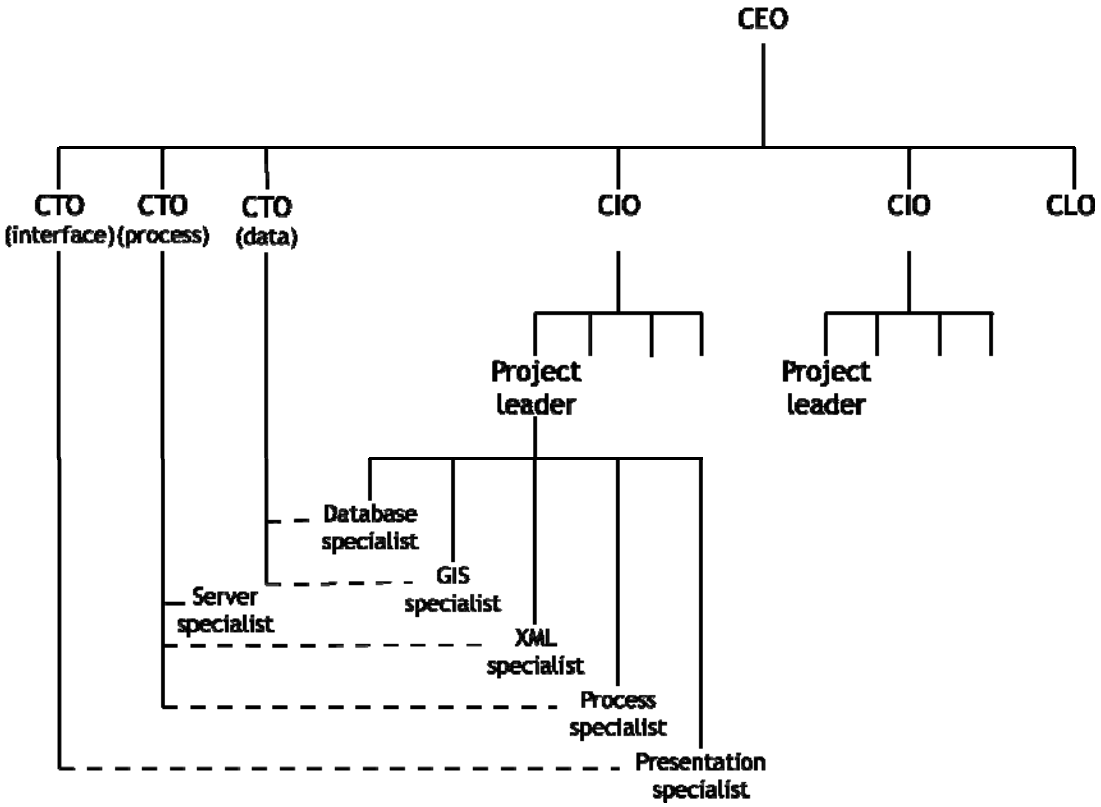

Figure 3. Potential Organizational Structure for Student Groups for a Course Project

by a community of developers that range from one to thousands [Raymond, 2005]. Open source typically means free, which is one advantage to using it. Beyond this consideration, many other features of OSS enable renewable projects to accomplish their goals.

1. Open source means that nothing is hidden from users. This visibility often comes with a spectrum of control that allows newcomers to learn about the technology while still letting advanced users add and extend the tools in powerful ways.

2. Projects created with open source tools become part of the community

3. Finished products can be released for public use.

4. Students are able to point to their accomplishments in a way that is difficult with traditional projects.

Both case studies use open source software extensively. Without such tools, projects would depend on proprietary software that prevents wide distribution and reuse. Because time frames typically limit most throwaway projects to unsophisticated or less useful undertakings, a community cannot be expected to form around it when costs are associated with its use. If, however, a project extends old tools and develops new systems that are given freely to others, it can evolve into a more sophisticated project that is worthwhile to students for learning and to society. While outside team members in a course project, may require a more closed version of the entire process, this consideration certainly does not prevent using open source software or reaping its benefits for all parties involved.

III. DESIGNING A RENEWABLE PROJECT

A generic framework for creating a renewable project requires two phases.

1. Provide a broad definition of requirements and goals for the entire life of the project. All multi-semester considerations should be enumerated and addressed before focusing on the project within any particular semester. For example:

    - How many iterations of development should be expected before the system is mature?

    - What tools are necessary to develop such a system?

    - How could the functionality be encapsulated and modularized?

2. The goals of individual semesters should be specified. Additionally, issues should be identified that will differ from semester to semester. For example, we know that the output of one semester's project will be part of the input of the next one. The impact of this cycle should be incorporated into course material.

These two phases repeat in a cycle (Figure 4). Between semesters, a review of progress should be made to provide feedback to all parties (students, instructor, and outside participants) and to gauge progress in the long-term objectives of the renewable project and identify errors and pitfalls to avoid in future iterations. Before semesters begin, objectives should be reassessed and adapted to the current environment. A final review after the life of the entire project could also provide insight into improving future projects.

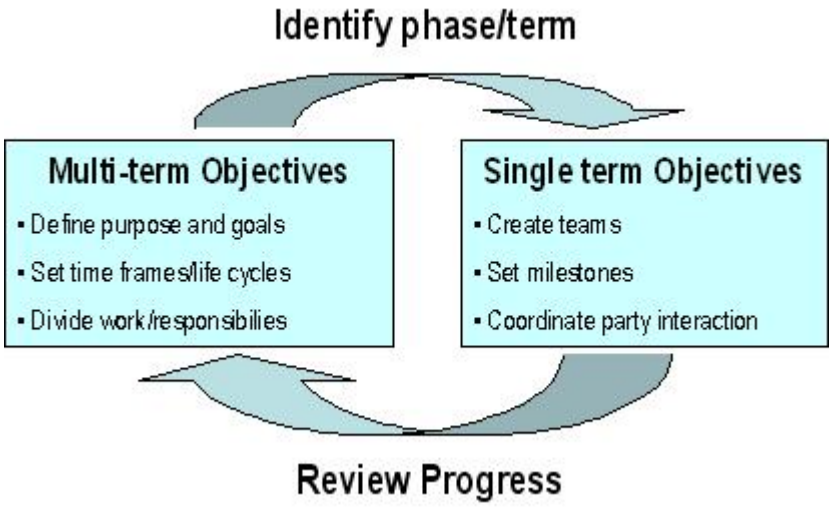

Figure 4. Developing Single and Multi-Semester Objectives.

In each case study, this creation and constant re-evaluation of objectives both across and within semesters contributes to the adaptability and ultimate success of each project. The development loop remains open so that errors are not repeated, and progress is made toward a greater goal that extends beyond the horizon of just one term.

IV. CASE STUDIES

We present two case studies on renewable projects. The first involves a course on XML where the students co-authored a textbook on the material using Wiki technology. After several iterations, this project is now being extended to classes in other countries. The product is an ever-evolving resource on XML that is partially translated into other languages with up-to-date information in a rapidly changing field.

The second project involves developing next generation tourism software under the banner of the Open Tourism Consortium [Watson et al., 2004]. Using open source solutions such as Apache Cocoon, students develop an open source system to deliver dynamic content for tourism authorities around the world. Modularization of work allows teams to learn a great deal about the particular technologies needed to develop such a system and about the requirements and responsibilities of working within an advanced organizational structure.

Both projects open the development loop by allowing students to delve deeply into extensions of previous work while learning the broad concepts of a course. We present a summary of the XML textbook project to show how it uses the renewable framework successfully, and examine the tourism system for examples of how to implement the three main components of the framework.

CREATING AN XML TEXTBOOK USING WIKI

A common problem for courses concerned with new technologies is that educational resources often lag behind the topic itself. That is, the technologies evolve faster than the educational resources such as textbooks (and often instructors) can keep up with them. The renewable solution to this problem is to turn the development and maintenance of instructional materials into the course itself. By dividing responsibility for content creation, editing, and quality assurance across all students in the class, everyone benefits from the most state-of-the-art material.

In January 2004, a course was implemented requiring students to co-author a Wiki book on XML. With Wiki technology [Leuf and Cunningham, 2005], a network of hyperlinked documents is maintained by multiple users. The tool allows authoring and publishing of content without overly

complicated processes. The technology lends itself well to collaboration for writing a book. Each student was responsible for writing and teaching a chapter and editing another. After its first iteration, the project was adopted in courses in Germany for its use as a text, and for improvements and extensions. Ultimately, the online book grew into an XML resource that is in the process of being translated into several languages. A screenshot of the project is shown in Figure 5.

This project embraces all three components of a renewable project:

1. The project keeps the development loop open by building on what came before it as well as leaving something behind for other groups. The input into each term is the accumulated work of all previous terms so that students can use an up-to-date XML textbook. This input serves two purposes: (1) a textbook for current students and (2) a starting point for adding new material such as new or updated chapters on XML or perhaps revised exercises. Each term also leaves an updated XML resource to the community to be used freely.

2. Roles are clearly delineated within the project to ensure accountability for each participant. By assigning students to write, teach, and edit different chapters, the breadth of XML knowledge is covered while depth is also provided.

3. By using a Wiki for editing and publishing the textbook, the project provides a free and portable architecture for other adopters.

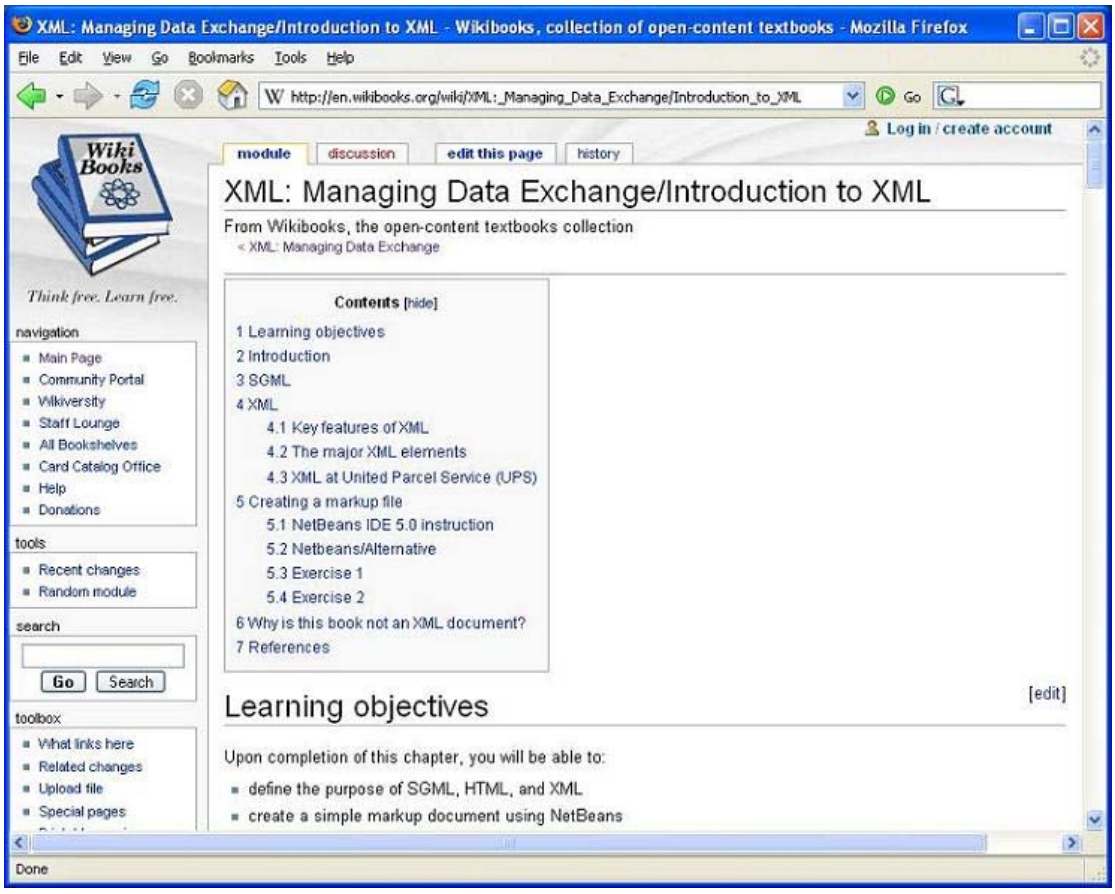

Figure 5. The XML Wiki Textbook

Project objectives both within and across terms are evaluated at specified times. By keeping goals realistic within terms and by adapting and learning from past experience, each semester

produces a higher quality resource for public use. Where a traditional project would rapidly become outdated, the XML textbook project evolves with technology.

A NEXT GENERATION TOURISM SYSTEM

The Open Tourism Consortium (OTC) proposes deploying information services for the three phases of travel (pre-tour, tour, and post-tour) [Watson et al., 2004]. By creating a standardized data model, markup languages for information exchange, and a content management system (CMS), the consortium aims to provide services that improve travel experiences for both host and guest. Since developing a modern content management system draws on relational database systems, XML technology, and state of the art web delivery and presentation technologies, it is an excellent candidate for an advanced project in a data management course. The traditional approach might use several teams attacking the problem with different strategies, but even the most successful team will normally shelve the project at the end of the term. By turning the development of the system into a renewable project, everyone involved can focus on creating a usable and robust system that the community at large can use.

Prior students developed the data model and markup language. The content management system is designed specifically to provide a turnkey open source program for tourist authorities to deliver dynamic web content from a standard data model. The development of this system, though, should provide students with exposure to all of the appropriate technologies. By constantly re-evaluating the objectives of the project as a whole and as it applies to a particular class or semester, a rather robust system can be developed that would otherwise be impossible in a traditional project. The amount of time to create a reliable and usable solution to such a problem necessarily requires multiple iterations not only to allow for the inexperience of the students but also to adapt to changing technologies.

After enumerating project goals, the project's first iteration saw students create a CMS from scratch. Using Java and open source web technologies such as Apache, students attempted to incorporate the OTC data model and markup language into something more than a proof of concept. While the concept was proven, it quickly became evident that the complexity of the project went beyond creating a CMS for OTC into creating a system that could be a CMS for any application. A post-term review of objectives identified the need to find an open source method that provides the necessary generic functionality so that the specifics of the OTC system could become the focus.

Apache Cocoon provides such a method [Cocoon 2005]. It is a dynamic web content delivery framework that offers all of the functionality necessary to customize a generic system for tourist authorities. As a project of the Apache Software Foundation, Cocoon is an open source with a large community of developers and users. The CMS also uses an open source database (MySQL) and operating system (Linux). The open standards created by the Open Tourism Consortium serve as appropriate models for the design of particular modules. By taking advantage of free but well-supported and robust tools such as these, the project can evolve from semester to semester without the concerns that proprietary software brings.

A key feature of the Cocoon framework is its emphasis on modularity and separation of concerns [Mazzochi, 2005]. This feature, of course, lends itself nicely to the second component of renewable projects. Teams can be formed in the classroom to work on data, logic, and presentation independently yet cooperatively. By appointing leaders and specifying interfaces between components, the project benefits from this specialization of labor and exposes students to the realities of large scale development.

This project emphasizes maintaining an open project life cycle. The long-term success of the project depends on openness because such an undertaking is not possible in a single semester. By attacking smaller pieces of the problem one class at a time, progress can be made toward a stable system with all of the necessary functionality to make it useful to the community at large. The OTC CMS already implemented the beginnings of each major area (data, logic, and

presentation) as a demonstration of proof of concept. Future classes will refine these same areas by implementing more of the data model or providing alternative and more attractive presentations. They can also provide entirely new and more complex functionality within these areas.

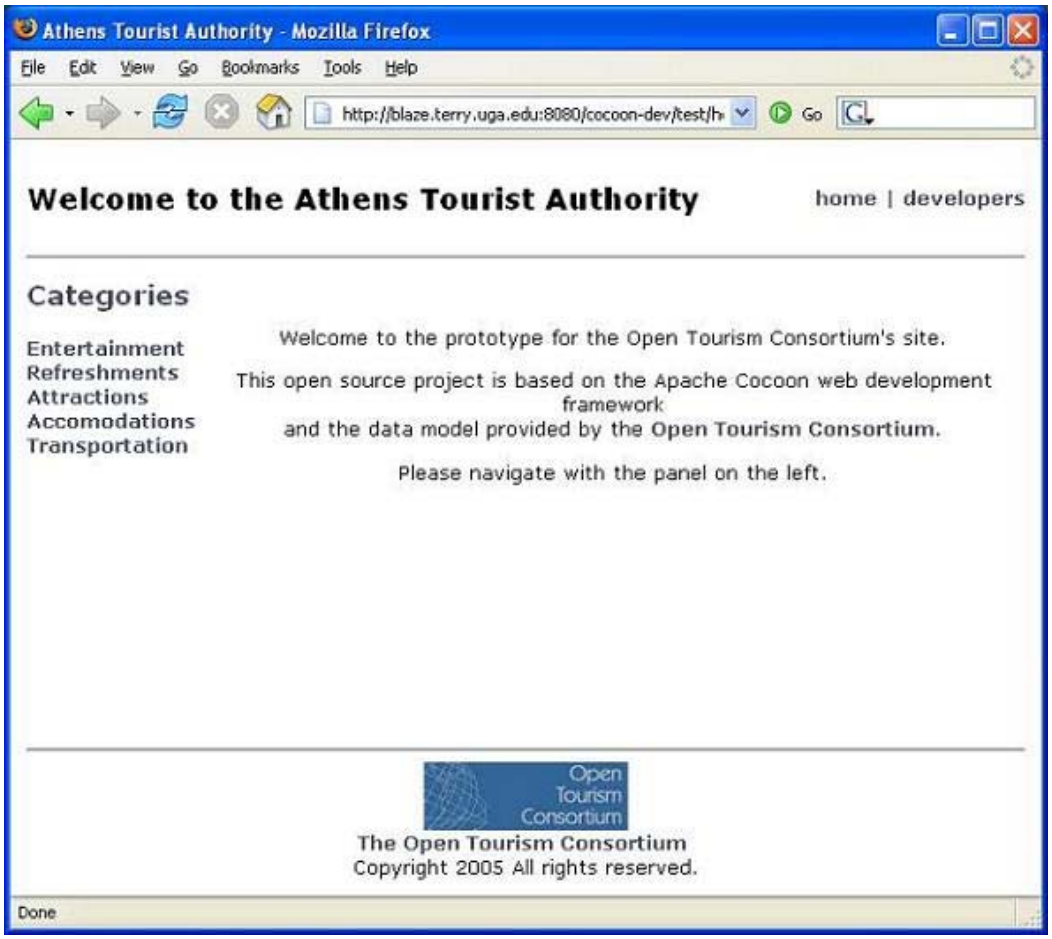

Figure 6. The OTC CMS Using Apache Cocoon

Creating a usable system such as the OTC CMS requires more than just a single project within a single term. By implementing all three components of the renewable framework, this project provided students with the traditional benefits of system development in a project and also with the additional experience of working in a realistic organizational structure for a complex project that necessarily considers what has come before and what will be left behind. In their course evaluations students often express how much they enjoy working on a project that is actually used after their contributions and to which they can point their prospective employers as evidence of their knowledge and abilities.

V. SUMMARY

By closing the development loop from class to class, traditional projects waste resources on throwaway systems. Renewable projects provide an attractive alternative to traditional course projects in advanced computing and technology courses. By embracing whole life cycles of projects, modularizing work to teams of students, and using open source software renewable projects create robust and usable systems for the community, as well as providing invaluable experience for students in developing a real system with real constraints. Renewable projects can extend beyond one class to involve multiple classes, possibly in different countries, working

simultaneously on developing a system. Students receive the additional experience of working in distributed, culturally diverse teams. Such experience will be valuable to them in a world of offshore outsourcing.

By renewing the work from semester to semester, a renewable project exposes participants to more than just the process of developing a system. When output becomes input, students learn more about the development cycle without sacrificing any of the process of traditional projects. We presented two case studies illustrating that renewable projects can be successful in teaching students leading-edge technology and in turning resources into products for the community, rather than disposing of them.

REFERENCES

Cocoon (2005). The Apache Cocoon Project: A Web Development Framework. http://cocoon.apache.org. Last consulted January 17th, 2006.

Leuf, B., and W. Cunningham (2001). *The Wiki Way: Quick Collaboration on the Web*, Boston: Addison-Wesley.

Mazzochi, Stefano (2005). "Past, Present, and Future of the Apache Cocoon Project", Apache Software Foundation http://cocoon.apache.org. Last consulted January 17th, 2006.

Raymond, Eric S. (2000). *The Cathedral and the Bazaar*. http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/. Last consulted January 17th, 2006.

Tucker, Allen B (1996). "Strategic Directions in Computer Science Education". *ACM Computing Surveys,* 28(4), pp 836-845.

Watson, R.T., et al (2004). "The Open Tourism Consortium: Laying the Foundations for the Future of Tourism," *European Management Journal* 22(3), pp 315-326.

XML: Managing Data Exchange (2005). http://en.wikibooks.org/wiki/XML

Editor's Note: The following reference list contains hyperlinks to World Wide Web pages. Readers, who have the ability to access the Web directly from their word processor or are reading the paper on the Web, can gain direct access to these linked references. Readers are warned, however, that

1. These links existed as of the date of publication but are not guaranteed to be working thereafter.
2. The contents of Web pages may change over time. Where version information is provided in the References, different versions may not contain the information or the conclusions referenced.
3. The author(s) of the Web pages, not AIS, is (are) responsible for the accuracy of their content.
4. The author(s) of this article, not AIS, is (are) responsible for the accuracy of the URL and version information.

ABOUT THE AUTHORS

Chris Bennett is a doctoral student in the department of Computer Science at the University of Georgia. His research focuses on algorithm selection, data mining, and computer science education.

Richard T. Watson is the J. Rex Fuqua Distinguished Chair for Internet Strategy and Director of the Center for Information Systems Leadership in the Terry College of Business, the University of Georgia. His publications appear in leading journals in several fields. He is the author of books on data management and electronic commerce. His current research focuses primarily on open source, electronic commerce, and IS leadership. He has given invited seminars in more than 20 countries for companies and universities. He is the Past President of AIS, a visiting professor at Agder University College, Norway and Fudan University, China, and a consulting editor to John Wiley & Sons.

Communications of the Association for Information Systems

ISSN: 1529-3181