Project Number. GFP0901

Walden Woods

An Interactive Qualifying Project Report:

submitted to the faculty of the

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the

Degree of Bachelor of Science

by:

Andrey Yamshchikov

Date: December 16, 2009

Approved:

Professor Gary F. Pollice

Abstract

This project establishes a solid foundation for a website that will serve as an electronic source of information about Walden Woods and Pond. The completed work includes a website accessible from the World Wide Web that enables contributors to add content in the form of various types of documents, such as HTML, pictures, etc. As a result it creates an intuitive webpage layout that promotes ease of navigability and simplicity of use of the resources.

Methodology

Introduction

This project took place during the Fall Semester at WPI – from beginning of September to the middle of December. Edmund Schofield sponsored the effort and time invested into this project; he also simulated a customer from whom the requirements were obtained. Professor of Practice Gary Pollice advised this project. Professor David Rawson assisted with communicating with the sponsor, while Professor David Brown was consulted regarding the improvement of the website's layout.

It should be noted that this project focused mainly on establishing groundwork for a website, rather than providing a website complete with functionality and content – which would have proved impractical in the allotted timeframe. The resulting website offers several essential functionalities (that concern content and user management) and serves as a basis for future work, such as digitizing and adding content or improving the key existing functionalities.

Background

Walden Pond and the woods that surround it make up the Walden Pond State Reservation in Concord, Massachusetts. The pond was formed by retreating glaciers over ten thousand years ago. This particular type of formation is referred to as a kettle or a kettle hole. Walden Woods is defined by droughty, sandy, acidic soils which make it unsuitable for farming in contrast to the surrounding lands that's been historically agricultural locations (What is Walden Woods?).

The area was made famous by Henry David Thoreau who spent two years (1845-1847) in a cabin – which he built himself – located in the woods near the pond. In his book *Walden, or Life in the Woods* he describes the time he spent living in Walden Woods. In his book Thoreau chose to reduce the two years he spent in Walden Woods to a single year in order to use the passage of the seasons to symbolize human development. Today the book is critically acclaimed and is regarded as a classic American novel.

In 1961 Walden's ecosystem was threatened when a significant portion of the woods was proposed to be leveled in order to commercially develop the land. The situation quickly turned into a court case and soon the decision to abort the development was reached. Years later, in 1990, the Walden Woods Project was initiated with the main goals to "preserve the land, literature and legacy of Henry David Thoreau to foster an ethic of environmental stewardship and social responsibility. The project accomplishes this mission through the integration of conservation, education, and research" (What is Walden Woods?).

Requirements

The requirements analysis helped gain an in-depth understanding of the task at hand. Requirements solicitation sessions produced two requirements categories: the target audience (i.e. the people who would want to use the website) and the content of the website including the functionalities provided by it.

Users

Interactions with the customer led to determining six main types of users: general user, authorized user, contributor, reviewer, administrator, and developer.

The main purpose of the website is to serve as an educational tool – in a broad sense of the phrase. That's not to say that the target audience is limited to students and teachers. The aim is to let *anyone* discover Walden's many intricacies – scientific, literary or simply the beauty of the place. An average person should benefit from a visit to the website which would help further the interest of the masses in the importance of saving the radiance of preserves such as Walden Woods.

It is also important that the website contains aspects that attract potential contributors from the scientific and literary communities. Analyzing Walden's delicate ecosystem and recording the results allows us to acquire more insight into its workings. Anyone interested in this type of research can use the website to share his or her findings with other visitors. That is the reason for introducing authorized users into the system. These users have an active account and credentials that they can use to login and manage settings from a page that is only accessible by registered users.

Contributors are people who express interest in providing more content for the website. Contributors could range from a scientist who volunteered to monitor particular parts of Walden's ecosystem and make the recorded results available through the website to a high school student who decided to share the photographs he or she took while visiting the preserve during a field trip.

The need for reviewers arises in order to verify the validity of the information provided by the contributors. Reviewers are trusted people who make sure the content submitted by the contributors is relevant to Walden and is with accordance to the mission statement of the website.

An administrator's role is to manage all the mentioned types of users.

Developer is the final user type. These are the users with an intimate knowledge of the inner-workings of the website (code, database, etc.) who are charged with general maintenance and improvement of the website.

Content

The website will provide information in a large variety of formats – text, table, graph, picture, and video – which calls for various presentation options. The approach used to satisfy this need arranges the information into categories and subcategories to resemble a tree hierarchy. Each category contains thematically relevant topics. A topic can be of any one type or combination of types of information. For example, imagine one of the categories is Geology with a subcategory Hydrology; one of the topics in Hydrology contains tabulated annual scientific data that describes the movement, distribution and quality of water in the preserve (Figure 1), while another Hydrology topic covers information pertaining to the shoreline of the pond – pictorially representing the seasonal changes and explaining them in textual form (Figure 2).

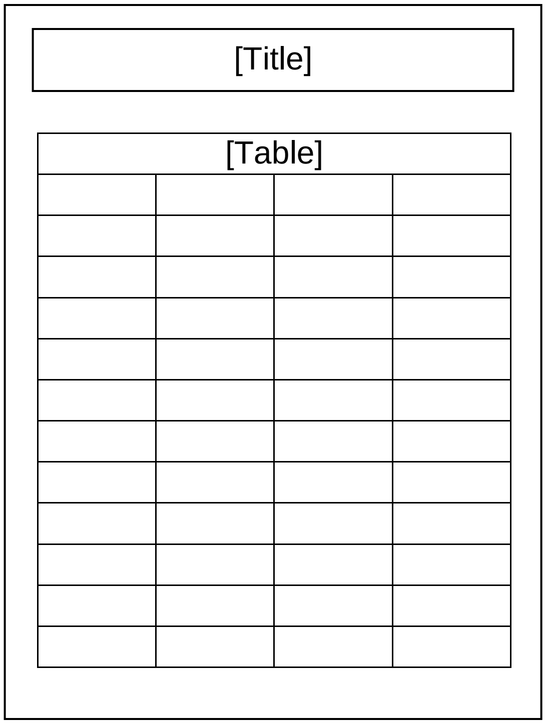


Figure 1

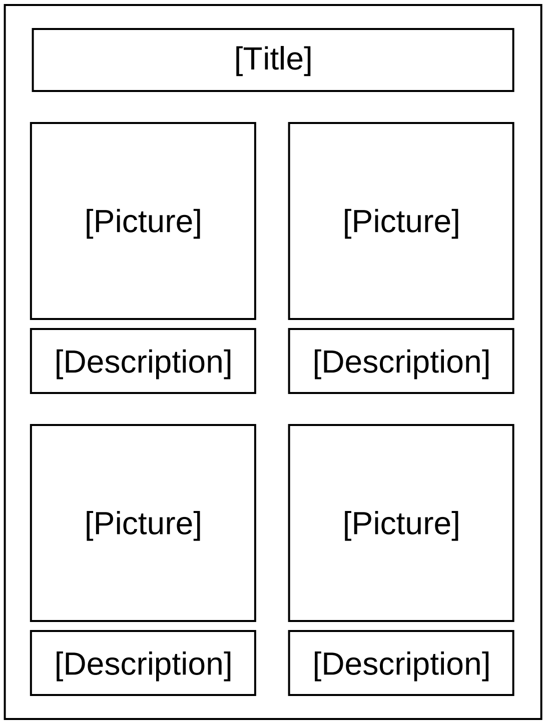


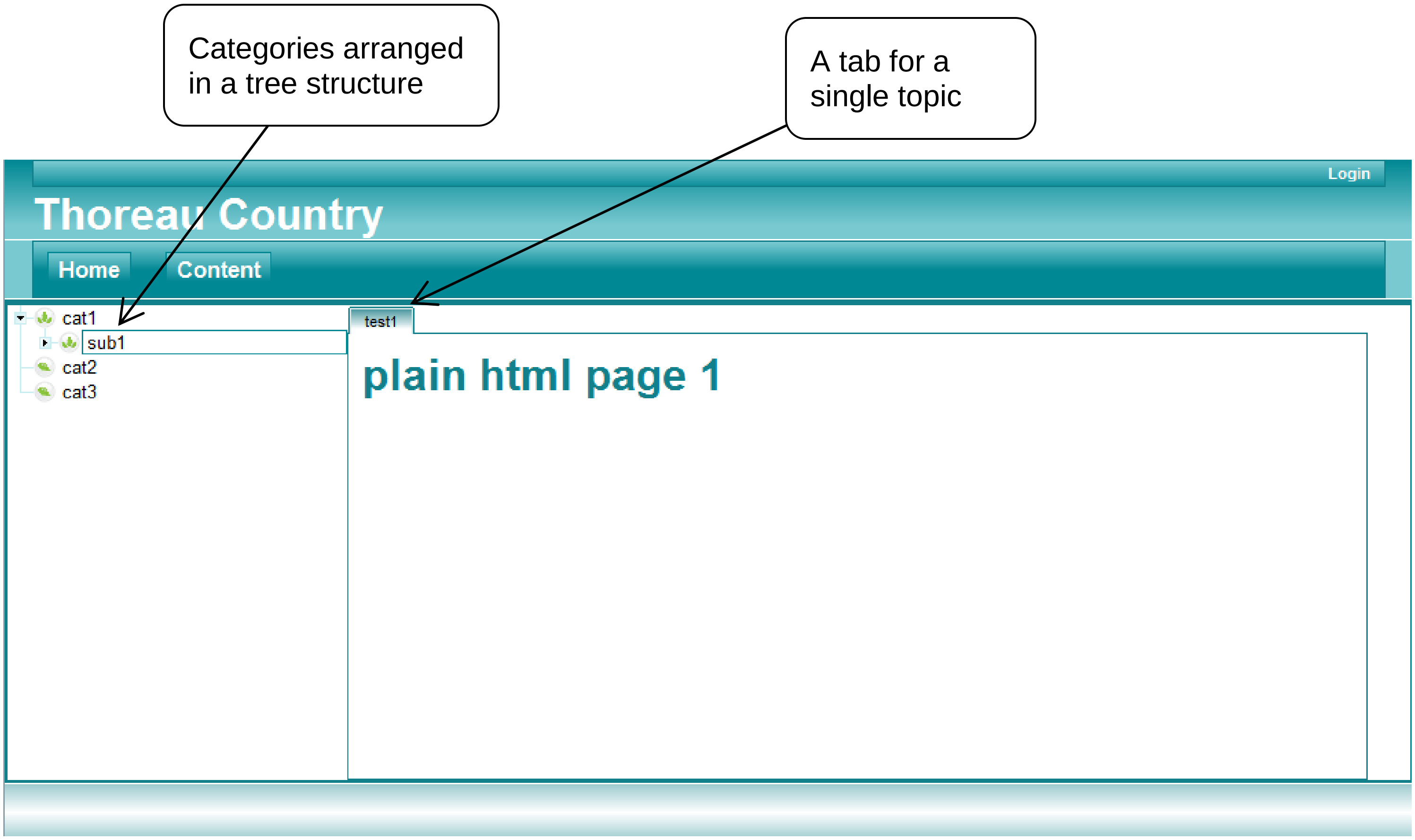
Figure 2

Layout

Due to the variance in the skill sets of the target audience, very few assumptions can be accurately made about the computer proficiency of an average visitor, therefore the navigability and the presentation of the content is intuitive and as simple as possible.

The website has three fundamental pages: home page, content page, and registered user's page. The home page contains a mission statement and a short overview of Walden Woods and Pond reservation. The content page is the place where the visitors look for the information of interest. The layout and controls on the registered user's page change depending on the role of the registered user – for example, an administrator would be able to create and modify users and roles, while a contributor would be able to submit new website content for review.

The layout of the content page of the website is as follows: on the left side of the page there is a tree-like view of categories and subcategories while the center of the page is be dominated by a panel that displays topics of the selected category; each topic is represented as a tab (Figure 3). This way when a visitor selects a category of interest on the left, the center panel changes accordingly to display the number of tabs that directly corresponds to the number of topics in the selected category.



Categories arranged in a tree structure

A tab for a single topic

plain html page 1

Figure 3

Technologies

The technologies employed in this IQP include a range of JBoss-developed web frameworks – JBoss Application Server 5, Seam 2.2, RichFaces 3.3, Hibernate 3 (Core, EntityManager and Annotations). All of the mentioned frameworks were developed, tested and deployed using Java 1.6.0. The backend also makes use of a MySQL database version 5.

These technologies were chosen based on a number of criteria: learning curve, setup time, and ease of upkeep. Because all the frameworks used in this project are based on the Java programming language the learning period for anyone already familiar with Java is fairly short. In the web-development community Java is popular due to its extensibility, ease of use and conformation to the Object-Oriented paradigm. That is why Java is a safe choice of the backend programming language.

Some alternatives include PHP and Ruby on Rails. Java has the upper-hand in each case – it is more robust and provides better scalability than either, has better separation of layers than PHP and is more stable than Ruby on Rails. PHP is simple and extremely easy to learn but is fairly limiting in choice of fully-functional, up-to-date frameworks that can be used in conjunction with it. Ruby on Rails is simply unstable between versions and was discarded as an option considering the amount of time it will take to fully implement the website (multiple projects in succession are planned for this website). Finally, enterprise Java and the frameworks associated with it are considered an industry standard by many large businesses (Fidelity Investments and Goldman Sachs) and despite opposing views rooted in fierce loyalties to other technologies despite any shortcomings (usually prescribed to the Ruby on Rails community) serve as a great example of the practical, real-world technologies.

A third competing technology that could have been used in the front-end development is Adobe Flex based on the Adobe Flash Platform. It's a viable alternative to RichFaces (or any other library that implements the JavaServer Faces standard) except for the integration with the backend. Seam provides a more clear and intuitive integration of RichFaces and backend Java, as opposed to BlazeDS an integrating platform similar to Seam that facilitates communication between Flex and the server-side Java.

Stability, extensibility, maintainability and scalability are all qualities that are positively associated with the way chosen Java technologies combine with each other.

RichFaces

One of the most current and popular technologies used in front-end web development is JavaServer Faces abbreviated JSF. JSF is a specification that was compiled and is often revised to allow for new functionalities by Sun Microsystems. JSF extensively uses the MVC (Model-View-Controller) design pattern. JBoss RichFaces is one of the most commonly used and widely recognized implementations of the JSF API. The RichFaces technology is a component library that provides user interface widgets for web applications. A worthy alternative to RichFaces is ICEfaces – an implementation of JSF developed by ICEsoft Technologies. Both implementations provide similar web UI components each with their own advantages and shortcomings, however, in order to avoid any possible bugs or conflicts with other JBoss technologies used to build the website, RichFaces were chosen over ICEfaces.

Data Management

MySQL is a free, open-source relational database management system (RDBMS). Its popularity, ease of use and many features fit in quite well with this project. The communication with the MySQL database is done through the Hibernate EntityManager as described below.

ORM – Object-Relational Mapping – is a very useful technology. Currently, most of the databases used in practice are relational, which means the data they contain can be organized into groups based on commonly shared traits. For example, a college that wants to keep track of students and faculty can use a relational database with three tables. First table would store common traits (as fields) that both students and professors share – first and last names, ID number, email, etc. However, there are some characteristics that are unique to only students or only professors – students are assigned to dorms while professors have offices – and these are stored in the other two tables. ORM is a technique that is used in software development to convert data retrieved from a database into objects that easily manageable within a scope of a program. Java Persistence API (JPA) is a specification of an ORM as well as a soon-to-be standard. JPA was put together by Sun, after Hibernate began to become popular. JPA combines the best ideas from other frameworks (Hibernate, JDO, etc.) to create a new persistence API that satisfies the needs of many Java programmers and greatly simplifies development of any application that takes advantage of a back-end database.

Hibernate EntityManager provides a powerful, high-performance implementation of JPA which is why this project takes advantage of it as an ORM tool.

Seam

Seam is an integration framework that joins together standard Java EE technologies with several nonstandard but widely-used technologies into a consistent, unified programming model. Java EE is a technology that expands on standard Java by providing tools for web-development. The Seam Framework makes use of Java EE 5.0 to provide a consistent and easy-to understand programming model for all components in an enterprise web application. It eliminates most of the boilerplate code and XML (Extensible Markup Language) configuration – extensively (ab)used in the Spring framework – from applications by using Java annotations. Seam solves the integration problem and opens web developers to useful tools that were too hard to use in the web applications before. Seam enables developers to use annotated POJOs (Plain Old Java Objects) – as opposed to Enterprise Java Beans – for all application components. Compared with applications developed in other web frameworks, Seam applications are conceptually simple and require significantly less code (in both Java and XML) for the same functionality (Yuan, Orshalick and Heute).

JBoss Application Server

JBoss AS is a free, open-source Java EE application server. An application server is a software component that is used for deployment and management of web-based applications, in other words it’s a program that makes a website accessible on the Web via the Internet. An application server is not to be confused with a physical computer that acts as a server; the latter is a hardware component (a more powerful version of a desktop PC) on which an application server is run. Embedded in the JBoss AS is Apache Tomcat which serves as the servlet container (which allows the three technologies mentioned earlier to work). As an application server, JBoss AS features clustering, distributed deployment, load balancing, caching, session-management, database connection pooling, etc.

For a better understanding of these technologies, consult the Suggested Reading Materials section at the end of this paper.

Web Hosting Service

Web hosting service is the final piece of the puzzle. Here's how the technologies come together (Figure 4): RichFaces, Hibernate and Seam are the three main frameworks used in the software development process, yielding compiled code which has to be deployed on the JBoss Application Server; when the code is deployed and run, some parts of it will make use of the MySQL database (note that the MySQL database is not deployed on the JBoss AS, it is simply another program). The only question that remains is where are the two latter applications (the application server and the database) running? This is where Arvixe comes in. Arvixe provide hardware needed by the software used in this project. Arvixe is a paid web hosting service that rents out its customers the physical servers (in contrast to the application servers). Arvixe was chosen from a group of web hosting services that allow customers to run their own application servers and take advantage of Java programming language. Arvixe has a unique combination of features including unlimited hard drive space, SSH access, comparatively low price, etc. that make it a reasonable choice for a web hosting service.

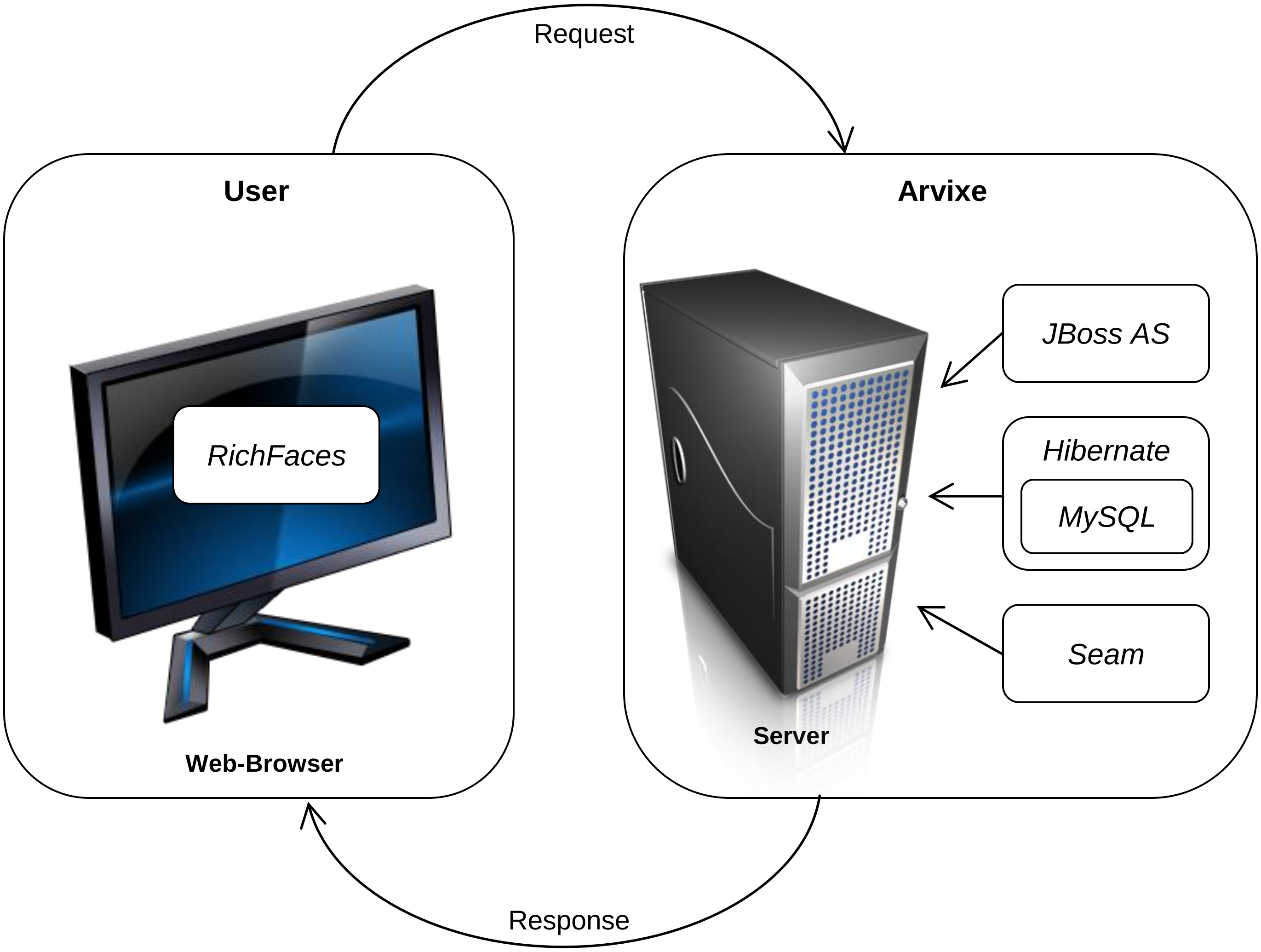


Figure 4

Issues

Obtaining the requirements from the customer proved to be a challenge due to the customer’s lack of experience in participating in the requirements analysis process. The main difficulties were associated with attempting to pinpoint different types of users and the content of the website. For example, the customer had difficulty understanding the difference between user roles and website users – more particularly that a user can assume multiple roles.

Arvixe, the web hosting service provider proved to be a major encumbrance throughout the course of this project due to continuous service discrepancies and the technical support’s ineffectiveness as well as lack of knowledge in the subject matter pertaining to server setup. Also, lengthy periods of times between inquiries and responses concerning service issues further impeded the progress.

Future Work

There are several improvements that can be made in order to further increase the efficiency of the website. One such improvement is search. Adding robust search functionality would greatly benefit the website and make it much easier for visitors to locate the information they are looking for. One of the best ways to implement searching is with the help of Hibernate Search. Hibernate developers built on Apache Lucene to bring the power of full text search engines to the persistence domain model through transparent configuration and a common API. Hibernate Search eliminates structural mismatches (object/index translation), duplication mismatches (changes synchronized with the database, index access is transparently optimized), API mismatches (the index can be queried using regular queries used in Hibernate) (Hibernate).

Another suggested improvement is to allow the content (documents with information about Walden Woods) to be updated using a web UI in order to avoid any possible confusions that are associated with modifying the file system on the server. Actions like adding new categories and topics could be done from a content management webpage and restricted to authorized users with appropriate roles.

Finally, the web hosting issue requires a resolution. Google App Engine is a worthy alternative to consider. It’s a platform for hosting web applications. Its main advantages are that it’s free of charge until the website becomes popular and the detail of customization of service associated with a cloud – since the application will be deployed in Google’s cloud, properties such as bandwidth, CPU time, number or requests can be modified to reach desired effects.

Results

The completed work resulted in a website with some key functionalities and needs to be populated with appropriate content – various types of information about Walden Woods. This website is available on the World Wide Web at <http://thoreaucountry.net>.

A visitor is greeted with the home page which contains a picture of Walden Woods and the mission statement provided by the sponsor. The user has an option of navigating to the content page where the information about the preserve is located or log in with username and password supplied by an administrator. If the visitor supplies valid credentials he has the option of submitting new content about Walden Woods by following the "Contribute" link. The information sent by contributors is marked for review and saved inside the "contribute" directory on the server. If the visitor has administrative privileges he or she is able to manage existing or create new users. An administrator is also able to manage roles and assign them to specific users.

The categories structure seen on the content page mirrors the file structure on the server. Each category (in tree-view) corresponds to a folder and each topic (presented as tabs) maps to an HTML file located within the appropriate directory.

This project succeeded in providing a concrete foundation for a website that will serve as an electronic source of information about Walden Woods and Pond. The criteria for a simple page layout and the ability to contribute with various types of information were also satisfied.

Works Cited

25 February 2009. Red Hat. 6 Dec. 2009 <https://www.hibernate.org>

What is Walen Woods? 6 Dec. 2009 <http://www.walden.org/conservation/waldenwoods.htm>

Yuan, Michael Juntao, Jacob Orshalick and Thomas Heute. *Seam Framework*. Boston: Prentice Hall, 2009

Suggested Reading Material

Bauer, Christian and Gaving King. *Java Persistence with Hibernate.* Greenwich: Manning, 2007

Katz, Max. *Practical RichFaces*. New York: Apress, 2008

Model-View-Controller. Sun Microsystems. 5 Dec 2009

http://java.sun.com/blueprints/patterns/MVC-detailed.html