CampusBook: an online book shop

Objectives

- To learn how to build enterprise-strength web based applications
- To comprehend server side architecture for these applications, specifically servlet, JSP, JDBC and JavaBeans technologies from the J2EE standard
- To analyze a given problem and propose a solution in relation to J2EE related technologies
- To implement these technologies within Tomcat, MySQL and UNIX environment
- To develop interpersonal and team problem solving skills

1. Project Overview

CampusBook is a non-profit online book shop for university students to sell and buy used text and reference books. To support the running and maintenance of this website, advertisement is needed for financial income.

The core requirement of this project involves creating a website using servlets, sessions, database connectivity and JSP. Following summarizes the features required.

Core/ Basic Requirements

- Random advertisement banner displayed each time the website is accessed. Image files will be provide for banners, or you can use images you like. Each banner is a link to a sponsor, which will be opened in a new window if it is clicked.
- Create member account,
- Fill in your profile,
- Log into account
- Add books to sell
- Lists of books to sell or buy
- Shopping Cart
- Public Mode
- Member Mode
- Administrator Mode

Bonus / Optional feature:

- Write reviews
- The review consists of a 1-5 star rating and a free text area

3. Required Functionalities

You are to create a fully functional system with the following modes:

Public Mode:

This is the mode for general public to visit the website. In this mode, anyone can view, search books available on CampusBook. Search can be based on book title, ISBN, and unit code. In public mode a link is always provided for members to log in.

Member Mode:

A member is an individual who can view, buy/sell books on CampusBook website. Member is a registered user who can perform actions listed below in addition to viewing the site but does not posses the privileges an administrator does.

- **Actions:** Add books to sell, list books have been ordered, buy books, create/edit profile

Member Profile will hold:

- User details
- **Attributes:** Name, Surname, Email, Password, Postcode, Gender, phone number will be used to register.

Possible scenario for the user:

Any user can open the website and view content anytime. However to be able to sell and/or buy books, he/she has to become a member.

When a user enters the required details (Name, Surname, Nickname Email, Password, Postcode, Gender, phone number) and becomes a member he can customize his profile.

A member can also add list of books for sale, and view books that have been ordered.

For each member the following details are also displayed his date registered, Items for sale or ordered. Members and place selected books in the shopping cart. Items in the shopping cart can be deleted before check out.

In your submission, the system should have at lease three members, and each member has at lease three books listed for sale or buy.

Administrator Mode:

Administrator is the only one who can perform the following actions which normal member cannot.

- **Actions:** can view, add, and delete any book listed on the website. An administrator can also delete member profiles.

Furthermore, the system should take these into consideration:

- User Authentication
- Form Validation

- Use of Sessions
- Data Structures
- Design Patterns (see below)
- Code documentation

You must implement the majority of your program using servlet and JSP. Any other incorporating technologies would certainly enhance your system.

Design Patterns

It is highly recommended that you follow some logic and structure when creating your program; otherwise, you are guaranteed you will be spending too much time shuffling your code. Well structured code means adhering to a component of your program that accomplishes a certain task while another component accomplishes something else- all while all of this may be recurring throughout the lifecycle of your project. A viable standard solution to this problem of recurrence is called design patterns. One such pattern that is widely used is Model View Controller or the acronym MVC for short. Each of the phrases of MVC is implemented using a certain server side Java technology within the context of our subject. The 'Controller' portion is the servlet, which controls the flow of the program. This means when a request is received from the user, the servlet decides where the request should be sent for processing. Once processed, the servlet receives a response back and it gets sent to the user. The 'View' portion is the JSP or HTML page, which is used for presentation, such as when sending a response back to the user. The 'Model' portion is referred to a plain old Java object (or POJO) or JavaBeans, which does most of the modeling of your program (i.e. heavy computations). You should also be cautious in the places where a servlet or a JSP is used. A rule of thumb—if a response contains a lot of static or any presentation material such as texts, images, etc. then using JSP will suffice otherwise using a servlet is a good idea. Remember its sole purpose is to control the flow of your application.

Some guidelines in developing your program- synonymous to SDLC:

- Model the problem by developing use cases and draw a domain class diagram (i.e. using UML)
- Create a database and create dummy records and input them into MySQL
- Using use cases as hints create static pages per use case.
- Write code to link the static pages to database (conforming to MVC architecture)

4. Tasks and Marking Scheme

Task 1 [30 marks]: Design and Documentation

- Your designs including page layout, database design (ERD/RDM, or descriptions of database design and tables), system architecture, UML class diagram where applied. [20 marks]
- A screen shot and test case for each mode. [6 marks]
- A help manual in one page to explain how to install and use your system [4 marks]

Task 2 [70 marks]: Implementation

- Functionalities related to Public Mode [5 marks]
- Functionalities related to Member Mode [25 marks]
- Functionalities related to Administrator Mode [25 marks]
- Functionalities related to Shopping Cart [10 marks]
- User interface [5 marks]

As explained earlier there will be an execution test for the implementation phase of this assignment; please refer to page 1 for more details.

5. Reference

- http://www.textbookexchange.com.au/