CSCI 4050/6050

Summer 2017

Term Project

Online Bookstore System

Over the past several years, ecommerce has transformed how people buy and sell online. In fact, online shopping has become so popular that many vendors sell only online with no physical location. An online bookstore can reach customers anywhere in the world, with no physical product, vendors can boast truly instant delivery for instance purchasing of digital media such as downloadable music and movies.

In this project, you are required to develop an online bookstore system for a respected client, the system is not intended to be used by a specific bookstore, and thus, its functionalities must be general to serve a large range of bookstores.

The software will enable the clients (Business) to have real time statistics of sales and book inventory, automatically produce End of Day sales reports and End of Day low-inventory notices. It will enable the **users (customers**) to view the real time inventory and extract book information. The system allows users to register in the store by providing their information (Name, date of birth, gender, email address …). Only registered users can place orders online and pay using different online payment methods. Orders are to be shipped to the customer or to be picked up in-store. Customers can also access their personalized account profile, order history, etc.

The system **administrator has** additional functionalities compared to the users of the system, he can add, delete or update member information, book information (title, category, publisher,..) order information and promotions.

Lastly, the software will enable authorized **vendors** to have access to part of the system – it will allow any vendor to update book information, add/remove new books to/from the inventory and updated prices.

Setting the Stage

The contract for the development of the system will be signed once the detailed requirements document is ready. Subsequently, we will design and implement the system according to the agreed-upon requirements. Once the system is implemented and delivered, it will be tested according to the acceptance ` developed software components will undergo thorough *unit, subsystem*, and *integration testing*.

The customer will participate in *acceptance testing* as well as a *demonstration* of the completed system.

 Deadlines for the remaining project deliverables will be announced at a later time. The final project demonstration will be given towards the end of the Summer semester, 2017 (the exact date will be announced later).

High Level System Requirements:

1. The syatem must provide a browse or vie catalog facility that shows the list of available books sorted by title. The list should include book title, author and the price.
2. The system must provide an interface to search for and view books that are available in the store categorized by subject, title, ISBN, or author.
3. The system must allow a selection facility to choose a book for purchase.
4. The system must allow people to register for the system by giving their name, email address, address and birthdate. Each person must provide a unique user name ~~and password, as well.~~
5. Registered users can subscribe/ unsubscribe for promotions and latest news offered by the bookstore adminstrators.
6. The user's registration is verified by entering a verification code that is generated by the system and sent to the user's email address upon completing the registration form.
7. The user must be able to modify his/her user profile, or even remove his or her registration.
8. A registered user must be able to login and logout from the system.
9. The system must provide a shopping cart and shall allow users to add/ remove books to the shopping cart. Users can view and update their shopping cart at any time.
10. ~~Registered users unwilling to provide online payment information can reserve books up to 5 days to be purchased and picked up at the bookstore.~~
11. The system shall provide a secured check out facility where users provide their payment information.
12. The system must send order confirmation to the customer's email. Order confirmation should include Customer name, Confirmation number, Order ID, Order Date, Shipping address (if applicable), ordered items and total amount in dollars.
13. The system must allow registered users to use their promotion codes in order to take advantage of current discount promotions.
14. ~~The system must allow different payment methods using cash (in store only), credit card and checks.~~
15. The system must allow administrators to add, delete or update member information, book information (title, category, publisher,..) and order information.
16. ~~The system must enable authorized vendors (publishers) to have access to part of the system – it will allow any vendor to update only his published book information by adding/removing books to/from the inventory and update prices.~~
17. The system must enable business owners to pull predefined reports; namely End of Day sales reports, low-inventory notices, book sales report and publisher sales report.
18. The system must have an easy-to-use user interface (UI) with screens designed for each part of the system's functionality and suitable for different types of users (Administrators, vendors, users and registered users). The user interface must either be a standalone Graphical User Interface (GUI) or one accessible from a common Web browser (such as the Mozilla Firefox, Google Chrome, Safari, and Internet Explorer), for all the users.
19. The system must provide multi-user access, assuring correct concurrent behavior. The system should maintain suitable authorization information and authenticate access.
20. User authentication should be implemented by checking user name and password.

21. The system must use a persistent data store (MySQL RDBMS) for all of the relevant data.
22. The system should use accepted standards whenever possible (HTML, CGI, Servlet API, JDBC, ODBC, SQL, etc.). The system must be coded in either Java or C++, possibly incorporating some smaller parts coded in other languages, such as PHP and JavaScript, if needed.