Web Development

Module Title:	Web Development
Assignment Type:	Main Assignment
Project Title:	Design and Implement a Library Web Application
Project Date:	11 October 2017
Assignment Compiler:	Sean Cahill
Weighting:	Marked out of 30, worth 30% of overall
Due Date:	December 15th 2017
Method of Submission:	Moodle uploader on course page
Feedback Method:	Provided once graded through Moodle Feedback

Assignment Introduction

The College has asked for an online Library to be created for the current library books in stock and any new ones that are purchased. The system currently used is a paper based system and they would like the system automated.

Design and implement a web based application for the College to allow students to login and search for books and to check out books from the library.

The system will be available in the library via a desktop computer located by reception.

User Accounts

As the system will be used by a number of different people, different user account types must be created. Depending on the account type, different options in the system will be available to them. This can be done by redirecting the user to the correct page once logged in.

- **Student** - A student should be able to log into the system and search for a book from the library. They should be able to check out a book and view all their currently checkout books along with the due date

- **Admin** - The system admin should be able to login and view all books currently checked out, indicating which student has checked out the book. They should also be able to view all books that have been checked out and books that are past their due date. They should also be able to input a new book to the library.

System Requirements

- When user opens the application, they should be greeted with two main options, register on the system for a new account or to login to the system with an existing account.

- Students who register must enter a username, their Student Id and a password.

- A login page for students and Admin to access the library. Students should be validated against a users table with username and password. Student should be directed to appropriate home page.

- A secret Admin account has been created. Admin can login with username: Admin and password: Admin. This combination will bypass the validation against the users table and proceed to Admin home page, with appropriate session set.

- All of the pages which the user is viewing should always be validated to ensure that a user is not on a page without having a validated account.

- After a Student has logged into the system, they should be presented with four main options on their home page.

    1. Search the library for books using title, including wildcard search.
    2. Checkout a book from the library if it is available.
    3. View their checked out books, displaying due date.
    4. Logout of the system, invalidating the current session.

- After an Admin has logged into the system, they should be presented with five main options on their home page.

    5. View Checked out books
    6. Check a book back in
    7. View books past their due date with student id displayed.
    8. Add a new book to the system
    9. Logout of the system, invalidating the current session.

- Checked out books should be stored with today's date and return date +1 week

- Storing passwords in plain text is a security issue that should be avoided. When users register for a new account, the password that they enter should be hashed and salted.

- All input fields should be validated using client side validation, checking for length and blank values.

- All input fields should be validated using server side validation, checking for length and blank values.

- After any user has logged in, a session should be started storing their username and a token to ensure that the current session is valid

- After a user has finished with the system, an option should exist to logout and destroy the current session.

- As the college is sensitive to automated attacks, a CAPTCHA should be added onto the registration page to ensure that automated bots cannot register.

- Wireframes for the system should be developed, to ensure that the college will sign off on the overall structure of the program before the development process is started.

Data Requirements

User

Student must enter

username: non blank, more than 3 characters

student_id: non blank, 7 digits exactly

password: combination of characters and digits between 6 and 10 max

Books

title: non blank

author: non blank

isbn: exactly 10 digits

Note

Planning is the key to this assignment, do not jump into the development process straight away. Ensure you think about the different pages that are needed and what the database structure will look like. After this has been completed, then you should start development.

Deliverables

- Library application source code
- Wireframes for the application
- Provide a database dump file for creation and insert of your database

All files must be zipped up as single file and uploaded to Moodle before the deadline. No late submissions will be accepted. All code that is created **MUST BE YOUR OWN CODE** this is an individual assignment, not a group assignment.

Sample Books

Below is a sample list of books which the user can view and/or checkout from the library.

ID	Title	Author	ISBN
1	Generation X	Douglas Coupland	0349108390
2	Introducing HTML5	Remy Sharp	0321687299
3	Handcrafted CSS	Dan Cederholm	0321643380
4	Bulletproof Web Design	Dan Cederholm	0321509021
5	The Tipping Point	Malcolm Gladwell	0349113467

Additional Marks

The College has asked for an enhanced security version of the application for Administration Users.

Remove the hard coded Admin user and password. Create a new table which contains a username and password. This will be the table against which the Admin login should validate against.

The login screen should now have the option of logging in as Admin or Student via a radio button on the screen. The appropriate validation will be used based on this choice and the user directed accordingly to the required home page.

In addition the system should be enhanced to ensure student cannot register more than once.

Marking Scheme

Description	**Grade**
Registration page for students	1
Login page for 2 different user account types Admin and Student. Each must be routed to correct home page on successful login. Login for Admin to use secret username and password.	2
Server side validation of all input fields, providing correct error messages if input is not valid	2
Session started when a user logs into the system and destroyed when the user logs out of the system	1
Client side validation of all input fields including reg ex. on isbn field	2
Student book search list should not allow checkout of book already checked out	2
Student can search for books, including wildcard search by title, and checkout books that are available with due date stored +1 week	4
Students can view all their checked out books with due date	2
Admin staff can insert new books to the library	2
Admin staff can view checked out books with student id displayed and check them back in	2
Admin staff can view all checked out books overdue (more than 1 week) with student id displayed	2

Password hashing for all passwords that are stored in the database. During login, the passwords entered by the user are also validated against the hash.	2
CAPTCHA added successfully to registration page	2
Wireframes for application	1
Enhanced security for Admin user and Student. Version 2.0	3