Web Site Application Fall 2014

Assignment 5

This Assignment uses knowledge gained throughout the entire course – it is your final assignment. This assignment is worth 24% of your final grade. This assignment is due: **Wednesday December 10th 2014 11:59pm Chicago Time**. It will take you several hours to complete. You will need to make sure you have access to a computer that you can install and run Grails, Eclipse and the JDK - ideally on your personal machine. Any students enrolled in the 562 sections must complete the additional work described in the graduate extension section.

This is an individual assignment

Objectives

- Outline why Object-Oriented-Programming is essential for creating advanced and reusable functionality
- Explain the concept of scaffolding and 'rails' and the advantages it affords developers
- Create a basic web application using Grails and Groovy
- Describe the components and advantages of the Model View Controller architecture

Contents

Development Project

You are going to finish your basic grails application that was started in the fourth assignment. Your grails application will be a complete course management system – similar to the basic functions of blackboard.

Design Brief

You have already built a basic working framework with a scaffolded grails application. You may continue to use certain elements of scaffolding – but using the grails commands to generate views for editing will be required to implement the following requirements.

Your course application will need to handle the following use cases (requirements)

- Users must be able to register themselves for your site
    - Your system should check to see if a username already exists
- Users must be able to log in to your application
    - You should only need to provide a username and password for the entire session
- Users must be able to register for courses
    - If a user has already registered for a course
- Users must be able to view their own enrolments
- The website should not look like a standard grails application and should incorporate a new logo and design elements on the main layout
- The homepage should show a welcome message and a link or login form

Requirements

- You must create a Grails application that implements the above requirements an uses the MVC design pattern.
    - You should reuse your existing course, user and enrolment domain classes – you may add additional fields to these classes as necessary
    - You should reuse your existing controller for course and user and add additional actions utilizing the **session** and **params** objects

Graduates must complete all the requirements listed above and additionally provide the following extra requirements

- You must implement actions in your controllers (or create new controllers) to add support for the following additional use cases:
    - Students must be able to search the catalog of courses and have returned to them the top 5 results
        - You may use the grails *searchable* plugin to achieve this result, or implement a basic search algorithm yourself
        - The search field should be implemented as a partial page and included in the main layout for the entire site and fit with the sites theme

Tips

There is one additional config file BootStrap.groovy. Within this file you can create sample data that will appear in your database even after a fresh restart when using the in-memory h2 data. This code is executed on every startup of your application.

e.g

```
import bookstoreclass2.Author
import bookstoreclass2.Book

class BootStrap {

    def init = { servletContext ->

        def aSampleAuthor = new Author(firstName:"Jason",lastName:"Lambert")
        aSampleAuthor.save(flush:true)
        def aSampleBook = new Book(title:"Sample Book", author:aSampleAuthor)
        aSampleBook.save(flush:true)

    }
    def destroy = {
    }
}
```

Design Paper

This is a mandatory section of the assignment – You are to submit a design document detailing your solution to the following design brief. In your design you should diagram and explain how you would implement the solution using Grails or PHP – specifically describing:

- MVC elements and their relationships
- How to build and link the MVC elements in a working application
- Application code layout
- Major visual layout considerations
- Security and access control
- Deployment Strategy (technology and databases)
- Additional assumptions and questions you would need answered to continue

Your design brief should be around 1500-2000 words and include appropriate diagrams to illustrate your design. A fellow student or software engineer should be able to take your design brief and implement the solution.

[A great (but optional) practice would be for you to implement this system yourself – but this is not required for this assignment]

Design Brief

A community library needs an inventory management system. They have several kinds of inventory items – including books, Blu-ray disks, and magazine – the library can have several copies of each. The library manager needs to be able to add new inventory items and mark old ones not for loan when they are retired. Library desk operators need to be able to scan inventory barcodes and record the loan date and the due date and the person who loaned the item. For books the library wants to keep track of which Authors are the most popular and have a function to determine how many Books from a given Author are on loan. Books, magazines and disks are all organized into library sections (for a given articles dewy decimal number) and managers should be able to manage library sections, viewing the total number of available books in each section. Finally users of the library should be able to log in to the system and view their loaned items, ideally sorted by type and when they are due.

Late Policy

There are no extensions for this assignment. Submissions more than 1 day late will likely not be graded as grades are due a short time after the final - so please submit early.

Submission Details

You are to submit a zip file containing your project source code and your PDF containing your individual solution to blackboard by the due date. Your ZIP should be named
YOURFULLNAME_YOURSTUDENTIDNO_AS5.zip

Grading Details

Both the design paper and programming assignment are worth 120 points each – for a total of 240 points available in this assignment.

Programming Segment:

Submission Specifications				10
	D	0	Significant Details Violated	
	B	6	Minor Details Violated	
	A	10	No Errors	
Code Efficiency				20
	D	0	Many things in the code could have been done better	
	C	12	Code uses poor approaches in some places	
	B	16	Code uses poor pattern once	
	A	20	No errors, and code uses best approaches	
Program Correctness	D	0	Limited use of correct patterns	40
Design	C	24	some correct patterns	
	B	32	mostly correct patterns	
	A	40	All patterns used	
Program Correctness	D	0	limited features	50
Features	C	30	some features	
	B	40	most features	
	A	50	all features	

Design Brief Segment:

Submission Specifications				10
	D	0	Significant Details Violated	
	B	6	Minor Details Violated	
	A	10	No Errors	
Accompanying Material				20
	D	0	There were no accompanying materials	
	C	12	Material difficult or hard to understand how it relates	
	B	16	Diagrams and materials had errors or were partially misleading	
	A	20	Diagrams used well and communicate ideas efficiently	
Communication	D	0	Text is poorly written and missing clarity	40
	C	24	Text is hard to understand or poorly communicates ideas	
	B	32	Text communicates ideas but lacking clarity	
	A	40	Text communicates ideas clearly	
Solution Correctness	D	0	Many features asked for are missing in the final design	50
Features	C	30	some features are defined well but many are detailed incorrectly	
	B	40	all but a few features are correctly detailed with a solution	
	A	50	all features are detailed with solutions that will work	