1. Course Code

2249

2. Course Title

Web Application Development

3. Teacher

WANNOUS, Muhammad

4. Term

Spring 1

5. Course Requirements (Courses / Knowledge prerequisite for this course)

Courses:
[1] Data Structures and Object Oriented Programming (Java) (2291)
[2] Fundamentals of Database Systems (2205)

Skills
[1] Use of the command-line and IDE
[2] Software installation and setup
[3] Programming (the programming language used is Java)

6. Course Overview and Objectives

This course introduces the way the web operates and the design of web applications in the fashionable programming languages Java and JavaScript.
It starts with an introduction to the server-client architecture and the HTTP (and maybe FTP) protocol. It then continues to cover the design of web application for the client-side (HTML, CSS, and JavaScript) and server-side (Java).
The course also covers the deployment of web applications in Apache Tomcat HTTP server. The course includes a project to complete by its end.

7. Course Outline

1 Course orientation and development environment tools and setup.
2 How the web works, the server-client architecture, & URLs
3 TCP protocols for the web (HTTP and FTP)
4 Client side (HTML & CSS)
5 Client side (Responsive Design & HTML forms)
6 Client side (JavaScript)
7 Exercise-1: designing a login, dashboard, and file upload pages
8 Server side (Servlets & the deployment descriptor)
9 Server side (reading input parameters and uploaded files)
10 Excercise-2: implement authentication and files uploading logic
11 Server side (Sessions, application context, listeners & filters)
12 Server side (combining HTML and Servlets in JSP pages)
13 Server side (MVC design pattern)
14 Exercise-3: use a controller to authenticate users & upload files. Use session and context
15 Final project: image repository.
16

8. Textbooks (Required Books for this course)

For this course, a set of lecture slides, handouts, and reports will be distributed in timely manner through Moodle.

9. Reference Books (optional books for further study)

Servlets & JSP: A Tutorial, 2nd edition, 9781771970273
JavaScript & jQuery, 9781449399023
www.w3schools.com

10. Course Goals (Attainment Targets)

(1) Describe how the web works and its protocols
(2) Describe the technologies used to construct the client-side components
(3) Practice designing web pages using HTML, CSS, and JavaScript
(4) Describe the technologies used to construct the server-side components
(5) Practice using Java to create server side web application
(6) Practice deploying a web application in a web server
(7)
(8)

11. Correspondence relationship between Educational goals and Course goals

Educational goals of the school			Course Goals
High level ICT skills	Basic academic skills		(1) (2) (4)
	Specialized knowledge and literacy		(3) (5) (6)
Human skill (Tankyu skill)	Ability to continually improve own strengths		
	Ability to discover and resolve the problem in society	Problem setting	
		Hypothesis planning	
		Hypothesis testing	
		Practice	
	Fundamental Competencies for Working Persons	Ability to step forward	
		Ability to think through	
		Ability to work in a team	
Professional ethics			

12. Evaluation

Goals	Evaluation method & point allocation					
	examination	Quiz	Reports	Presentation	Deliverables	Other
(1)		◯				
(2)		◯				
(3)					◯	
(4)		◯				
(5)					◯	
(6)					◯	
Allocation		40			60	

13. Evaluation Criteria

Examination	

Quiz	This course will include up to four quizzes. Each quiz consists of several questions of different types, simple answer, multiple choices, space-filling, …etc. The questions are to verify the knowledge a student has acquired regarding specific lectures. The quiz is limited in time, but a student can try it twice.
Reports	
Presentation	
Deliverables	The course has three exercises and one project. One exercise involves designing a web page and coding it in HTML, CSS, and JavaScript. The second exercise involves writing Java Servlets that complete the logic necessary to achieve certain functions. The last exercise involves writing a simple application that utilizes the MVC design pattern. The instructor will provide the information related to each exercise in timely manner. As for the final project, it involves designing a complete web application that has a number of functions. Each student should deliver a web archive file for the exercises and the final project, and the instructure will deploy it on his computer for testing.
Other	

14. Active Learning

Hourly percentage of active learning within the whole class time	70%

1	Active learning such as problem solving assignment using the knowledge and skills acquired in class.	All the time
2	Active learning such as group works and discussions.	Sometimes
3	Outcome presentations and feedbacks.	Not at all
4	Students actively make decisions on how the class should be conducted.	Not at all

15. Notes

This course contains both theoretical and practical parts. Be prepared for designing web pages and coding (in JavaScript and in Java).
Quizzes and projects have deadlines and they won't be postponed unless a serious issue occurs.

16. Course plan

(Notice) This plan is tentative and might be changed at the time of delivery

Lesson 1: (Course orientation and development environment tools and setup)
(Discussion and Lecture, 45 minutes & Practice 45 minutes)

[1] Course syllabus
[2] Grading
[3] Development environment
  + Java Development Kit (JDK)
  + NetBeans IDE
  + SQLite
  + Apache Tomcat

Lesson 2: (How the web works, the server-client architecture, & URLs)
(Discussion and Lecture, 90 minutes)

[1] How the WEB works?
[2] The server-client architecture
  + The web browser (client)
  + Apache tomcat (server)
[3] Deploying a web application
  + Create a web application in NetBeans
  + Build a WAR file for the project
  + Deploy the WAR file in tomcat
[4] URLs

Lesson 3: (TCP protocols for the web (HTTP and FTP))   (Practice 90 minutes)

[1] HTTP Protocol
  + Request
  + Response
  + Methods
  + Parameters

Lesson 4: (Client side (HTML & CSS))   (Lecture 30 minutes, Practice 60 minutes)

[1] HTML
  + Tags
  + Attributes
  + Forms
[2] CSS
  + classes

Lesson 5: (Client side (Responsive Design & HTML forms))
(Lecture 30 minutes, Practice 60 minutes)

[1] Responsive design
   + Principles
   + Bootstrap
[2] HTML forms
   + Passing parameters to the server
   + File uploading

Lesson 6: (Client side (JavaScript)) (Lecture 30 minutes, Practice 60 minutes)

[1] JavaScript
   + Basics
   + Document Object Model
   + Changing attributes
   + Adding nodes
   + Objects
   + JavaScript Object Notation

Lesson 7: (Exercise-1: designing a login, dashboard, and file upload pages)
(Practice, 90 minutes)

In this session, we will practice designing three pages:
1- A login page
2- User dashboard
3- File upload page

Lesson 8: (Server side (Servlets & the deployment descriptor))
(Lecture 30 minutes, Practice 60 minutes)

[1] Java Servlets
   + HttpServlet class
   + Class methods
[2] The deployment descriptor (web.xml)
   + Registering Servlets
   + Welcome page

Lesson 9 : (Server side (reading input parameters and uploaded files))
(Lecture 30 minutes, Practice 60 minutes)

[1] Reading the input parameters
   + Single-value parameter
   + Multi-values parameters
[2] Reading uploaded files.

Lesson 10 : (Excercise-2: implement authentication and files uploading logic)
(Practice, 90 minutes)

In this session, we will write the server side logic associated with the three pages we designed in Exercise-1
1- A login logic
2- User dashboard logic
3- File upload logic

Lesson 11 : (Server side (Sessions, application context, listeners & filters))
(Lecture 30 minutes, Practice 60 minutes)

[1] Session
   + Storing and retrieving objects
   + Destroying sessions
[2] Application Context
   + Storing and retrieving objects
[3] Listeners
[4] Filters
   + Filter chain

Lesson 12 : (Server side (combining HTML and Servlets in JSP pages))
(Lecture 30 minutes, Practice 60 minutes)

[1] JSP pages
   + Scriptlets
   + JSTL
   + Expression Language

Lesson 13-14 : (Server side (MVC design pattern))
(Lecture 30 minutes, Practice 60 minutes)

[1] Model View Controller
   + The model
   + The view (JSP)
   + The controller
   + The dispatcher

Lesson 14 : (Exercise-3: use a controller to authenticate users & upload files. Use session and context)
(Practice, 90 minutes)

In this session, we will convert the server side logic in Exercise-2 to utilize MVC design pattern
1- A login controller
2- User dashboard controller
3- File upload controller
4- The dispatching Servlet

Lesson 15 : (Final project: image repository)

(Lecture 30 minutes, Practice 60 minutes)

This session is dedicated to discussing the details of the final project and answering any questions on it. The project is about creating an image repository where each user can upload several images and writes the objects in every image. All users can search for images containing certain objects.