Las Positas College
3000 Campus Hill Drive
Livermore, CA 94551-7650
(925) 424-1000
(925) 443-0742 (Fax)

Course Outline for CS 44

ADVANCED WEB PROGRAMMING

Effective: Fall

I. CATALOG DESCRIPTION:
CS 44 — ADVANCED WEB PROGRAMMING — 4.00 units

The objective of this course is to develop skills and understanding in designing eCommerce websites. This is a programming course that goes beyond mere “formatting” of web pages found with HyperText Markup Language (HTML). This course extends web page “functionality” with interactivity, multimedia, security, and database capability using prior knowledge of a scripting language (HTML, JavaScript, etc.). The participant will learn about database sorting and filtering capabilities of eXtensible Markup Language (XML) that identifies data fields (e.g. refer to auto manufacturers). The participant will also learn eXtensible Stylesheet Language (XSL) -- which specifies the presentation of a class of XML documents by describing how an instance of the class is transformed into an XML document that uses the formatting vocabulary. XSL is based on and extends the Document Style Semantics and Specification Language (DSSSL) and the Cascading Style Sheet, level 1 (CSS1) standards. Additional topics include good design principles, examples of scripts (JavaScript, ASP, ActiveX, VBScript, Servlets, JSP, Perl or CGI), discussion of security (SET, SSL etc.), and examples of "good, bad, ugly" eCommerce websites.

3.00 Units Lecture 1.00 Units Lab

Prerequisite
CS 37 - Web Programming
with a minimum grade of C

Grading Methods:
Letter or P/NP

Discipline:

	MIN
Lecture Hours:	54.00
Lab Hours:	54.00
Total Hours:	108.00

II. NUMBER OF TIMES COURSE MAY BE TAKEN FOR CREDIT: 1

III. PREREQUISITE AND/OR ADVISORY SKILLS:

Before entering the course a student should be able to:

A. CS37
1. GENERIC: These outcomes are being developed throughout the entire programming sequence. Upon completion of the course, to an intermediate level, students should be able to: Programming Skills
2. explain and apply the basic and advanced concepts of HTML programming;
3. explain and apply the fundamental concepts of scripting;
4. present the elements and features of the website development environment;
5. explain and use the web page interface design process;
6. define and use decision and repetition structures in JavaScript;
7. define and use functions, recursion and storage classes in JavaScript;
8. use operators and functions using HTML/JavaScript syntax and logic;
9. define and use dynamic data structures using HTML/JavaScript;
10. define and explain trends in HTML/JavaScript standards;
11. write, compile, test and debug scripts within an HTML document;
12. present the characteristics of object-oriented programming using JavaScript;
13. define and use data types and variables in JavaScript;
14. define and use arrays in JavaScript;
15. define and use constructors in JavaScript functions;
16. define and use inheritance mechanisms in Cascading Style Sheets (CSS);
17. define and use user interfaces using the HTML form tags;
18. define and use file Input/Output (I/O) in JavaScript;
19. define and develop class modules under CSS;
20. develop and use event-driven applications (HTML, form tags, JavaScript);
21. embed one CSS within JavaScript;
22. Documentation

23. write in a concise and precise form appropriate for technical documentation;
24. explain and use techniques of HTML/JavaScript technical documentation
25. adhere to industry and organization standards for HTML/JavaScript documentation;
26. Testing and Debugging
27. develop testing procedures;
28. test programs, and document errors and solutions;
29. select testing tools and develop test system;
30. User Interface Design
31. define the requirements for the user interface in a typical eCommerce website;
32. perform user interface tests, and troubleshoot problems;
33. Problem Solving
34. use a wide range of troubleshooting methods and tools to isolate problems;
35. select the appropriate approach to identify causes of the problem based on the given situation;
36. perform systematic analysis to identify problem causes using the best available tools and processes.
37. SPECIFIC: These outcomes are detailed specifically for this course. Upon completion of the course students should be able to: work in a web page environment.
38. write JavaScript to detect browser.
39. use web page support programs.
40. use “Static” HTML tags
41. use JavaScript controls.
42. enhance HTML pages with “dynamic” through embedded JavaScript codes.
43. use JavaScript/HTML for scrolling.
44. enhance HTML pages through JavaScript cookies, arrays, and frames.
45. use JavaScript/HTML for scrolling for messages, forms, and pop-up windows.
46. enhance HTML Tags for Image and Form Objects through “client side” JavaScript with an introduction to “server side” capabilities.
47. discuss HTML/JavaScript Standards and Trends on web page publishing.

IV. MEASURABLE OBJECTIVES:

Upon completion of this course, the student should be able to:

A. GENERIC: These outcomes are being developed throughout the entire programming sequence. Upon completion of the course, to an advanced level, students should be able to: Programming Languages
   1. Explain and apply the basic and advanced concepts of XML/XSL programming
   2. Present the elements and features of the website development environment
   3. Explain and use the design process for eCommerce
   4. Define and use decision and repetition structures
   5. Define and use functions, recursion and storage classes
   6. Use operators and functions using XML/XSL syntax and logic
   7. Define and use dynamic data structures using XML/XSL
   8. Define and explain trends in XML/XSL programming standards
   9. Write, compile, test and debug XML/XSL programs
   10. Present the characteristics of object-oriented programming
   11. Define and use data types and variables
   12. Define and use arrays
   13. Define and use pointers
   14. Define and use inheritance mechanisms in Object Oriented Programming (OOP)
   15. Define and use user interfaces
   16. Define and use file Input/Output (I/O)
   17. Define and develop class modules
   18. Develop and use event-driven programs (HTML, form tags for data entry
   19. Combine XML with other technologies (JavaScript, ActiveX, etc.)

B. Database Design
   1. Explain database design concepts and the role of database components
   2. Model data and design XML database structure
   3. Explain the use of XML databases and information in the business environment
   4. Develop XML database business applications

C. Systems Design
   1. Specify major subsystems and interfaces
   2. Develop detail design specifications
   3. Select design methodology and tools
   4. Identify maintenance requirements for XML/XSL interface.
   5. Perform feasibility studies of design alternatives (e.g. Java, JavaScript, ActiveX, etc.)
   6. Identify physical requirements for systems implementation
   7. Prepare and conduct design reviews
   8. Identify impact on existing systems
   9. Perform usability testing and human factor analysis
   10. Develop test plan

D. Technical Documentation
   1. Write in a concise and precise form appropriate for technical documentation
   2. Explain and use the processes and techniques of XML/XSL technical documentation
   3. Prepare materials written to convey specific technical XML/XSL problems, their related issues, and their solutions
   4. Explain the purpose of logs, reports, training manuals and other forms of technical documentation
   5. Adhere to XML/XSL documentation industry and organization guidelines and standards

E. Testing and Debugging
   1. Select debugging and testing methodology, and develop comprehensive and systematic test plan
   2. Select program debugging tools and techniques
   3. Develop testing procedures
   4. Conduct tests in the most efficient way
   5. Test programs, and document errors and solutions
   6. Select testing tools and develop test system

F. User Interface Design
   1. Define the requirements for the user interface in a typical eCommerce website
   2. Perform user interface tests, and troubleshoot problems

G. Problem Solving
   1. Recognize a wide range of problems, and assess their impact on the system
   2. Use a wide range of troubleshooting methods and tools to isolate problems

H. SPECIFIC: These outcomes are detailed specifically for this course. Upon completion of the course students should be able to: Discuss Internet, WWW, and eCommerce.

I. Use Basic HTML – headers, text styling, linking, images, fonts, special characters
J. Use Advanced HTML – lists, tables, forms, maps, meta tags, frames.
K. Use JavaScript – control structures, functions, arrays, objects.
L. Use the fundamentals of XML syntax and grammar.
M. Create Data Type Definitions.
N. Use Customized Markup Languages.
O. Work with Client-side XML and Server-side XML.
P. Work with DHTML: CSS, objects, events, filters, data binding, ActiveX controls.
Q. Discuss trends in Electronic Commerce and Security, XML, Active Server Pages

V. CONTENT:
A. Introduction – Computers, Internet, WWW, eCommerce
B. Review of Basic HTML – headers, text styling, linking, images,
C. fonts, special characters
D. Review of Advanced HTML – lists, tables, forms, maps, meta tags,
E. frames
F. Review of JavaScript – control structures, functions, arrays, objects
G. Fundamentals of XML syntax and grammar
H. Data Type Definitions
I. Creating Customized Markup Languages
J. Client-side XML
K. Server-side XML
L. DHTML – CSS, objects, events, filters, data binding, ActiveX controls
M. Trends in Electronic Commerce and Security, XML, Active Server Pages

VI. METHODS OF INSTRUCTION:
A. **Lecture** -
B. **Demonstration** -
C. **Projects** - Optional: Programming project completed in teams
D. **Lab** - Lab Programming Assignments
E. **Discussion** -

VII. TYPICAL ASSIGNMENTS:
A. Create an XML application allowing users to view records using a series of buttons 1. [|<] First Record 2. [<] Previous Record 3. [>] Next Record 4. [>|] Last Record 5. You will have to create a separate data file. Display your output in a "static" HTML table and call the data file with the command. 6. NOTE: HTML assumes the data file is in the same directory and on the same drive as the calling HTML web page. 7. You will have entered the records and stored them as: a. Employee ID b. Employee First Name c. Employee Last Name d. Hours Worked e. Pay Per Hour f. Federal Tax Rate g. State Tax Rate

8. The buttons are from the "static" form tag 

9. XML recognizes the following button onclick commands: a. 'xmldso.recordset.movefirst()' b. 'xmldso.recordset.moveprevious()' c. 'xmldso.recordset.movenext()' d. 'xmldso.recordset.movelast()' 10. The Next Record and the Previous Record function requires a