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

Course Outline for CS 37

WEB PROGRAMMING

Effective: Spring 2018

I. CATALOG DESCRIPTION:
CS 37 — WEB PROGRAMMING — 4.00 units

[This needs to be completely rewritten -- web programming for HTML5 / Javascript / 5 dozen frameworks era!]

3.00 Units Lecture 1.00 Units Lab

Prerequisite
CS 7 - Introduction to Computer Programming Concepts
or

Strongly Recommended
CS 1 - Computing Fundamentals I

Grading Methods:
Letter or P/NP

Discipline:
- Computer Science

	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. CS7
1. Design simple algorithms to solve a variety programming problems.
2. Design and implement programs of short to medium length, using standard elements of programming languages such as variables, input/output, control structures, functions/methods and arrays.
3. Analyze and investigate program behavior to effectively alter or debug existing code.
4. Design and implement specific program steps and components to achieve desired program behavior.
5. Design and implement simple graphical and command line user interfaces implementing the students algorithms.

Before entering this course, it is strongly recommended that the student should be able to:

A. CS1
1. Design, create and compile C++ programs within multiple development environments and operating systems, including the use of command-line tools in Unix/Linux.

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 intermediate level, students should be able to: Programming Skills
1. explain and apply the basic and advanced concepts of HTML programming;
2. explain and apply the fundamental concepts of scripting;
3. present the elements and features of the website development environment;
4. explain and use the web page interface design process;
5. define and use decision and repetition structures in JavaScript;
6. define and use functions, recursion and storage classes in JavaScript;
7. use operators and functions using HTML/JavaScript syntax and logic;
8. define and use dynamic data structures using HTML/JavaScript;
9. define and explain trends in HTML/JavaScript standards;
10. write, compile, test and debug scripts within an HTML document;
11. present the characteristics of object-oriented programming using JavaScript;

12. define and use data types and variables in JavaScript;
13. define and use arrays in JavaScript;
14. define and use constructors in JavaScript functions;
15. define and use inheritance mechanisms in Cascading Style Sheets (CSS);
16. define and use user interfaces using the HTML form tags;
17. define and use file Input/Output (I/O) in JavaScript;
18. define and develop class modules under CSS;
19. develop and use event-driven applications (HTML, form tags, JavaScript);
20. embed one CSS within JavaScript;

B. Documentation
   1. write in a concise and precise form appropriate for technical documentation;
   2. explain and use techniques of HTML/JavaScript technical documentation
   3. adhere to industry and organization standards for HTML/JavaScript documentation;

C. Testing and Debugging
   1. develop testing procedures;
   2. test programs, and document errors and solutions;
   3. select testing tools and develop test system;

D. User Interface Design
   1. define the requirements for the user interface in a typical eCommerce website;
   2. perform user interface tests, and troubleshoot problems;

E. Problem Solving
   1. use a wide range of troubleshooting methods and tools to isolate problems;
   2. select the appropriate approach to identify causes of the problem based on the given situation;
   3. perform systematic analysis to identify problem causes using the best available tools and processes.

F. 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.

G. write JavaScript to detect browser.

H. use web page support programs.

I. use “Static” HTML tags

J. use JavaScript controls.

K. enhance HTML pages with “dynamic” through embedded JavaScript codes.

L. use JavaScript/HTML for scrolling.

M. enhance HTML pages through JavaScript cookies, arrays, and frames.

N. use JavaScript/HTML for scrolling for messages, forms, and pop-up windows.

O. enhance HTML Tags for Image and Form Objects through “client side” JavaScript with an introduction to “server side” capabilities.

P. discuss HTML/JavaScript Standards and Trends on web page publishing.

V. CONTENT:

A. Web page environment
   1. Netscape or
   2. Internet Explorer

B. JavaScript to detect browser type and to shift automatically to appropriate pages depending on browser type (meta tags)

C. Web page support programs
   1. web page editor
   2. graphic image editor
   3. FTP
   4. lab capabilities
   5. free web storage sites

D. "Static" HTML tags
   1. text formatting
   2. image placement
   3. URL links
   4. background colors and images
   5. object spacing tags
   6. mailt
   7. embedding multimedia sound and movie clips

E. JavaScript controls
   1. push buttons
   2. pull down menu lists
   3. text boxes
   4. radio buttons
   5. checkbox buttons
   6. dynamic date, time, label “inner text” presentation

F. Enhance HTML pages with “dynamic” through embedded JavaScript codes using
   1. loops
   2. arithmetic statements
   3. IF
   4. functions
   5. passing argument parameters

G. Use JavaScript/HTML for scrolling
   1. messages
   2. forms
   3. pop-up windows

H. Enhance HTML Tags for Image and Form Objects through “client side” JavaScript with an introduction to “server side” capabilities

I. Enhance HTML pages through JavaScript
   1. Cookies
   2. Arrays
   3. Frames

J. HTML/JavaScript Standards and Trends on web page publishing

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 special effects with HTML Tags 1. Discussion: Aside from the standard background colors or images, hypertext links, embedded graphic (i.e. jpg or gif) images, frames, and assorted things you can do with the text on a standard web page, what attracts the short attention span of the typical net surfer? We need a “gimmick”, something that “grabs” your attention. a. Marquees - A banner that moves across your screen displaying some message. A human eye will actually spend a few precious seconds to read that moving marquee. There are 3 ways to write a marquee: i. Using JavaScript or ActiveX for a marquee to run anywhere on the web page including the bottom status bar. Javascript works on both Netscape and Internet Explorer. The drawback is learning to code JavaScript or VBScript with an ActiveX Control. ii. Using Microsoft Internet Explorer’s built in tag . Unfortunately this tag works only on Microsoft Internet Explorer. iii. Using the Gif Construction application. b. Dissolving graphical images - Using the Gift Construction application, you can select an image and create a looping dissolve effect that is sure to grab attention. c. Background music – using the Music producer (or finding a MIDI tune from the web), insert a MIDI tune into your HTML web page. You probably want make sure that the console appears when the page is loaded, the music plays automatically upon page opening, Width is 50 and Length is 200 (so the console appears). d. Image maps - Tired of the row upon row of tags? There are several options: i. WINDOWS compatible Map This. ii. The built in Map Editor in the Web Page Creator (WPC). You will need a graphic picture (GIF) for the basic map. 2. Create yet another web page theme with the following capabilities: a. Embed an image map to at least 3 different related web sites. b. Select a main image that symbolizes your web page theme and using the Gif Constructor application, dissolve it. When we look at it, a “dynamite” special effect occurs. c. Create a marquee somewhere on your web page. If you use Microsoft Internet Explorer, be sure to note on your web page “Microsoft Internet Explorer Only”. You should try the Gift Constructor. d. Embed a sound file (use the embed src tag). You might wish to leave the controls so the viewers can turn off the sound if they your choice “annoying”. You might wish to use the autostart=false option to give them the choice to turn it on or off. B. Special effects with Cascading Style Sheet 1. Discussion: Cascading Style Sheets (CSS) has the capability of changing the appearance of your web page according to what your user does. Examples include the JavaScript codes – onemouseout, onmouseover or onclick events – in response to the user’s mouse activity. Text appearance (font size, font type, font color, vertical alignment, horizontal alignment, character format, paragraph format, page layout, etc.) and graphic image appearance (foreground, background, text wrap, etc.) can be controlled using Cascading Style Sheets. 2. Your assignment is as follows: a. Change the text with the codes onmouseover and onmouseout. b. Change lines of text by positioning your mouse over an image graphic (Microsoft Internet Explorer’s innertext command). c. Reposition a static image and text without using the HTML , „

or HTML table structure anywhere on your web page space. d. To accomplish this assignment: i. Determine a theme plus at least 3 related URL links to your theme. ii. Create or obtain at least 2 related graphics (i.e. in .jpg or .gif formats) iii. To reposition an object, use the following code – Style=”position:absolute; top:400; left:20;” iv. To change text on an image, use the following code – onmouseover=”Saying2()” onmouseout=”Saying1()” v. To control the timer, use the following code – setTimeout(“go()”,5*1000); vi. To define the CSS Styles directly in the tag itself, use the following code vii. Transitions are activated with a

VIII. EVALUATION:

A. **Methods**

B. **Frequency**

1. Frequency of evaluation
   a. Recommend 2 or 3 exams plus final examination
   b. Recommend programming assignment to cover each topic within course content. Contents can be combined.
2. Types of Exam Questions
   a. Write the code to reposition an object.
   b. Write the code to control the timer.
   c. Write the code to change text on an image.
   d. Transitions are activated with a

IX. TYPICAL TEXTS:

1. Felke-Morris, Terry. *Web Development and Design Foundations with HTML5.* 7 ed., Addison-Wesley, 2015.
2. Welling, Luke, and Laura Thomson. *PHP and MySQL Web Development.* 5 ed., Addison-Wesley Professional, 2015.

X. OTHER MATERIALS REQUIRED OF STUDENTS:

A. It is recommended that students have a portable data-storage device (i.e. USB drive) or maintain an active cloud-storage account to facilitate saving and transfer of their work.