

**UNIVERSITY OF NEBRASKA AT OMAHA
COURSE SYLLABUS**

Department and Course Number	CIST 1300
Course Title	Introduction to Web Development
Course Coordinator	Robert Fulkerson
Total Credits	3
Repeat for Credit?	No
Date of Last Revision	March 30, 2014

1.0 Course Description Information

1.1 Catalog description:

This course will provide students with a practical introduction to web development. By learning the basic skills needed to develop an interactive website, students will develop an understanding of the web development task and an appreciation of the importance of the Internet in both business and academic environments. Specific technical topics to be covered include HTML, CSS, the Unix/Linux operating system, web server software, and a programming language. As part of the class, each student will develop a working website.

1.2 Prerequisites of the course:

MATH 1310

1.3 Overview of content and purpose of the course:

With the wide proliferation of "computers in your pocket" in the form of "smart phones" and a seemingly endless barrage of other everyday devices such as portable gaming devices, built-in car computers, gaming consoles, televisions and even kitchen appliances, knowledge of how technology works is no longer optional to be competently functional in today's fast-paced, high-tech, always-on society.

This course is an introduction to the primary "programming" languages of the World Wide Web, which are HTML and CSS. Students will learn how to put together a standards-compliant web site that will be accurately viewable in any web browser across any platform, including desktop and laptop computers and mobile devices such as "smart phones" and tablets such as the iPad.

Students will then learn about how the World Wide Web works "behind the scenes" by learning how to write computer programs and connect their web pages to data sources, thereby allowing them to create "dynamic" web sites instead of simple, "static" web sites.

Through this gradual build-up of knowledge, students gain an appreciation for the discipline required to create usable and useful websites that they use on a daily basis.

1.4 Unusual circumstances of the course.
None

2.0 Course Justification Information

2.1 Anticipated audience / demand:

The course is intended for all students interested in computers and web development.

2.2 Indicate how often this course will be offered and the anticipated enrollment:

This course is offered every semester, including during the summer. Typical enrollments based on previous semesters:

Fall Semester: 110 students

Spring Semester: 90 students

Summer Semester: 20 students

2.3 If it is a significant change to an existing course, please explain why it is needed:

None

3.0 List of performance objectives stated in learning outcomes in a student's perspective:

3.1 A working knowledge of a command-line oriented operating system.

3.2 The ability to create a standards-compliant HTML5 web page.

3.3 The ability to create standards-compliant CSS documents.

3.4 The ability to write simple programs in the Perl programming language.

4.0 Content and Organization Information

4.1 Introduction and Overview 1

4.2 Unix 0.5

4.2.1 Overview of Unix

4.2.2 ssh

4.2.3 mkdir

4.2.4 chmod

4.2.5 directory structure

4.2.6 cd

4.2.7 ls

4.2.8 mv

4.2.9 rm

4.2.10 vim

4.3 HTML5 2

4.3.1 Overview

4.3.2 Web browsers

4.3.3	Web servers	
4.3.4	Block vs. Inline elements	
4.3.5	Content vs. Structure	
4.3.6	URLs, relative and absolute	
4.3.7	Minimal page (<!DOCTYPE>, <html>, <head>, <title>, <meta>, <body>)	
4.3.8	Headers (<h1>, <h2>, <h3>, <h4>, <h5>, <h6>)	
4.3.9	Sectioning elements (<section>, <article>, <header>, <footer>, etc)	
4.3.10	Paragraphs, line breaks and blocks (<p>, , <div>,)	
4.3.11	Naming and labeling elements (id, class)	
4.3.12	Comments (<!-- -->)	
4.3.13	Formatting (, <i>, , , <code>, <tt>, <sample>, <pre>)	
4.3.14	Block quotes (<blockquote>)	
4.3.15	More formatting (<sup>, <sub>, <ins>,)	
4.3.16	Abbreviations and acronyms (<abbr>, <acronym>)	
4.3.17	Images ()	
4.3.18	Image types (JPEG, GIF, PNG)	
4.3.19	Links (<a href>, <a name>)	
4.3.20	Lists (, , <dl>, , <dt>, <dd>)	
4.3.21	Tables (<table>, <tr>, <td>, <th>)	
4.3.22	Audio and video (<audio>, <video>)	
4.3.23	Audio and video formats supported in HTML5	
4.4	CSS	1.5
4.4.1	Overview	
4.4.2	Inline vs. Embedded vs. External	
4.4.3	Inline CSS via style attribute	
4.4.4	CSS Predefined Values	
4.4.5	CSS Lengths / Percentages (em, px, in, cm, mm, pt, pc)	
4.4.6	RGB triplets for color	
4.4.7	Common properties used for formatting (border, border-color, width, padding, font-size, font-family, background-color, border-top, border-bottom, etc)	
4.4.8	Embedded CSS via <style> tag	
4.4.9	Style sheets and rules	
4.4.10	External CSS via <link> tag	
4.4.11	Pseudo-classes	
4.4.12	What is meant by “Cascading” style sheets	
4.4.13	Box Model of CSS Layout	
4.4.14	Positioning (absolute, relative, static, fixed)	
4.5	Perl	6
4.5.1	Overview	
4.5.2	Machine language, Assembly Language, High-level Language	
4.5.3	Compiling	
4.5.4	Basic output using print, printf	

4.5.5	Types of programming errors (syntax, runtime, semantic)	
4.5.6	Scalar data type	
4.5.7	Basic input using <>	
4.5.8	Variable interpolation	
4.5.9	Escape characters	
4.5.10	chomp() and chop()	
4.5.11	Basic arithmetic (+, -, *, /, %, **, .., x)	
4.5.12	Precedence of operators	
4.5.13	Compound assignment operators	
4.5.14	Unary increment and decrement operators	
4.5.15	if selection	
4.5.16	if/else selection	
4.5.17	Boolean expressions (&&, , !)	
4.5.18	Uninitialized variables	
4.5.19	while repetition	
4.5.20	Array data type	
4.5.21	foreach repetition	
4.5.22	for repetition	
4.5.23	Data::Dumper for debugging	
4.6	HTML5 Forms	1
4.6.1	Overview	
4.6.2	Submit and reset buttons	
4.6.3	Checkboxes and radio buttons	
4.6.4	Text fields, password fields, text areas	
4.6.5	Combo boxes	
4.6.6	Date/time, file upload, color picker, etc.	
4.7	CGI	2
4.7.1	Overview	
4.7.2	Static content vs. dynamic content	
4.7.3	CGI defined	
4.7.4	Basic CGI using simple Content-Type: text/html	
4.7.5	CGI.pm overview	
4.7.6	CGI.pm to create basic HTML5	
4.7.7	CGI::Carp for debugging	
4.7.8	CGI.pm to create HTML5 forms	
4.7.9	CGI.pm to process forms	

5.0 Teaching Methodology Information

5.1 Methods:

The material is taught through lectures based on a common set of slides, assignments and exams for all instructors. Some sections may use hands-on laboratory classrooms for each lecture or for special lectures throughout the semester.

5.2 Student role:

Students are encouraged to read the material in the books in advance of the lecture so that new concepts and terms are not “sprung on them”. They are encouraged to participate in class discussions and ask questions when the material is unclear since all of the material is cumulative and builds on understanding of the material that has come before it in the course. Students are also encouraged to try out all of the example code demonstrated in lecture to familiarize themselves with the concepts being covered.

6.0 Evaluation Information

6.1 Describe the typical types of student projects that will be the basis for evaluating student performance:

Each student will write complete approximately 15 assignments on the various topics as they are covered during the semester. These assignments will include creating HTML5 web pages, CSS style sheets and Perl programs. Additionally, the students will take a mid-term and a final exam along with quizzes as needed.

6.2 Describe the typical basis for determining the final grade (e.g. weighting of various student projects):

Homework assignments: 70%

Exams: 30%

6.3 Grading type:

This section usually provides a mapping from a numeric score to a letter grade. The numeric score could be an overall percentage figure (based on section 6.2) or some number of points, if points are given for each activity. For example:

Percent	Grade	Percent	Grade
97 – 100	A+	77 – 79	C+
93 – 96	A	73 – 76	C
90 – 92	A–	70 – 72	C–
87 – 89	B+	67 – 69	D+
83 – 86	B	63 – 66	D
80 – 82	B–	60 – 62	D–
		0–59	F

7.0 Resource Material Information

7.1 Textbooks and/or other required readings used in course:

7.1.1 Castro, Elizabeth. *HTML & CSS Visual QuickStart Guide*. Berkeley, CA:

Peachpit, 2013. Print.

7.1.2 Schwartz, Randal L., Tom Phoenix, and Brian D. Foy. *Learning Perl*. Beijing:

O'Reilly, 2011. Print.

7.2 Other student suggested reading materials:

7.2.1 "HTML & CSS - W3C." *World Wide Web Consortium (W3C)*. Web. 19 Apr. 2011. <<http://www.w3.org/standards/webdesign/htmlcss>>.

7.2.2 *Perl.com*. Web. 19 Apr. 2011. <<http://perl.com>>.

7.2.3 "HTML5 Tutorial." *W3Schools Online Web Tutorials*. Web. 19 Apr. 2011. <<http://w3schools.com/HTML5/>>.

7.3 Current bibliography and other resources:

7.3.1 Castro, Elizabeth. *HTML & CSS: A Visual QuickStart Guide*. Berkeley, CA: Peachpit, 2013. Print.

7.3.2 "CGI - Perldoc.perl.org." *Perl Programming Documentation - Perldoc.perl.org*. Web. 19 Apr. 2011. <<http://perldoc.perl.org/CGI.html>>.

7.3.3 "CSS Tutorial." *W3Schools Online Web Tutorials*. Web. 19 Apr. 2011. <<http://w3schools.com/css/>>.

7.3.4 Harris, Andrew. *HTML, HTML5, and CSS All-in-one for Dummies*. Hoboken, NJ: Wiley, 2011. Print.

7.3.5 "HTML & CSS - W3C." *World Wide Web Consortium (W3C)*. Web. 19 Apr. 2011. <<http://www.w3.org/standards/webdesign/htmlcss>>.

7.3.6 "Perl 6." *Perl 6*. Web. 19 Apr. 2011. <<http://perl6.org/>>.

7.3.7 *Perl.com*. Web. 19 Apr. 2011. <<http://perl.com>>.

7.3.8 Schwartz, Randal L., Tom Phoenix, and Brian D. Foy. *Learning Perl*. Beijing: O'Reilly, 2008. Print.

7.3.9 "HTML5 Tutorial." *W3Schools Online Web Tutorials*. Web. 19 Apr. 2011. <<http://w3schools.com/HTML5/>>.

7.3.10 "HTML5." *Wikipedia, the Free Encyclopedia*. Web. 19 Apr. 2011. <<http://en.wikipedia.org/wiki/HTML5>>.

8.0 Other Information:

8.1 Accommodations statement:

8.2 Other:

8.3 Author(s):

Robert Fulkerson

9.0 Computer Science Accreditation Board (CSAB) Category Content (class time in hours):

<i>CSAB Category</i>	<i>Core</i>	<i>Advanced</i>
Data structures	6	0
Computer organization and architecture	0	0
Algorithms and software design	0	0
Concepts of programming languages	37.5	0

10.0 Oral and Written Communications:

Every student is required to submit at least 0 written reports (not including exams, tests, quizzes, or commented programs) to typically 0 pages and to make 0 oral presentations of typically 0 minutes duration. Include only material that is graded for grammar, spelling, style, and so forth, as well as for technical content, completeness, and accuracy.

11.0 Social and Ethical Issues:

None

12.0 Theoretical content:

Please list the types of theoretical material covered, and estimate the time devoted to such coverage.

None

13.0 Problem analysis:

Please describe the analysis experiences common to all course sections.

Students will learn a number of fundamental computer concepts and skills. At the end of the course, they will be able to apply these skills to the development of their own dynamic, data-driven web site.

14.0 Solution design:

Please describe the design experiences common to all course sections.

This course will require the student to apply a set of initially unfamiliar tools to a diverse set of problems. As the student works through each problem, he/she will gain experience and competence in the broader skill of problem solving.

CHANGE HISTORY

<i>Date</i>	<i>Change</i>	<i>By whom</i>	<i>Comments</i>
04/22/2004	Initial ABET version	Pauley	
03/09/2005	Change to initial course of a sequence	Brown	Also formatted the course to be suitable for General Education
08/18/2005	Reorganization of Content and Organization section to make the course more web page centric. Also added a semester project.	Pauley	
09/11/2006	Further reorganization of Content and Organization to make the course even more web page centric.	Pauley	
04/21/2008	Revision to reflect course conversion to Perl since Fall 2007	Fulkerson	
04/24/2008	Revision to reflect CSCI 1300 cross-listing and removal of "database" in course description	Fulkerson	
11/17/2008	Update to include CS Program outcomes	Fulkerson	
04/19/2011	Update to new CCMS format	Fulkerson	
03/30/2014	Updated for HTML5 and CSS changes	Fulkerson	

This table is used to relate performance objectives (in section 3 of the syllabus) to the ABET program outcomes. List the appropriate performance objectives in the left column. In the body of the table, use S and X to indicate the relationship between the performance objective and the ABET program outcomes. Leave cells blank if there is no relationship. Add additional rows to the table as needed.

S – Strong relationship

X – Contributing relationship

Course objective	(a) knowledge of discipline	(b) analyze problem, define requirements	(c) design and implement solution	(d) function on a team	(e) ethical issues	(f) communicate effectively	(g) analyze impact of computing	(h) continued professional development	(i) Current techniques and tools	(j) apply foundations	(k) apply design and development principles
A working knowledge of a command-line oriented operating system.					S				S		
The ability to create a standards-compliant HTML5 web page.		X	X		S				S		
The ability to create standards-compliant CSS documents.		X	X		S	X		S	S		
The ability to write simple programs in the Perl programming language.		X	X		S	X		S	S		