

CIS 58.52 Course Outline as of Spring 2004**CATALOG INFORMATION**

Dept and Nbr: CIS 58.52 Title: XML - INTRO FOR WEB DEV

Full Title: XML - An Introduction for Web Developers

Last Reviewed: 4/14/2003

| Units | | Course Hours per Week | | Nbr of Weeks | Course Hours Total | |
|---------|------|-----------------------|------|--------------|--------------------|-------|
| Maximum | 3.00 | Lecture Scheduled | 2.00 | 17.5 | Lecture Scheduled | 35.00 |
| Minimum | 3.00 | Lab Scheduled | 0 | 17 | Lab Scheduled | 0 |
| | | Contact DHR | 3.50 | | Contact DHR | 61.25 |
| | | Contact Total | 5.50 | | Contact Total | 96.25 |
| | | Non-contact DHR | 0 | | Non-contact DHR | 0 |

Total Out of Class Hours: 70.00

Total Student Learning Hours: 166.25

Title 5 Category: AA Degree Applicable

Grading: Grade or P/NP

Repeatability: 00 - Two Repeats if Grade was D, F, NC, or NP

Also Listed As:

Formerly:

Catalog Description:

This course introduces students to the Extensible Mark-up Language, XML. Students will learn how XML can be used to define industry specific Internet mark-up languages such as HTML (Hypertext Markup Language) and HDML (Handheld Device Mark-up Language). Students will be given projects requiring research, implementation and deployment of valid XHTML web pages created with XML applications which are linked to Cascading Style Sheets. The course is designed for students who wish to learn about advanced internet publishing technologies.

Prerequisites/Corequisites:

Course Completion of CS 50.11B (or CIS 58.51B or CIS 84.42B)

Recommended Preparation:

Knowledge of computer programming and completion of ENGL 100A or 100 or ESL 100.

Limits on Enrollment:**Schedule of Classes Information:**

Description: This course introduces students to the Extensible Markup Language (XML) and the Extensible Style Language. XML syntax features will be covered as well as document type definitions (DTDs), schemas, and XSLT for document rendering. The course is designed for

students who wish to learn about advanced internet publishing technologies. (Grade or P/NP)
Prerequisites/Corequisites: Course Completion of CS 50.11B (or CIS 58.51B or CIS 84.42B)
Recommended: Knowledge of computer programming and completion of ENGL 100A or 100 or ESL 100.

Limits on Enrollment:

Transfer Credit: CSU;

Repeatability: Two Repeats if Grade was D, F, NC, or NP

ARTICULATION, MAJOR, and CERTIFICATION INFORMATION:

| | | | |
|----------------------|----------------------|----------------------|---------------------|
| AS Degree: | Area | Effective: | Inactive: |
| CSU GE: | Transfer Area | Effective: | Inactive: |
| IGETC: | Transfer Area | Effective: | Inactive: |
| CSU Transfer: | Transferable | Effective: Fall 2002 | Inactive: Fall 2010 |
| UC Transfer: | | Effective: | Inactive: |

CID:

Certificate/Major Applicable:

Not Certificate/Major Applicable

COURSE CONTENT

Outcomes and Objectives:

Upon completion of the course, students will be able to:

1. Produce XHTML documents which will validate error-free according to the W3C XHTML validator.
2. Produce XHTML compliant style sheets which will validate error-free according to the W3C CSS validator.
3. Produce short XML applications which will validate error-free according the W3C XML validator.
4. Experiment with and create well-formed and valid XML documents.
5. Combine XML and CSS to produce valid XHTML documents.
6. Create and utilize simple Document Type Definition (DTD) files.
7. Successfully prepare data for electronic data interchange (EDI) and transmit the data over the World Wide Web.

Topics and Scope:

1. Introduction
 - a. The Problem with HTML
 - b. The Power of XML
 - c. XML Helpers
 - d. XML in the Real World
 - e. What is XML?
 - f. How are HTML and XML related?
 - g. How can XML enhance HTML?
 - h. XHTML as a Bridge to XML

2. Writing Well-Formed XML
 - a. Rules for Writing XML
 - b. Writing Non-Empty Elements
 - c. Nesting Elements
 - d. Adding Attributes
 - e. Using Empty Elements
 - f. Writing Comments
 - g. Displaying Tags as Text
 - h. Errors in XML
3. XML in the Browser: Cascading Style Sheets (CSS)
 - a. History of styling HTML
 - b. Style Sheets and HTML
 - c. External styles in HTML
 - d. Internal XML Style Sheets
 - e. Using Media in Style Sheets
 - f. Defining Elements as Block-Level or Inline
 - g. Layout with CSS
 - h. Formatting Text with CSS
4. XSLT
 - a. Transforming XML with XSLT
 - b. Beginning an XSLT Stylesheet
 - c. Creating the Root Template
 - d. Outputting HTML Code
 - e. Outputting a Node's Content
 - f. Creating and Applying Template Rules
 - g. Batch-Processing Nodes
 - h. Processing Nodes Conditionally
 - i. Sorting Nodes Before Processing
 - j. Generating Attributes
5. XPath
 - a. Determining the Current Node
 - b. Referring to the Current Node
 - c. Selecting a Node's Descendants
 - d. Disregarding the Current Node
 - e. Selecting a Node's Attributes
 - f. Selecting Subsets
6. The Document Object Model (DOM)
 - a. Declaring an Internal DTD
 - b. Writing an External DTD
 - c. Naming an External DTD
 - d. Declaring a Personal External DTD
 - e. Declaring a Public External DTD
7. Defining Elements and Attributes in a DTD
 - a. Defining Elements and Element Types
 - b. Defining Choices
 - c. Defining How Many Units
 - d. About Attributes
 - e. Defining Simple and Unique Value Attributes
 - f. Referencing and Restricting Attributes
8. Entities and Notations in DTDs
 - a. Creating Shortcuts for Text
 - b. Using Shortcuts for Text

- c. Shortcuts for Text in External Files
- d. Creating and Using Shortcuts for DTDs
- e. Creating Entities for Unparsed Content
- f. Embedding Unparsed Content
- 9. XML Schema
 - a. Simple and Complex Types
 - b. Local and Global Declarations
 - c. Beginning a Simple Schema
 - d. Indicating a Simple Schema's Location
 - e. Annotating Schemas
- 10. Defining and Using Simple and Complex Types
 - a. Declaring an Element with a Type
 - b. Deriving and Using Custom Simple Types
 - c. Specifying a Set of Acceptable Values, Pattern and Ranges
 - d. Limiting a Simple Type
 - e. Creating List Types
 - f. Predefining an Element's Content
 - g. Limiting Elements
 - h. Defining and Referencing Named Groups
 - i. Referencing Already Defined Elements
 - j. Basing Complex Types on Complex Types
 - k. Defining and Declaring Basic Complex Types
 - l. Elements with Anonymous Complex Types
 - m. Declaring and Requiring Attributes
 - n. Predefining an Attribute's Content
 - o. Defining and Referencing Attribute Groups
- 11. Namespaces, Schemas, and Validation
 - a. Designing and Declaring a Namespace
 - b. Namespaces for Individual Elements
 - c. Namespaces, Attributes, DTDs, and Valid Documents
 - d. Schemas and Namespaces
 - e. Populating a Namespace
 - f. Adding Declared Elements
 - g. Referencing Components in Schemas
 - h. The Schema of Schemas as the Default
 - i. Namespaces and Validating XML
 - j. Indicating and Locating a Schema
- 12. Links and Images: XLink and XPointer
 - a. Creating a Simple Link and a Linkset
 - b. Defining Reference Points and Connections
 - c. Using a Linkset
 - d. Linking to Part of a File
 - e. Creating the Simplest XPointer
 - f. Creating Walking XPointers
 - g. Creating an XPointer Range

Assignment:

- 1. Create 2-4 error-free Cascading Style Sheets (CSS).
- 2. Create 4-6 error-free XHTML documents.
- 3. Create 10-12 short error-free XML applications.
- 4. Integrate Style Sheets & XML Applications to produce XHTML documents.

5. Use the W3C XHTML, CSS, & XML Validators to test the validity of and correct all class work prior to submission for grading.
6. Use the World Wide Web to locate 3-5 on-line XML resources and submit these to a class resource page maintained by the instructor.
7. Complete a web page report on current and/or projected uses of XML for web page developers.
8. Take quizzes and exams.
9. Read from textbook, 20-40 pages per week.

Methods of Evaluation/Basis of Grade:

Writing: Assessment tools that demonstrate writing skills and/or require students to select, organize and explain ideas in writing.

None, This is a degree applicable course but assessment tools based on writing are not included because problem solving assessments are more appropriate for this course.

Writing
0 - 0%

Problem Solving: Assessment tools, other than exams, that demonstrate competence in computational or non-computational problem solving skills.

Homework problems, Web-based projects

Problem solving
25 - 60%

Skill Demonstrations: All skill-based and physical demonstrations used for assessment purposes including skill performance exams.

Performance exams, Create web pages and scripts

Skill Demonstrations
20 - 50%

Exams: All forms of formal testing, other than skill performance exams.

Multiple choice, True/false, short answer, script correction

Exams
20 - 30%

Other: Includes any assessment tools that do not logically fit into the above categories.

None

Other Category
0 - 0%

Representative Textbooks and Materials:

1. "Beginning XML", by David Hunter - Wrox Press 2001
2. "Learning XML", by Erik T. Ray - O'Reilly 2001