CIS 311: Interactive Web Development

Spring 2017

Instructor: Stefan Bund, MS
Office: Building 164 Room 3017
E-mail: slbund at cpp.edu
Phone: 949.433.7487

1. Class Meetings and Office Hours, Holidays
Class meetings:
Section 1:
Tuesday, Thursday from 1 pm until 2:50 pm, Building 98C, room 4-35
Office Hours: Tuesday, Thursday 4 - 5:50 pm, office above

Section 2:
Tuesday, Thursday from 6 pm until 7:50 pm, Building 98C, room 4-35
Office Hours: Tuesday, Thursday 4 - 5:50 pm, office above

Holidays for Spring 2017
March 31st, May 29th

2. Course Description

From our catalog:
The Design and development of web applications for business. Principles and applications of modern website design. Use of client-side scripting for website dynamics and interactivity. Development of server-side scripts for three-tier web applications. 4 units lectures/problem solving.

Instruction will focus upon core protocols supporting browser markup and web-based applications: html 5, CSS, javascript, server side scripting, cloud infrastructures and internet-facing data bases. The course entails a hands-on basis for learning, with deliverables in the form of projects and programming exercises. Students should use the course as a means to discover a potential major (application development) while gathering deep understanding about the security and infrastructure demands of employee and public-facing web applications. Additional discussion explores the corporate strategic impact of web development, and organizational/cultural change.

3. Learning Objectives

Students successfully completing this course should have acquired the ability to:
  • Understand different types of web applications and how they work.
  • Analyze and translate user needs and requirements into a software architectural model.
  • Create wireframes and prototypes of user-centered and SEO-friendly web sites
  • Create structure and content for web sites using standards-compliant HTML and HTML 5.

- Create styles for web sites using Cascading Style Sheets (CSS).
- Interact with users using client-side Javascript.
- Understand how ASP.NET works to enable dynamic contents and web applications.
- Control the flow of scripts, use conditional statements, loops, strings and arrays, write reusable modules in Visual Basic.
- Collect, validate and process information entered by users via web forms.
- Build dynamic web site that connect to a text file and database, insert, present and modify data in sophisticated ways.
- Understand the basics of web analytics and optimization, including the role of content, link building, social media marketing, usability, accessibility, other search engine and conversion optimization tactics.

4. Prerequisites

A minimum grade of C (2.0) in CIS 304, 305

5. Textbook and Software

Recommended Textbooks:

Responsive Design text at our bookstore

Required Web Development Software:

For client-side development: A text editor. Options include: Notepad (plain mode, default on all Windows), TextEdit (plain mode, default on all Mac) notepad ++ (freeware), and Dreamweaver (student discount available at bookstore). Eclipse is also excellent, especially with the JSDT

For server-side development: Microsoft Visual Studio 2010 Premium (free and available on CIS dept website -> MSDNAA Download) with built-in MS SQL server Compact. Alternatively you can install Visual Web Developer 2010 Express

6. Exams, Projects and Assignments

Exams: No formal sit-down exams. One project is due at term's end, by close of final exam.

Quizzes. Each week one quiz is given, worth 10%. In cases where the student does not score above 50% on the quiz, the instructor will inspect their micro project, posted on their blog. Seven weekly micro-projects are due, with strict deadlines each Thursday at **midnight**.

Make-up policy: There will be no make-up exams except for serious and compelling reasons that are substantiated with formal documents. For example, medical cases have to be substantiated with valid doctor or hospital note stating that the student is too ill to attend the exam.

Late assignments or projects: Late assignments or projects submitted within one week after due date and time will be penalized 50%. Late assignments or projects submitted more than one week late will not be accepted. If the student submits an assignment or project late, the student is responsible to send an email to notify the instructor upon submission; otherwise the assignment or project will not be graded.

Tutoring: For free tutoring on campus, contact the Learning Center in the library.

7. Grading

Grade	Percentage
A	93.00-100.00
A-	90.00-92.99
B+	87.00-89.99
B	83.00-86.99
B-	80.00-82.99
C+	77.00-79.99
C	73.00-76.99
C-	70.00-72.99
D+	67.00-69.99
D	63.00-66.99
D-	60.00-62.99
F	0-59.99

For a detailed breakdown of assignments, weight and their details, please see section 11, 'Tentative Course Schedule and Graded Items.'

8. Class Communication [IMPORTANT]

E-mail: All emails must be sent to the instructor with a Cal Poly email account, must be signed with the student's first and last name, and must have "CIS 311" in the subject line, or it may not be read or responded. Please consult the syllabus before sending emails. **Messages sent through Blackboard will not be read.**

Assignment submission
All assignments are to be hosted on Amazon Web Services (AWS). A youthe video is to be produced, detailing your work, lasting no more than one minute, with code displayed at all times during the video. A link to the video is to be posted to a blog (blogger, Tumblr, wordpress, other).

Create your blog during week one of the course, then submit the link to the home page of your blog to the instructor, using this link:
  • https://goo.gl/forms/Mjz59CZbJmUgdaKx2
  • SUBMIT THIS ONCE

To review
1. post your micro project on AWS
2. record and post your mp video on youtube

3. post a link to your video to your blog
4. repeat, weekly for each micro project, due at the schedule below

Github: Course materials including announcements, lecture presentations, assignments, projects documents, classroom exercises and solutions, and grades will be posted on Github. All graded assignments and projects will be visible in Blackboard Gradebook. Our github will be:

• https://github.com/stefanbund/311

Refer to the Github page above for all homework, learning materials, lesson plans and other details critical for your success in 311.

Subject to Change: This syllabus and class schedule are subject to change. If the student is absent from class, it is the student's responsibility to check on announcements made and make up the work while absent.

9. Course Policies

Classroom environment: The classroom is a special environment in which students and faculty come together to promote learning and growth. It is essential to this learning environment that respect for the rights of others seeking to learn, respect for the professionalism of the instructor, and the general goals of academic freedom are maintained. Student conduct which disrupts the learning process shall not be tolerated and may lead to disciplinary action and/or removal from class.

Using laptops, cellphones and other electronic devices:
• Using laptops during the class for anything other than this class, personal conversations, talking or texting on cell phones or other distracting behavior are prohibited.
• As a courtesy to all, please turn off all cell phones and pagers during class. If the student needs to be reached for family medical or significant work-related issues, the student must present evidence to the instructor before the class starts.
• Absolutely no cellphones or other electronic devices may be used during an exam or quiz.
Attendance:
• Arrive on time.
• Always whisper when the instructor is lecturing.
• If the student needs to leave early, the student must let the instructor know before the class starts, and choose a seat that minimizes disruption to the class when leaving.
• If the student has to miss the class, the student must send an email to let instructor know before class and explain the reason.
• If the student is sick and contagious, the student should not come to the class and risk getting others sick. Email the instructor before class to inform him of your decision.

Student responsibilities:

Each student is responsible for the successful completion and submission of all assignments and projects.

The instructor will not review your assignments or projects before grading for the entire class to ensure fairness. The instructor will, however, help you understand the expectations and clarify the requirements. Spot assessments will help you to outline questions and receive pre-due date feedback.

The instructor will not debug assignments or projects for individual student. The instructor will, however, help you gain knowledge and skills in analysis and design, problem solving, coding, testing and debugging, and answer specific questions about course topics. Make sure you have spent significant and reasonable amount of time and effort in research and working on your own before asking help. **To help you in this effort, many code examples are posted on our course GitHub.**

Team Work
No team work will take place during 311 this term.

10. University Policies

Students with Disabilities: Upon identifying themselves to the instructor and the university, students with disabilities will receive reasonable accommodation for learning and evaluation. For more information, contact Services to Students with Disabilities at http://dsa.csupomona.edu/drc/.

Academic Integrity: Students should understand or seek clarification about expectations for academic integrity in this course (including no cheating, plagiarism, or inappropriate collaboration); neither give nor receive unauthorized aid on examinations or other course work that is used by the instructor as the basis of grading; take responsibility to monitor academic dishonesty in any form and to report it to the instructor or other appropriate official for action.

Cheating and Plagiarism: Cheating is the actual or attempted practice of fraudulent or deceptive acts for the purpose of improving one's grade or obtaining course credit; such acts also include assisting another student to do so. Typically, such acts occur in relation to examinations. However, it is the intent of this definition that the term 'cheating' not be limited to examination situations only, but that it include any and all actions by a student that are intended to gain an unearned academic advantage by fraudulent or deceptive means. Plagiarism is a specific form of cheating which consists of the misuse of the published and/or unpublished works of others by misrepresenting the material (i.e., their intellectual property) so used as one's own work. Penalties for cheating and plagiarism range from a 0 or F on a particular assignment, through an F for the course, to expulsion from the university.

Computing Resources: At Cal Poly Pomona, computers and communications links to remote resources are recognized as being integral to the education and research experience. Every student is required to have his/her own laptop/computer or have other access to a computer with all the recommended software for this course. Find out more about how to access to the university's information resources from Information Technology Services.

Copyright Policy: Copyright laws and fair use policies protect the rights of those who have produced the material. The copy in this course has been provided for private study, scholarship, or research. Other uses may require permission from the copyright holder. The user of this work is responsible for adhering to copyright law of the U.S. (Title 17, U.S. Code). A full description of

Cal Poly Pomona's copyright policy is included in the University's Intellectual Property policy. The course web site contains material protected by copyrights held by the instructor, other individuals or institutions. Such material is used for educational purposes in accord with copyright law and/or with permission given by the owners of the original material. Students may download one copy of the materials on any single computer for non-commercial, personal, or educational purposes only, provided that (1) do not modify it, (2) use it only for the duration of this course, and (3) include both this notice and any copyright notice originally included with the material. Beyond this use, no material from the course web site may be copied, reproduced, re-published, uploaded, posted, transmitted, or distributed in any way without the permission of the original copyright holder. The instructor assumes no responsibility for individuals who improperly use copyrighted material placed on the web site.

11. Tentative Course Schedule and Graded Items

Term Grading

The Term will contain 7 micro-projects, each worth 10%, and due at the close of each unit in the Lesson Plan. A Final term project will take place worth 20%.

Weekly Quizzes	Unit	Weight (%)	Due
	1	10	4 / 6
	2	10	4 /13
	3	10	4 / 20
	4	10	4 /27
	5	10	5 / 4
	6	10	5 / 11
	7	10	5 / 18
	8	extra credit	5 / 25
Final Project		20	TBA on cpp final exam schedule, due midnight that day

MP 8 is optional, and can be submitted to replace a grade on a prior MP.

Term Final Project

Students may choose from one of the following Tracks for the final project.

1. **Interactive Web Product.** The final project compiles each micro-project into a cohesive interactive web product. Compile a product video on a promotional page of the product's website. ASP.NET or the AWSJSDK is to be used, but not both, due to hosting limitations.

Students should plan the project within the first month of the course, then seek to develop small facets of the project with each mini-project. Since each mini-project awards conceptual development, the final should complete whatever concept was initiated during the term.

Grading rubric, for AWS projects:

A. Product is suitable to compete a Cal Poly entrepreneurship competition. A working prototype should exist with a demonstration account. A video should demonstrate the website in a clear and honest manner, with narration and instruction (not a marketing presentation). Product should possess a goal, be conceived the solve a problem, and lay out the rationale for its design. The product should operate smoothly, and bring the user through a basic business process where the concept is can be discussed by an outside observer. The site should be hosted publicly on Amazon web services simple storage service (S3), and **employ all techniques** taught during the term.
B. Missing one element from an A project
C. Missing two elements from an A project
D. Missing three elements from an A project
E. Missing four or more elements from an A project.

Grading Rubric, for ASP.NET projects:

A. Project completes an ASP.NET project with several key elements. Elements include integration of SQL Server, deployment of a web container, integration of 5 SQL queries, producing HTML markup, integration of a responsive CSS in a uniform fashion on all pages, evidence of input / update and delete on database contents, a basic security mechanism.
B. AProject is missing 10% of the A criteria.
C. Project is missing 20% of the A criteria.
D. Project is missing 30% of the A criteria.
E. Project is missing 40% of the A criteria.

Discussion

For students who aim to major in development at cal poly, the ASP.NET project is recommended, whereas for students in separate specializations, the AWS is recommended. Since the AWS will leverage the term's code production, this is the less time-intensive version, as the ASP project will likely require 15-20 hours of effort at the close of the course.

Micro-Projects

For each unit in the following matrix, a small project is made by the student to practice and implement the technology. Source code will be presented and made available online on our course GitHub. Students will be expected to show competency implementing the code demonstrated for the unit in some working metaphor, or micro-project.

Deliverable. The coding practices taught in the unit should be present on each page in some functional, explainable way. The student should show that they have learned how to use the skills, then apply them in a very simple demonstration. An outside observer should show them moving data then display it in HTML markup, in a very simple business process.

Each student will use a public blogging service to describe their work, then link to their project. The assignment is a blog post, completed by the due date, which links to their functioning micro-project on the web.

Best Practices
For your micro-project, understand that *A* projects are small portfolio pieces, adequate to show employers a wide range of web development skills.

Due dates
Micro-projects must be submitted by end of the next unit. For example, class ends on a Thursday, then the micro-project will be at midnight. Your micro-project must be online and accessible for full credit.

Grading Rubric (for each weekly micro project)
A
Student accomplishes the week's micro project goals, per the table "Weekly micro project goals," on the following page.

B
Student accomplishes 85% of the coding goal

C
Student accomplishes 75% of the coding goal

D
Student accomplishes 65% of the coding goal

E
Student accomplishes 55% of the coding goal or below

Lesson Plan, Lecture Topics and Session Objectives

Months	Class sessions	Lecture Topic	Unit # on github
March	28	**Basic Javascript**. Basic user interface interactivity, HTML elements	1
	30	Responsive project element demonstration and discussion. Navigation, URLs and basic UI for HTML 5, using bootstrap CSS.	
April	4	**Javascript objects**. Manipulating member variables and functions in javascript core language	2
	6	Responsive project element demonstration and discussion. Alerts, labeling, wells and other means of displaying data to users.	

Months	Class sessions	Lecture Topic	Unit # on github
	11	**Handling Data with Javascript Objects**. Javascript Object Notation (JSON), java objects, arrays, array traversal, functions, forEach iteration, Document Object Model (DOM) and javascript, text nodes	3
	13	Responsive project element demonstration and discussion. List elements, tabular outputs, menus with dynamic content.	
	18	**Cloud Security and Querying NoSql Databases**. OAUTH, Amazon Web Services SDK, Identity and Access Management (IAM), DynamoDB and DocumentClient, event-driven javascript, Web Identity Federation, Third Party Authentication Systems and Cloud Computing	4
	20	Responsive project element demonstration and discussion. The HTML div and CSS styles. CSS grids, page layout and row-based container elements in CSS	
	25	**Integrating services into client-side javascript**. HTML Elements integrated with cloud computing APIs	
	27	Responsive project element demonstration and discussion. Continued work in developing means to display interactive output from data, in rows	
May	2	**Basic Get, Put, Update, Delete on NoSql cloud environments**. Transferring cloud JSON into HTML elements	5
	4	Responsive project element demonstration and discussion. Triggering navigational elements via javascript and the window object in the HTML DOM	
	9	**Persisting client-side data with cookies and adding interactivity to interactions on the web. URL parameters in the DOM**	6
	11	Responsive project element demonstration and discussion. Integration of prior weeks' discussion into cookies. Leveraging data transfer via URL parameters	
	16	**HTML FILE protocols and uploading images to the cloud**	7
	18	Responsive project element demonstration and discussion. The responsive image, img tag, and the row of divs. Building interactivity to links which involve data transferred over URL parameters, including images.	
	23	Final Project work time. Get personal consultation, coaching and time to discuss project.	8
	25	Final Project work time. Get personal consultation, coaching and time to discuss project.	

Months	Class sessions	Lecture Topic	Unit # on github
	30	Final Project work time. Get personal consultation, coaching and time to discuss project.	
June	1	Final Project work time. Get personal consultation, coaching and time to discuss project.	
	TBA	final exam, final project submission due date	

Weekly micro project goals

Microproject	Objective	Consult
1	Be sure to try examples 1 through 12 in our unit 1 .pdf tutorial on basic javascript in our github. Create four .html pages, where responsive design is taking place (bootstrap). Use a drop-down navigation menu on each page. Enable each page to contain text inputs and paragraph elements which respond to the user's input.	http://getbootstrap.com/getting-started/#examples-navbars and unit 1 code examples. Be sure to view our youtube, also.
2	**Add one javascript object to each of your new pages**. Each object should have a different name, but their data content may be different. When the visitor enters each page, they should be able to click a button which displays the object's data inside a bootstrap panel. Set the title and panel body using content from your object on each page.	http://getbootstrap.com/examples/theme/#
3	For every page where you implemented a javascript object in unit 2, add a JSON array of data, as we did in our samples. For those pages, implement a row of columns using bootstrap, then add div elements for each row of the data. Your data array in JSON should appear within several columns.	http://getbootstrap.com/examples/grid/
4	**Begin with the pages you created for unit 3**. Implement a query on pages where you handle JSON. Create one table in dynamoDB for each query (3 or 4, likely). Present the results of your queries in your bootstrap columns, as in unit 3. You may also produce more than one query on a page, associated with different buttons.	

5	Begin with the query results you got for each query in MP 4. For each div you create, with query results, add an HTML href, or link element. This element should contain results from your query, to construct a valid URL. Hence, you should create a column in your DynamoDB which contains a string sufficient to create a hyperlink. Then, when this column data loads within the page, a link is constructed, concatenating other URL strings. The link should be underlined. Create the link in a responsive way.	http://getbootstrap.com/examples/theme/#
6	TBA	
7	TBA	
8	TBA	