Syllabus for CIS 311: Interactive Web Development

California State Polytechnic University, Pomona
Spring 2018

Instructor: Stefan Bund, MS
Office: Building 164 Room 3027
E-mail: slbund at cpp.edu (replace at with @)
Phone: 949 433 7487 (no texts accepted, email please)
Office Hours: Tuesday, Thursday 4 - 5:50 pm, office above

This syllabus governs all policy for the course.

Guide to the Syllabus

Class Meetings and Office Hours, Holidays

Term Dates

Classes run each week from March 27th until June 7, Tuesday and Thursdays.
Final exam dates for 311 sections:

1-3 pm class final exam is on	Thursday, June 7 between 1:30 and 1:30 pm
6 - 8 pm class final exam is on	Thursday, June 7 between 6 and 8 pm

Class Meeting Times and Locations

class session , time	meets in room…	between
311 1 pm session	CLA 98 C 4-035	1-3 pm
311 6 pm session	Building 9 room 333	6 -8 pm

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.

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.

Prerequisites

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

Textbook and Software

Optional Textbooks:

Responsive Design text at our bookstore

Required Web Development Software for assignments, one of the following:

- Eclipse with the JSDT installed
- Net Beans or other full featured editor used in CIS 234, capable of making HTML pages

For the final project please acquire via DreamSpark:

- Microsoft Visual Studio 2017 (free and available on CIS dept website -> MSDNAA Download) with built-in MS SQL server.

Exams, Projects and Assignments

This term, students will form into teams to perform Group Projects (GP). A digest of all Group Project assignments, or GPs, are posted here:

- https://github.com/stefanbund/311/blob/master/SPRING%20311%20GP%20PLAN.pdf

GP Submission
Each week a team member will send in the results of your team's GP. The team will go over your submission either during office hours (after class) or during class.

The grade weighting for these important assignments are detailed below.

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. Athletic events are acceptable also, and will require documentation from the team, per our university norms.

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 for sending 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 CIS department in the CBA Administration Building.

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.'

Class Communication and Getting Help

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, especially in the area of microprojects. **Messages sent through Blackboard will not be read.**

Coding Questions
In cases where you have a technical question, please post this to our Github at

- `https://github.com/stefanbund/311/issues`

You may also send the professor an email at `slbund at cpp dot edu`.

How to Win at Emails to Your Professor
Expect that your email is read on a mobile device, to return a reply to you very quickly and build value in your experience. Thus, compose your email carefully:
1. include screenshots of errors in the Javascript console, in your Browser
2. run Inspect Element on the running page, screen shot the code you are running
3. error codes given by your IDE, browser are really helpful!
4. include only a short number of sentences, and try to limit the email to one or two precise questions. This guarantees a quick reply.
5. The closer you ask questions to a deadline, the harder it is to get a fast reply. Schedule your work so you get help efficiently.

General Course Expectations on Help and Technical Assistance
- Your process should include reading the syllabus, reading the assignment, reviewing our learning materials, trying code, collecting results, **then** asking about running code. This will dramatically speed-up your experience. I would prefer not to answer your email with `please read syllabus page 19 subsection 18, paragraph 3, clause 18.`
- Please understand that very general questions are harder to answer, so emails with the subject line `PLEASE HELP` (multiple emoji), or `I AM CLUELESS` (emoji 1 and 2) are harder to manage, and are likely to be delayed.
- Emails with precise subject lines like `DynamoDB Error in AWS`, or `Authentication Failure on Facebook` will be **potentially** answered quickly throughout the day, 7 days a week.

Course materials
Lecture presentations, assignments, projects documents, classroom exercises and solutions, will be posted on Github. All graded assignments and projects will be visible in Blackboard Gradebook. Our github repository is:

- `https://github.com/stefanbund/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.

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. **Whenever you have questions related to your studies**, please send an email to the faculty member, or attend his office hours, posted on page one, above.

The instructor will not debug assignments or projects for an 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. **It is always appropriate to discuss bugs and other complications with the instructor, however, the process of fixing problems rests with each student, exclusively.** Make sure you have spent significant time and effort in researching and working on your own before asking help. **To help you in this effort, many code examples are posted on our course GitHub, and to support your learning, each lecture is posted on our youtube channel. Be sure to watch each youtube instructional item at least once!**

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.

In assignments where code is required, students who allow their work to be copied will receive a zero on the first episode of cheating. Forewith, meetings with the university will follow, to escalate, potentially, to explusion. All code you submit must be coded by you without copying from another source.

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.

Tentative Course Schedule and Graded Items

Weekly GP

The course grade is calculated using the items, below.

GP and final project	Weight (%)
1	5
2	5
3	5
4	5
5	5
6	5
7	5
8	5
ASP.NET Project	**10**
Total points	**50**

Each GP should be submitted via email (slbund at cpp dot edu), each week. Each week, a team leader (or group) will present the prior week's deliverable in-person. Should take 5 minutes per team, extended into office hours.

Term Final Project

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 is to be used. Students will integrate resources from Microsoft in creating their individual project. Resources for the project are located at this address,

- https://docs.microsoft.com/en-us/aspnet/core/tutorials/first-mvc-app/index

This link will also be posted to the course GitHub page. Students should plan to devote 15 hours to the project, minimum. Grading is based on the percentage of the tutorial which you have finished and documented.

How to Submit Your Final Project for Grading

1. **Public blog**. Students will demonstrate their project on their term blog, where their micro projects are posted.
2. **Submission deliverable**.
    1. Students will copy/paste the code they used to produce the site in **a single .pdf document**.
    2. In this document, screen shots of each working UI are to be inserted, to accompany their code. The document should show the working product with the code they used to produce it.
    3. A dropbox will be sent to students on the due date, via CPP email, to assist with their submission.
    4. Only one submission per student may take place, and must be submitted by the closing time of the final exam, as posted on the school's final exam schedule (and on on our GitHub, for convenience).
    5. Multiple documents, unorganized images, or departure from rule 1 results in a zero.
    6. Your submission's filename must contain your full name, and the words, '311 ASP project'.

Grading Rubric, for ASP.NET projects:

A. Project completes an ASP.NET project with several key elements. Elements include
    1. integration of SQL Server
    2. deployment of a web container (IIS) on a personal device (laptop / PC)
    3. integration of SQL queries producing HTML markup
    4. integration of a responsive CSS in a uniform fashion on all pages,
    5. evidence of input / update and delete on database contents
    6. a basic security mechanism.
    7. use of URL parameters
    8. use of the ASP.NET model / view paradigm
B. Project 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.

Grading Rubric (for each weekly group 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

Month	Class sessions	Unit # on github	Lecture Topics
mar	27	1	introduction to clouds: hosting, image uploads. Javascript fundamentals. OAUTH and securing web traffic, building authentication mechanisms via Javascript. APIs for Javascript. Responsive web design explained via Bootstrap CSS. Synchronizing code to the cloud via AWS S3. Hosting explained and demonstrated.
	29		demonstrations continue, answer questions, group work time, individual mentoring and instruction
april	3	2	NoSQL databases in the cloud. Inserting data. Managing restful responses from the cloud in Javascript. Building UI in Javascript. Document Object Model (DOM), Windows, Locations, Documents, Elements. Programming for navigation.
	5		demonstrations continue, answer questions, group work time, individual mentoring and instruction
	10	3	NoSQL getting items. Continuation of restful programming, API use and handling JSON.
	12		demonstrations continue, answer questions, group work time, individual mentoring and instruction
	17	4	NoSQL databases, editing rows. Continuation of rest, JSON, APIs in JS.
	19		demonstrations continue, answer questions, group work time, individual mentoring and instruction
	24	5	URL parameters, chaining functions in javascript. Integrate URL parameters with editing data in the cloud. Involving page navigation with this skillset.
	26		demonstrations continue, answer questions, group work time, individual mentoring and instruction
may	1	6	Query NoSQL databases. Develop skills iterating collections, page DOM, programming html elements without CSS.
	3		demonstrations continue, answer questions, group work time, individual mentoring and instruction
	8	7	Manage rich data sets into rich markup via CSS. Create responsive output from highly available, secure and resilient databases in the cloud. Ensure hosting is not deniable, offers role-based access control (RBAC) via OAUTH.
	10		demonstrations continue, answer questions, group work time, individual mentoring and instruction

Month	Class sessions	Unit # on github	Lecture Topics
	15	8	GP project refinements, catch up, make up week. No new lecture.
	17		demonstrations continue, answer questions, group work time, individual mentoring and instruction
	22	FP	Integrate relational data via ASP.NET, Microsoft frameworks. Integrate responsive CSS into the ASP.NET MVC paradigm.
	24		demonstrations continue, answer questions, group work time, individual mentoring and instruction
	29	FP	Continue to apply ASP.NET toward a project of your design. Lecture is continuation, reiteration.
	31		demonstrations continue, answer questions, group work time, individual mentoring and instruction
june	7		final exam: present your final project