

Syllabus for CIS 3110: Client-side Web Development

**California State Polytechnic University, Pomona
Fall 2018**

Instructor: Stefan Bund, MS
Office: Room 3027, Building 163, 3rd floor
E-mail: sbund at [cpp.edu](mailto:sbund@cpp.edu) (replace at with @)
Course Website: <https://github.com/stefanbund/311>
Phone: 949 433 7487 (no texts accepted, email please)
Office Hours: Tuesday, Thursday 1:50 - 3:50 pm, office above
Sections: This syllabus covers two sections of 3110, 7pm and 8:30 pm starts
This syllabus governs all policy for the course. It is subject to change under certain conditions.

Guide to the Syllabus

Class Meetings and Office Hours, Holidays	2
Course Description	2
Learning Objectives	3
Prerequisites	3
Textbook and Software	3
Exams, Projects and Assignments	4
Grading	4
Class Communication and Getting Help	5
Course Policies	6
Student responsibilities:	6
University Policies	7
Tentative Course Schedule and Graded Items	8
Weekly GP	8
Term Final Project	8
Grading Rubric (for each weekly group project)	9
Lesson Plan	10
Pedagogical Notes	12

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:

7 pm class final exam is on	TBA
8:30 pm class final exam is on	TBA

Class Meeting Times and Locations

class session , time	meets in room...	between
311 7 pm session	CLA 98 C 4-035	7 - 8:15 pm
311 8:30 pm session	CLA 98 C 4-032	8:30 - 9:45 pm

Course Description

From our catalog:

Design and development of interactive and responsive websites using client-side technologies. Application of current HTML standards and multimedia elements for content. Use of modern CSS techniques for presentation style. Integration of JavaScript and JavaScript libraries for rich user experience and interactivity.

Major technologies include HTML 5, CSS 3, Javascript, JQuery and React.js frameworks.

Learning Objectives

Students successfully completing this course should have acquired the ability to:

- Integrate HTML 5, CSS 3, Javascript, JQuery and React into cohesive web application front end user experiences.

Prerequisites

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

Textbook and Software

Textbooks will be free, and available online without cost. Multiple sources will be used, including reference material on the Mozilla Developer Network, at <https://developer.mozilla.org/en-US/>

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

- Atom

- Notepad++
- Eclipse, with the JSDT installed
- NetBeans, with React and JQuery libraries available

Any other IDE may be used, so long as the environment compiles React.js, JQuery, Javascript and has syntax correction for open web technologies such as HTML 5.

Exams, Projects and Assignments

Teams will be formed to create a collaborative environment. Projects are assigned for the teams to complete. There are individual deliverables for each team assignment, to be completed by each team member, individually.

- **8 team deliverables are due this term.** There is no late nor makeup policy for each. Each assignment will be required to be complete, in order to do the next assignment. Hence, teams will find that completing the team deliverable will be convenient every two weeks. Teams will show the Instructor the deliverable during class work time, or during office hours. Grading is to take place with the group present, or a group envoy.
- **1 Final Exam will take place**, where teams will present a group project. The final project will capitalize on the prior 8 team assignments.

Make-up policy

Teams may submit their assignments at any time during the term.

Late assignments or projects

Teams may submit their assignments at any time during the term. Groups may take longer to complete elements which require more time. However, each team must respect the time limits implicit in each assignment, and make plans to submit work periodically. Missing due dates will ultimately make subsequent assignments impossible to do by the close of the term.

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 the section below, '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 to in a convenient time frame. Please consult the syllabus and our github before sending emails, especially in the area of assignments. **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 `Javascript Type Error, or Syntax Error line 11` will be **potentially** answered quickly throughout the day, 5 days a week.
- Emails will not be replied to during Sunday, and will likely not be read on Saturday, due to family commitments (small children, etc)

Course materials

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

- <https://github.com/stefanbund/311>

Video demonstrations will be posted at this github address.

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, even when team work is expected.

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 feedback, pre-due date. **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.

It is the responsibility of the Lecturer to present the material, and make learning materials accessible. The student must then ask questions, and take responsibility for their individual learning. The Instructor is not a mind reader, and must be told what things need explanation in more individual ways. Understand that if you do not make effort to ask questions, and help yourself learn, all deadlines apply, whether you understand the material or not.

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 video instructional item at least once!**

In cases where students have not attended class or have not spent time on the assignments, the Lecturer will refer them to the video-recorded lectures. The student will not receive a personal, private rendition of the lecture. They will be told to watch the videos, and refer to reference materials. Private lessons and tutorials will not take place, in cases where students are not attending, participating with the class.

In assignments where coding takes place, students will receive demonstration code, but will be expected to render code in a unique way, without copying from others. In cases where teams supply deliverable work, each team member may supply guidance, advice or discussion, but under no circumstances may they supply code. Groups who plagiarize across its membership will not receive credit for a deliverable. Source code will be inspected at any time.

In cases where teams do not function well together, the following protocol will take place.

1. a team member must submit an email documenting a lack of team work or any issues related to team productivity.
2. the Lecturer will ask for a manifest, or list of all team work, and to whom work is delegated.
3. the assignment will be graded based on the manifest in (2), or in some manner which separates the contribution of each team member into individual scores.

In all cases where no complaint is present, the group will be graded ensemble, with same scores applying to all.

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 expulsion. 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.

Deliverable	Weight (%)
1	10
2	10
3	10
4	10
5	10
6	10
7	10
8	10
Final Team Presentation	20
Total points	100

Each team deliverable will be accepted during office or class times, and graded in the presence of the student team members. In cases where the assignment is not complete, the team may opt to continue work on it, but will be graded provisionally at the time of submission.

Term Final Project

Throughout the term each team will aggregately build up a web experience. At the close of the term the team should demonstrate a thematic engagement, with a specific set of features, in line with subject matter taught. Grading rubric for the project will proceed as follows.

1. a theme and goal is chosen, and a goal is achieved. The project may cover any domain of mainstream practice related to websites. The theme and general mode of expression must be appropriate to the University community. For example, the team must be comfortable presenting their concept to a diverse audience. In all cases, the technique used to produce the site will supercede the messages, content and overall goals of the site.
2. the site must be deployed to a public web container, and hosted at a public URL and it must be viewable by the general public.

3. all topics and technical skills taught in the course must be reflected in the site. Please refer to the Lesson Plan, below for a checklist of all topics which must be exhibited in the project.
4. Work on the site must be equally shared between group members. In cases where an imbalance of work is performed, grading will follow proportionally.
5. The project should identify a problem and attempt to solve it by providing a web experience. Some goals may contain providing web-based tools to supply social interactions, inter-organizational communications, the posting or pooling of knowledge. Other goals may be presented and discussed, but the team must receive approval by the faculty, in order to proceed, and must elevate their goals to be inline with the class. In certain cases, some students may propose personal projects, which require the use of skills taught in the class. In these cases, the faculty member will guide the project so that it becomes feasible within timelines.
6. The team delivers their project for Instructor review at the time of the final exam.

Grading Rubric for Final Projects

A projects will complete each item above.

B projects will complete between 80 to 89% of the objectives.

C projects will complete between 70 to 79% of the objectives.

D projects will complete between 60 to 69% of the objectives.

E/F projects will complete between 50 to 59% of the objectives, or less.

Grading Rubric (for each weekly group project)

A: Student accomplishes the week's project goals, per the assignment posted under phases 1 through 8 in our github repository.

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

CS 3110 Lesson Plan and Deliverable Due Dates

week	domain	topics	project, phase # due	deliverable weight (%)	dates
1	HTML	intro DOM, head, html, body, h, p tags, text inputs. Use of head in CSS, AWS hosting, URLs in practice/theory.			Aug 23
	CSS	2 simple css classes, called in html and manipulated via js. Selectors			
	JS	getElementById(), createElement(), appendChild, setAttribute to CSS classes. DOM manipulation. Variables, arithmetic, if-else. Functions in javascript, scope. onClick and other event handles. Adding elements to the DOM via createElement.			
2		Team based trouble shooting and lab environment, with focus on deliverable due	1	10	Aug 28, 30
3	HTML	Lay out a more developed page, using CSS classes throughout.			Sept 4, 6
	CSS	Parameters and tags, more of. Images, widths. Develop discussion of common elements in CSS classes, available in HTML 5.			
	JS / JQ	Create links, more if-else. Grab images from the web, using URLs. For-loops, generating a list of images using arrays and iteration strategies in javascript. Replacing DOM manipulations with JQuery. Introducing UI elements via JQueryUI.			
4		Team based trouble shooting and lab environment, with focus on deliverable due	2	10	Sept 11, 13
5	HTML	Divs, page layout, Div box model. Information architecture and introduction to web design.			Sept 18, 20
	CSS	Configuration of Div tags / box model. Grids in CSS for page layout. Prelude to Bootstrap.			
	JS /JQ	Accessing data in cookies. JQuery intensive, adjusting all parts of the page. URL Parameters and programmatic navigation.			
6		Team based trouble shooting and lab environment, with focus on deliverable due	3	10	Sept 25, 27

7	HTML	Introduction to responsive design, making up for inherent shortcomings in div configurations.			10 /2, 10 /4
	CSS	Bootstrap and responsive, grid-based systems.			
	JS / JQ	Continue using arrays and for-loop iteration. Using JQuery to add elements to the page during iteration.			
8		Team based trouble shooting and lab environment, with focus on deliverable due	4	10	10 /9, 10/11
9	REACT	App.js and app layout. Navigation via URL and window.location (not routing), integrating multiple components. Applying object oriented practices in React.js via main.js and components. Adding simple UIs via JQuery UI, in React pages.			10 /16, 10 / 18
10		Team based trouble shooting and lab environment, with focus on deliverable due	5	10	10/ 23, 10 / 25
11	REACT	Stateful components, this and super(). Properties (props). Compositions with multiple components. Data types in React. Exchanging data between UI and Component. Creating JQuery opportunities to adjust DOM via React. Adding CSS to React components.			10/30 , 11/1
12		Team based trouble shooting and lab environment, with focus on deliverable due	6	10	11 / 6, 11/8
13	REACT	Event-driven UI in React. Button, inputs, forced updates to components. DOM queries, component lifecycle. Binding UI input to components, state and property manipulation via UI. Functions and button events. Marshalling data and components during array iteration. Tabular HTML and data.			11/13, 11/15
14		Team based trouble shooting and lab environment, with focus on deliverable due	7	10	11/20, Thanksgiving holiday
15	REACT	Project meetings. Time to work out group issues, strategy work with Lecturer.			11/27, 11/29
16		Team based trouble shooting and lab environment, with focus on deliverable due	8	10	12/4, 12/6
17	EXAM	Submit your team project in a presentation setting, with Lecturer as audience. Provide a working website with static data features.		20	12/10 - 12/16 TBD

Pedagogical Notes

The Fall 2018 and Spring semesters will contain several teaching experiments. These will gauge successful teaching methods and will address these questions:

- When using video to capture lectures, what kinds of lecture formats create successful video content? Video is understandably kind to students who wish to replay lectures, but given that students must replay videos for technical demonstrations, how can the instructor format live lectures with respect to effective video production?
- Fall term will present two lecture types, 1) with fidelity to content, providing a doctrinal perspective. A second lecture will follow, where a call and response technique will guide the class in a repetitive, learner-focused approach. Both videos will be available, but with different purposes. The first, doctrinal video will assist students who have grasped the call/response lecture, but need to implement ideas in more professional ways. The second video will serve to review or refresh students.

Larger learning questions include 1) does the prevalence of video get used by students? 2) how do high achieving students leverage the combination of source code and video content to self-manage and improve self-reliant learning, when tutors and teachers are not present?