Web Development CS 290-400 Syllabus Winter 2021

Course Description

In this course, we will mostly talk about the frontend web development skills and a few backend techniques. We will be covering topics like HTML, CSS, JavaScript, Node.js and MySQL. We will spend most of the time talking about JavaScript.

Prerequisites: CS 162 or CS 165

Credits: 4

Terms Offered: Every Term

Instructors: Luyao Zhang

Email: zhangluy@oregonstate.edu

Communication Policy

- **Always use your OSU email to contact us. The Canvas mailbox does not work very well.**
- **When you send us an email, you must include the tag "[CS 290]" in your email subject.**

Please use the email addresses above to contact the instructor and TAs. You should expect a response to emails within 48 hours. Emails sent over the weekend sometimes take longer to respond to.

Post all course-related questions on the Piazza board so the whole class may benefit from our conversation. Please sign up yourself on Piazza. You can use the course Team group to ask questions as well.

For grading questions and regrading request, please contact the TA who graded your assignment by posting privately on Piazza. You can include key words like "regrading Assignment # @ TA's name" in the subject to highlight it. Do not post re-grading request on Team. You should expect your grade to be posted after one week of the due time. If you submit the assignment late, it may take longer for your grade to be released.

The instructional team will be using the class mailing list extensively to communicate with you. We will also frequently post information on Piazza. It is your responsibility to keep up-to-date with these communiques and they are considered part of the required learning material.

Course Topics

This course is divided into three main sections which are largely addressed sequentially:

Layout and Styling

The first portion of the class focus on the static layout and styling of a web page (**HTML/CSS**). For some, this may be a review if you have done web publishing in the past. There is quite a bit of information to take in here but the problems to solve are not that intricate.
Client Side Interaction

The second portion of the class focuses on **JavaScript** and making interactive web pages in the browser. Things like forms that will display an error message if a password is too short or creating a drop down menus are things that will be covered in this portion of the class.

Server Side Interaction

In this portion of the class, we look at using a very simple **database** to store data between website visits. The technologies we will be using this term are **Node.JS** and **MySQL**. In addition, we look at how we can track a user and data from page to page which is a critical first step in designing more complex systems like shopping carts for an online shopping website.

Measurable Student Learning Outcomes:

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

1. **Describe** the architectural elements of effective web applications, as well as key threats to relevant architectural quality attributes.
2. **Demonstrate** implementation of extensive custom functionality across multiple tiers.
3. **Evaluate** which architectural strategies to apply, and in what manner, to address a given set of quality requirements, with particular emphasis on… 1) Scalability, 2) Usability, 3) Security.

Course Schedule

You can see the assignment due dates on Canvas directly.

Topics by Weeks

Week	Topic(s)	Due
1	Web Overview and Intro to Tools	HW1 Node.js and Git Course policies quiz
2	HyperText Markup Language and Cascading Style Sheets	HW2 HTML/CSS
3	Introduction to JavaScript	Activity1 JS Environment Activity2 JS Functions Activity3 JS Objects
4	JS Functions and Objects	HW3 Higher-Order Functions and Objects Activity4 Fixing Closure Loop
5	JavaScript and the DOM	HW4 DOM and Events
6	JavaScript and HTTP (forms)	Activity5 Ajax Interactions Project Proposal

7	Intro to Node.js	HW5 GET and POST checker
8	Sessions and HTTP	Activity6 Practice Project
9	Database Interaction	
10	Wrap up	HW6 Database interactions and UI Sample quiz

Textbooks

Required: Eloquent JavaScript Luhttp://eloquentjavascript.net/

Optional: Jon Duckett's "HTML and CSS design and build websites", ISBN-13: 978-1118008188, ISBN-10: 1118008189

Assignments

This course has three types of assignments:

- **HW Assignment:** You will have six assignments this term, each is given one week to finish. You will write a website or make some functions working using required techniques to meet the constraints. The assignments will be graded on how well they meet the requirements. Please refer to the rubric for expectations.
- **Activity:** The activities are much easier assignments, often graded on effort and not correctness. You need to show that you did the required practice and tried your best to make the code working. Please refer to the rubric for expectations.
- **Project:** This is a research-based project. You need to build a website based on the knowledge you learn in this course to meet some requirements. Please refer to the rubric for expectations.
- **Quiz**: After you go over all the information in the start here module, you need to take the course syllabus quiz and get the full point to unlock the week 1 module

PLEASE NOTE: If you submit the incorrect files/assignment you will NOT be given a chance to resubmit and will receive a 0 for the assignment. It is important that you download your submission after it uploads to verify that everything is as you expected. There will be no exceptions.

Exam

This course has one proctored exam -- the final exam. You can find out more about proctoring at the central Ecampus page on [tests and proctoring.](#)

The final exam window will run from the **Sunday** at the start of finals week through **Thursday** of finals week. If you are unable to take the exam in that window, you must make arrangements **prior to the end of the 2nd week** of classes. Beyond this deadline, only emergency situations will be considered for alternate testing times.

This course will use an automated online proctoring system called Proctorio, where your exam session is recorded for instructor review. You will **not** need to schedule proctoring appointments, and there is **no cost to you** to use Proctorio.

Please note that a functioning webcam and microphone are required for using Proctorio. If you do not have these, you will need to locate and submit an alternative proctor through the [exams and proctoring form](#) and pay for any associated proctoring fees.

Grading Policy

Grade letter	Percentage floor
A	93
A-	90
B+	87
B	83
B-	80
C+	77
C	73
C-	70
D+	67
D	63
D-	60
F	0

Grade Weighting

•Activities/Exercise - 25%
•Homework Assignments - 50%
•Project - 10%
•Final Exam - 15%

Accommodations

"Accommodations are collaborative efforts between students, faculty and Disability Access Services (DAS). Students with accommodations approved by DAS are responsible for contacting the faculty member in charge of the course prior to or during the first week of the term to discuss accommodations. Students who believe they are eligible for accommodations but who have not yet obtained approval through DAS should contact DAS immediately at 541-737-4098."

Students with documented disabilities who may need accommodations, who have any emergency medical information the instructor should be aware of, or who need special arrangements in the event of an evacuation, should make an appointment with the instructor **as early as possible**, and no later than the first week of the term. Class materials will be made available in an accessible format upon request.

If you have a really tough situation that might affect your progress a lot (illness, job duties, family emergency...), you should contact the instructor immediately. **Don't wait until the due date or even past the due date to explain your personal situations and ask for extensions**. If you are not sure whether to ask for it, better do it.

Late Policy

Requests for extensions are considered on a case by case basis. Non-emergency requests must be submitted **via email at least 72 hours before the due time**. (Not having enough time to get the assignment done does not, by itself constitute an emergency, sorry!). **If you don't know if you will need an extension but might, you should ask for one.**

Time elapsed past due date	Penalty Applied (if no extension is granted)
T < 24 hours	-10%
T < 48 hours	-20%
T < 1 week	-30%

Bonus Day

You have **3 bonus days** that you can apply to any activities or assignments (except the final assignment). You can use it all at once for one assignment (if you are late for 3 days) or split it and use one day each for three assignments (no "half" day). If multiple assignments are due the same day, then you will need to use a bonus day for EACH assignment due.

How to apply the bonus day: leave a comment on Canvas under that assignment submission, saying that you would like to apply x bonus days for this late submission, and you have y bonus days left after that. When TAs are applying the late penalty, they will look at your submission time and your comments. If you do not leave a comment there, TAs will directly apply the late penalty.

Extra Credit

There is extra credit in some of the assignments. Try to get them when possible. At the end of the term, I will give extra credit (0.5 to 2 maximum points) to those who are active on Piazza answering questions and sharing notes with others.

Code Sharing

You will not get in trouble for sharing code with your classmates in order to solve problems. The communication guide actually mandates that you share **portions of your code** if you want to ask a good question. If you are worried that you are posting too much code, mark it private and ask the instructor to review it. Note that this is a more permissive policy than the standard policy for the program.

You will get a great deal of trouble if you copy code without citing it. See the policy on plagiarism. Code from lectures is not your own, code from Stack Overflow is not your own, code from the Mozilla documentation is not your own. If it is not your code, you must cite it. If you cite it, you must provide documentation in very great detail of what it is doing so that I know you understand the code you are using.

Academic Misconduct

The Code of Student Conduct prohibits Academic Misconduct and defines it as:

Any action that misrepresents a student or group's work, knowledge, or achievement, provides a potential or actual inequitable advantage, or compromises the integrity of the educational process.

To support understanding of what can be included in this definition, the Code further classifies and describes examples of Academic Misconduct, including cheating, plagiarism, assisting and others. See the Code of Student Conduct for details.

You are expected to do your own work and demonstrate academic integrity in every aspect of this course. Familiarize yourself with the standards set forth in the OSU Code of Student Conduct Section 4.2. You must only access sources and resources authorized by the instructor. You may not show your work to any other current or future students without the instructor's authorization. Violations of these expectations or the Code of Student Conduct will be reported to the Office of Student Conduct and Community Standards. If there is any question about whether an act constitutes academic misconduct, it is your responsibility to seek clarification and approval from the instructor prior to acting.

Code Style

In this course, we follow the Google style guide for the HTML/CSS/JS code:

https://google.github.io/styleguide/htmlcssguide.html (Links to an external site.)

https://google.github.io/styleguide/jsguide.html