CS 290: Web Development

Spring 2022

http://web.engr.oregonstate.edu/~hessro/teaching/cs290-sp22

How to design and implement a multi-tier application using web technologies: Creation of extensive custom client- and server-side code, consistent with achieving a high-quality software architecture.

Instructor	Rob Hess (hessro@oregonstate.edu)
TAs	Please refer to the course website linked above
Office Hours	Rob: Please refer to the course website linked above TAs: Please refer to the course website linked above

Lectures	**MW 10:00–11:50am (US/Pacific), LINC 128**
	All lectures will be held in-person in the room listed above. Lectures will also be broadcast via Zoom (details below), in case you can't make it to the in-person lecture for any reason.
	
	Note that most lecture meetings will consist of approximately 80 minutes of lecture (though many will be longer than that). For these lectures, the remaining 30 minutes will be used either for open lab time during which I will be available for course-related questions or, sometimes, to dive more deeply into a particular topic from lecture in a more informal, less structured way.
	
	Join Zoom Meeting
	https://oregonstate.zoom.us/j/99481044210?pwd=OFZaY1Bsak9qTXpSdkYyL3Jqbk16dz09
	
	Password: 483702
	
	Phone Dial-In Information
	    +1 971 247 1195 US (Portland)
	    +1 253 215 8782 US (Tacoma)
	    +1 301 715 8592 US (Washington DC)
	
	Meeting ID: 994 8104 4210

Lecture Info and Guidelines	The primary mode of delivery for all lectures in this course will be in-person. However, every lecture will also be broadcast remotely via Zoom, so those who, for any reason, can't make it to the in-person

lecture can still attend lectures. Please see below for additional information and guidelines about lectures for this course.

- **Lectures will be in-person-optimized –** All lectures will be formatted to optimize the in-person lecture experience, so you will have the best lecture experience if you attend in person. For example, it may or may not be possible to easily field questions from Zoom attendees, and precedence will be given to in-person questions. Thus, I strongly encourage you to attend the in-person lecture unless you have a compelling reason not to.
- **Connecting to remote lectures –** Connection information is listed above for remote lectures on Zoom. Remote lectures will be restricted to require attendees to log into Zoom using their OSU account. When you initially connect to the lecture, you will be placed into a "waiting room" until the lecture begins.
- **Lecture content –** All lectures will consist mainly of live coding and whiteboarding. Lecture content will be outlined in a set of lecture notes posted on the course website linked above.
- **Lecture recordings –** The remote version of each lecture will be recorded and posted on Canvas for later offline viewing. Code created during lectures along with other lecture content will be posted on the course website linked above.

Prerequisites   CS 162

Course Content
- Designing and implementing static web pages using HTML and CSS
- Client-side scripting and interactive web pages
- Serving static content using a custom Node.js-based server
- Serving dynamically generated content using server-side routing and templating
- Storing site data in a back end data store and accessing it from the client via an API

Student Learning Outcomes   At the completion of this course, students will be able to:
- **Describe** the architectural elements of effective web applications, as well as key threats to relevant architectural quality attributes
- **Demonstrate** implementation of extensive custom functionality across multiple tiers
- **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

Learning Resources   There is no textbook for this course. There will be required articles and other material from the internet posted on the course website as the course progresses.

Evaluation of Student Performance

Final grades will be comprised of the following weighted components:
- 50% – Programming assignments
- 10% – Code blog
  - For each programming assignment, you will write and publish a post in a code blog that you maintain. You'll use the skills you learn in this class to develop the code blog yourself (i.e. you'll write the HTML, CSS, etc. for it instead of using a service like Tumblr, Wordpress, etc.). Each code blog entry will be graded on a pass/fail basis (i.e. you'll either get 1 point out of 1 or 0 points out of 1).
- 40% – Final project
  - In lieu of a final exam, you'll demonstrate your mastery of the skills you've learned by the end of this course by working with a small team to develop a complete web application that serves dynamic, interactive content based on data stored in a database on the back end.

Grades will be assigned using the [standard ranges](#), after rounding. For example:
- ≥ 93% = A
- 90-92% = A-
- 87-89% = B+
- 83-86% = B
- 80-82% = B-
- etc.

Final grades will not be curved.

REMINDER: A passing grade for core classes in CS is a C or above. A C-, 72 or below, is not a passing grade for CS majors.

Grading Demos for Programming Assignments

In this course, programming assignments will be graded by giving a live demonstration of your work to a TA. Each grading demo will be about 10 minutes and will involve compiling and/or running your submitted code for the TA to show them how it works. You will also walk through your source code to explain to the TA how key pieces of functionality are implemented. At the end of the demo, your TA will assign you a grade for the assignment based on a grading rubric.

All grading demos this term will be in person by default. However, the TAs will be able to perform remote grading demos via Zoom **by request only**, on an as-needed basis. If you need to request a remote grading demo, please contact the TA at least 24 hours in advance.

It is your responsibility to sign up for a grading demo for each assignment. Links to sign up for grading demos will be posted on the "TAs" page of the course website.

Importantly, grading demos are the only way to have your assignments graded in this course. Grading demos must be completed by the demo deadline listed for the corresponding assignment. If you fail to complete a grading demo for an assignment before the listed deadline, you will receive a grade of 0 for that assignment. In addition, if you fail to show up for a grading demo appointment for which you signed up, you may, at the TA's discretion, receive a 10% penalty on the assignment.

Other Course Policies

- **Late work.** Programming assignments submitted after the due date may not be accepted without documentation of a valid medical or family emergency and will receive a grade of 0.
- **Contesting grades.** If you have a problem with a programming assignment grade, you must contact your TA via email within one week of receiving your grade. After one week, you may not dispute a programming assignment grade.

Course Communication

In this course, we will use Campuswire as our main platform for communication. Campuswire includes a Q & A forum as well as live chat rooms. If you have a question about the course, please post it in one of these places. I'd like to keep all Q&A in one place for this course, so it's easy for everyone to find and access, so please plan on using Campuswire instead of another platform that not everyone knows about (e.g. please use the chat rooms on Campuswire instead of setting up a separate Discord server for the course).

You will receive an email at your @oregonstate.edu email address with information about how to join our Campuswire space for this course. You can also find a Campuswire link for this course on the course website linked above and in Canvas.

Please note that, while you are free to send me (Hess) regular email, I typically check my email only once a day (or less), so please be patient in looking for my response to your email. It may be helpful to set your expectations by imagining that your email and my response are both being delivered via postal mail. ✍🏼📬

Importantly, **do not send me an email if you need a timely response to something**, since I may not see your email for a while. Instead, speak to me in person, either after lecture or in office hours, or slip a brief note (with contact info!) under my office door.

Classroom Community

In this course, we will strive to create an affirming climate for all students, including underrepresented and marginalized individuals and groups. We welcome differences including (but not limited to) age, color, ethnicity, national origin, gender, physical or mental ability, religion, socioeconomic background, veteran status, sexual orientation,

and parent status. For more about the community we seek to establish in this course, please see this page:

http://web.engr.oregonstate.edu/~hessro/teaching/community

About In-Person Lectures and COVID-19

Though restrictions are starting to be relaxed, COVID-19 is still a thing. None of us wants to become infected with COVID-19, and we all want things to continue becoming more and more normal.

With this in mind, please do not attend in-person lectures or other in-person course activities if you are experiencing any symptoms of COVID-19. As described above, remote and recorded versions of every lecture will be available if you need to miss the in-person lecture for any reason. Detailed lecture notes and other relevant materials will also be posted on the course website linked above. Please use all of these resources to keep up with the lecture material if you need to miss any of our in-person lectures this term. Please consult OSU's quarantine and isolation guidelines to help determine your next steps if you become ill.

Expectations for Student Conduct

In this course, you are encouraged to collaborate with your fellow classmates to discuss concepts and high-level approaches to programming assignments. However, you are expected to do your own programming work and may not work with other students to write code for programming assignments. You may never copy code written by another person, whether that person is a student in the course or not, though using small code snippets (e.g. from Stack Overflow) is generally OK, as long as you cite the source of the snippet (e.g. by including a link in a code comment).

At the instructor's discretion, any student whose work is deemed to be too similar to another person's work will receive a zero for the assignment in question, and the offense will be reported as academic dishonesty to the Office of Student Conduct. In the case of shared work, both the student copying the work and the student sharing the work will both receive a zero and be reported for academic dishonesty.

A second violation of these rules will result in the student receiving an F for the course and a second report of academic dishonesty to the Office of Student Conduct.

See this page for more details on OSU's Student Conduct Code: https://beav.es/codeofconduct.

Reach Out for Success

University students encounter setbacks from time to time. If you encounter difficulties and need assistance, it's important to reach out. Consider discussing the situation with an instructor or academic

advisor. Learn about resources that assist with wellness and academic success at oregonstate.edu/ReachOut. If you are in immediate crisis, please contact the Crisis Text Line by texting OREGON to 741-741 or call the National Suicide Prevention Lifeline at 1-800-273-TALK (8255).

Statement Regarding Students with Disabilities	Accommodations for students with disabilities are determined and approved by Disability Access Services (DAS). If you, as a student, believe you are eligible for accommodations but have not obtained approval please contact DAS immediately at 541-737-4098 or at http://ds.oregonstate.edu. DAS notifies students and faculty members of approved academic accommodations and coordinates implementation of those accommodations. While not required, students and faculty members are encouraged to discuss details of the implementation of individual accommodations.
Religious Holidays	Oregon State University strives to respect all religious practices. If you have religious holidays that are in conflict with any of the requirements of this class, please see the instructor immediately to discuss alternative arrangements.
Student Bill of Rights	OSU has twelve established student rights. They include due process in all university disciplinary processes, an equal opportunity to learn, and grading in accordance with the course syllabus: https://asosu.oregonstate.edu/advocacy/rights.