Syllabus for CS304, Databases with Web Interfaces

Welcome to what I think will be an exciting class! This page describes the structure, format and requirements of the class.

About this Course¶

If you haven't already, please read about this course

Educational Goals¶

The major concepts and topics are listed in the "About" page, above. The major goals of this course are that students

1. understand those concepts and topics
2. be able to build a web application using those concepts and techniques
3. be able to work in pairs and teams to accomplish a joint goal
4. to build community

The first ones are typical of CS 304 for many years, but the last is new. In this pandemic era, it is easy to feel isolated, cut off from your fellow students. I'm glad to be teaching in person again, instead of zoom. Nevertheless, one of my goals is to fight that feeling of isolation: to try to build friendships and relationships. Please help me with this.

Instructor¶

My name is Scott Anderson. Feel free to call me Scott. My office is in E116 and my office phone is extension 3249, but the best way to reach me is by email at `scott.anderson@wellesley.edu`. At this point, my office hours are currently as follows. All times are ET, Eastern Time.

- MTRF 3:45-4:30pm (except some Tuesdays)
- Wednesdays 3:30-6pm

These OH will be held *either* on Zoom or in my office, E116: your choice. I will be in my office. You are *welcome* to come there, or to meet me on Zoom. Again, it's your choice.

I occasionally will have to cancel OH, particularly on Tuesday afternoons when I have a monthly committee meeting.

If these times don't work for you, *please* let me know. You can always schedule an appointment.

Note that you an also ask questions by email. Sometimes it's easier to talk over zoom, but often an email will suffice.

I love to talk with students one-on-one — it's the best part of my job — so please visit me during office hours. You don't even need to have a question!

Office Hours Zoom Link¶

My office hours will always use my personal zoom link. They are configured to use a waiting room so feel free to stop by even if you don't have an appointment (though making an appointment will minimize your wait):

- https://wellesley.zoom.us/my/wc.scott.anderson
- https://wellesley.zoom.us/j/6913533005

These links are synonymous.

I've set up bookable appointments (15 minute slots) during my office hours. That will allow you to have some idea whether I'm available and be able to reserve some of my time. *However*, you don't need to make an appointment in order to visit. If someone else already has that slot and nothing better is available, you can come then and hope that the other person's question takes less than 15 minutes. I'll be using a waiting room, so you'll just wait in the waiting room until I'm available. Or, contact me for an alternative time.

Scott's Calendar¶

You can book OH appointments via my Google Calendar:

Scott's Calendar

Since my calendar will also have the class sessions on it, with zoom links, you can use that to get to our class sessions.

Tutors¶

We have two excellent tutors this semester who will be holding café hours in E101:

- Arushi Bhandari (abhardar), Tuesdays 8pm-10pm
- Sara Clark (sclark4), at TBD

Prerequisites¶

CS 304 has *CS 230 Data Structures* as a prerequisite. Students with equivalent programming ability may also be admitted. If you do not have this prerequisite, contact me ASAP. You should probably drop the class.

Class is In Person¶

Class time will be in person; there is no remote option via zoom. If, for any reason, you're unable to come to class, you can get notes from someone who was there (as usual) or talk to me.

Class will be done in a "flipped" format (see below), so you'll spend much of the time working on assignments with your partner. So you'll want to attend.

Flipped Format¶

In the COVID-19 era, I'm teaching this course in a flipped classroom format. That means we will spend our class time on interaction (discussions, Q&A) and on coding. The coding will be working on homework assignments and project work. It's unlikely that you will complete the assignments solely during class, but hopefully you will make good progress on it. Furthermore, I will be available to help when you get stuck. In some ways, class time becomes office hours. (I'll also hold regular office hours.)

A flipped classroom also means that I will expect you to prepare for class *before* class. That means doing the reading, watching the videos, and taking some quizzes. I have always asked students to prepare for class by doing reading before class, but the balance is much changed now. I will be avoiding lecturing as much as possible, so that we can maximize time to work on assignments.

My concern about a flipped classroom, though, is that if a student comes to class unprepared, they may not be able to get much out of class. If this happens to you, possibly through no fault of your own, you won't be able to contribute to whatever joint work you are doing with your partner on an assignment. You won't be able to ask the pertinent questions that you would ordinarily be able to do. In fact, the best thing for you to spend class time on is probably doing the reading that you weren't able to do. Hopefully, it'll be a one-time event, and you'll be able to catch up for the next class, which is probably the next *day*. So there's not a lot of time.

The format of the course is four days a week instead of the usual two days a week. That means that deadlines for assignments come fast and furious.

Schedule¶

The current schedule is online. My goal is to make sure everyone is comfortable with a topic before going on, so we may spend more time on these topics than I have allocated. If so, topics will be moved later, and the last topics will be omitted. I expect we will not fall more than a couple of days behind, but we'll see.

The schedule lists:

- *readings*, which should be done before that day's class. There will usually be an associated Sakai quiz, which will be mailed out beforehand. Look for that. The readings will mention if there is an associated video.
- activities for that day of class (usually shown for brevity as "do") on the schedule. No need to read that before class, but you're welcome to if you like.
- assignments (both homework assignments and project deadlines).

Please take a look at it and make sure it's clear.

Videos¶

Most of the asynchronous content of this course is presented in webpages like this one. I will also record videos where I have a demo of a procedure or result that really requires it. My current plan is to upload those videos both to the Sakai site for this course and to my YouTube channel as unlisted videos (so you can only find them with the direct link). The links will be posted to the Sakai site in the "Resources" tool. Hopefully, you'll be able to view one or the other of those. If you are not able to, please contact me ASAP.

Policies¶

Please take a couple of minutes to read my policies. My policies cover questions like attendance, extensions, grading standards, collaboration and the like.

Evaluation¶

Student work will consist of homework assignments, and a semester project. They are weighted as follows:

component	weight
homework	35
project	55
quizzes	10

Most assignments will be pair assignments, where I will assign you a partner (randomly) and you'll work collaboratively on it. That means you'll work together on each line of code, not dividing up the work and working independently. That allows you to bounce ideas off each other, build rapport, and have a second pair of eyes to avoid and fix mistakes.

Quizzes¶

We will be using the quiz feature of Sakai to administer online quizzes before most, maybe all, lecture classes. These will be a few (3-5) multiple-choice questions that are based solely on the readings for that lecture. The questions are not intended to be difficult; rather, we want to make your pre-class reading more effective and educational by making you a more engaged reader. These quiz questions will be accumulated over the semester into the equivalent of one test. (For example, if we had four questions per lecture for 25 lectures, each question would be worth 1 point.)

Note that each quiz usually takes students only a few minutes to complete. I allow 30 minutes just so that there's no time pressure. Nonetheless, there is not enough time to start the quiz and *then* start the reading. So, do the reading first, and then do the quiz.

Part of my inspiration for the idea of using quizzes is this NY Times article entitled To Really Learn, Quit Studying and Take a Test

Homework Assignments¶

There will be eight homework assignments. The assignments will usually all have the same weight; but I will give additional weight to those that are more time-consuming. The schedule lists the assignments and due dates as currently planned.

Coding Style¶

The bulk of the grade for each assignment will be for getting the code working, but coding style is still important. Assignments will link to the following page of relevant coding criteria.

For each coding assignment, 70% of the points were for *functionality* (does the code work?) and 30% was allocated to code *quality* (is the code readable, secure and efficient?). This meant that working code would earn a passing score, even if the coding style was awful (in practice, it rarely was).

Gradescope¶

To ensure consistent and timely grading, I've used Gradescope. At the beginning of the term, you'll get an email inviting you to join (if you don't already have an account). When you are done with an assignment, you'll upload a short PDF to Gradescope for that assignment, saying who you worked with and any other pertinent facts. These will be described in the assignment.

If you work with a partner, only one partner uploads a PDF for the two of you, but that person must be sure to associate the partner with the submission. In other words, if Alice and Betty work together, and Alice uploads, Alice will click a few more buttons to associate Betty's gradescope account with the submission. (Gradescope doesn't read the PDF, so it won't see what you wrote there.)

Grading¶

There has been a lot of discussion recently about equity in grading policies and how that may be different from equality. As an extreme example, if I based the grade on a single high-stakes exam at the end, it would treat everyone *equally*, but might not be "fair." A student who doesn't "test well" (maybe has an anxiety condition or something else) or just has a bad day, can end up with a grade that doesn't fairly represent their knowledge of the material.

In some cases, I am willing for an assignment to be fixed and re-submitted. I want to be judicious about this, though, because a policy of unlimited do-overs favors students who have lots of time, and that isn't always equitable. (I recall a student who thanked me for imposing a time-limit on the take-home exam because they felt disadvantaged compared to other CS majors who has a lot more time available.)