SYLLABUS

Internet Software Development
CS 3320 — Spring 2016

Course Overview

From the course catalog:

> A course providing foundations for the construction and design
> of static and dynamic Web pages with database applications.
> This will include server-side and client-side software
> development.

In this class, you will learn what is called *full-stack web development*: the
front-end (displaying and formatting information in the user's web browser
and enabling them to interact with that data) and the back-end (code on
the web server to prepare, handle, and store the data, facilitate interactions
between the users and the broader world).

We'll be doing this with a few key technologies:

- The Python programming language
- The Flask toolkit for building web applications in Python
- HTML5 and CSS3, current technologies for displaying information in
  web browsers
- JavaScript and the JQuery library for making web application
  interactive

There are many different technologies that can be used for web
development, many of the key ideas are similar and a great deal of the
knowledge you gain in this class will transfer to whatever technology you
use down the road.

By the end of this class, it is my hope that you can:

- Format information in structured HTML for display to users
- Style that HTML using CSS to achieve a desired appearance
- Organize content in a web site or application
- Produce web content from data stored on a server, and respond to
  user actions around that data
- Develop interactive browser-based web applications
- Facilitate web-based communication between multiple users
- Develop web applications and content that work on a variety of
  devices

This class will be a lot of work — you'll be learning how to develop web applications by doing it, repeatedly — but hopefully it'll also be a lot of fun.

Course Logistics

Number	CS 3320
Title	Internet Software Development
Credits	3
Prerequisites	CS 2308 with a C or better
Class Meetings	Tuesdays and Thursdays, 11a–12:20p, in Derrick 240
Web site	`https://cs3320.ekstrandom.net`
	TRACS (for grading and assignment submission)

Instructor

Michael Ekstrand

Office	Comal 307F
E-mail	`ekstrand@txstate.edu`
Skype	mdekstrand
Phone	(512) 245-7523
Office Hours	MW 11a–12p
	Tu 1–2p
	Th 1-3p
	Other times by appointment

Course Structure

There are several main components of the course:

- Biweekly class sessions
- Readings (from textbook and online resources)
- Quizzes
- Projects and assignments

Class Sessions

Class meets twice a week, Tuesdays and Thursdays 11:00 AM–12:20 PM in Derrick 240.

Tuesday during the second week of class (January 26), we will be forming work teams. While you should attend class in general, you **must** be present for this class session.

Readings

The textbooks for the class are:

- *HTML & CSS* by Jon Duckett (published by Wiley, ISBN 978-1-118-00818-8)
- *Javascript & JQuery* by Jon Duckett (published by Wiley, ISBN 978-1-118-53164-8)
- *Flask Web Development* by Miguel Grinberg (published by O'Reilly, ISBN 978-1-449-37262-0)

The first two books are available together under ISBN 978-1-118090744-3; this is how I have asked the bookstore to order them.

None of the books are formally required. *Flask Web Development* is a useful resource if you like having a physical book to learn from, or prefer to read two different takes on the same subject; however, the Flask toolkit has excellent online documentation, and all the details you will need can be found there. The Duckett texts will provide much more unique value. While there is good online reference documentation on the details of individual HTML, CSS, and JavaScript elements, comprehensive learning resources that are organized pedagogically and show you how the pieces fit together is harder to come by. *HTML & CSS* and *JavaScript & JQuery* are quite excellent in this regard, and I recommend that you read them.

In addition, we will be making use of a number of online resources, including:

- The Python documentation <https://docs.python.org/3.4/>
- The Flask documentation <http://flask.pocoo.org/docs/0.10/>
- The Jinja2 documentation <http://jinja.pocoo.org/docs/>
- The Mozilla Developer Network <https://developer.mozilla.org/en-US/>, for HTML, JavaScript, and CSS references
- The jQuery documentation <http://api.jquery.com/>
- Version Control by Example <http://ericsink.com/vcbe/html/index.html>

Links to these and additional resources are available on the course web site.

Quizzes

We will have 5 quizzes throughout the course of the semester. Each quiz will be written to take about 30 minutes. Quiz weeks are listed in the schedule; each quiz will be on Thursday of the specified week.

Your lowest quiz score will be dropped.

Projects and Assignments

The bulk of the work in this class will be in a series of team projects. You will work with the same team throughout the course; we will form teams in the second week of class. It's my intention to integrate the project work with the class sessions so that some class time is spent discussing the projects and material with your team.

These projects are supplemented with a small number of individual assignments.

Grading

Overall course grades will be computed as follows:

Component	Weight
Quizzes	30%
Solo Assignments	25%
Group Projects	45%

Course Policies

Announcements

From time to time, I will make announcements regarding the course materials, structure (such as assignment due dates, changes, or corrections), etc., either in lecture or via TRACS announcements. You are responsible for these announcements. Make sure that you check TRACS regularly, or have TRACS announcements delivered to your e-mail (and check that).

Recording and Sharing of Course Materials

You may make audio or video recording of my lectures for your own personal study, and you may share such recordings with your classmates in this offering of the course. You **may not** make such recordings publicly available, or share them with other students not in the course this term, without my prior written consent.

After each lecture, I will make the following available on TRACS or the course web site:

- My lecture notes
- Lecture slides, if I used any
- Example code and data sets

You may download and use these, and retain copies after the course has completed, for your own personal use and study. Please do not share or reproduce them without asking me first, unless they are clearly labeled for redistribution (e.g. example code bearing an open source license).

Late Work

Late work for this class handled on the basis of *late days*. These are intended to accommodate most ordinary needs for extensions or late submissions.

Each student has 3 late days that they can apply to the solo assignments, and each team has three late days that they can apply to their projects. Each late day extends the deadline by 24 hours; late days can be combined on a single submission, so you can turn it in 2 days late by using 2 late days. You do not need approval in advance to use late days, but do need to state in your submission's README file how many days you are using.

With the exception of the late days, no late work will be accepted. You must take quizzes at their scheduled times.

Exceptions to this policy will only be granted in extreme circumstances. Any requests for individual exceptions must be submitted by e-mail so that I have a record of the request and my response.

Cheating and Academic Integrity

As both a programmer and a student, you are expected to do your own work, attribute sources, and respect the legal and moral rights of others with respect to their work; as a student, you are also required to abide by the University Honor Code <http://www.txstate.edu/honorcodecouncil>. While I aim to allow you to make reasonable use of resources, cheating (including copying code, using unauthorized resources during quizzes, etc.) will not be tolerated. If you are found to be cheating, the penalty may range from an F on the assignment to an F on the course, and will also be reported to the university.

External Resources

You may consult external resources such as other books and web sites for understanding how to complete assignments or projects. In in the README

file accompanying each assignment or project, list any external resources you used; if they are available online, provide the URL. You do not need to cite any of the textbooks or the official documentation for the software we are using; in general, pages listed on the course web site's 'Resources' section can be used without citation.

Besides the course forums on TRACS, you may ask questions related to completing the project on publicly accessible discussion forums such as Stack Overflow or publicly-archived mailing lists. Provide URLs to the forum discussion on the relevant web site or archive (Google Groups works well for newsgroup archives) with your project deliverable submission. Sites that require payment or registration in order to view results do not qualify as 'publicly accessible'.

Restrict your questions to questions about how to go about a particular sub-portion of the problem, how something works, why something you are trying doesn't work, or other specific difficulties. Do not ask "how do I solve <the problem description>?", or similarly direct translations of the project requirements, or for specific code. Questions should be written to fill in a gap in your understanding that will then enable you to continue your work, not to get a solution to the assignment.

Conduct

You are expected to behave in a civil, respectful manner in all class interactions, both in official meetings such as lectures and out-of-classroom activities such as project group meetings and study sessions, and to contribute to a constructive learning environment.

Texas State policy (PPS 4.02 <http://www.provost.txstate.edu/pps/policy-and-procedure-statements/4-teaching/pps4-02.html>) describes general behaviors that are disruptive. In addition, the Hacker School Social Rules <https://www.recurse.com/manual#sub-sec-social-rules> are a good source of guidance on how to maintain a constructive and educational environment.

If you experience or witness harassment of any form, please let me know.

Disability Accommodations

If you need particular accommodations to be able to fully participate in this course, please talk with me as soon as possible. I may ask that you provide

documentation from the Office of Disability Services, so if you have such documentation please bring it.

Schedule

Following is a tenatitive schedule for the class. Exact topics and dates may be updated as the course progresses.

Week	Date	Topic	Due
1	Jan. 18	Introducing the Web and HTML	A0
2	Jan. 25	Hypertext	A1
3	Feb. 1	Visual Styling	Q1, P1
4	Feb. 8	Hyper-Text Plumbing	
5	Feb. 15	Data Storage and the Service-Oriented Architecture	P2
6	Feb. 22	Users and Sessions	Q2
7	Feb. 29	JavaScript and REST	A2
8	Mar. 7	Locally Initiated Interactivity	P3
B	Mar. 14	*Spring Break*	
9	Mar. 21	Files and Pictures	Q3
10	Mar. 28	Going Mobile	
11	Apr. 4	TBD	P4
12	Apr. 11	Integrating Third-Party Services	Q4
13	Apr. 18	TBD	A3
14	Apr. 25	Beyond the Web	Q5
F	May. 2	*Final Presentations*	P5

Unless otherwise specified, each assignment is due **before class on Tuesday of the following week**. This means that the first assignment, A0, is due at 10:30 AM on January 26.

Key for the due column:

A*n*   Solo Assignment (4)

P*n*   Team Project (5)

Q*n*   Quiz (5)

This class does not have a final exam. We will use the final exam period (11:00 AM–1:30 PM on Thursday, May 5) for final project demos.