17-437 / 17-637: Web Application Development Spring 2019 Syllabus

January 22, 2019

This course will introduce concepts in programming web application servers. At the conclusion of this course you will understand the fundamental concepts of software engineering and how they apply to web application design and programming, will know the modern tools used to program web application servers, and will be able to produce substantial web applications as part of a team. This course will introduce web application concepts primarily using Django/Python technologies, and you will be able to generalize these concepts to other web application technologies and tools.

During the first part of the semester we will have a series of homework assignments in which you build an increasingly sophisticated web application. The second part of the course will focus on a larger project, in which you will design and implement a substantial dynamic web site of your choice as part of a project team. At the conclusion of your project you will demonstrate your web site to the course staff. There will be a single test: a final exam.

In Spring semesters, the course is offered “lecture-style”. (In Fall semesters, the course is offered “flipped-style” -- video lectures are homework assignments, class meetings are labs.)

Prerequisites

Students will be working with a significant amount of code, so students need to have good programming skills. Ability to program in Python is assumed. If you do not know the Python language, you must have the ability to learn it quickly on your own (and you will need to do so). For undergraduates, either of 15-213 or 17-214 (or their various cross-listings) is required. Prerequisite courses for graduate students are not enforced, but they must have the commensurate background.

Students will need a reasonably modern laptop on which to do assignments and to participate in classroom exercises/demonstrations. Laptops may be of the Windows, MacOS, or Linux variety. Students will need to install and run the Chrome browser as well as Git, Python 3 and Django 2 software.

Course Communications

- Canvas will be used to provide links to lecture slides, videos, and sample code
- Piazza will be used for announcements and Q&A
- GitHub will be used for homework submission

Course Staff

Instructor:

Jeffrey Eppinger (eppinger@cmu.edu), Office: WEH 5124

Teaching Assistants:

Arihant Jain (arihantj@andrew.cmu.edu)
Roy Li (tianqil1@andrew.cmu.edu)
Adithya Raghuraman (araghura@andrew.cmu.edu)
Marat Valiev (mvaliev@andrew.cmu.edu)
Echo Wang (echow1@andrew.cmu.edu)

TA Office Hours will be posted on Piazza and will usually be held in NSH 3001.

Course Meeting Times

This course will meet on Tuesdays and Thursdays at 3pm in the Giant Eagle Auditorium (Baker Hall Room A51). Final project presentations will be scheduled outside of normal meeting times, during the week of April 22nd.

Important Dates

- Homework will be due on Mondays, every week, for the first half of the course.
- You need to be present, in-class, for quizzes and your team's sprint presentations.
- You need to be present for your team's final project presentation (aka the demo).
- You must be present for the final exam. If you must schedule travel plans before the registrar announces the final exam schedule, do not leave before Tuesday, May 14th.

Textbooks

This course has no required textbooks as information about the topics covered in this course is readily available on the internet, but you might find the following texts to be useful references:

- **Mastering Django.**
  - An outdated (based mostly Django 1.8 and Django 1.11) is at https://djangobook.com/the-django-book
  - An upcoming version (Mastering Django 2.0) is being posted here**:** https://djangobook.com/mastering-django-2-book
- **Pro Git.** Chacon. Apress, 2009.
  - Free online at http://git-scm.com/book
  - **Pro Git. Chacon. Apress, 2009.**
- **Software Engineering, 10th edition.** Ian Sommerville. Pearson, 2015.

Course Topics

The expected lecture schedule, subject to change, is as follows:

Date	Lecture Topic	Date	Lecture Topic
1/15	Introduction	3/12	(Spring Break)
1/17	HTML & CSS	3/14	(Spring Break)
1/22	JavaScript & DOM	3/19	Bootstrap
1/24	HTTP & Django	3/21	Transactions
1/29	Cookies & Sessions	3/26	(Sprint #1 Presentations)
1/31	(Quiz & Discussion)	3/28	(Sprint #1 Presentations)
2/5	Forms & Templates	4/2	Internationalization
2/7	Authentication	4/4	Security
2/12	Models	4/9	(Sprint #2 Presentations)
2/13	Files & Images	4/11	(Sprint #2 Presentations)
2/19	AJAX	4/16	Scalability
2/21	jQuery & WebSockets	4/18	(Spring Carnival)
2/26	Cloud Deployment	4/23	(Project Demos)
2/28	Databases & E-mail	4/25	(Project Demos)
3/5	(Project Proposals)	4/30	Review for Final
3/7	(Project Proposals)	5/2	Best Project Awards

Video Records

- We will post video recordings of most lectures (barring unforeseen circumstances).
- You may NOT make your own records of class meetings.

Grading

Your course grade will be computed approximately (and likely exactly) as follows:

- 30% Homework
- 1% Quizzes
- 40% Final project
- 29% Final exam

Currently, we are planning just the one quiz on January 31st.

Late Homework Policy

We understand that normal student-life events, including projects and exams in other courses, can interfere with your ability to complete your work on time. Therefore, in most cases, you may request a short extension on your homework assignment, with minimal penalty, as follows:

- **Request the extension via Piazza before 11:59pm on the due date**
  - Create a private question
  - Provide your Andrew ID
  - Specify when you want to turn in your assignment
  - Tell us the reason why you need the extension

Responses

- Requests for small numbers of late days will generally be granted.
- Requests made by 4pm on the due date will usually get a response before after 5pm. Requests posted after 4pm but before 11:59pm may not receive a response until the next day. Requests made after the due date will not be accepted (unless you get a letter from the Dean of Student Affairs indicating that you have extenuating circumstances).

Counting late days

- We will keep track of the number of late days we grant to you.
- Students requesting a second extension must do so before the end of their first extension. The number of late days used in a second extension will be counted 2:1. (So a grant of 2 additional late days in a second extension will be counted as 4 more days in our totals.)

Penalties

- Generally, for short extensions, there will be no points deducted on homework assignments.
- Students with the fewer late days will be able to sign up earlier for project demo times.
- Students who are granted late days give up their right to complain about having the highest course average in their grade bracket. The more late days you use, the greater the likelihood you will have the highest course average in your grade bracket.

We reserve the right to update this policy if it's not working out.

Late Project Policy

If you are unable to demonstrate your course project at the scheduled final presentation time during "demo week", you may discuss with the professor the possibility of demonstrating your project the following week with a reduction in your project grade.

Collaboration Policy

You should read and abide by the University Policy on Academic Integrity, http://www.cmu.edu/policies/student-and-student-life/academic-integrity.html.

For homework assignments, you are encouraged to talk with and share ideas with other students, including examining and critiquing others' solutions. You must independently create and turn in your own unique work. In particular, you may not copy another student's files or let another student copy your files. You may use external resources (books, internet sites, etc.) as references, but you may not copy files or substantial parts of files from external resources, and you must clearly cite any external resources you use. Citations should be in a `README.md` file at the top-level folder of each homework assignment and project you turn in via GitHub.

You are encouraged to collaborate with your partner and with other students for your course project. All project deliverables, however, must be completed by you and your partner. You may not copy another project's documents or code for your project solution, or use substantial external code or documents obtained from any third party such as an Internet site.

Here are some examples of behavior that are inappropriate:

- Copying files or parts of files (such as source code, written text, or unit tests) from another person or source.
- Copying (or retyping) files or parts of files with minor modifications such as style changes or minor logic modifications.
- Allowing someone else to copy your code or written assignment, in draft or in final form.
- Getting help that you do not fully understand, and from someone whom you do not acknowledge on your solution.
- Writing, using, or submitting a program that attempts to alter or erase grading information or otherwise compromise security of course resources.
- Copying someone else's files containing draft solutions, even if the file permissions are incorrectly set to allow it.
- Lying to course staff.
- Copying prose or programs directly.
- Giving copies of work to others.
- Making your work publicly available in a way that other students (current or future) can access your solutions, even if others' access is accidental or incidental to your goals.
- Coaching others step-by-step without them understanding your help.

There are of course some gray areas, such as receiving help you don't fully understand or copying generic, boilerplate UI designs or configurations from the internet. In general, you should ask the instructor if you have any questions or concerns about the policy, or if you are unsure about the appropriateness of your own past or potential future actions. ***When in doubt, ask the instructor.***

The minimum penalty for violating this policy will be a zero grade for the assignment in question, and ***all*** cases will be referred to the appropriate university disciplinary board. Be warned that the university disciplinary actions for cheating can be very harsh, especially in response to cheating by a graduate student. Note: There is no statute of limitations for violations of the collaboration

policy; penalties may be assessed (and referred to the university disciplinary board) after you have completed the course, and some requirements of the collaboration policy (such as restrictions on posting your solutions) extend beyond your completion of the course.

Well-being Statement

Be well. Try not to stress about this course. This course is actually meant to be fun. Our late policies are designed to reduce stress. If you're so behind that you're about to cheat (or help someone else cheat), please go to sleep and come see me in the morning (or tell someone else to do this). We'll work something out.