American University
Web Programming, CSC 435, Spring 2018

Basic Info:

Instructor: Prof. Bei Xiao, American University.

Contact: bxiao@american.edu,

Office: Myers Technology Room 204.

Lecture Time: Tuesday/Friday 4:05-520pm

Location: DMTI 121

Office hours: Thursday, 4-5pm,
Friday 5:30-6:30pm
or by appointment (48 hours advance).

Prerequisite: Introduction to Computer Science 1 (CSC 280).
CSC 435 is a rigorous (and fun) upper-division CS course that assumes familiarity with both basic computer science concepts as well as solid programming abilities in a command-line environment. **If you haven't taken CSC 280 or if you are not familiar with one (Python, C or Java) programing language, it is recommended that you learn ONE programming language first before you take this class.**

Course Content and learning objectives

The Web is intertwined with our lives. No matter if you decide to be a politician, a musician, a small business owner, a scientist, an artist, a doctor, a journalist, or purely an interesting person, web design is an extremely useful skill. Web development is useful in advertising (personal homepages), entrepreneurship, (web API), journalism (news media and blog), multi-media (Pinterest), social networks (Facebook LinkedIn), e-commerce (Amazon), user experiences and crowd-sourcing (Amazon Mechanical Turk), and online games (Phantasy star online).

This course is an introduction to programming for the World Wide Web. We will cover all the major pieces of how websites work. This will include the relationship between clients and servers, how web pages are constructed, and how the Internet works. We will NOT focus on how to stylize a website. Students who are interested in web design should take courses from the Graphic Design Program.

We will examine several technologies in various depths, though our majority of time will spend on JavaScript and its libraries.

- HyperText Markup Language (HTML) for authoring web pages
- Cascading Style Sheets (CSS) for supplying stylistic information to web pages
- **JavaScript and its libraries (jQuery, D3) for creating interactive web pages**
- PHP Hypertext Processor for generating dynamic pages on a web server
- Asynchronous JavaScript and XML (Ajax) for enhanced web interaction and applications
- JSON for transferring data
- Structured Query Language (MySQL) for interacting with databases

- Git and Github tools for version control and web hosting.

Course website:

The course Github is here:

https://github.com/fruittree/CSC435WebProgramming

Lecture slides, notes, and homework solutions will be posted here.

Learning outcomes:

The course is NOT simply teaching you how to build a web side. You can learn this from free online tutorials. However, the expected learning outcome of the course is to have a deep understand the fundamental technology and principles behind web applications so that you can adapt to any new programming languages and APIs you might encounter in your career. You will also be expected to learn how to debug your own code, which is an important skill of software engineering. Of course, you will be acquiring skills about building a dynamic user-friendly website that interacts with database.

Textbooks (not required):

The course has no required official textbooks but I will assign weekly readings (free online) through Blackboard.

However, I find the following textbooks useful as references.

1. JavaScript, The good part. Douglas Crockford. O'Reilly Media, 2008.
2. Stepp/Miller/Kirst. Web Programming Step by Step, Second Edition. ISBN 978-1-105-57878-6.

Free online tutorial:
https://bento.io/tracks

Computers and Software:

The recommended software for the course is the Chrome web browser or Firefox web browser with Firebug add-on. The recommended text editor is Sublime text 3. The course website will have places to download these software.

You can use any IDE for web developing. I recommend

http://brackets.io/

We will also demonstrate how to install Git via command line. Git is extremely important for developer to do version control.

Later in the semester, you should download Ampps in order to test your server-side script on your local machine.

http://www.ampps.com/downloads

You do not need to upload your HTML or JavaScript file to a web server to see your webpage. But at some point, we will talk about testing your webpage on the Internet by uploading your files on to free web server.

Grading:

Graded work will receive categorized point values, with the following categories and their respective weights:

50% individual assignments (approximately 5-6 projects)

15% Mid-term exam (multiple choices and short programming exercises)

10% Creative project

15% Final Project

5% Quiz

5% In-class exercises. These are short exercises, such a simple maze.

There won't be in-class final exam for this course. But there will be a final presentation of your final project, which will meet on the final exam day. Please arrange your travel accordingly.

We will have weekly in-class quizzes (randomly timed without advance notice). Quizzes are usually drawn from lectures and assigned readings. It is very important that you finish the assigned readings.

We don't have a dedicated lab session. But we will do many in-class exercises in the last 15 mins of the class. These exercises are not turned but you are expected to show your results to your instructor in class.

Grading Scale listed below:

94-100%	A	Excellent
91-93%	A-	
88-90%	B+	
84-87%	B	Good
80-83%	B-	
77-79%	C+	
73-76%	C	Acceptable
71-72%	C-	(cut off to receive credits for CS major)
60-70%	D	Poor
0-59%	F	Fail

I generally do not allow arguing for grades unless you find a calculation error. If your percentage is 85.6, for example, it will be round up to 86%, but if it is 85.4%, it will be round up to 85%. I will try my best to estimate your grade percentage as the semester goes along.

Again, honor students who wish to complete honor supplements needs to acquire additional 20% of the grades. Please talk to me in office hours if you wish to pursue this.

Creative Project (New!!):

To give you a chance to play around with the technologies we're learning, this semester one of your homework assignments will be to write a user response website from scratch, entirely of your own design.

Each week, there will be some requirements that you have to meet, but the requirements don't force you to have any particular content, layout, color scheme, feel, etc – you'll make that yourself.

The end project of your creative website will be an interactive crowd-sourced user-response web app. At the end, the working project can also be sent to Amazon Mechanical Turk or other server to collect real-time user responses from the crowd. What type of data do you want to collect? It is entirely up to you (e.g. neighborhood safety rating, movie reviews, labeling photos, political surveys, etc). This project will allow you to learn how to use front-end design, create interactive user experiment and collect data, real-time data visualization, back-end data storage and analysis, as well as interacting with databases and whatever tools you came across.

Attendance policy: It is an interactive class with lots of live demos and discussions. Missing a class must be accompanied by written medical proof in advance (at least 3 days) and athletes must submit written evidence before they miss classes for sports events. Without any advanced written notice, **missing 2 classes will result in zero attendance score**. **Missing one class without written request will result in 2% reduction in attendance score.**

In-class quiz, which will be randomly assigned, cannot be made up.

Late Policy: **Homework assignments are usually individual programming projects.** All assignments will be submitted electronically on Blackboard, and will be due at **11:59 pm on the due date**. No assignments are accepted via email.

Assignments must be submitted by the **due date to receive credits.**

Each late homework will receive 5% automatic reduction for each late day. No homework is accepted 5 days after the due date.

Email Policy:

You can email me if you have questions regarding home-works. But you must write to me at least 48 hours to expect an answer. No homework is accepted via Email.

Policy of collaborative work:

Programming assignments must be completed individually unless teamwork is specified. You may discuss an assignment in general terms with other students, including a general discussion of how to approach the problem, but all code you submit must be your own. Any help you receive from classmates should be limited and should never involve details of how to code a solution.

You must abide by the following:

- You may not work as a partner with another student on an assignment.
- You may not show another student your solution to an assignment, nor look at his/her solution.
- You may not have another person "walk you through" an assignment, describe in detail how to solve it, or sit with you as you write it.
- You also may not provide such help to another student. This includes current or former students, tutors, friends, TAs, web site forums, or anyone else.
- You can discuss general ideas and ask specific questions either in office hour or post on Pizza.

We enforce our policies by running detection software during the quarter over all programs, including ones from past quarters. Please contact me if you are unsure whether a particular behavior falls within the policy.

Academic Integrity

Plagiarism and academic misconduct are defined in the University Academic Integrity Code. You should be familiar with what constitutes academic dishonesty. For example, it is extremely forbidden to share code and answers during an exam.

In particular, you should observe the following rules: only high-level discussions are allowed (i.e., not relating to a single line of code), and you have to declare whom you discussed with.

CSC 435 Course Schedule (Tentative, depends on class progress and students'backround):

Please refer to course website for updated information.

Weeks	Lecture contents	Projects
1. Jan 16-19	Internet, Web, Server, Client, HTML	
2. Jan 22-26	More on HTML	
3. Jan 29-Feb 2	CSS, Git and web development environment	Project 1 out
4. Feb 5-9	Start with JavaScript	
5. Feb 12-16	More on JavaScript	Project 1 (Recipe) due
6. Feb 19-23	jQuery	Project 2 Out
7. Feb 26 -March 2	jQuery, D3	Project 2 Due
8. March 5- March 9	D3, data visualization	Project 3
9. March 12-16	Spring break	
10. March 19-23	Review, Mid-term Exam	Project 3 Due
11. March 26-30	More on PHP	Project 4
12. March 28- April 1	Relational Database SQL	
13. April 2-6	User account, Log-in sessions; cookies	Project 4 Due
14. April 9-13	More on cookies and session management	Project 5 Out
15. April 16-20	Ajax Client server interactions. JSON	Project 5 Due
16. April 23- 27	jQuery Ajax, Local Storage, Amazon Mechanical Turk.	Creative Project Final Due
17. May 1	Final Project due and presentations	