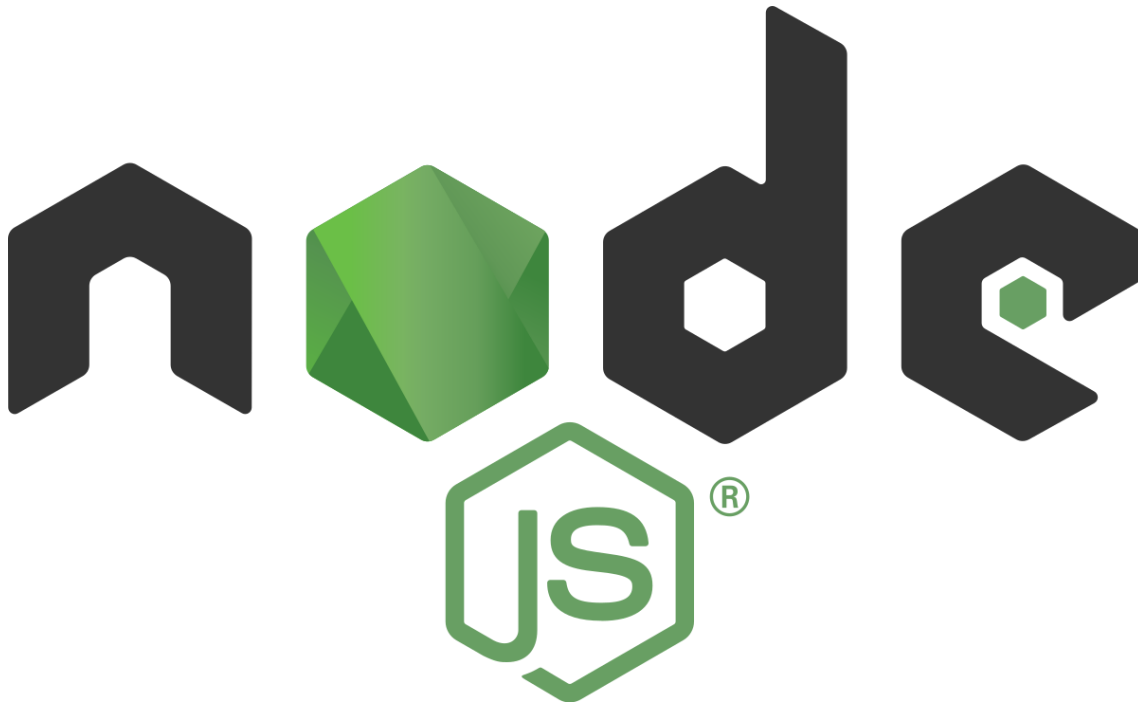# 389Kspring20

CMSC389K: Full-stack Web Development with Node.js

Course Description

This course provides a comprehensive, practical introduction to modern full-stack web development using JavaScript and Node.js. The course will start with basic HTML/CSS/JavaScript. Then, we will move into Node.js and learn how to deploy a website from there. We will learn about Express.js (server-side development module) and MongoDB (database) in order to create a complete web application.

Course Details

- **Course**: CMSC389K
- **Prerequisites**: C- or better in CMSC216 and CMSC250
- **Credits**: 1
- **Seats**: 30
- **Lecture Time**: Fridays, 11:00AM-11:50AM
- **Location**: CSI 2120
- **Semester**: Spring 2020
- **Textbook**: None
- **Course Facilitators (Section 0101)**: Camille Stacho and Robert Choe
- **Course Facilitators (Section 0201)**: Jordan Mess and Elijah Schwelling
- **Faculty Advisor**: John Dickerson
- **Syllabus Last Updated**: September 31 2019

Topics Covered

- HTML and CSS
- JavaScript (+ECMAScript 2015 (E6))
  - Variables, data types, expressions, operators
  - Conditionals, iteration, statements
  - Functions
    - Functional Programming
    - Callbacks
  - Interaction with DOM
    - DOM Manipulation
    - Event Handling
  - jQuery + AJAX
    - Dynamic interaction with the DOM
    - Making API Calls
- Node.js
  - File manipulation
  - Modules
- Express.js
  - API (Application Program Interface)
  - Web Application Framework for Node.js
- MongoDB
  - Nonrelational Database
  - Persistent Data
- Javascript Libraries
  - Chart.js
  - Typed.js
- Socket.io
  - Real-time communication between server and clients
- Now
  - Application deployment
- Git
  - Version Control

Grading

Grades will be maintained on the CS Department [grades server](#).

You are responsible for all material discussed in lecture and posted on the class repository, including announcements, deadlines, policies, etc.

Your final course grade will be determined according to the following percentages:

Percentage	Title	Description
45%	Projects	Weekly individual projects that teach practical skills and real life applications.
10%	Participation	Small assignments to be completed in groups.

20%	Midterm	Midterm Project
25%	Final Project	Final project to demonstrate mastery of all topics learned and apply knowledge to create a new application from scratch. Group Project!

Any request for reconsideration of any grading on coursework must be submitted within one week of when it is returned. No requests will be considered afterwards.

Week	Topic	Assignment
1 (1/31)	Intro + HTML/CSS	Personal Website (P1)
2 (2/7)	Javascript Language w/ DOM	P1 DUE, JS Function Implementation (P2)
3 (2/14)	jQuery and AJAX	P2 DUE, 50 State Game (P3)
4 (2/21)	Node.js	
5 (2/28)	Express.js	P3 DUE, Pokémon API (P4)
6 (3/6)	Express.js (cont.)	
7 (3/13)	Express.js (cont.) / Handlebars.js	P4 DUE & Midterm Project
8 (3/20)	Spring break	
9 (3/27)	Databases	Midterm Project DUE
10 (4/3)	Web Sockets	
11 (4/10)	Midterm Project Presentation	Final Project OUT
12 (4/17)	Deployment & Advanced Git	
13 (4/24)	Javscript Libraries (Chart.js / Typed.js)	
14 (5/1)	Javascript Libraries	
15 (5/8)	Presentations	Final Project DUE

Projects

Projects must be submitted electronically following the instructions given in each project assignment. Projects may not be submitted by any other means (e.g., please do not email your projects to us). It is your responsibility to test your program and verify that it works properly before submitting. All projects are due at 11:59 PM on the day indicated on the project assignment. Please read submission details thoroughly.

In-Class Assignments

At the end of most of the classes there will be a 5-10 min group (max 3 people) exercise for extra practice and application of the material taught. Each assignment counts for 1% of your grade. You can drop at most 2 in-class assignments.

Late policy (IMPORTANT)

Projects may be submitted up to 24 hours late for a 7% penalty and **will not be accepted afterwards unless you talked with us for an extension**. If you submit both on-time & late, your project will receive the

maximum of the penalty-adjusted scores. You may submit multiple times, but only your most recent submission will be evaluated unless it is late. We give students ample time to do all the projects (1-2 weeks).

Unlike lower-level programming classes, we will not provide you with test cases (e.g., public tests) before projects are due. You will be responsible for developing your own tests and for using appropriate testing techniques. Also, we expect your projects to use proper style and documentation.

Outside-of-class communication with course staff

We will interact with students outside of class in primarily two ways: online at Piazza and office hours by appointment. Email should only be used for emergencies and non-class related questions (e.g. projects).

Instructor:

Dr. John Dickerson - john@cs.umd.edu

TAs:

[Camille Stacho] - camstach@gmail.com

[Robert Choe] - robert.d.choe67@gmail.com

[Elijah Schwelling] - eschwe333@gmail.com

[Jordan Mess] - jmess@gmail.com

Excused Absence and Academic Accommodations

See the section titled "Attendance, Absences, or Missed Assignments" available at Course Related Policies.

Disability Support Accommodations

See the section titled "Accessibility" available at Course Related Policies.

Academic Integrity

Note that academic dishonesty includes not only cheating, fabrication, and plagiarism, but also includes helping other students commit acts of academic dishonesty by allowing them to obtain copies of your work. In short, all submitted work must be your own. Cases of academic dishonesty will be pursued to the fullest extent possible as stipulated by the Office of Student Conduct.

It is very important for you to be aware of the consequences of cheating, fabrication, facilitation, and plagiarism. For more information on the Code of Academic Integrity or the Student Honor Council, please visit http://www.shc.umd.edu.

Course Evaluations

If you have a suggestion for improving this class, don't hesitate to tell the instructor or TAs during the semester. At the end of the semester, please don't forget to provide your feedback using the campus-wide CourseEvalUM system. Your comments will help make this class better.