

## COURSE SYLLABUS

<b>Instructor</b>	Abdulmalek Al-Gahmi
<b>Contact Information</b>	Office: TE 110-E Email: aalgahmi@weber.edu Phone: 801-626-7915
<b>Class meeting</b>	<b>Tue/Thu</b> 9:00 -10:20am @ EH 313
<b>Office Hours</b>	<b>Mon/Wed:</b> 11:30am -12:00pm <b>Tue/Thu:</b> 10:30am - 12:30pm
<b>Pre-requisites</b>	WEB 3200: Dynamic Languages for Web Development
<b>Credit Hours</b>	3

### Course Description

MEAN is a collection of JavaScript-based technologies — MongoDB, Express.js, AngularJS, and Node.js — used to develop web applications. This course introduces development techniques that capitalize on the strengths of every layer in the MEAN stack.

### Learning Objectives

At the conclusion of this class, the students will be able to:

- Differentiate between front-end and back-end web development;
- List the pros and cons of Single Page Applications (SPA's);
- Create and manage a NoSQL database using MongoDB;
- Build a full-stack web application using the MEAN stack;
- Build and consume a REST API; and
- Deploy a full-stack application build using the MEAN stack and use GitHub to manage its source code.

## LEARNING RESOURCES

### Textbook

No textbook is required for this class, but the following books are useful as references and are available for FREE:



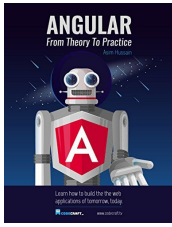
**Eloquent JavaScript** (available online for free)

**Edition:** 2nd

**By:** Marijn Haverbeke

**ISBN:** 978-1593275846

**Online:** <http://eloquentjavascript.net>

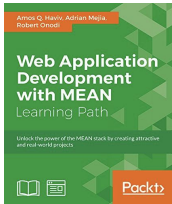


**Angular 5: From Theory To Practice** (available online for free at Amazon)

**Edition:** Kindle

**By:** Asim Hussain

**Online:** [An Amazon Kindle Book](#)



**Web Application Development with MEAN** (available online via the library)

**By:** Amos Q. Haviv; Adrian Mejia; Robert Onodi

**Pub. Date:** October 28, 2016

**ISBN:** 978-1-78712-172-0

**Online:** [Safari Books Online](#)

## Canvas

Canvas is where course modules, assignments, grades, and announcements will reside. It can be accessed from <https://canvas.weber.edu>. For Canvas-related technical support, please click the HELP link in the top right corner of your screen. You can also call WSU Online at (801) 626-6499 or email [wsuonline@weber.edu](mailto:wsuonline@weber.edu).

## Online Resources

### Software (the MEAN stack)

- **M:** MongoDB Community - <https://www.mongodb.com/community>
- **E:** Express - <https://expressjs.com/>
- **A:** Angular - <https://angular.io/>
- **N:** Node.js - <https://nodejs.org/en/>

### Development Tools

Text Editors:

- Adobe Brackets: <http://brackets.io/>
- Visual Studio Code: <https://code.visualstudio.com>

Extensions/Plugins (available both above editors):

- Emmet (for improving HTML and CSS workflow): <https://emmet.io/>

### Online Resources

- Emmet Cheat Sheet: <https://docs.emmet.io/cheatsheet-a5.pdf>
- Codepen: <https://codepen.io/>
- Plunker: <https://plnkr.co/>
- GitHub: <https://github.com/>
- Git Cheat Sheet:  
<https://services.github.com/on-demand/downloads/github-git-cheat-sheet.pdf>

## LEARNING ACTIVITIES

Class will consist of lectures, discussions, quizzes, programming assignments, and a final project. Students are expected to attend classes, read the weekly-assigned materials, actively participate in class activities, and submit assignments on time.

### Readings

Weekly reading assignments will be posted to Canvas. These readings may be from the reference books above or in the form of relevant online articles, tutorials, and/or blog posts. Students are highly recommended to read the assigned materials before coming to class.

### Quizzes

There will be two quizzes accounting for 10% of the final grade and testing your JavaScript and TypeScript skills.

### Assignments

There will be 6 programming assignments accounting for 60% of the final grade. These assignments are typically given on Tuesdays and are due on Friday of the week after.

### Final Project

The final project is worth 30% and is where you put everything you learned in this class together by using all the components of the MEAN stack to create a fully functional web application of your choosing.

### Schedule

The following is a tentative high-level weekly schedule of this class; it is subject to change at any time. Always refer to Canvas for more details and due dates.

Week of	Topic	Quiz	Assignment
August 27	Introductions, command-line; npm; GitHub		
September 3	JavaScript review	1	1
September 10	Full-stack web development		
September 17	MongoDB		2
September 24	Building a data model		
October 1	Building a REST API		3
October 8	Building a dynamic site		
October 15	Forms		4
October 22	ES6 and TypeScript	2	
October 29	Angular		5

November 5	Building an SPA		
November 12	Sessions and user authentication		6
November 19	Final Project		
November 26	Final Project		
December 3	Final Project		
December 10	Final Project		

## GRADING

### Scale

The final grade will be calculated based on the following scale with the passing grade being C or above.

<b>A:</b> 100 - 94	<b>B-:</b> <84 - 80	<b>D+:</b> <70 - 67
<b>A-:</b> <94 - 90	<b>C+:</b> <80 - 77	<b>D:</b> <67 - 64
<b>B+:</b> <90 - 87	<b>C:</b> <77 - 74	<b>D-:</b> <64 - 60
<b>B:</b> <87 - 84	<b>C-:</b> <74 - 70	<b>E:</b> <60

### Distribution

The final grade is broken down as:

**10%** Quizzes

**60%** Assignments

**30%** Final Project

### Software Engineering Note

Up to 10% of the submitted programs' grades will be towards making sure that these programs are easily readable, clearly documented or commented on, and properly indented.

## POLICIES/STATEMENTS

### Extra Credit

No extra credit is available beyond what is already specified above.

### Late Policy

Quizzes cannot be made up unless arrangements are made to take them ahead of time. You will be able to submit one late assignment for full credit and after that all late assignments will be given half credit.

### Allocated Time

You should anticipate spending two to three hours of study per week for each credit hour of a university course. Computer and programming classes typically require time in the upper range. An online class is not easier; it is harder. You have to do more reading and learning on your own.

### **Tips for Success**

- One cannot learn a new software and/or a programming language by just reading textbooks or watching videos; practice is critical when learning. So write as many programs as you can.
- The most effective way to get a C or above in this class is to stay current with the course topics and submit assignments on time. Your grades will be based on the degree to which you fulfilled the requirements of this course and not on you needing to get a C or above.

### **Incomplete Grades**

An "Incomplete" may be given only when the student, having satisfactorily completed approximately 80% of the required work, is unable to complete the class work for a legitimate reason (such as illness or accident) and can reasonably finish on his/her own.

### **Course Fees**

Course fees are designed to cover the costs of lab equipment maintenance and replacement including desktop and server computer systems and software; consumable materials and supplies; and support for lab aides, student tutors, and online instructional resources.

### **Academic Dishonesty**

Students are expected to maintain academic ethics and integrity in regards to performing their own work. The WSU Student Code specifically prohibit the following activities:

- a. Cheating, which includes but is not limited to the following examples:
  - i) Copying from another student's test;
  - ii) Using materials during a test not authorized by the person giving the test;
  - iii) Collaborating with any other person during a test without authorization;
  - iv) Knowingly obtaining, using, buying, selling, transporting, or soliciting in whole or in part the contents of any test without authorization of the appropriate University official;
  - v) Bribing any other person to obtain any test;
  - vi) Soliciting or receiving unauthorized information about any test;
  - vii) Substituting for another student or permitting any other person to substitute for oneself to take a test;
  - viii) Knowingly obtaining academic credit for work that is not one's own regardless of the source of the work;
  - ix) Knowingly involved in arranging fraudulent academic credit or false transcripts.
- b. Plagiarism, which is the unacknowledged (uncited) use of any other person's or group's ideas or work. This includes purchased or borrowed papers;
- c. Collusion, which is the unauthorized collaboration with another person in preparing work offered for credit;
- d. Falsification, which is the intentional and unauthorized altering or inventing of any information or citation in an academic exercise, activity, or record-keeping process;
- e. Giving, selling, or receiving unauthorized course or test information;

- f. Using any unauthorized resource or aid in the preparation or completion of any course work, exercise, or activity;
- g. Infringing on the copyright law of the United States which prohibits the making of reproductions of copyrighted material except under certain specified conditions.

School of Computing policy dictates that any verifiable evidence of student academic cheating, as defined and determined by the instructor, will result in:

1. an automatic failing grade for the class and
2. a report to the Dean of Students that will include the student's name and a description of the student's dishonest conduct.

### **Accommodations for Disabilities**

Any student requiring accommodations or services due to a disability must contact Disability Services in Room 181 of the Student Services Center (or Room 256 at the Davis Campus). Disability Services can also arrange to provide course materials (including this syllabus) in alternative formats upon request. You can also call 801-626-6413 (Ogden) or 801-395-3442 (Davis) or visit <http://www.weber.edu/ssd> for more details.

### **Disclaimer**

The instructor reserves the right to make changes to this syllabus, as he sees fit, anytime during this class.