



Professional Applications and Frameworks in Web Development

ITP 405x (3 Units)

Objective

Provide students with the necessary skills to build structured, maintainable, scalable, and tested web sites and applications using frameworks and tools common in the industry.

Concepts

The course will also cover the differences between traditional server-side technologies like PHP and Ruby and how they differ from Node.js, an asynchronous server-side alternative with JavaScript. You will learn how to build traditional server-side rendered applications with PHP as well as API driven applications with a frontend JavaScript framework.

Prerequisites

ITP 300 or CSCI 351 (or sufficient experience). You should be proficient with building dynamic web pages using HTML, CSS, SQL, and PHP.

Lecture

3 hours / week

Course Structure

Students are expected to participate in lecture discussions and critiques, complete weekly assignments and projects, and manage and complete individual class projects.

Students are responsible for completing assignments and projects by stated deadlines. Students will upload most assignments to Github or Bitbucket.

Reading

PHP Object Oriented Solutions by David Powers: Apress, 2013 – There is a free version provided by USC. Visit [this link](#). Click the link next to Electronic Access that says "» SpringerLink - An electronic book accessible through the World Wide Web; click for information". Log in with your USC credentials, and then you can download the book in PDF format.

Articles – Each week there will be mandatory online articles to read.

Grading

Assignments: 35%

Class participation, attendance, and labs: 10%

Written exam: 20%
Individual final project: 35%

Final course grade is determined by standard formulas:

A	100% - 93%
A-	92% - 90%
B+	89% - 87%
B	86% - 83%
B-	82% - 80%
C+	79% - 77%
C	76% - 73%
C-	72% - 70%
D+	69% - 67%
D	66% - 63%
F	62% and below

Policies

It is the responsibility of the student to make sure projects and assignments are turned in on time. Make sure you follow the procedures outlined in each assignment or project.

Late projects will be reduced a letter grade per day after the assignment was due. No projects will be accepted later than five days from the due date.

Academic Integrity

The use of unauthorized material, communication with fellow students during an examination, attempting to benefit from the work of another student, and similar behavior that defeats the intent of an examination or other class work is unacceptable to the University. It is often difficult to distinguish between a culpable act and inadvertent behavior resulting from the nervous tension accompanying examinations. When the professor determines that a violation has occurred, appropriate action, as determined by the instructor, will be taken.

Although working together is encouraged, all work claimed as yours must in fact be your own effort. Students who plagiarize the work of other students will receive zero points and possibly be referred to Student Judicial Affairs and Community Standards (SJACS).

All students should read, understand, and abide by the University Student Conduct Code listed in SCampus, and available at:

<http://www.usc.edu/student-affairs/SJACS/nonacademicreview.html>

Students with Disabilities

Any student requesting academic accommodations based on a disability is required to register with Disability Services and Programs (DSP) each semester. A letter of verification for approved accommodations can be obtained from DSP. Please be sure the letter is delivered to me (or to your TA) as early in the semester as possible. DSP is

located in STU 301 and is open 8:30 a.m. - 5:00 p.m., Monday through Friday. The phone number for DSP is (213) 740-0776.

Course Outline

1/12	Class Introduction Traditional vs. API driven web applications Lab: Installing PHP >= 5.4 and Composer Lab: Installing Git and setting up Github Reading: PHP Object Oriented Solutions – Chapter 1 & 2
1/19 (PHP)	HTTP lifecycle Database-driven web pages review PHP Data Objects (PDO) PHP namespaces Break Lab: Installing Laravel Lab: PHP autoloading Assignment
1/26 (PHP)	SQL joins Break What is MVC? Introduction to Laravel Controllers, views, and database access Assignment
2/2 (PHP)	CRUD Break Flash messages, data validation Authentication Assignment
2/9 (PHP)	Object Relational Mapping (ORM) with Eloquent Assignment
2/16 (PHP)	REST Working with 3 rd party APIs Service classes Server-side caching Assignment
2/23 (PHP)	Building an API Assignment
3/1 (PHP)	Testing overview & terminology

Unit testing with PHPUnit
Reading: JavaScript basics
Assignment

3/8 (PHP) Test Doubles with Mockery
Dependency injection
Reading: JavaScript basics
Lab: Installing Node.js, Bower, and Ember CLI
Assignment

3/15 Spring Break

3/22 (JS) Client-side frameworks with Ember
Assignment

3/29 Written exam

4/5 (JS) Client-side frameworks with Ember
Assignment

4/8 Last day to drop with a mark of "W"

4/12 (PHP) Securing APIs

4/19 (Node) Introduction to asynchronous programming with Node.js
Introduction to Express
Break
Working with web services
Reading: CommonJS modules and NPM
Reading: Flow control patterns
Optional Reading: Testing in Node with Mocha and Chai
Assignment

4/26 (Node) Database access with Sequelize ORM
Reading: Deploying to Heroku
Reading: Continuous integration with Travis CI

Final Projects due Saturday at 10am of the last week of class