CSPP 52553 Web Development

Spring Quarter, 2013

Department of Computer Science
University of Chicago

Dates:	Mondays, April 1, 2013 - June 14, 2013
Lecture Location:	The Starter League, Merchandise Mart
Website:	www.cspp52553.com

Lecturer:	Jeff Cohen
Email:	jeff@starterleague.com
Office:	Starter League Offices, Merchandise Mart
Office Hours:	Mondays, 3:00-4:00

TA:	Raghu Betina
Email:	raghu@starterleague.com
Office Hours:	TBA

Course Description

This course provides students with an introduction to web development, with an emphasis on the pragmatic skills needed to build live, functioning web applications.

Learning Goals

Specifically, students will learn how to:

1. Build a live website or web application and deploy it to the public internet
2. Use the Ruby on Rails framework to rapidly build a web application
3. Write software using the Ruby programming language
4. Use a relational database to provide content for dynamic websites
5. Follow industry best-practices of modern web software development
6. Troubleshoot and resolve the most common problems with web applications

We will use the Ruby language and the Rails framework to immerse students into the challenge of building a live, database-backed web application deployed at a public web address.

Students will learn fundamental domain modeling skills, agile software techniques and best practices, Javascript and AJAX, profiling/optimization, and server-side and client-side debugging.

Course Organization

Lecture sessions contain built-in lab sessions, in which students will gain practical skills by working in iMac computers in the classroom. Each session is therefore a combination of brief demonstrations, lecture material, and "pair programming" labs intended to be both enjoyable and provide a dynamic learning experience.

Grades are based on a combination of factors:

- (20%) Weekly homework (weeks 1 through 5 only);
- (30%) A midterm exam, administered during week 6
- (50%) A final project, due by the end of the course

In-Class Projects

Throughout the quarter, we will develop a couple of web applications as a class. Examples may include:

- A "mashup app" that includes live data from Facebook and YouTube
- A photo sharing site
- An airline reservation system
- An online newspaper

Homework

For the first five weeks, students are required to complete short coding assignments that are submitted electronically. Instructions for homework submission will be provided on the first day of class.

Midterm Exam

One midterm exam will be administered during the final hour on February 13, 2013. The midterm will be a combination of multiple-choice and short programming assessments.

Final Project

Final projects are done in pairs. Students are highly encouraged to find a partner to collaborate with for the final project, but solo projects are also acceptable. Three choices for a final project will be announced in class. Some examples might be: an e-commerce site, and book club web application, or an online event invitation system.

Grading Policies
Grades on homework, the midterm, and the final project are based on various factors depending on the assignment. However five objective criteria are paramount, and are listed here, in decreasing order of importance:

1. Working code (zero defects)
2. Optimal use of built-in Ruby classes and algorithms
3. Proper application of Rails classes and methods
4. Adherence to agile principles such as DRY, YAGNI, and SRP
5. Adoption of Ruby idiomatic style

Suggested Reading

Course notes will be provided, but students may wish to augment their learning with any of the following texts:

- *Learn to Program*, 2nd Edition, Chris Pine, Pragmatic Bookshelf 2009.
- *Rails Tutorial*, 2nd Edition, Michael Hartl, Addison-Wesley 2012.
- *Agile Web Development with Rails*, 5th Edition, Sam Ruby et. al. Pragmatic Bookshelf 2010.
- *Programming Ruby 1.9*, 10th Anniversary Edition, Dave Thomas. Pragmatic Bookshelf, 2010.

Calendar

Week	Topics	Due
1 **Wednesday Jan 9**	Introduction to Ruby. Data structures and enumeration. Basic HTTP protocol. JSON data format and parsing. Unit testing in Ruby.	
2 **Wednesday Jan 16**	Introduction to the Rails framework. The MVC architecture pattern. Development tools.	*Homework #1 Due*
3 **Wednesday Jan 23**	Receiving user input. Forms.	*Homework #2 Due*

Week	Topics	Due
4 **Wednesday Jan 30**	Relational databases. Models and schema migrations. Business logic. MIME types and custom responders. Rails resources.	*Homework #3 Due*
5 **Wednesday Feb 6**	ActiveRecord query interface. Advanced database migrations. Model associations (part 1).	*Homework #4 Due* **Project Declaration Due**
6 **Wednesday Feb 13**	Model associations (part 2). Advanced queries. Browser security. HTTP cookies and sessions. **MIDTERM DURING FINAL HOUR.**	*Homework #5 Due*
7 **Wednesday Feb 20**	User authorization and security. RubyGems.	**Project Review Due**
8	Search functionality. Pagination. Sending email.	
9	Javascript, Ajax, CoffeeScript	
10	Profiling, caching techniques, debugging, rake tasks	
11	**FINAL PROJECT DUE**	