

# CSc 397B: Web Application Development with PHP and MySQL

*Syllabus - Fall 2007*

## **Description**

Introduction to developing secure, dynamic, database-driven web applications using PHP and MySQL.

## **Credit**

1 unit, Pass/Fail.

## **Prerequisite Courses**

C SC 127B or C SC 227.

## **Instructors**

Course instructor: Justin Samuel <jsamuel@cs.arizona.edu>

Instructor of record: Rick Mercer <mercer@cs.arizona.edu>

All official university actions (adds, drops, etc.) must go through Rick.

## **Grading**

Pass/Fail only. "Pass" will be provided to students with satisfactory attendance (see attendance policy below) who complete an acceptable level of course work throughout the semester.

It should be pretty hard for a student to get a "Fail", but not showing up for the lecture more than the attendance policy allows, not attempting the activities or projects assigned during the course, and generally blowing off the course could do it.

If at any point you have concern about whether you are passing the course, please talk to the instructors.

## **Lecture/Lab**

Mondays from 5:00 to 6:50pm in Gould-Simpson 930.

The scheduled time will be a combination of lecture and lab. We plan to usually format this two-hour period with the first hour being more lecture-heavy and the second hour being a lab period where you will be able to work on activities related to the preceding lecture.

## **Homework**

It is likely that there will be very little of your time required outside of class. This is a two-hour period for a one-unit course. If at any time you feel that the time expected of you outside of class is greater than it should be, please do let us know. It is possible that we have misjudged the time required to complete various activities.

## **Attendance**

Attending class time is extremely important for this class. Attendance will be recorded. Missing three or more lectures will result in a failing grade unless other provisions have been made with the instructors. Please talk to us if you are having

problems attending.

The fact that most of the lecture will be in the first hour doesn't mean that there will be no lecture or useful information discussed in the second hour. It does mean, however, that if there is a week where you have to leave early, you'll still benefit from showing up for as much of the two-hour period as possible. (Don't skip the whole class that week just because you need to leave early!)

**Required texts:**

None.

**Academic Integrity:**

Students are responsible for understanding and complying with the University's Code of Academic Integrity. The Code is found at <http://dos.web.arizona.edu/uapolicies/> The full text is available from the Office of the Dean of Students in Room 203 Old Main.

**Course Objectives**

Students will learn the platform-neutral fundamentals of secure, dynamic web application development. Students will also learn how to implement a web application using one specific set of open source server-side tools: PHP and MySQL.

Upon completion of the course, students will be able to efficiently continue to expand their web development knowledge on their own with the solid foundation gained in the course. Students will also have a level of web application security knowledge exceeding that of many web developers in industry.

**Topics Covered**

The following topics will be covered in the course, some in more depth than others:

- Running PHP, MySQL and Apache locally for development purposes.
- Using a professional IDE (Eclipse) for PHP development.
- Understanding PHP syntax.
- Writing object-oriented code in PHP 5.
- Using PHP application frameworks.
- Using templating systems for separation of code and design.
- Handling upload of files and performing basic image processing in PHP.
- Understanding important PHP configuration settings.
- Writing basic HTML.
- Using Cascading Style Sheets (CSS).
- Using HTML forms to allow website users to submit data.
- Protecting HTML forms from abuse by automated bots.
- The difference between GET and POST requests and when to use each.
- The usage and purpose of JavaScript and AJAX.
- Using basic JavaScript.
- Developing scalable web applications.
- Performing website user authentication.
- Working with sessions and store user data between sessions.
- Using cookies.

- Writing basic SQL queries to add, manipulate and retrieve data from a database.
- Designing simple relational databases in MySQL.
- Using PHPMyAdmin for managing a MySQL database.
- Using and generating XML in PHP.
- Deploying websites.
- Fundamental security considerations related to web applications (see below).

### Security Topics Covered

The following security-related topics will be covered in the course:

- Cross-Site Scripting (XSS) attacks and prevention.
- Cross-Site Request Forgery (CSRF/XSRF) attacks and prevention.
- SQL Injection attacks and prevention.
- HTTP Header Injection attacks and prevention.
- Header Injection in dynamically constructed emails.
- The use of SSL and SSL Certificates for communication through HTTPS.
- The use of SSH for secure file transfer using SFTP.
- Dangers associated with allowing user-uploaded files.
- Storage of sensitive data.
- Security issues specific to shared hosting environments.
- The importance of thinking like an attacker for designing secure applications (or, really, any secure system in general).
- Practices of full disclosure and “responsible” disclosure of vulnerabilities.

### Timeline / Schedule

Below is a preliminary timeline for the course. It is highly subject to change and many topics will be more spread out than shown.

Week	Date	Topics
1	Aug 20	About the course/go through syllabus. What's a web application? Why are we using PHP and MySQL? WAMP setup. Eclipse+PDT setup. First insecure PHP script. First secure PHP script. Security topic: Vulnerability research and disclosure. Security topic: Examples of the kinds of mistakes you won't make when you are done with this course.
2	Aug 27	Introduction to PHP. Introduction to HTML. Introduction to Cascading Style Sheets (CSS). HTML forms. Handling form input with PHP. Risks and rewards of independent vulnerability research.

		Security topic: Cross Site Scripting (XSS) overview and non-persistent XSS.
3	Sep 3	Labor Day. No class.
4	Sep 10	<p>What is client-side scripting?</p> <p>What is JavaScript?</p> <p>Introduction to JavaScript.</p> <p>OOP in PHP.</p> <p>User authentication.</p> <p>Sessions and session-scope data.</p> <p>Application-scope data.</p> <p>Cookies.</p> <p>Security topic: Password storage.</p> <p>Security topic: JavaScript and XSS.</p> <p>Security topic: Session hijacking.</p>
5	Sep 17	<p>What's a database and what's an RDBMS?</p> <p>Introduction to SQL.</p> <p>Primary keys in database tables.</p> <p>SQL statements: SELECT and INSERT.</p> <p>Introduction to PHPMysqlAdmin.</p> <p>Creating a database in PHPMysqlAdmin.</p> <p>Accessing a database through PHP.</p> <p>Security topic: SQL injection.</p>
6	Sep 24	<p>Catch up if we're behind schedule (we probably are already).</p> <p>SQL statements: UPDATE and DELETE.</p> <p>Using a database abstraction layer.</p> <p>Security topic: Persistent XSS.</p>
7	Oct 1	<p>Foreign keys and relating database tables.</p> <p>More advanced SQL.</p> <p>Security topic: XSS filter evasion</p> <p>Security topic: Cross Site Request Forgery (CSRF).</p>
8	Oct 8	<p>What is AJAX?</p> <p>AJAX libraries.</p> <p>Sending email from PHP.</p> <p>Security topic: Email header injection.</p> <p>Security topic: DOM-based XSS.</p>
9	Oct 15	<p>File submissions through HTML forms.</p> <p>Using PHP to store form-uploaded files.</p> <p>Image processing in PHP using GD.</p> <p>Security topic: Dangers with file uploads.</p>
10	Oct 22	<p>Catch up if we're behind schedule (we probably are still).</p> <p>Security topic: Extracting data from a database through SQL injection.</p> <p>Security topic: Browser history theft through XSS.</p> <p>Security topic: Intranet attacks through XSS.</p>

11	Oct 29	<p>What are frameworks and why use them?</p> <p>PHP frameworks and libraries:</p> <ul style="list-style-type: none"> <li>- PEAR</li> <li>- Zend Framework</li> <li>- CakePHP</li> </ul> <p>Using random scripts off the 'Net.</p> <p>Security topic: Trusting other people's code.</p> <p>Security topic: eval().</p>
12	Nov 5	<p>What's templating and why use it?</p> <p>Templating systems:</p> <ul style="list-style-type: none"> <li>- PHP itself</li> <li>- Smarty</li> </ul> <p>Accessing a site over SSL.</p> <p>E-commerce and credit card processing.</p> <p>Introduction to XML.</p> <p>XML handling in PHP using SimpleXML.</p> <p>Security topic: Protecting against automated bots.</p> <p>Security topic: Storage of sensitive data.</p>
13	Nov 12	Veterans Day observed. No class.
14	Nov 19	<p>Getting your site up on the Internet.</p> <p>Domain registration.</p> <p>Introduction to DNS.</p> <p>Finding a website hosting provider.</p> <p>Uploading files to your website.</p> <p>Hosting control panels.</p> <p>Being your own host: Running servers and VPSs.</p> <p>Getting an SSL certificate.</p> <p>Security topic: Sniffed plain-text traffic (ftp and http).</p> <p>Security topic: insecure hosts.</p>
15	Nov 26	<p>Catch up if we're behind schedule (we're probably halfway through the course).</p> <p>Error handling, logging and developer notification.</p> <p>Considerations for highly-trafficked websites.</p> <p>Server-side caching and static vs. dynamic delivery.</p> <p>Scalability.</p> <p>Security topic: Risks of improper error handling.</p>
16	Dec 3	<p>Odds and ends that were forgotten or there was no time for.</p> <p>Additional tools that you may find useful.</p> <p>Security topic: redirection pages and header injection.</p>