Lewis University Fall 2016

CPSC 24700 Web and Distributed Programming

Course Syllabus
MWF 2pm-2:50pm, Room AS-106-A

Instructor

Dr. Piotr Szczurek, Assistant Professor, Computer and Mathematical Sciences

Office location:	Room AS-114-A
Office hours:	Monday, Friday, 12pm-2pm, Wednesday, 1pm-2pm
Office phone:	815-588-7083
Email:	szczurpi@lewisu.edu

Course Description

Languages and technologies for programming and leveraging web-based computer services securely. Languages include PHP, Perl, JavaScript, Java, Ruby, CSS, and HTML5. Technologies include relational databases, web services, Hadoop, and cloud computing platforms. This course teaches students how to develop useful applications using a variety of distributed data and programming models. (3 Credits)

Prerequisites

CPSC 20000 (Introduction to Computer Science) or programming experience

Course Objectives

On the successful completion of this course students will be able to:

- Understand the ideas of distributed computing and the World Wide Web
- Create web pages with HTML5 and CSS
- Have and understanding of the best Web design practices
- Create dynamic and interactive web sites using JavaScript
- Have an understanding of XML and Web services
- Have an understanding of basic database concepts and make simple SQL queries
- Write PHP scripts to process forms and interact with databases
- Understand basic concepts of cloud computing
- Understand the Hadoop framework and writing MapReduce programs

Student Learning Outcomes

- Students will be able to develop dynamic Web pages using latest technologies
- Students will be able to utilize databases with server-side scripts
- Students will be able to understand the concepts of distributed computing

Relationship to Mission

"Lewis University is a Catholic University in the Lasallian Tradition. Our Mission is integrated into all aspects of University life, including this course. This course embraces the Mission of the University by fostering an environment in which each student is respected as an individual within a community of learners. In the spirit of the vision of Lewis University, the goals and objectives of this course seek to prepare students to be successful, life-long learners who are intellectually engaged, ethically grounded, socially responsible, and globally aware."

Textbook

Robert W. Sebesta, Programming the World Wide Web 8th Edition, Addison-Wesley, 2015
(ISBN 978-0-13-377598-3).

Course Requirements

Projects

You will have 5 programming projects due during the semester (see course outline on the last page for the due dates). The purpose of these assignments is for you to be able to apply the knowledge learned from the lectures and the readings in order to develop concrete programming solutions. The assignment descriptions will be posted on BlackBoard. You will have to submit your solution files through BlackBoard in the format specified in the assignment description. **No emailed submissions will be accepted**.

The assignments will be graded based on the following rubric:

Category	Unsatisfactory (0-1 points)	Satisfactory (2-3 points)	Excellent (4 points)
Webpage/ Program Correctness	• Program/script does not load/execute due to errors • Incorrect results for most or all input •Lack of appropriate interface content	• Program/script works and completes most tasks appropriately • Program/script fails to work for special cases	• Program/script runs and completes all required tasks • Handles special cases • Executes/loads without errors • Contains all specified content
Programming Style	• No name, date, or assignment title included • Poor use of white space (indentation, blank lines). • Disorganized and messy • No or few comments in the source code • Poor use of variables (improper scope/visibility, ambiguous naming).	• Includes name, date, and assignment title. • White space makes program fairly easy to read. • Well organized code. • Some comments missing in the source code or too many comments • Good use of variables (few issues with scope/visibility or unambiguous naming).	• Includes name, date, and assignment title. • Excellent use of white space. • Perfectly organized code. • Source code is commented throughout when needed • Excellent use of variables (no issues with scope/visibility or unambiguous naming).
Following Specifications	• Incorrect filenames • Incorrect class name or other specified identifier names • Source code organization different from requirements • Additional requirements not satisfied	• Correct filenames and class names •Few issues with other specified identifier names • Source code organization close to requirements • Some additional requirements not satisfied	• Correct filenames, class names, and other specified identifier names • Source code organization satisfies all requirements • All additional requirements satisfied
Webpage/ User Interface Design	• Lack of necessary styling of interface elements • Disorganized element placing • Website/program hard to use/navigate	• Most elements have the proper styling applied • Placement of elements is well organized • Website/program is easy to use/navigate	• All elements have appropriate styling applied • Placement of elements is well thought out and cleanly organized • Website/program is naturally easy to use/navigate

The project grade will be recorded as a percentage of the maximum number of points acquired. For example, if you received 3 points for both the correctness and programming style categories with an excellent for the following specification and design categories, you earn 3+3+4+4=14 points. This will be recorded as 14/16 = 87.5%.

NOTE: To receive any credit, the submitted project must be an honest attempt at a solution to the assignment. **Irrelevant or plagiarized submissions will receive zero points!**

Readings

Before each lecture, you are required to read the required chapter of the book as specified in the course outline section at the end of this syllabus.

Lectures/Labs

The course will consist of days with regular lectures with in-class examples and lab days in which you will work on exercises. You are required to attend all the lectures and participate in any lab activities. The main purpose of these activities is to confirm your understanding of the lecture material. The exercises will typically require you to write a program that executes some task. You are required to save these programs somewhere where you can access it immediately for next time. You should also make sure to complete any missed exercises and have your code ready before the next class.

Quizzes

There will be 6 quizzes posted on BlackBoard throughout the semester. The main purpose of these quizzes is to test your understanding of material covered by the lectures and the reading materials. You will take these quizzes on your own time through BlackBoard by the required due date (see course outline on the last page). Each quiz consists of 5-10 true/false, multiple choice, or fill-in-the-blank type questions. You will have exactly 20 minutes to complete each quiz. You may retake the quiz twice, but you may have different questions. Your grade will be based on the quiz with the highest score.

Exams

There will be two in-class midterm exams and a final exam. All exam scores will be normalized such that the highest score becomes 100% and the average score is not less than 80%. In case of the second midterm and the final, the exam will focus mainly on material covered since the last exam, but may include questions from material covered by previous exams. A study guide for each exam will be provided approximately a week before the exam is scheduled. The guides will allow you to get familiar with the type of questions you may be asked on the exam and see the topics that are included on the exam.

Grading Policies

Your final grade in the course will be based on the following:

Projects	40%
Quizzes	10%
Midterm Exam 1	15%
Midterm Exam 2	15%
Final Exam	20%

Final course letter grade will be determined using the following scale:

A	94-100	C	74-76
A-	90-93	C-	70-73
B+	87-89	D+	67-69
B	84-86	D	64-66
B-	80-83	D-	60-63
C+	77-79	F	59 and below

Course Policies

Class attendance is mandatory. It is your responsibility to know what goes on in class. Students must turn in all assignments and take all scheduled tests. Extensions for assignments and make-up tests will not be given. **Late assignments will not be accepted**.

Any form of plagiarism will result in a severe consequence for all parties involved. Please see the "Lewis University Copyright and Intellectual Property Guidelines" document, which is posted at http://www.lewisu.edu/academics/library/index.htm. Specifically for this course, the policy is that you are not allowed for anybody to look at your code or for you to look at someone else's code. I reserve the right to use code plagiarism detection software to check for plagiarism occurrences. In any situation in which I have reason to believe that you have copied program code or allowed anyone else to use your code, you will receive, at the minimum, a grade of zero for that particular assignment and possibly an F grade for the entire course. So while you may and in fact are encouraged to discuss the assignments with other students, you need to be careful to make sure you write your own code.

You are required to periodically check BlackBoard for any new announcements or changes to the course. I may post an announcement about the exam day and time or the programming assignment, etc. Such announcements will usually also go to your email, so if you check that at least once a day, you should be fine.

Additional Materials

Lecture videos, assignments and other course materials will be posted on the course's Blackboard site (http://lewisuniversity.blackboard.com) throughout the semester. All assignments must be submitted using the BlackBoard assignment area. You must make sure that your email address is recorded on Blackboard, because I use Blackboard's email facility to send a message to the entire class.

Getting Help

For many of you, this course will be the first time you are required to do significant amounts of programming. At times you may feel overwhelmed by the assignments or the lectures. You should know that this is normal. It is essential that in those cases, you immediately get help either by asking other students, arranging for tutoring sessions, or by notifying me.

You should know that I am always happy to help you and you should feel free to contact me at any time. You should start on the assignments early to see if you can understand what is required and how to get started. If, after spending some time, you cannot figure this out, let me know immediately and I will help you through it.

The best way to get in touch with me is through e-mail (listed on the first page). Unless something unexpected occurs, I will reply to you within 24 hours. Note that this means you cannot wait until the last day to ask a question about an assignment that is due. Although I try to answer questions as soon as possible, I cannot guarantee an immediate response. The alternative way to get in touch with me is to visit my office during office hours or call me.

How to Be Successful in This Course

1. Start on project assignments early
2. Ask for help if you are stuck
3. Come to class and do all in-class exercises
4. Read the textbook and review the lecture slides

Assistance

"If you have a disability that may require consideration by your instructor and you have not previously submitted documentation to the staff in the Leckrone Academic Resource Center (LARC), please make an appointment with Denise Rich, Director of Academic Support Services in LARC (x5593). It is recommended that you address this prior to the start of class or within the first week of class. If you need accommodations for successful participation in class activities prior to your appointment in LARC, you should provide information in writing to your instructor that includes suggestions for assistance in participating in and completing class assignments. It is not necessary to disclose the nature of your disability to your instructor. For more information about academic support services, visit the LARC website at: www.lewisu.edu/larc."

Course Outline (subject to change)

Week	Lecture Dates	Description	Reading	Assignments
1	8/29, 8/31, 9/2	Course overview, Introduction to Distributed Computing and the Internet	Ch. 1	
2	9/7, 9/9 No class on Monday (Labor Day)	Making web pages - HTML5	Ch. 2	
3	9/12, 9/14, 9/16	Making web pages - HTML5 (cont.) LAB #1 – 9/14 Styling web pages - CSS	Ch. 2, 3	Quiz 1 DUE 9/12
4	9/19, 9/21, 9/23	Styling web pages - CSS (cont.) LAB #2 – 9/23	Ch. 3	Project 1 DUE 9/23
5	9/26, 9/28, 9/30	Creating dynamic web pages - JavaScript basics	Ch. 4	Quiz 2 DUE 9/26
6	10/3, 10/5 No class on Friday (Fall Break)	Creating dynamic web pages - JavaScript basics (cont.) LAB #3 – 10/3 Using JavaScript with HTML	Ch. 4, 5	
7	10/10, 10/12, **10/14**	Creating dynamic web pages - Using JavaScript with HTML (cont.) LAB #4 – 10/12 **MIDTERM EXAM 1**	Ch. 5	Quiz 3 DUE 10/10
8	10/17, 10/19, 10/21	Creating dynamic web pages - Using JavaScript with CSS (cont.) LAB #5 – 10/21	Ch. 6	Project 2 DUE 10/21
9	10/24, 10/26, 10/28	XML, Web services	Ch. 7	
10	10/31, 11/2, 11/4	Ajax	Ch. 10	Quiz 4 DUE 10/31
11	11/7, 11/9, 11/11	Server side scripting – PHP LAB #6 – 11/11	Ch. 9	Project 3 DUE 11/11
12	11/14, 11/16, 11/18	Server side scripting - PHP (cont.) LAB #7 – 11/18	Ch. 9	Quiz 5 DUE 11/14 Project 4 DUE 11/20
13	11/21 No classes on Wednesday or Friday (Thanksgiving Break)	**MIDTERM EXAM 2**		
14	11/28, 11/30, 12/2	Introduction to Relational Databases LAB #8 – 12/2	Ch. 13	
15	12/5, 12/7, 12/9	Cloud computing fundamentals Hadoop / MapReduce Framework		Quiz 6 DUE 12/5 Project 5 DUE 12/9
16	See below for a schedule	**FINAL EXAM**		

FINAL EXAM SCHEDULE:
Monday, 12/12, 4:00pm-6:00pm