Web Software Development

CS 346 - Fall 2019

Instructor:	Hannah Hillberg
Email:	hillbergh@uwosh.edu
Office:	Halsey 217
Office Hours:	1:00-3:00pm MW
	3:00-4:00pm TuTh

Lecture:	Th	1:20-2:50pm	Halsey 202
Lab:	Tu	1:20-2:50pm	Halsey 101C

Course Information

An introduction to the tools for developing internet applications. Topics covered include: Internet history, the HyperText Markup Language, graphic images and manipulation, multimedia, programming in the JavaScript and PHP languages.
Credits: This is a 3 credit course.
Prerequisites: CS262 with a grade of C or better

Course Website: UWO Canvas (http://canvas.uwosh.edu)
You should check Canvas on a regular basis - it will contain lecture notes, handouts, assignments, announcements, and grades. I'll do my best to let you know when something new and important comes up, but it is your responsibility to check the website frequently for information that you might not get otherwise.

Required Textbook: COMP SCI 346: Web Software Development, Online book by Zybooks.
Subscription Instructions:
1. Create an account at learn.zybooks.com. You are required to use your uwosh email as your login. You may need to also provide your ID number.
2. Enter zyBook code: UWOSHCOMPSCI346HillbergFall2019
3. Subscribe. A subscription is $58 and will last until December 27, 2019. Students will be able to subscribe until December 2, 2019.

Labs

You will have weekly labs. Labs are generally short tasks that can be accomplished in a few hours. Labs will cover the material we cover in the previous class period (Thursday) to ensure you are staying current with the material, but you will have about a week to complete them. Labs will be posted shortly following the class period of the material covered (Thursday), and must be completed before the following ***class*** session (the following Thursday). Thus, you may start working on it soon after class, and it is due soon after the designated lab session. No late labs will be accepted. We will also be doing peer reviews of each other's labs, which will be due at the same time as the following lab (before the next class session.

Project

You will have one semester-long project to complete in this course. It can be a group project with teams of up to 3. Guidelines for the project will be provided near the beginning of the semester.

Course Grading Policy

Your final grade for this course will be based on four components, namely a project, labs, exams, and participation and challenge activities. Your overall numerical grade for the course will be computed as the weighted sum of the component grades using the following weights:

Component	**Weight**
Exams (3)	30%
Project	30%
Labs	30%
Participation and Challenge Activities	10%

Your letter grade for the course will be computed as follows:

Numerical Score	Grade	Numerical Score	Grade
>=92	A	72-78	C
90-92	A-	70-72	C-
88-90	B+	68-70	D+
82-88	B	62-68	D
80-82	B-	60-62	D-
78-80	C+	<60	D

While this overall grading scheme is fixed, I will be happy to discuss any issue you may have with individual grades. If you notice a mistake or have a question regarding a specific grade, please come and talk to me as soon as possible. Do not wait until the end of the semester to bring up grading issues.

***Tentative* Exam Dates:**

- Exam 1 – Tuesday, October 8
- Exam 2 – Tuesday, November 5
- Exam 3 – Tuesday, December 10

Academic Dishonesty
Academic dishonesty of any kind will not be tolerated. All assignments, labs, and exams are to be completed individually. While discussion of ideas and problems with fellow students is encouraged, all projects and labs must be done individually. In certain circumstances, code fragments from the instructor may be provided to eliminate tedious coding or to provide a common framework for all students. All other code must be original. Online resources may be used to help you understand the material, but you may not copy online code nor can you "borrow" code from other students, past or present.

Any suspected academic dishonesty will be dealt with on a case-by-case basis. Any clarification of what does or does not constitute academic dishonesty must take place *before* you turn in questionable work. For clarification on what constitutes academic dishonesty, contact me or consult the printed policy in the UWO Student Discipline Code, Chapter UWS 14.

Accessibility
It is the University's policy to provide, on a flexible and individual basis, reasonable accommodations to students who have documented disabilities that may affect their ability to participate in course activities or to meet course requirements. Students with disabilities must be registered with Services for Students with Disabilities or Project Success and receive an Accommodation Recommendation form to receive accommodations. Services for Students with Disabilities is located in 125 Dempsey Hall.

It is also the policy and practice of UW Oshkosh to create inclusive learning environments. If there are aspects of the instruction or design of this course that result in barriers to your inclusion, please notify me as soon as possible. You are also welcome to contact Services for Students with Disabilities at 920-424-3100 or dean1@uwosh.edu. For more information, visit the Services for Students with Disabilities website at http://www.uwosh.edu/deanofstudents/disability-services.

Course Outcomes
Students should be able to do the following upon successful completion of the course:

- Explain the key concepts associated with internet architecture that facilitates web application development.
- Explain the basic components of web architecture and describe how web browsers and servers work in tandem.
- Describe a web engineering framework to support the development of web-based applications.
- Apply the web engineering process to the development of a moderately complex web
- application.
- Utilize an integrated development environment to construct and deploy a web application.
- Construct and validate web pages.
- Design and implement client-side application logic with selected scripting languages.

- Design and implement server-side application logic with selected technologies.
- Design and implement the model-view-controller architecture for web-based applications.
- Design and construct web pages that interact with persistent storage.
- Read and apply web standards to the design and creation of web-based applications.
- Identify trends in web technologies and develop an evaluation strategy for assessing emerging web technologies.
- Work effectively with a small team of web developers to produce a web application.