CSE316 : Fundamentals of Software Development

Syllabus

Term: Fall 2025
Instructor: *Tony Mione*
Course Meeting Times / Location: *Mon & Wed, 9:00-10:20 AM – B204*
Office: *B425*
Phone: *+82 032-626-1226*
Email: *antonino.mione@sunykorea.ac.kr*
Office Hours:

Mon, Tue, Wed : 4:00 - 5:00 PM
Tue, Wed, Thurs : 11:00 AM - 12:00 Noon
(or by appointment) [B425]

Course Homepage:
https://www3.cs.stonybrook.edu/~amione/CSE316_Course/index.html

Brigthspace: ***https://mycourses.stonybrook.edu/d2l/home/2058753***

Recommended Texts:

- *Freeman, Eric, Robson, Elisabeth, Bates, Bert, and Sierra, Kathy*, "Head First Design Patterns", O'Reilly Media Inc, 2014. ISBN: 978-0-596-00712-6.
- *Accomazzo, Anthony, Lerner, Ari, Murray, Nate, Allsopp, Clay, Guttman, David, McGuinnis, Tyler*, "Fullstack React: The Complete Guide to ReactJS and Friends", Fullstack.io, 2020, ISBN: 978-0991344628
- *Hallie, Lydia, Osmani, Addy*, "Learning Patterns: Patterns for building powerful web apps with Vanilla Javascript and React", patterns.dev, 2022.
- *Other readings as specified*

Course Organization

This course is an introduction to systematic design, development and testing of software systems, including event-driven and Web programming, information management, software design and development fundamentals, and the application of these skills to the construction of large robust programs. Includes weekly assignments and projects, which provide students with experience in the practice of design and programming.

Course Objectives/Outcomes

Upon completion of the course, students are expected to possess:

- An ability to use event-driven programming in the construction of Web software.
- An ability to create a logical database design and implement it as a physical schema for a software application.
- An ability to apply algorithmic and software design/development principles to build large programs.
- An ability to design and implement a software testing plan.
- An ability to apply security principles to develop robust programs.

Prerequisites

- C or higher in CSE 216 or CSE 260
- CSE major

Grades and Evaluation

Your grade in the course will be based on the following work:
Attendance/Participation – 5% [25 points] – Based on attendance at class.
Assignments - 45% [225 points] - Programming assignments (4) which will provide practice in various development skills and software engineering practices. These assignments are to be completed individually.
Project – 25% [125 points] A development project that will help students practice and demonstrate comprehensive skills learned during the semester.
Midterm Exam – 10% [50 points] – a midterm exam to check understanding of the concepts presented in readings, lecture slides and demonstrations.
Final Exam – 15% [75 points] - A cumulative final exam will provide questions that will cover the key concepts taught during the entire semester.

Final Grade Calculation

The final grade is based on the accumulated points from all exams, assignments, and the semester project (with the entire class comprised of 500 points). Letter grades are given on the following scale:

Letter	Minimum Percentage	Minimum 'points'
A	93	465
A-	90	450
B+	87	435
B	83	415
B-	80	400
C+	77	385
C	73	365
C-	70	350
D+	67	335
D	60	300

Attendance

The range of topics covered in this course is extensive, and due to the limited lecture time, these topics are covered in an intensive manner. MOE guidelines also dictate that missing more than 20% of the classes in a course requires issuing an 'F' for the class. Therefore, attendance at all classes are mandatory in order to keep up, perform well, and receive a passing grade

- Attendance will be taken during each lecture session.
- A sheet is passed around that must be signed
    - ONLY Sign for yourself! Signing for anyone else is fraud and is a breach of academic integrity. If caught it will be reported!
    - **Make sure you sign the sheet during class! I cannot mark you present based on an email after class saying "I was at class and forgot to sign. Please mark me present."**
- ***If a student has over 20% unexcused absences, the final course grade will be an F.***

Note also that 5% of the course grade is based on attendance. So missing even 5 classes (not enough to fail the course), will lose perhaps 1% of grade which could be the difference between the grade you want and what you earn.

Re-grading

Requests for a review or regrade on an assignment, project, or test must be made within 1 calendar week of receiving the grade with feedback.

Important note: On assignments and/or the project: Regrading will ONLY be considered in cases where I may have run the code incorrectly. Upon correcting the code, if functionality is shown to work, the grade will be adjusted. I will NOT regrade on the basis of the assignment being 'vague' or failing to state a specific behavior. I will try to make the specifications as precise as possible but if something seems ambiguous, you should ask in class, in person during office hours, or by email. If there is enough confusion, I will announce more specific guidelines to the class either in person or by email.

Course Schedule

Following is a tentative schedule for the class topics:

Week/Day	Lecture Topics	Readings	Tests/Vids/Assgn
W1: 8/25	Course Overview / Web Overview Web Application Structure		Web Development 2023: A Practical Guide
8.27	Web Technologies - HTML/CSS	HTML Tutorial (and Reference) CSS Tutorial (and reference) CSS Layout Tutorial	HTML Tut Vid CSS Tut Vid
W2: 9/1	Javascript	Javascript Tutorial	JS Tut Vid A1G
9/3	Javascript		
W3: 9/8	The DOM, JQuery, JSON	How Web Apps Work: Browser, HTML, and CSS (Optional) JQuery Tutorial	JQuery Tut Vid [1st of 4] DOM Tut Vid [1st of 4]
9/10	Bootstrap		Bootstrap Video Tutorial
W4: 9/15	Typescript		Typescript tutorial
9/17	Software Process, Requirements	Good Requirements	A1D
W5: 9/22	CI/CD / Docker Intro		
9/24	React.js	React Tutorial	React Video Tutorial [Mosh] React Tutorial Video A2G
W6: 9/29	React.js		
10/1	Version Control – Github		Git Video Tutorial
W7: 10/6,10/8	Chuseok : No class		
W8: 10/13	NodeJS and Express	Understanding JavaScript Promises	Node.js Tut Vid
10/15	Midterm		
W9: 10/20	API Design		Proj Given
10/22	Information Mgmt/Relational Model		
W10: 10/27	Database Design/Normalization		A2D A3G / Proj Proposal Due
10/29	SQL / Using SQL from Javascript	SQL tutorial [W3 Schools] SQL tutorial Query Documents Tutorial	SQL Beginner Course MySQL w/Node.js Tutorial
W11: 11/3			
11/5	Software Testing : Unit Testing/Integration Testing / CI_CD		Project Req/Test Plan Due
W12: 11/10			
11/12	NoSQL/MongoDB/GraphQL	Using Database with Mongoose Tutorial 6 Rules of Thumb for MongoDB Schema Design	Guide to NoSQL MongoDB A3D A4G
W13: 11/17	Intro to Web Security		
11/19	Web Security – Authentication and Authorization	Sign-in form best practices Sign-up form best practices 13 best practices for user accounts	
W14: 11/24	JSONWeb Tokens Certificates		
11/26	Software Design / UML / Design Patterns : Creational Patterns	Freeman: Chap 1, 4, 5	Proj Due
W15: 12/1	Design Patterns : Structural Patterns	Freeman: Chap 3, 4, 7	A4D
12/3	Design Patterns : Behavioral Patterns	Freeman: Chap 6, 8, 9, 10	
12/8	Final Exam [9:00-11:30]		

Academic Dishonesty

You may *discuss* the homework assignments with anyone you like, however each students' *assignment (including coding)* which they submit must be their own work, and **only** their own work. **Any evidence that source code or solutions have been copied, shared, or transmitted *in any way* (this includes using source code downloaded from the Internet (i.e. Chegg), source code written by an AI like ChatGPT, or code written by other students in previous semesters!) will be regarded as evidence of academic dishonesty and will be reported to the administration.**

Guidelines for Assignments

When working on programming assignments, you must work only with others whose understanding of the material is approximately equal to yours. In this situation, working together to find a good approach for solving a programming problem is cooperation; listening while someone dictates a solution is cheating. You must limit collaboration to a high-level discussion of solution strategies, and stop short of actually writing down a group answer. Anything that you hand in, whether it is a written problem or a computer program, must be written in your own words. If you base your solution on any other written solution, you are cheating.

Guidelines for the Term Project

The term project will be developed by teams of 2 students so, obviously, it is okay to work together and share code within your team for the final project.

Guidelines for Taking Exams

When taking an exam, you must work completely independently of everyone else. Any collaboration here, of course, is cheating. All examinations will be closed-notes and closed-book. No electronic devices of any kind will be permitted to be used during exams. All cell phones must be silenced or turned off during exams. You will be allowed one (1) sheet of notes, both sides (8.5 x 11 or A4). **Any additional note sheets found will be confiscated and you may be reported to academic administration for a violation of academic integrity which could cause you to fail the course.**

General Guidelines

Be advised that any evidence of academic dishonesty will be treated with utmost seriousness. *We do not distinguish between cheaters who copy others' work and cheaters who allow their work to be copied.*

If you cheat, you will be given an F on the assignment. Any incidence of cheating will be reported to Academic Affairs. If you have any questions about what constitutes cheating, please ask.

Students with Disabilities

If you have a physical, psychological, medical or learning disability that may impact your course work, please let the instructor know. Reasonable accommodation will be provided if necessary and appropriate. All information and documentation are confidential.

Critical Incident Management

The University expects students to respect the rights, privileges, and property of other people. Faculty are required to report to the Office of Judicial Affairs any disruptive behavior that interrupts their ability to teach, compromises the safety of the learning environment, or inhibits students' ability to learn.