

CSC648.848.05 SOFTWARE ENGINEERING

Department:	Computer Science	Semester:	Fall 2024
Course:	CSC648-848	Section:	05
Modality:	In-person Instruction	Units:	3
Instructor:	Jose Ortiz	Email:	jortizco@sfsu.edu
Lecture Time:	Thursday 4:00PM - 6:45PM	Room:	HSS Building 380

Course Website: All the course materials will be available through Canvas. <https://canvas.sfsu.edu/>

Office Hours: Tuesdays and Thursdays from 2:00pm to 3:00pm (Online). No appointment is needed, but an email required. Important details and links are posted on Canvas

Prerequisites:

- CSC 648: CSC 413 and CSC 317 with grades of C or better.
- CSC 848: Graduate standing*
 - * Conditionally classified grads who had CSC413 or CSC317 as a condition must clear this first, no concurrent enrollment allowed
- All the prerequisites listed above are strictly enforced. Concurrent enrollment is not allowed.
- International visiting students - contact instructor

Catalog description:

Practical methods and tools for SW Engineering, including organizational teamwork. Course is paired with CSC 848. Students who have completed CSC 848 may not take CSC 648 for credit.

Expanded Description:

- Introduction and motivation for Software Engineering
- Overview of several basic SE methodologies with emphasis on Extreme, Iterative and Incremental Development and User Centered Design
- Usability and UI design principles and practice
- Basic components of SW Engineering process: Planning; Requirements and Specifications; Iterative Design, Rapid Prototyping, Mockups
- Software Design; Coding and documentation techniques (high level only)
- SW Engineering related to Web application development
- Open source SW development and management (NEW)
- Software Configuration Management, Delivery, Installation, and Documentation
- System Design, Database Architecture, and Network Application Protocols
- Software Metrics, Performance and Usability Measurements
- Software QA and Testing

- Software Maintenance
- Project Management issues
- Teamwork and Communication as integral part of SW Engineering
- Issues related to global SW engineering
- Basics of IP, licensing, digital rights management and copyright
- SW Engineering ethics
- Real life examples and cases from instructor and students
- Guest and student presentations
- Final team project including several milestones, interaction with instructor, and final demo and delivery.

Course Objectives and Role Program:

This is a capstone course and paired class with graduate version CSC 848 where students integrate and apply all the knowledge they attained in the program. It offers comprehensive and advanced coverage of practical team oriented methods and tools of full life-cycle SW development. Special emphasis will be on Iterative, Incremental, Agile and User Centered Design team practices and on global SW engineering. Cornerstone of the class is student team project (in teams of 5-6 students) where students engage in a team project in order to experience and practice key aspects of full SW engineering life-cycle in setting that simulates small SW company. Class teaching is coordinated with student team project needs.

Learning Outcomes:

- Knowledge of modern Agile SE engineering methods/processes and best practices for team SW development, and their appropriate application
- Knowledge of basic components and tools for team full life-cycle SW development from use cases and initial specifications to design, implementation, testing, evaluation as well as code management and deployment on cloud systems
- Knowledge of best practices for the design, implementation and evaluation for delivering easy to use and maintainable SW
- Knowledge of basic SW dependability metrics, quality metrics, and basic architectural models
- Ability to constantly iterate and re-prioritize goals based on user needs, budget, schedule and resources
- Development of significant SE teamwork and project based experience, as close as possible to real life

Method of Evaluation:

- **Final Exam:** 30% (multiple choice questions testing basic knowledge of SW Engineering concepts learned in class)
- **Team Project and Technical Documentation:** 70% (Design and development of a SW application by student teams.)
 - This project is divided into five milestones that are graded using the following metrics:
 - * SE Process (how well the team followed SE best practices, M1-M5) 35%
 - * SE Product and its prototypes (how good the product the team developed is) 35%

Tentative Schedule:

Week	Topic	Assignments
1	Intro to the Course, Teams and Repositories	Milestone 1 and TAP Assigned
2	Use Cases and High Level Requirements	Tech Stack Approval Due
3	Competitive Analysis and Product Validation	
4	Software Processes, XP, and Agile Methodologies	Milestone 1 and TAP Due, Milestone 2 and VP Assigned
5	System Design and Network Application Protocols	
6	Database Architecture	
7	Product Deployment and Containerization	
8	Introduction UI/UX and UCD Design	Milestone 2 and Prototype 2 Due, Milestone 3 Assigned
9	Advanced UI/UX and Human Interaction	
10	Integration of AI tools and AI Ethics	
11	UX/UI and Localization Testing	Milestone 3 and HP Due, Milestone 4 and BP Assigned
12	QA Testing	
13	Intellectual Property and Security	
14	Thanksgiving Recess. No Classes	
15	Demo Presentation	Milestone 4 and BP Due, Milestone 5 and FP Assigned
16	Final Exam Review	Milestone 5 and FP Due
17	Final Exam (Thursday, December 19, 5:00PM - 7:00PM)	

Notes

- M[X] = Milestone [X]
- UX/UI = Usability and User Interface (Front-end Development)
- UCD = User Centered Design Process
- QA = Quality Assurance
- SW = Software
- AI = Artificial Intelligence
- TAP = Throw Away Prototype
- VP = Vertical Prototype
- HP = Horizontal Prototype
- BP = Beta Prototype
- FP = Final Product
- XP = Extreme Programming

Grades: The final letter grade for the course will be computed using the following distribution. I reserve the right to grade more leniently than this, but I will not grade more strictly.

Highest	Lowest	Letter Grade
100	93	A
92.99	90	A-
89.99	87	B+
86.99	83	B
82.99	80	B-
79.99	77	C+
76.99	73	C
72.99	70	C-
69.99	67	D+
66.99	63	D
62.99	60	D-
59.99	0	F

Final Group Project

Objective: By working on developing a SW application in a team setting, students will experience and exercise all steps in full SW Engineering life-cycle simulating real SW company environment. All teams will choose/work in their own application. Usually a tree tier media rich WWW application which will be approved by the instructor in order to force the teams to engage in full cycle of SW development - from initial design to delivery.

Organization: Teams will be selected by instructor based on mandatory student skills and experience survey (used only for team building) in order to build teams with approximately equal skills and experience profile. Team roles will be rotated for every milestone, and each team will have team-lead, front-end lead, back-end lead, database lead (usually the back-end lead fills this role) and GitHub master. Instructors will serve as client, CEO, CTO and/or coach depending on the milestone.

SW tools and environment: Students will have some choice in deciding specific SW stack and cloud systems from a set recommended by instructors (first team milestone). All SW and tools as well as cloud servers for deployment will be open source and free

Deliverable: These include a) Final documents from all the required milestones from the team project (after instructors feedback has been incorporated), b) code and code management repository (e.g. GitHub; and c) demo of the final product to the whole class. These materials can also serve as student portfolio for job search.

Grading Policy:

- **Milestones:** All team members will receive the same grade, except for those who do not meet participation requirements. The team grade is determined by two components: the SE Process (adherence to best practices in software engineering) and the SW Product (quality of the team's software deliverable). Grades assigned to the initial version (version 1) of each milestone are provisional. These grades can be improved based on the grade awarded to the revised version (version 2), provided that the instructor's feedback has been appropriately addressed.

- **Prototypes:** Unlike milestones, grades for prototypes are final and will not be subject to revision unless a grading error by the instructor or grader is identified.

Class attendance and student responsibilities: Full participation in all team activities and attendance at all team sessions are mandatory. Students are required to read and respond promptly to all class-related emails from the instructor and their team members. Student participation and engagement in the team project will be regularly monitored through team meetings with the instructor, communications with team leads, and activity in code management systems such as GitHub. If a student's participation is found to be inadequate, their grade will be adjusted downward from the team's overall grade in that milestone, prototypes or/and the final version of the software project.

All other mandatory class policies will be introduced in the class in introductory slides during the first lecture of the semester

Textbook: In this class, you won't need a textbook, as technology in this industry evolves rapidly, making books quickly outdated. I will provide all the resources you need to succeed in this course. All the resources and tools provided are industry standards, widely used by the most popular and successful companies in the software industry. We also review and update the course materials every semester to ensure we stay ahead of the latest technological changes.

Course Policy:

- **Attendance:** Students are expected to attend all scheduled lectures in this course.
 - Attendance is vital for understanding course material, participating in class discussions, and achieving academic success.
 - Participation in lectures is not limited to mere attendance but also includes active engagement in class discussions, asking questions, and contributing constructively to the learning process.
 - Students who miss lectures without extenuating circumstances and compelling documentation will lose participation points. These points may contribute to the overall course grade and serve as a reflection of their commitment to the learning process.
 - Students missing a lecture are responsible for what they miss in class.
 - Students missing a lecture must notify their instructors at least 24 hours before the lecture when they encounter extenuating circumstances that may affect their attendance. Communication should be clear, timely, and respectful.
- **Milestones:** this class has five mandatory milestones assignments. The following are their policies:
 - In this course late work will not be accepted under any circumstance. Teams will be given more than enough time to complete and revise their work by its deadline.
 - All your work in this class must be uploaded to the team's Github class repository to be considered for grading.
 - All assignments are due on the designated due date, as specified by the instructor, and it is the student's responsibility to keep track of all assignment due dates and plan accordingly.
 - Teams in this class are fully responsible for their submissions. If a blank or incomplete assignment (missing components or sections) is submitted, it will be graded accordingly, and the team will forfeit the right to request a revision. We enforce these rules because, in the real software industry, meeting delivery dates agreed upon with clients is crucial and directly impacts the reputation of the company or team responsible for the product or milestones. Always double-check your submissions to ensure that what you intended to upload is indeed what was uploaded.

- Teams will turn in two versions of the same milestone; version 1 and 2. The first version will get a temporal grade, the second version (after all the revisions from the feedback provided are applied) will get a final grade. Versions 2 of a milestone are due by the deadline of the Version 1 of the next milestone. For instance, M2V2 will be due by the deadline of M3V1.
- **Prototypes:** Students in this class are required to implement incremental prototypes that align with the technical documentation created for the corresponding milestone. Each prototype represents a step in the incremental development of your final product.
- **Exams:** Generally, there will be no make-up exams and no incomplete grades given. If any of the scheduled exam dates are in conflict with your religious observances, you must notify your instructor during the first two weeks of the semester. Drop and withdrawal from courses are student responsibilities. Exams will be taken in a paper format, in the classroom, and proctored by the instructor. Exams are 30 multiple choice problems with some True or False problems.
- **Class Notes:** Students may not capture audio, photos or video from class sessions on their own devices without the explicit permission of the instructor and everyone present, unless part of a DPRC-authorized accommodation;
- **Office Hours Policies:** Office hours are a valuable resource provided by instructors to support students' learning and offer additional assistance outside of regular lectures. They are intended to foster a deeper understanding of course materials, answer specific questions, provide guidance on assignments, and address any concerns related to the course. It is important to emphasize that office hours are not intended to serve as a replacement for missed lectures. Instructors do not typically use office hours to re-lecture or repeat the content covered during a missed class.
- **Final Grades:** I create a lot of opportunities for students to do well in this course, and all posted grades are final unless there was a verifiable mistake in the grade calculation. Fairness demands, and university rules, require that all students be marked according to the same standards, so that all students receive the marks they earned. I can't just raise your final grade for unjustifiable reasons, so I will not respond to any such requests.
- **Incomplete Grades:** The instructor does not assign incomplete grades. Students are responsible for dropping the course, requesting a withdrawal, or selecting the Credit/No Credit option if they are unable to meet the minimum requirements for the course.
- **Syllabus Updates:** This syllabus and schedule are subject to change. The official syllabus will be maintained at the course website. The instructor will make an announcement every time a policy from the syllabus is updated during the semester. It is the student responsibility to check on the site frequently and even when revisions are made while you are absent.
- **Academic Integrity:** Any form of cheating or plagiarism will incur very serious consequences. Students may not post any course materials to any third-party sites (such as Chegg) or post any recordings, screenshots, audio or chat transcripts in any setting outside the class, and that violations of this are subject to student disciplinary action. In addition, the use of artificial intelligence (AI) or other automated writing tools to complete assignments is strictly prohibited in this class. In addition, any evidence of the use of AI will be considered a violation of academic integrity and will be met with a failing grade for the assignment and may result in further disciplinary action. It is the responsibility of each student to ensure that all work submitted for this class is their own original work, written and completed without the use of AI or other automated writing tools. Any instances of cheating, deceit, fabrication, forgery, plagiarism, unauthorized altering of records or submitting false documents, unauthorized collaboration, unauthorized submission of work previously given credit, or other forms of academic misconduct will be assigned a grade penalty, likely an F or a grade of zero. Failing one or more assignments or examinations for reasons of academic integrity violations may result in a final

class grade of F. Students may not withdraw from classes in which they have committed academic misconduct. Consequences for violations of academic integrity may exceed an F on the assignment, examination, or class as determined by the Academic Integrity Review Committee. Members of our academic community have a responsibility to develop an awareness of academic integrity, to cultivate skills to realize honesty in academic and community work, and to sustain actively academic honor as a core value of our community. Students are expected to engage in behaviors that reflect well upon the university. In addition to attending to one's own actions, the Standards for Student Conduct require that students who witness academic dishonesty notify their faculty/instructor, department chair, or the Office of Student Conduct. Supporting academic integrity enhances the reputation of the University and the value attributed to degrees awarded by the University.

– To learn more about the university policies regarding academic integrity and other policies:

1. Academic Dishonesty: <https://conduct.sfsu.edu/academic-integrity>
2. SFSU Code of Student Conduct: <https://conduct.sfsu.edu/standards>
3. Plagiarism: <https://conduct.sfsu.edu/plagiarism>
4. CS Department Student Policies: <https://cs.sfsu.edu/grads/student-policies>

- **Students with disabilities:** Students with disabilities who need reasonable accommodations are encouraged to contact the instructor. The Disability Programs and Resource Center (DPRC) is available to facilitate the reasonable accommodations process. The DPRC is located in the Student Service Building and can be reached by telephone (voice 415-338-2472, video phone 415-335-7210) or by email (dprc@sfsu.edu).

SF State fosters a campus free of sexual violence including sexual harassment, domestic violence, dating violence, stalking, and/or any form of sex or gender discrimination. If you disclose a personal experience as an SF State student, the course instructor is required to notify the Title IX Coordinator by completing the report form available at <http://titleix.sfsu.edu>, emailing vpasem@sfsu.edu or calling 338-2032. To disclose any such violence confidentially, contact: The SAFE Place - (415) 338-2208; http://www.sfsu.edu/~safe_plc/ Counseling and Psychological Services Center - (415) 338-2208; <http://psyservs.sfsu.edu/> For more information on your rights and available resources: <http://titleix.sfsu.edu>

- **Religious and cultural holidays policy** The religious and cultural holidays policy (S09-212) states that the faculty of San Francisco State University shall accommodate students wishing to observe religious and cultural holidays when such observances require students to be absent from class activities. It is the responsibility of the student to inform the instructor, in writing, about such holidays during the first two weeks of the class each semester. If such holidays occur during the first two weeks of the semester, the student must notify the instructor, in writing, at least three days before the date that they will be absent. It is the responsibility of the instructor to make every reasonable effort to honor the student request without penalty, and of the student to make up the work missed.
- **Academic calendar policy:** The Academic Calendar can be found at <https://webapps.sfsu.edu/public/webcal/acadcalendar>