

## CS 4311: Software Engineering II Fall 2021.

Class Time: TR 1:30-2:50 pm; Chemistry Computer Scildg G.0208  
Instructor: Dr. Oscar A Mondragon  
Email: oamondragon@utep.edu  
Office Hours: W 9:00-10:00 am  
Mangmt Meeting: M 11:00 am – 1:00 pm  
TA: Jose E Soto jesoto4@miners.utep.edu

**Prerequisites:** CS 4310 with a grade of C or better. You should be in your last two semesters as an undergraduate.

**Text Books:** Wirfs-Brock, R. Wilkerson, and L. Wiener, *Designing Object-Oriented Software*, Prentice Hall, 1990.

**Resources:** Shari Lawrence Pfleeger, Joanne Atlee, *Software Engineering: Theory and Practice*, 4<sup>th</sup> Edition, Prentice Hall, 2009.

### COVID-19 PRECAUTION STATEMENT

Please stay home if you have been diagnosed with COVID-19 or are experiencing COVID-19 symptoms. If you are feeling unwell, please let me know as soon as possible, so that we can work on appropriate accommodations. If you have tested positive for COVID-19, you are encouraged to report your results to [covidaction@utep.edu](mailto:covidaction@utep.edu), so that the Dean of Students Office can provide you with support and help with communication with your professors. The Student Health Center is equipped to provide COVID-19 testing.

The Center for Disease Control and Prevention recommends that people in areas of substantial or high COVID-19 transmission wear face masks when indoors in groups of people. The best way that Miners can take care of Miners is to get the vaccine. If you still need the vaccine, it is widely available in the El Paso area, and will be available at no charge on campus during the first week of classes. For more information about the current rates, testing, and vaccinations, please visit [epstrong.org](http://epstrong.org).

**Course Description:** Methodologies, approaches, and techniques associated with software design, implementation, and testing of a software system; other topics include cooperative teamwork, project management, and documentation; second semester of a two-semester capstone project in which students design and implement a real-world application specified in CS4310.

**Goals:** To provide computer science students with software analysis and design techniques that result in the development of maintainable and reliable software that meets the customer's needs. In practical terms, this course is meant to provide students with an approximation of a real-world experience of software development.

### **Topics:**

- Configuration management
- Software design analysis techniques

- High-level software design
- Software design specification
- Software implementation
- Software validation and verification
- Software process improvement

## **Learning Outcomes**

### **Level 1: Knowledge and Comprehension:**

Level 1 outcomes are those in which the student has been exposed to the terms and concepts at a basic level and can supply basic definitions. Upon successful completion of this course, students will be able to:

- a. Articulate design principles, including cohesion and coupling, encapsulation, and information hiding.
- b. Describe software design concerns related to maintenance.
- c. Describe different software architectural styles, such as blackboard, event systems, layered system, and pipe and filters

### **Level 2: Application and Analysis:**

Level 2 outcomes are those in which the student can apply the material in familiar situations, e.g., can work a problem of familiar structure with minor changes in the details.

Upon successful completion of this course, students will be able to:

- a. Apply different diagramming techniques for an architectural design.
- b. Apply different textual and diagramming techniques for producing a detailed design of a system.
- c. Relate general strategies to identify and implement appropriate software architecture styles (including distributed and cloud) for the system under development.
- d. Relate general strategies for creating a design of a system.
- e. Distinguish between the different levels of cohesion and coupling.
- f. Use software development and maintenance tools, such as software documents creation and editing tools, GUI generators, comprehension and analysis tools, supporting activities tools (configuration management tools), verification and validation tools, and security vulnerability analysis tools.
- g. Describe differences between unit, integration, system, and acceptance testing.
- h. Apply black testing techniques to develop test cases for a variety of test coverages.
- i. Apply white-box testing techniques to develop test cases for a variety of test coverages.
- j. Apply static and dynamic techniques to analyze non-functional properties, including common security vulnerabilities such as password weakness, over/underflows, and race conditions.
- k. Engage in self-directed study to learn new techniques and tools for software design, implementation, and/or testing.

### **Level 3: Synthesis and Evaluation**

Level 3 outcomes are those in which the student can apply the material in new situations. This is the highest level of mastery. Upon successful completion of this course, students will be able to:

- a. Conduct a technical review of software design, implementation, and V&V.
- b. Create and implement a software configuration management plan.
- c. Create an architecture design and a detailed design for a software system.
- d. Construct software from a detailed design.
- e. Develop a test plan for a software system.
- f. Demonstrate an ability to orally present a software design and implementation.
- g. Compose software design-related documents that are grammatically correct and technically sound.
- h. Apply effective techniques for collaboration and problem-solving within a team.

**Course Objectives:** This course will provide students with the fundamentals of the design and implementation of software systems, emphasizing the principles and methods used to develop and verify software systems. On completion of CS4311, students should be able to discuss and demonstrate approaches, techniques, or methods for creating high-level and detailed designs, hierarchical factoring of object systems, develop verification plans, be familiar with IEEE standards, and have experience in planning and implementing a large project.

**Examinations:** Examinations are assumed to be closed book, closed notes, in class, unless otherwise specified. Make-up examinations are not given. If you miss an examination for a legitimate reason (such as illness, death in the family, participation in a college sponsored activity), then your comprehensive final examination will be counted extra to compensate for the missed work. If you miss an examination without a legitimate reason, a grade of 0 will be recorded for that examination. If you know you will be missing an exam date due to a college sponsored activity, you may arrange to take the exam in advance.

**Teams:** We believe that the ability to work with other software developers is essential. Therefore, students will be required to work effectively in teams throughout the semester. Students will be assigned to lead teams (role manager) for particular work products and/or project development support activities. The lead (role manager) is responsible for: identifying task to build assigned work product and/or project support activity; facilitating the assignment of tasks; and integrating the work product. Team leader and planning manager ensure the team meets the deadlines; monitor work progress; verify and validate work products, provide project status to management.

## Empowering self-directed teams (policy):

Students will work on teams. Students should develop soft skills to properly and effectively work as team members. Team members are expected to have a positive attitude, deliver quality work on time, participate in peer reviews, and support other team members. Because you are working in a self-managed team, your team has empowerment and it is expected that the team should use it professionally and ethically. If a team member starts not working, performing poorly, breaking ground rules, or being irresponsible, the team must let the member know about the problem and **acknowledge a warning as soon as possible** (verbally during a meeting or by sending an email). If the team member is continuously breaking ground rules and negatively affecting the team, the team must let the member know about the problem and acknowledge a yellow card by sending him/her an email and copying the instructor (**show a yellow card**). The team member should let the team know a strategy to improve performance and request help from the team and instructor if needed. If the team member continues performing poorly or irresponsible negatively affecting the team, the team has the empowerment to remove him/her from the team by showing him/her a red card (**show a red card**). *A student can only receive credit for the team project work if he/she is a contributing member of a team.* Yellow and red cards are designated by consensus by all the team members but the affected party. The team leader shall send an e-mail to the instructor and all team members including the affected team member with a brief explanation of the circumstances.

**Grading Summary:** Final grades in this course will be determined by combining grades for two components: exams and projects/homework. The sequence described below will be used to determine your final grade.

1. The exam average will be computed by summing 20% of midterm 1, 30% of midterm 2, and 40% of the comprehensive final., and 10% of quizzes.
2. The project grade will be computed by combining the grades for the Class, Responsibility, Collaboration, Hierarchy, High-level Design, Protocols, Prototype, Subsystem design, Test Plan, Configuration Management Plan, Implementation, Testing Report, and Final Presentation. Project grades may be strongly influenced by the Guidance Team's assessment of a student's contribution to his/her team.
3. For each student whose grade has not yet been assigned, a final score will be computed by summing 45% of the exam grade, 45% of the project grade, and 10% of the class participation (team survey evaluation and management team evaluation). The final grade in the course will nominally be assigned according to the scale A: 90-100, B: 80-89, C: 70-79, D: 60-69.

### CS 4311 Software Engineering II Project Grade Distribution

The project grade weights 45% of the final SE II grade. Here is the breakdown of how the project grade is being calculated.

	Deliverables	Weights
1.	Software Configuration Management Plan	5%
2.	Prototype Demos	10%
3.	CRC	10%
4.	Subsystem	5%
5.	Protocol	5%
6.	SDD	20%
7.	Implementation	20%
8.	Test Plan	10%
9.	Final Presentation	15%
	<b>Total:</b>	<b>100%</b>

**General policies:** *Use of electronics in class:* UTEP supports the use of technology for learning. Laptops can be an asset to some students and help them in their note-taking and learning. Students will be allowed to use laptops in this class provided they follow the rules described below. Failure to follow these guidelines will result in suspension of laptop privileges in class.

- Charge your laptop batteries fully before coming to class.
- Set your laptop volume control to mute or off before coming to class.
- Keep your laptop closed during presentations and other specific in-class activities.
- Do not engage in unauthorized communication or entertainment (web surfing, instant messaging, chat room chatting, DVD viewing, music playing, game playing, etc.) during class unless it is part of the lesson.

Cell Phones are nearly universal in our modern culture. Under normal circumstances, however, you are expected to refrain from using cell phones during class time. Your cell phone should be set to silent mode or turned off before class. Under no circumstance will you be allowed to use text messaging (sending or receiving) or web browsing features of your phone while you are in class. In an emergency, there may be a genuine, rare need for you to use a cell phone during class time. In this case, you will excuse yourself from class and leave the classroom to answer an incoming call. You will not be permitted back into the classroom for the remainder of class. It is never permissible to place an outgoing call while you are in class. Other Wireless Communications Devices are not allowed in class.

*Time:* This course is time intensive. You must start assignments early and work steadily to be successful.

*Grading errors:* We are only humans. We can and will make mistakes. You have one week after graded material is returned to the class to rectify any grading errors or to argue for additional credit. After the week has passed, no changes in grades will be made.

*Class Attendance and Participation:* As a college student, you have the freedom to choose whether or not to attend class. However, in this course we are committed to cooperative techniques, which can only work if students attend regularly and on-time. Part of what we

are encouraging in this course is the establishment of professional behavior. Therefore, we will take attendance. **Your final grade will be lowered by one point for each unexcused absence above three.** For the purposes of this class, you will be counted as absent if you are not present when we take attendance. If you feel that you must interact with people using cell phones, PDAs, Blackberries, email, twitter, chat, or any other electronic means, you are free to do so outside of class. If we find you doing these things in the classroom, we will ask you to leave, and to avoid disturbing the rest of your classmates, you should not return until the start of the next class.

*Office hours:* We expect you to meet with us outside of class time to discuss the course material. You and your team may contact the Guidance Team by office phone, in person, or email to arrange a suitable time to meet.

Let us make one more point here: we are available to *assist* you in solving problems, not to *think or do* work for you. Office meetings are for helping you by clarifying material and for assisting you with problems you are encountering. It is not for repeating things you missed when you skipped class. You should come to office appointments prepared. The harder you work at it, the harder we will work to help you.

*Incomplete:* Students receive a grade of Incomplete only under extraordinary circumstances: when they have substantially completed the course work with a passing grade, but cannot finish the course for a legitimate reason. Legitimate reasons include severe illnesses and debilitating accidents. Class or workloads that are too demanding are NOT legitimate reasons.

*Lecture material:* You are expected to preview lecture material BEFORE coming to class, including reading the assigned material from the text book. Some material may not be in the text: references will be provided, but you are responsible for the content

*Academic dishonesty:* Cheating is defined as submitting work under your name that was not done entirely by you for individual assignments or by your team for team assignments. (This includes taking programs from the web or cutting text from web pages and pasting them into documents, even if the source is cited). Cheating will not be tolerated--those caught cheating will be reported to the Dean of Students. You should be aware of the Standards of Conduct posted at [http://www.utep.edu/vpfa/student\\_affairs/student/studindex/htm](http://www.utep.edu/vpfa/student_affairs/student/studindex/htm).

*Disabilities:* If you have a disability and need classroom accommodations, please contact The Center for Accommodations and Support Services (CASS) at 747-5148, or by email to [cass@utep.edu](mailto:cass@utep.edu), or visit their office located in UTEP Union East, Room 106. For additional information, please visit the CASS website at [www.sa.utep.edu/cass](http://www.sa.utep.edu/cass).

*HELP:* Please confer with us if you experience difficulty with any aspect of the course – we are here to help you to learn. If you request help via email, make sure to write HELP in the subject line. Call us. Send us mail. Ask us questions.