

CS 4311: Software Engineering II

Spring 2022

CRN: 24892

Class Time: TR 3:00 PM -4:20 PM in CCSB G.0208

Website: Blackboard

Instructor: Yoonsik Cheon (x-8028, ycheon@utep.edu); office hours: TR 4:30 PM – 5:20 PM in CCS 3.0606)

TA: Jose Soto (jesoto4@miners.utep.edu; office hours: TBA in CS dungeon)

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

Textbooks:

Frank Tsui, Orland Karam, and Barbara Bernal, *Essential of Software Engineering, 3rd Edition*, Jones & Bartlett Learning, 2014. Ebook is available for free from the utep.edu domain (use VPN at home):
<https://learning.oreilly.com/library/view/essentials-of-software/9781449691998/>

R. Wirfs-Brock, R. Wilkerson, and L. Wiener, *Designing Object-Oriented Software*, Prentice Hall, 1990 (or R. Wirfs-Brock and A. McKean, *Object Design: Roles, Responsibilities, and Collaborations*, Addison-Wesley, 2002)

Resources:

M. Seidl, M. Scholz, C. Huemer, and G. Kappel, *UML@Classroom: An Introduction to Object-Oriented Modeling*, Springer, 2015. Free ebook through UTEP library (use VPN at home)

Scott W. Amber, *The Elements of UML 2.0 Style*, Cambridge University Press, 2005

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

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.

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 for particular assignments. The lead is responsible for: coordinating meetings and completing the meeting records; assigning tasks and recording task assignments; collecting documentation of individual work, including rough drafts; ensuring the team meets the deadlines; organizing the team notebook; maintaining backup copies of work; and binding

deliverables.

Examinations: There will be one mid-term exam and the final. The mid-term exam will take place during the regular class session, and the final exam will take place on the date specified by the university. Makeup exams will be given only when you have unusual circumstances, such as incapacitating illness or presenting a research paper at a conference. If you believe that you have an unusual circumstance that warrants a makeup exam, notify us as soon as possible. If you will be attending a conference or a university-sponsored activity, you must make arrangements for a makeup exam *in advance*. If you miss an examination without a legitimate reason, a grade of 0 will be recorded for that examination.

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

1. The exam average will be computed by summing 50% of exam 1 and 50% of the final. A student with an exam average below 60 will receive an F for the course. A student with an exam average between 60 and 65% will receive a D for the course.
2. The project grade will be computed by combining the grades for various project assignments (see Appendix A). Project grades may be strongly influenced by the Guidance Team's assessment of a student's contribution to his/her team. A student with a project grade below 60 will receive an F for the course and no further grade computation will be made. A student with a project grade between 60 and 65% will receive a D for the course unless the student has already received a grade of F for the exams.
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 lessons (quizzes and exercises) and class participation. 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, F: 0-59.

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 the 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 the 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, smartphones, email, Facebook, 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. Classes 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 textbooks. Some material may not be in the textbooks: references will be provided, but you are responsible for the contents.

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 Office of Student Conduct and Conflict Resolution (OSCCR). You are expected to conduct yourself in a professional and courteous manner, as prescribed by the Handbook of Operating Procedures: Student Conduct and Discipline (<https://www.utep.edu/hoop/section-2/student-conduct-and-discipline.html>).

Accommodations: 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, 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.

COVID-19 Precautions: 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, 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 <http://epstrong.org>.

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.

Appendix A: Project Grade Distribution

The project grade will be computed by combining the grades for various project assignments. Here is the break down of how the project grade is being calculated.

	Deliverables	Weights
1.	Software Configuration Management (SCM) Plan	5%
2.	Prototype Demos	10%
3.	Classes, Responsibilities and Collaborations (CRC)	10%
4.	Subsystem	5%
5.	Protocol	5%
6.	Software Design Document (SDD)	20%
7.	Implementation	20%
8.	Test Plan	10%
9.	Final Presentation	15%
	Total:	100%

Appendix B: Detailed Schedule

Dates		Topics	Readings	Project
Week 1	Jan. 18, 20	Introduction to CS 4311 Configuration management	Chapter 11	
Week 2	Jan. 25, 27	Process improvement Design principles	Section 4.5; handout	SCM plan
Week 3	Feb. 1, 3	Software architecture	Sections 7.1-7.2	
Week 4	Feb. 8, 10	CRC Group work on CRC	Handout Chapters 3-4 [W90]	
Week 5	Feb. 15, 17	Collaboration Cohesion and coupling	Chapter 5 [W90] Section 8.3	CRC
Week 6	Feb. 22, 24	Contract Subsystems	Chapter 6 [W90] Chapter 7 [W90]	
Week 7	Mar. 1, 3	Prototype demo 1 Exam 1		Demo 1
Week 8	Mar. 8, 10	Protocols Object Constraint Language	Chapter 8 [W90] Handout	Subsystem
Week 9	Mar. 15, 17	Spring break		
Week 10	Mar. 22, 24	V&V overview Unit testing	Sections 10.1-10.2 Section 10.3	Protocol
Week 11	Mar. 29, 31	Unit testing OO testing		SDD
Week 12	Apr. 5, 7	Prototype demo 2		Demo 2
Week 13	Apr. 12, 14	Integration and system testing Test plan	Handout	
Week 14	Apr. 19, 21	Non-testing V&V Program verification	Sections 10.5-10.7 Handout	Test plan
Week 15	Apr. 26, 28	Verification or project work		
Week 16	May 3, 5	Final Presentation		Presentation
Week 17	May 12	Final at 4:00 PM – 6:45 PM		

[W90] R. Wirfs-Brock, R. Wilkerson, and L. Wiener, *Designing Object-Oriented Software*, Prentice Hall, 1990.

Important Dates

January 17: Dr. Martin Luther King, Jr. holiday – university closed
 January 18: Classes begin
 February 2: Census day
 March 3: Exam 1
 March 14-18: Spring break
 March 25: Cesar Chavez holiday – no classes
 April 1: Drop/withdrawal deadline
 April 15: Study day
 May 5: Last day of classes
 May 6: Dead day
 May 12: Final on Thursday at 4:00 PM - 6:45 PM

Appendix C: 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. The material has been presented only at a superficial level. 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 notations for producing a detailed design of a system.
- c) Apply general strategies for creating a design of a system.
- d) Apply general strategies to identify and implement appropriate software architecture styles (including distributed and cloud) for the system under development.
- 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 students 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 architectural 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.