Master Course Syllabus

For additional course information, including prerequisites, corequisites, and course fees, please refer to the Catalog: https://catalog.uvu.edu/

Semester: Spring
Year: 2025
Course Prefix: CS
Course and Section #: 3450-002
Course Title: Principles and Patterns of Software Design
Credits: 3

Course Description

Gives students familiarity with modern principles and practices of software design. Emphasizes design patterns, including their motivation and the design principles on which they are based.

Course Attributes

This course has the following attributes:

☐ General Education Requirements
☐ Global/Intercultural Graduation Requirements
☐ Writing Enriched Graduation Requirements
☒ Discipline Core Requirements in Program
☐ Elective Core Requirements in Program
☐ Open Elective

Other: *Click here to enter text.*

Instructor Information

Instructor Name: Tim Severance

Student Learning Outcomes

Upon completion of this course, students should be able to:

1. Apply the key principles that underly the design of quality software,
2. Identify which patterns support which design patterns,
3. Detect when design principles are violated and which patterns are needed to improve a design,
4. Comprehend the design patterns in common use in professional software development,
5. Produce designs that are orthogonal and reusable,
6. Discuss designs at a high level using patterns terminology,
7. Identify patterns that balance the forces in a given design problem.

Course Materials and Texts

Freeman and Robson, Head First Design Patterns, 2nd ed., 2020, O'Reilly, 978-1-492-07800-5

Optional texts include:

- Gamma et al, *Design Patterns*, Addison-Wesley, 1994. (the famous "Gang of Four" book)
- Fowler, *UML Distilled,* Addison-Wesley, 2004.
- Fowler, *Refactoring: Improving the Design of Existing Code,* Addison-Wesley, 1999.
- Kierevsky, *Refactoring to Patterns,* Addison-Wesley, 2005.
- Martin, *Agile Software Development*, Prentice-Hall, 2003.
- Coplien, *Software Patterns*, SIGS Books, 1995
- Buschmann et al, *Pattern-Oriented Software Architecture, Vol. 4,* Wiley 2007
- Utas, *Robust Communication Software,* Wiley 2005
- Hanmer, *Patterns for Fault-Tolerant Software,* Wiley, 2007
- Arjona, *Patterns for Parallel Software Design,* Wiley, 2010

Course Requirements

Course Assignments, Assessments, and Grading Policy

I won't take attendance. However, you need to come to class, every time. The material is challenging enough that if you miss class often, you will feel the Gandalf effect ("You shall not pass!") Quizzes will be given in class occasionally, partly as a way to encourage you to attend class. Quizzes cannot be made up.

Grading

Your performance on the following will determine your course grade:

- Homework 15%
- Programs 40%
- Quizzes 10%
- Reading 5%
- 1 Mid-term Exam 15%
- Final Exam 15%
- Extra Credit (up to 5% bonus)

(Note: I reserve the right to change the above weights, generally to improve your results.)

You are responsible for all material in the text, readings, and class discussions. Grades will be assigned according to the following schedule:

93 – 100% A
90 – 92% A-
87 – 89% B+
83 – 86% B
80 – 82% B-
77 – 79% C+
73 – 76% C
70 – 72% C-
67 – 69% D+
63 – 66% D
60 – 62% D-
0 – 59% E

You are responsible for knowing all important university deadlines for students (e.g., last day to add/drop, etc.)

Programming Assignments:
There are two types of assignments, homework assignments and programming projects. The homework assignments are small, and should take you a small amount of time. A few require writing a small amount of code. The programming projects are exactly that: a program you must write. Please note the following:

- You may write code in C++, C#, D, Java, Python, or Javascript. If it's anything other than these languages, see me first.
- Some assignments require associated UML diagrams. Please follow standard UML; you will lose points if your diagrams are not accurate UML. (Part of good design is communicating your design to others, and UML is a standard way of doing so.) Consult UML Distilled and the UML description sheet for details.
    - Note that Visual Studio does NOT generate correct UML class diagrams.
    - Hand-drawn are actually often easiest, but MAKE THEM LEGIBLE and COMPLETE!
- Programming assignments will be graded on the following:
    - Do they work?
    - Do they implement the specified pattern correctly?
    - Do they follow the principles we have studied to that point?
- For programs and homework that requires programming, submit:
    - An executable (runnable on Windows WITHOUT installing additional dlls.) (Java programmers: create an executable jar file.)
    - Source code. I cannot grade programs without examining the source code. (C++ programmers: be sure it is standard C++, not VS C++, in case I need to compile it.)
    - Do NOT submit a complete VS project. Find the .exe and source files yourself!
- This is a software design class! It's not just the patterns! While the programming projects are about one or two patterns, they are ALSO about general software design. They are design and implementation problems with substantial meat!

Assignments are due at 11:59 PM on the due date, except where noted. They should be submitted via Canvas. Paper materials (where applicable) are to be handed in at the beginning of class.

Late assignments*:** Except where noted, late assignments are accepted with a 5% penalty per day late***. **A key to success is to turn assignments in on time.**

Exams
The exams are hard. They are harder than the assignments. You will be expected to show mastery of the material; to show that you not only understand the design principles and patterns, but how and where to apply them.

Expect the exams to be hard. Got that? However, I tend to compensate by grading generously. Exam averages are typically around 81%.

One double-sided cue sheet is allowed for all tests. (Size: for tests in the testing center, ½ a sheet; for final exam in class, a full 8.5 x 11 sheet)

The Final is comprehensive. It will be given in class at the university-mandated exam time.

Required or Recommended Reading Assignments

Weekly reading will follow the general subject matter as outlined below in accordance with the chapters of the textbook.

General Description of the Subject Matter of Each Lecture or Discussion

1. Overview and Christopher Alexander
2. The Strategy Pattern
3. The Observer Pattern
4. The Decorator Pattern
5. Factory Patterns
6. The Singleton Pattern
7. The Command Pattern
8. The Adapter and Façade Patterns
9. The Iterator and Composite Patterns
10. The Template Method Pattern
11. The State Pattern
12. The Proxy Pattern
13. Memory Patterns
14. Reliability Patterns
15. Refactoring

Schedule

HW	**Project**	**Quiz**	**Reading**	**Week #**
1 Pattern			Fish, Orders	1
2 Strategy Puzzle			1 Strategy	2
	1 Strategy	1		3
3 Observer		2	2 Observer	4
	2 Observer	3	3 Decorator	5
	3 Decorator			6
4 Factory/Singleton		4	4Factory/5Singleton	7
	4 Command	5	6 Command	8
5 Adapter/Facade		6	7/8Adapt,Façad,Tem	9
				10 Spring Break
	5 Iterator	7	9 Iterator/Composite	11
		8		12
6 Visitor	6 Composite		12 Compound / MVC	13
		9		14
	7 Design Project		13 Conclusion	15

Required Course Syllabus Statements

Generative AI
The purpose of education is learning. Learning happens by honest effort; there are no shortcuts. Your role is (and will be, in your professional life) to understand tools and concepts and to use them to solve practical problems.

When you submit work, you are claiming that it is your own work and that you understand how and why it works. It is appropriate to use external resources, including web sites and other students, to identify useful tools and concepts and to learn how to use them. It is not appropriate to copy work from external resources, regardless of how little of the work is copied. Uploading code to a website that makes it available to others is plagiarism, whether or not you intend to make it available. Do not read another student's code unless explicitly authorized. If you are unsure about what is or is not acceptable, ask your instructor.

Use of ChatGPT (or other similar platforms) as a learning assistant, not as a crutch. If you use it, cite it at the top of your code. You are responsible to make sure that any code or content does what it is supposed to do and says what you want it to say. Don't accept anything it generates at face value without checking it critically. These days potential employers will expect you to know how to use tools like ChatGPT to generate code, so it is a skill we need to teach you. If it helps you learn some things faster, GREAT because we can spend class time on more interesting topics. Just remember: If you REALLY want to be good, work for it.

Using Remote Testing Software
☒ This course does not use remote testing software.

☐ This course uses remote testing software. Remote test-takers may choose their remote testing locations. Please note, however, that the testing software used for this may conduct a brief scan of remote test-takers' immediate surroundings, may require use of a webcam while taking an exam, may require the microphone be on while taking an exam, or may require other practices to confirm academic honesty. Test-takers therefore shall have no expectation of privacy in their test-taking location during, or immediately preceding, remote testing. If a student strongly objects to using test-taking software, the student should contact the instructor at the beginning of the semester to determine whether alternative testing arrangements are feasible. Alternatives are not guaranteed.

Required University Syllabus Statements

Accommodations/Students with Disabilities
Students needing accommodations due to a permanent or temporary disability, pregnancy or pregnancy-related conditions may contact UVU Accessibility Services at accessibilityservices@uvu.edu or 801-863-8747.

Accessibility Services is located on the Orem Campus in BA 110.

Deaf/Hard of Hearing students requesting ASL interpreters or transcribers can contact Accessibility Services to set up accommodations. Deaf/Hard of Hearing services can be contacted at DHHservices@uvu.edu

DHH is located on the Orem Campus in BA 112.

Academic Integrity

At Utah Valley University, faculty and students operate in an atmosphere of mutual trust. Maintaining an atmosphere of academic integrity allows for free exchange of ideas and enables all members of the community to achieve their highest potential. Our goal is to foster an intellectual atmosphere that produces scholars of integrity and imaginative thought. In all academic work, the ideas and contributions of others must be appropriately acknowledged and UVU students are expected to produce their own original academic work.

Faculty and students share the responsibility of ensuring the honesty and fairness of the intellectual environment at UVU. Students have a responsibility to promote academic integrity at the university by not participating in or facilitating others' participation in any act of academic dishonesty. As members of the academic community, students must become familiar with their rights and responsibilities. In each course, they are responsible for knowing the requirements and restrictions regarding research and writing, assessments, collaborative work, the use of study aids, the appropriateness of assistance, and other issues. Likewise, instructors are responsible to clearly state expectations and model best practices.

Further information on what constitutes academic dishonesty is detailed in UVU Policy 541: *Student Code of Conduct*.

Equity and Title IX

Utah Valley University does not discriminate on the basis of race, color, religion, national origin, sex, sexual orientation, gender identity, gender expression, age (40 and over), disability, veteran status, pregnancy, childbirth, or pregnancy-related conditions, citizenship, genetic information, or other basis protected by applicable law, including Title IX and 34 C.F.R. Part 106, in employment, treatment, admission, access to educational programs and activities, or other University benefits or services. Inquiries about nondiscrimination at UVU may be directed to the U.S. Department of Education's Office for Civil Rights or UVU's Title IX Coordinator at 801-863-7999 – TitleIX@uvu.edu – 800 W University Pkwy, Orem, 84058, Suite BA 203.

Religious Accommodation

UVU values and acknowledges the array of worldviews, faiths, and religions represented in our student body, and as such provides supportive accommodations for students. Religious belief or conscience broadly includes religious, non-religious, theistic, or non-theistic moral or ethical beliefs as well as participation in religious holidays, observances, or activities. Accommodations may include scheduling or due-date modifications or make-up assignments for missed class work.
To seek a religious accommodation, a student must provide written notice to the instructor and the Director of Accessibility Services at accessibilityservices@uvu.edu. If the accommodation relates to a scheduling conflict, the notice should include the date, time, and brief description of the difficulty posed by the conflict. Such requests should be made as soon as the student is aware of the prospective scheduling conflict.

While religious expression is welcome throughout campus, UVU also has a specially dedicated space for meditation, prayer, reflection, or other forms of religious expression.