CS-6600-I03 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 #: 6600-I03
Course Title: Graduate Project I
Credits: 3

Course Description

Teaches the design and development of a walking skeleton with students participating in all aspects of software development, including: requirements elicitation, architecture, design, implementation, testing, and deployment. First semester of a two-semester capstone course.

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: Jingpeng Tang
Jing

Student Learning Outcomes

1	analyze the system using feedback from customers and other stakeholders to create a set of requirements.
2	design a large software system.
3	create a walking skeleton of a large software system.
4	lead a software project.
5	create clear design (internal) documentation

Course Materials and Texts

No

Course Requirements

Course Assignments, Assessments, and Grading Policy

CS 6600 and CS 6610 are consecutive courses for a two-semester project. The student will finish a proposal in CS 6600. The proposal will cover analyzing a large system to create a set of requirements, designing a large software system, creating a walking skeleton of a large software system, leading a software project, and creating clear design (internal) documentation. This is the preparation for CS 6610 for the following deliverable.

Deliverable #1:

Creation of an original software artifact, of enough size and complexity to demonstrate the mastery of designing and developing a large system. A project must demonstrate in-depth knowledge of one or more areas of computer science and go well beyond the material covered by the required graduate and undergraduate courses. The MCS is particularly interested in projects that work like end products rather than incidental pieces of experimental research (e.g., just an algorithm) given our focus on practical and applied education.

While the number of lines of source code in a project is but one measure of complexity, students should expect to produce a project with at least 5,000 lines of code. Projects of a smaller size can be acceptable but only if the overall complexity of the project is of sufficient depth and breadth to warrant a smaller project.

Creativity is the most straightforward way to add to the perceived complexity of a project. Creativity is highly desirable on the part of students and can be demonstrated by the design and implementation of new algorithms, data structures, or in the novel concept of the application. The creative elements of a project need not be of the quality to be publishable in a tier 1 journal, but it must represent original elements conceived by the student and/or mentor. When in doubt, go **BIG**!

It is completely acceptable to produce "yet another" implementation of a type of software that has been done before. However, when replicating others' work it is essential that the student clearly demonstrate their knowledge of prior work and document this knowledge with references and an analysis of how they have incorporated lessons learned from prior work into their project. Simply implementing the project will not be sufficient to show the depth and breadth of knowledge necessary to receive an MCS degree.

Combining an MCS project with something from work is acceptable, but what a student does for the MCS must represent their individual effort, not the combined efforts of a larger group at work. Additionally, students should be intimately aware of UVU's policies and procedures on intellectual property before engaging in a work-related project.

Deliverable #2:

Writing of an original paper, which clearly demonstrates mastery of technical writing and of the subject matter. The MCS will provide students with a MS Word .docx template or LaTeX template to follow for creating their original paper. Finished papers will be published online as a record of the student's accomplishment and to provide insight and ideas to future MCS students of what they can and should accomplish. Students should expect a project with sufficient size and complexity to take ten (10) to twenty (20) pages to adequately document. Students should consider taking advantage of the UVU Writing Center to ensure their paper is of professional quality. While the technical content of the paper is of utmost importance, spelling and grammatical errors cannot be tolerated. The basic format of the template will include the following sections:

- a title, which contains the name of the project, the name of the student, the name of the Faculty Mentor, the names of the MCS committee members who reviewed the project, and the MCS Director's name,
- an abstract, which contains a concise (250 words or less) summation of the entire paper,
- an introduction, which contains a detailed overview of the problem solved by the project,
- the body, which contains a comprehensive explanation of the design and implementation for the project. This should include, but is not limited to, subsections showing structural diagrams and behavioral diagrams for critical sections of the project, screen shots, example output, a detailed explanation of the results, and an analysis of prior work when replicating others' work,
- user's manual, which contains instructions for installing and running the project of sufficient detail that someone other than the student could install and use the software if they had prior knowledge of the domain,
- a conclusion, which contains the student's personal views on the success and/or failure of the project, what specific things were learned in creating the project, what knowledge would have been helpful to have before starting the project, and a summary of the important results detailed in the body,
- a reference section, which contains citations to books, articles, papers, websites, or other material used in creating the project,
- and copyright notices and authorizations that will allow the Computer Science Department/UVU to distribute the paper.

Deliverable #3:
Presentation of an oral report on the project, which demonstrates mastery of presenting technical material. Students are strongly encouraged to use PowerPoint or a similar tool to create visual material for their oral report. The oral report should include the structural and behavioral diagrams from the original paper as well as a live demonstration of the project. Immediately after their oral report, the student will be notified if their project has been accepted or rejected, and what if any changes must be made to the project before final acceptance. Once all suggested modifications are

made and documented, an updated version of the written report must be submitted to the MCS Director for publication on the MCS website.

Criteria for judging the acceptability of an MCS Graduate Project:
The Faculty Mentor, committee members, and Director shall be given two (2) weeks to review the written report prior to the presentation of the oral report to allow feedback to be given and changes to be made. While the software artifact represents the largest commitment of time and will be weighted the heaviest, poor or even moderate quality in the written or oral report will result in a rejection. If a rejection occurs, specific, written guidelines will be provided which outline the steps the student must take to remedy the rejection. In judging whether a project is acceptable, the Faculty Mentor, committee members, and MCS Director shall consider:

- the size of the project, 5,000+ lines of code.
- the novel or creative elements of the project,
- the depth and breadth of technical knowledge needed to complete the project,
- the level of skill and mastery of the material demonstrated in the written paper,
- the level of skill and mastery of the material demonstrated in the oral presentation,
- and if all the deliverables are consistent with the expectations of the MCS as well as the student being ready for employment as a senior developer/tech leader/chief architect who can design, implement, document, and present a large-scale system.

Grading Scale:
The following grading standards will be used in this class:

Grade	**Result**
A	Pass Oral Defense with or without minor changes
T	Continuing Project work that must be completed to pass the defense
E	Did not pass Oral Defense

Assignment Categories

Activity	Percent
Project software and deliverables	100
Total	100%

- *Note: SRI submission at the end of the semester is encouraged. I appreciate your constructive feedback to improve my teaching.*

- Late Work Statement:
  Work is finished when it is finished. Any "penalties do not make sense in this context.

Required or Recommended Reading Assignments
Not applicable.

General Description of the Subject Matter of Each Lecture or Discussion
Students meet with advisor in a regular base to discuss the selected project. Student and advisor decide the frequency and length of the meeting.

Required Course Syllabus Statements

Generative AI
ChatGPT (and similar Tools) in This Course In General: Use ChatGPT 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. Just remember: If you REALLY want to be good, work for it. Does your instructor REALLY expect you to use GEN AI in this class? REALLY? Yes!
Suggestions for using it responsibly:

1. **Concept Clarification:** If you're stuck on a concept like Bayes Nets, Minimax or Expected Value, GenAI can explain it in simpler terms or provide examples to help you understand better.

2. **Practice Problems:** GenAI can generate practice problems for you to solve on your own. After you attempt them, it can help you check your answers and understand any mistakes.

3. **Study Tips:** GenAI can offer strategies for studying various AI algorithms effectively, such as how to break down complex problems or how to organize your study sessions.

4. **Resource Recommendations:** GenAI can suggest other textbooks, online courses, or websites that provide additional explanations and practice problems.

5. **Homework Guidance:** While GenAI could but should not do your homework for you, it can help you understand the questions and guide you on how to approach solving them.

6. **Exams:** GenAI is not to be used on Exams. For remote exams such as in Canvas, your instructor leaves people on their honor not to cheat in any form.

7. **Debugging and Understanding Code:** GenAI should not be used to write blanket write code for you with no work on your part--the goal is to train **you**, not it--but it can help you find and fix problems and suggest improvements.

Actual use of AI, including Generative AI depends heavily on the project being done.

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.