

Syllabus

EECS 497 - Human-Centered Software Design and Development
*ENGR 455-400 - Multidisciplinary Software Development**
Winter 2021

**Note: If you are enrolled in ENGR 255/355, you have been included in this course simply for your own interest. At this time, the only work that you will be responsible for completing will be managed exclusively by MDP.*

Class Times

There are no official class times as lectures will be delivered asynchronously via lecture recordings. However, there will be synchronous office hours (see below). In order to track your virtual “attendance” in each lecture, you will need to fill out a short lecture summary using the form linked on Canvas as proof that you “attended” the virtual lecture.

Course Staff

Jeff Ringenberg – jringenb@umich.edu
Office Hour(s): Mondays, 1:00-2:30p EDT/EST
Office Hour(s) location: [Zoom \(click here\)!](#)

Ju Young (Lisa) Kim – kimjuyou@umich.edu
Office Hour(s): TBD
Office Hour(s) location: [Zoom \(click here\)!](#)

Course Overview

"When you design for everyone, you design for no one."

This phrase describes one of the key reasons why software design (and development) can quickly lead to failure. In this course, we will address this problem by covering a wide variety of topics that are significant in the creation of great software but that are frequently overlooked. The number of topics that we could potentially cover in this course is quite large. In fact, there are **so many** important topics that we can't reasonably cover them all in one semester.

For EECS 497:

For ENGR 455-400:

As a part of MDP, you are required to take a second semester of this course to learn more

If you are interested in learning more about human-centered software for a second semester and/or you would like to continue working on your project for an additional semester(s), please contact course staff about the possibility of pursuing this through a directed study.

about multidisciplinary software development and to continue the work on your sponsored project.

The end goal of this course is for teams to participate in the complete software design and development cycle, from idea incubation to product delivery, and to create successful software products that incorporate effective design strategies and project management methodologies.

Team Composition

For EECS 497:

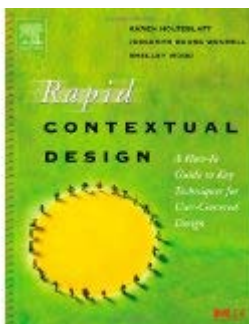
Teams will be composed of 3-5 students. In order to foster and apply effective design methodologies, **EECS 497 projects must incorporate components of the “not-you philosophy”**. Briefly, this means that significant elements of each project must be associated with users that differ from the project’s team members in a substantial way.

For ENGR 455-400:

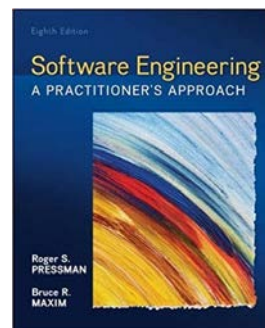
Teams will vary in size and will consist of a sub-team of CS students and a second Domain sub-team of non-CS students chosen to fulfill the requirements of individual projects in the Multidisciplinary Design Program (MDP). All projects will include the participation of a real customer and be developed according to their requirements.

Reference Textbooks

Note: **You don’t need to purchase these**, however, you are required to learn the material that is covered in lecture.



Rapid Contextual Design
Holtzblatt, Wendell, and Wood, 2008



Software Engineering: A Practitioner's
Approach (8th Edition)

ISBN: 978-0123540515

McGraw-Hill Education, 2014

ISBN: 978-0078022128

Selected online readings may also be posted separately on Canvas.

Main Course Topics

Under the above course model, the first (main) semester will cover critical topics in the early stages of software creation such as customer discovery, contextual inquiry, prototyping, process models, project scheduling, UI/UX design, development, and testing/debugging. A second semester would continue with topics such as security, change management, maintenance, business considerations, case studies, intellectual property, and entrepreneurship. Finally, several other “non-traditional” subjects may be covered at various points during either semester to further your ability to be the best programmer possible. These include topics such as team dynamics, creative problem solving, dealing with failure, leadership, and conflict resolution.

A full list of topics can be found in the Course Schedule.

Grading Scheme - EECS 497 Only

Category	% of Final Grade
Initial Project Concept Document	2%
Lecture Summaries	5%
Self and Team Evaluations	5%
Project Planning Document	7%
User Requirements Document	9%
Lo-Fi Prototype	12%
Testing and Debugging Report	15%
Final Demo	20%
Exam	25%

Factors such as class and team participation may be used to adjust your final grade, especially if it falls on a borderline. There is a ‘gray area’ of several points around each specific numeric grade, within which a \pm system is used. Two people getting the same numeric grade might therefore receive different letter grades for the course. If the student is in one of these gray areas, their grade may go up or down depending upon whether course performance has been improving or declining, or whether participation in group work has been sufficient or inadequate.

Working with your team remotely

One of the main goals for this course is to make it possible for you to complete this course without needing to meet with anyone in person. That said, face-to-face interactions are absolutely critical to effective teams in general. Therefore, your team should look over the following list of resources to see what tools can be used to help facilitate more interactive forms of communication. I suspect that you know of many more resources and you are free to use them as well. If you do know of any other tools, please share them with us on Piazza. With that, here's the list:

- [Zoom](#): Pretty much the de facto standard for video conferencing in the COVID-era.
- [BlueJeans](#): Another video conferencing option
- [Hangouts Meet](#): Yet another video conferencing option
- [GitLab](#): I suspect that many of you have already used this (or GitHub)
- [Jira Software](#): A more “commercial” tool for project management
- [Microsoft Teams](#): Another tool for team management

Ethics, Decency, and Student Conduct

While the topics of ethics and decency are quite complex, especially when it comes to the development of software, the goal of this course is for each student team to create projects that would be deemed as ethical and decent from a practical viewpoint. Given that users can generally make bad decisions that violate this policy regardless of the application, we simply require that any project that could be used in such a manner that would be deemed unethical and/or indecent should have modest safeguards put in place to address any concerns. In other words, your projects should not actively encourage unethical and/or indecent behavior nor should they present this type of material in any form.

In addition, in accordance with the [General Standards of Conduct for Engineering Students \(link to external site\)](#) and the University's [Statement of Student Rights and Responsibility \(link to external site\)](#), we are committed to a policy of equal opportunity and expect all students in this course, regardless of academic major, to follow any and all standards that have been established in the aforementioned links. Please feel free to contact us with any problems, concerns, or suggestions.

We expect that all students will treat each other with [R.E.S.P.E.C.T.](#)

Cheating Policies

You may be required to complete some of your coursework individually. In these cases, you are encouraged to form groups and to discuss your work at the conceptual level with your fellow

students. However, translating those ideas into computational artifacts and any related write-up should be performed without help from fellow students. Copying from a current or former student, a former instructor, or an instructor from another section, either manually or electronically, is considered cheating. In addition, it is unacceptable to provide your work to another student. However, you may (and are encouraged) to employ solutions supplied by your current instructors or the course website.

All individual assignments, quizzes, and exams must be carried out completely on your own, without aid from any of your fellow students.