

**UNIVERSITY OF MICHIGAN**  
**College of Engineering**  
**Curriculum Committee Meeting**  
**Tuesday, November 21, 2023**

**Attending:** Achilleas Anastasopoulos, Jack Baker, Robert Bordley, Yavuz Bozer, Roger De Roo, Chris Fidkowski, Fei Gao, Saadet Albayrak Guralp, Amir Kamil, Leena Lalwani, Xiaogan Liang, Cameron Louttit, Emmanuelle Marquis, Radoslaw Michalowski, Yulin Pan, Mika Panagou, Eric Rutherford, Rachael Schmedlen, Ben Spector, Roxanne Walker

**Support Staff:** Stacie Benison, Betsy Dodge, Matthew Faunce

**Call to Order: 1:35pm**

**Adjourned: 2:47pm**

**Agenda:**

1. Approval of 10.24.2023 Meeting Minutes - Page 2 - **APPROVED**
2. URO ROB CARFs Discrepancies and Requests– Informative Item – Page 4
  - a. Situation summary: A student was unable to register for ROB 450 when they had seemingly met the prerequisites. The enforced prerequisites on the course were, "Junior standing and [TCHNCLCM 350; (C or better)] and [ONE of ROB 310, 311, 320, 330 or 340; (C or better)]." When the Robotics Department had included "junior standing", they were under the impression that this meant at least junior standing but after speaking with the URO, they discovered that this is what is restricting senior students from enrolling. Michael Shearon assisted Robotics in updating the enforced prerequisite to read "Junior or senior standing and [TCHNCLCM 350; (C or better)] and [ONE of ROB 310, 311, 320, 330 or 340; (C or better)]" so that seniors could register for the Winter 2024 term. A Modification CARF with these edits was submitted and sent to the RO Curriculum Office on 11.20.2023.
  - b. For future reference, when a department requests a credit exclusion between courses, the department needs to be aware that for the credit exclusion to work properly, a CARF updating the credit exclusion for each course included must be submitted.
  - c. You cannot have a credit exclusion based on one section of a course. This can be added to the Enforced Prerequisite section instead, for example: *No credit in ENGR 100, topic "Robotics Mechanisms (topic ID 29)"*
3. HLC Annual Audit Questions for the CoE Curriculum Committee – Informative Item – Xiaogan to Present - Page 8
  - a. Question 1 (How should departments handle courses that are taught in combination with other institutions? How should departments handle it when the course is taught at UM? Do courses that are a teaching collaborative need to follow CoE Policy for the Assignment of Credit Hours?) - **PENDING**
    - i. CCC Chair Xiaogan Liang discussed that the course design should fit the requirements of both institutions.
    - ii. ROB representative had mentioned that Chad Jenkins has been leading the effort on the UM side, he can offer more information on the syllabus, course materials associated with this course and the agreement between the two universities. CCC Chair Xiaogan Liang will send an email to Chad Jenkins requesting clarification.
    - iii. A question was asked if the course is offered residentially, and it was answered that for UM students, this will be offered online, and lectures are conducted synchronously at Florida A&M University.
    - iv. CSE representative noted that the course is currently scheduled Tuesday & Thursday 9:30am-10:45am for 3 credits and asked for clarification that the discrepancy in contact hours was 10 minutes per week. Given that courses at UM are scheduled to the end of the hour but release students 10 minutes early, the discrepancy appears to be 5 minutes in lecture time.

- v. A few CCC members agreed that the department should schedule the class at the UM campus until the end of the hour (in the case of the current class, that would be until 11am, or a total timeframe of 9:30-11am with the understanding that students will be let out 15 minutes early, knowing this is comparable to UM).
- b. Question 2 (When was the lab policy established?) - **RESOLVED**
  - i. The current CoE Policy for the Assignment of Credit Hours was approved October 13, 2020. There was no further response from CCC members with knowledge of earlier dates of the Credit Hour Policy, but it was understood that Labs met for 2 contact hours for each credit.
- c. Question 3 (Do labs need to be scheduled in a formal CoE Computer Lab space when the work can be done online?) – **PENDING**
  - i. Combined discussion below under question 4.
- d. Question 4 (Are the activities associated with the online, self-paced, asynchronous Canvas modules used in ENGR 101 and 110 and other departmental courses acceptable as CoE contact hours?) – **PENDING**
  - i. Questions posed regarding a shift to more virtual instruction:
    1. What does it mean to have a tutorial with interactive tutorial without further participation from a faculty member?
    2. What happens when there is a course with minimal faculty interaction?
    3. Will students see this as a deterrent to attending the University of Michigan if they can receive similar instruction through online methods or videos?
  - ii. ECE representative commented on the interactive aspect of lab courses, with an example of a course taught with a regular lab, but during COVID students were allowed to build their circuits at home. They questioned what is the difference between a lab versus a homework assignment? Are students required to come to the lab so that they can show their work, have faculty supervision, and have faculty retain an active role in the learning process? Concern was expressed regarding software labs in which the work is being done all at home regarding whether this could be considered a lab.
  - iii. CCC Chair Xiaogan Liang mentioned using Piazza for providing answers to student inquiries and mentioned that the nature of the course can determine what components may be most desired to count as contact hours. He noted that for some courses, if a student does not attend the lab, they won't have access to the necessary tools. But for software courses in which students can work from any location, providing student feedback through a platform such as Piazza or other platforms that allow faculty to respond in the moment can be sufficient for providing faculty interaction to students.
  - iv. CSE representative added that the current definition of contact hours does not capture most interactions between instructors and students and what instructors and students themselves think is effective. The CSE representative mentioned having a 16-minute average response time for EECS 280 for Piazza, and that 1400 of these responses were their own. Office hours have been effective for many students taking CSE courses as this is noted as one of their primary methods for interacting with faculty. When attendance at lectures has been made optional, students have been attending class less and have instead been focusing their time on attending office hours. From the point of view of staffing hours, staff find it is more effective for faculty and staff to spend more time on Piazza office hours than being in the room with students. It would be nice for the policy to account for the "on the ground reality" of how faculty members spend their time interacting with students.
    1. A question was raised regarding how the college can redefine the contact hours policy to consider reasonable interaction through real-time interaction of faculty with students in office hours.
- v. ENGR/ADUE representative Rachael Schemedlen noted that both ENGR 101 and 110 came to their current structure by going through the Foundational Course Initiative process to determine how best to deliver course content. In ENGR 101, students really like the way the online modules work and feel it is an effective way to learn.
  1. The CCC should consider defining what course activities can be defined as "faculty-led engagement."
  2. The CCC should provide guidance on what is considered "interactive computer-assisted instruction" and what's the difference between "interactive tutorial without any further participation" and "interactive tutorial".
- vi. IOE representative commented on seeing two faculty interaction paths, one being learning something new in a team setting, in which a physical lab plays a bigger role, and assisting students with their questions about course content, in which students typically receiving assistance through office hours.

- vii. CCC members had a discussion surrounding whether student questions answered by faculty, on any platform, would be considered contact hours, such as on platforms like email or Piazza.
    - 1. IOE representative raised that the Michigan Medicine Patient portal displays a message that lets customers know that their message exchanges with physicians may result in medical charges.
    - 2. It was noted that complications may arise in how to define faculty engagement through message responses to students in a tangible way.
  - viii. A question was raised regarding when students work on the software associated with ENGR 101. Are students able to have a question answered in real time? ENGR/ADUE representative Rachael Schemdlen mentioned that Prairie Learn is the platform used to deliver this course content, which combines Canvas and Slack. Rachael will confirm whether the platform has this functionality for faculty to respond to questions in real-time.
  - ix. A question was raised regarding how an “instructor” is defined for contact hours, whether this is defined primarily as the professor or if a GSI or student teaching assistants would count.
  - x. It was determined by CCC Chair Xiaogan Liang that departments should go to back to their departmental constituents to collect further feedback, and this topic will return to a future CCC meeting as well as the review of the current Credit Hour Policy.
- e. ACTION ITEM: Departments should review questions 1, 3, and 4 with key departmental constituents for feedback and these items will return to a future CCC meeting for a review of responses and a vote on any necessary policy changes or decisions.**
- 4. Updates for the CoE Curriculum Committee on the HLC Annual Audit Process – Action Item – Xiaogan to Present – Page 10
    - a. Update – feedback has now been received on both the MATSCIE and UARTS spreadsheets.
  - 5. Proposal to Update CoE Minor Policy – Double Counting Policy Discrepancy – Action Item – Betsy to Present - Page 12 – **UPDATE APPROVED**
    - a. A question was asked whether the number of courses counted between minors is checked, and the answer is yes, departments should be checking this information.
    - b. A question was raised, “Why are there differences between LSA and CoE’s minor policies?”
      - i. A comment was made by CSE representative regarding students who see minors as “coupon collecting”, or students taking advantage of double counting rules to earn additional minors. Alternatively, students who have fulfilled the course requirements needed for a minor should earn the certificate for the minor.
    - c. A comment was made by CCC Chair Xiaogan Liang regarding looking into whether CoE should propose to schedule a joint meeting between LSA and CoE to discuss their differing minors policies.
    - d. There was a unanimous vote and the policy update to the updated language was approved. The CoE RO will make the associated update to the policy.

PAGE	SUBJECT	COURSE #	ACTION	SUMMARY	EFFECTIVE TERM	MIN. GRADE REQ. FOR ENF. PREPREQ	Is Course on LSA Course Guide?	APPROVED	NOTES & REVISIONS	TABLED
19	EECS	440	NEW		FT 2024	C	NO	APPROVED		

**UNIVERSITY OF MICHIGAN**  
**College of Engineering**  
**Curriculum Committee Meeting**  
**Tuesday, October 24, 2023**

**Attending:** Achilleas Anastasopoulos, Sarah Barbrow, Robert Bordley, Yavuz Bozer, Chris Fidkowski, Fei Gao, Robin Fowler, Amir Kamil, Xiaogan Liang, Cameron Louttit, Frank Marsik, Radoslaw Michalowski, Eric Rutherford, Mika Panagou, Anchal Sareen, Ben Spector, Andy Tadd, Roxanne Walker

**Support Staff:** Mercedes Carmona, Betsy Dodge, Matthew Faunce

**Call to Order: 1:35 PM**

**Adjourned: 2:14 PM**

**Agenda:**

1. Approval of 10.10.2023 Meeting Minutes (Page 2) - APPROVED
2. Re-visit - Non-Attendance Drop Statement Proposal – Action Item (Page 6)
  - a. Undergraduate Education & ENGR: Two issues that arise from this are attendance in the first week or so of the term to establish enrollment in the course and continued attendance/participation in the course throughout the term. A revised policy would make sense for the following as: 1) team-based courses (such as ENGR 100, capstone design, etc.) 2) Lab components that require in person attendance 3) Lecture based courses with waitlists. LSA Policy implies to secure a spot in the course, students are expected to attend the first couple of course meetings and if that does not occur, then instructors have the discretion to drop the student if there is no legitimate reason provided for the student's absence. This process could already be happening currently, but CoE could formally adopt a policy like this for courses. Attendance is an issue for courses that require (team-based courses) vs those that are not as critical for the course/after the first week. Would it be realistic for a waitlisted student to join a class, catch up and be successful beyond the first week of classes? The policy would need to be effective by the 2<sup>nd</sup> or 3<sup>rd</sup> weeks of a term. Waitlisted students would be negatively impact by not attending after the first couple of weeks, hence there would need to be a policy so that teams are not left because of a student's absence. Overall, option 2 is supported to create a modified non-attendance statement and procedure.
  - b. CSE: If a student is dropped, this could cause further issues such as a student losing financial aid or student visa. In agreement, from previous meeting, that attendance should be made part of the grade.
  - c. ME: Also in agreement, but states that the instructor should not have the power to drop students. A clear statement needs to be made that will hopefully encourage students to make a commitment to their courses.
  - d. IOE: ABET visit taking priority so no feedback has been retrieved from the department but will follow back up with any information. How are we to enforce a statement if we agree to one? This could lead into more issues such as timing, how this impacts instructors. Making attendance part of the grade is a concern as how would you take attendance for larger courses with 90+ students. What would be the appropriate value for the grade? Are we to come up with this ourselves or would the policy have a specific percentage needed for a course? As a result, this would create a lot of work for everyone involved.
  - e. ROB: Similar concerns to IOE and how can a student be dropped soon enough in order for a waitlisted student to be a part of the course. The statement would affect courses differently. What is the overall goal for this statement as not everyone can attend courses due to working or other conflicts.
    - i. Main purpose is to lessen the waitlist for courses, students to take team-based courses seriously, and overall enforce attendance.
  - f. CEE: No support from the department for a statement.
  - g. CHE: No waitlist concerns, but still reluctant to have a policy in place. Attendance in lab courses or 100 level courses makes it difficult for the instructor to hand out equitable workload due to attendance issues with students. Avoid the policy, but communicate clearly to students, in classes that attendance is an issue, that attendance is going to be factored into the course's overall grade.

- i. Some agreement in an enforcement policy needed along with a statement.
- h. UG Student: Preventative type of measure is needed before a student is dropped. Perhaps, future course registration can have a flag of some sort that reflects the student had attendance issues so that the instructor is aware.
- i. Vote taken with the same options listed as the previous meeting. Option 3 (10/17 members) had the majority vote that departments/units are to not allow administration to drop students from classes based on non-attendance.
  - i. In 2<sup>nd</sup>, was Option 2 (7/17) to create a modified non-attendance statement and procedure, with details to be determined by the CoE CC.
- 3. Re-visit - Review of Professional or Creative Development Courses (PCDC) Degree Audit Rule – Action Item (Page 7)
  - a. UG Education & ENGR: PCDC is only optional for IB and HU is required, it seems that the HU designation should be the stronger determiner, thus I support Option #3.
    - i. CSE, NERS, & ROB: Also, in agreement with this option.
  - b. Vote taken with the same options listed as the previous meeting with Option 3 having a unanimous vote.
- 4. Contradicting Double Counting Policies – Action Item (Page 9)
  - a. UG Education & ENGR: I don't see why a course shouldn't be double counted for a plan of study that doesn't show up on the diploma. If the course is relevant to both the plan of study and major, minor, etc., it makes sense to count it. However, as Fred and Susan mentioned, I think we want to discourage students trying to game the system to get as many of these supplemental studies on their transcripts by taking as few courses as possible and trying to triple (or more) count them, so I think it should be limited to double counting only. I vote for Option 2, to either remove #6 or modify the language of the Supplemental Studies Policy.
  - b. ECE: Split credits among the supplemental studies and minors.
  - c. Minor policy states that credits cannot be shared with multiple minors.
    - i. Policy states: No course can count towards more than two majors, and no more than one major and one minor. A course can only count toward two minors if the credits received for that course are beyond the 128 credits required for the student's major.
  - d. Vote taken with two options listed from the document. Option 2 had a majority of votes to remove contradiction or modify the language for the Supplemental Studies Policy. Most members agree that the #6 bullet point is to be removed from the policy but keep the Double Counting Policy as is.

### EECS CARFs with Subject Changes to ECE or CSE – Bulk Review

PAGE	SUBJECT	COURSE #	ACTION	SUMMARY	EFFECTIVE TERM	MIN. GRADE REQ. FOR ENF. PREPREQ	Is Course on LSA Course Guide?	APPROVED	NOTES & REVISIONS	TABLED
12	SI	565	MOD	Change in Home Department, Subject Change for Cross Listing, Course Credit Type.	FT 2024	NO	YES	APPROVED	Cross listed with EECS 597 and LING 702. The Home Unit can modify the Course Description if needed.	

## URO ROB CARFs Discrepancies and Requests

Michael Shearon had emailed Betsy and the Engineering RO email account in regards to CARFs approved and submitted from CoE CC meetings on 9.26.2023 (ROB 511) and 10.10.2023 (ROB 203) as there were going to be registration issues that occurred due to the information listed on the CARFs submitted. The following is what was requested for the courses to be effective for registration:

### **ROB 511 WN24 CARF:**

Credit Exclusion: *Only 1 course may earn credit from ROB 320, ROB 380, ROB 511, and EECS 367.*

In order for the credit exclusion to work properly, we need to set up Credit Exclusion rules for ALL courses listed in the Credit Exclusion request for ROB 511. This means, we need CARFs for the following courses, requesting credit exclusions:

- ROB 320
- ROB 380
- EECS 367

(I can set up the rules in the system now so that they will be ready for Monday, but RO Curriculum will need new CARFs for the courses listed above.)

### **ROB 203 WN24 CARF:**

Credit Exclusion: *Only 1 course may earn credit from ENGR 100.850 and ROB 203.*

We cannot set up Credit Exclusions from specific class sections (ENGR 100, section 850). Credit Exclusions can only be set up for entire courses.

What we CAN do is prevent enrollment in a class if the student has taken another class (or class section). Looking at the WN24 offerings of ENGR 100, I see that section 850 has a topic ID of **29 "Robotics Mechanisms"**

- I would recommend changing the CARF for ROB 203 from a Credit Exclusion to an Enforced Prereq: *"No credit in ENGR 100, topic Robotics Mechanisms (topic ID 29)"*

- Then, for ENGR 100, Associated Class 85 (section 850) for WN24, CHANGE the Associated Class Requirement Group

- from the current 000039 "Engineering Undergraduate"

- to the **new ERG 020620: "Engineering Undergraduate; and no credit in ROB 203"**

If this all meets with your approval, here are your action items:

1. **TIME SENSITIVE: Let me know right away if you want to apply these changes. I'll then start working right away on my end.**

2. Send in new CARFs for ROB 320, 380, and EECS 367 with new Credit Exclusion of: *"Only 1 course may earn credit from ENGR 100.850 and ROB 203."*

3. **TIME SENSITIVE:** Change the CARF for ROB 203 from a Credit Exclusion to an Enforced Prereq: *"No credit in ENGR 100, topic Robotics Mechanisms (topic ID 29)"*

4. **TIME SENSITIVE:** Apply the new ERG 020620: *"Engineering Undergraduate; and no credit in ROB 203"* to WN24 ENGR 100, Associated Class 85 (section 850)

The ROB Department gave a verbal confirmation, per Chad Jenkins, that these changes were approved and to move forward for registration.

The ROB Department created CARFs to Michael's requests listed above for:

- ROB 380 – Cross listed with EECS 367
  - The EECS Department was included in conversations as they are now the cross listed department for this course. Previously EECS was the home department for this course, but the CARF modification submitted was to change the home department.
- ROB 320
- ROB 511

The ROB 203 CARF had a response from the CoE CC member for ENGR and UG Education (Rachael Schmedlen):

I have a couple of concerns regarding the ROB 103/203 CARF and moving forward with adding a credit exclusion to ENGR 100.850 (the robotics section). The ROB 103/203 CARF indicates that ROB 203 will be a "lab-only offering of Engineering 100-850" for students who took a different section.

1.) Adding a credit exemption to one section of ENGR could set an undesirable precedent that would allow other departments to create/offer courses with elements from their ENGR sections that would require credit exemptions. This could potentially put students at a disadvantage if they don't or cannot (e.g. the section is full) take a specific ENGR 100 section associated with the department they intend to declare their major. ENGR 100 is intended as a course for students to explore their technical and personal interests and every section adequately prepares students for any CoE major. It seems a fine line to tread by putting course qualifiers to individual sections, even if it is only for elective ROB (or departmental) courses.

2.) If ROB 203 is intended to be an elective lab course for ROB majors who took a different section, how would this course be counted towards the major? In the sample schedules listed in the [Robotics Undergraduate Program Guide](#) "technical electives" must be 300+ or 400+ level, so this lab would not fulfill the requirement and it isn't obvious where else it could count.

As an aside, the [Robotics Undergraduate Program Requirements webpage](#) lists as a core requirement, "Introduction to Engineering: Engineering 100 or introductory engineering equivalent". What is considered an "introductory engineering equivalent"? I'm not aware of what course(s) could substitute for ENGR 100 in the Robotics program.

Xiaogan responded with:

If ROB 203 is designed only for the students who have already taken Engr.100 but didn't get the credits from Engr.100 Sec.850, we could place an enforced prerequisite like **"Engr.100, but no credit in Engr.100 Sec. 850"** only for ROB 203. In this way, we may not need to put **"no credit in ROB 203"** for Engr.100 Sec.850 (i.e., no students can take ROB203 prior to Engr. 100). This will not generate any bias for first-year students to select among different Engr.100 sections. Some departments do have the post-Engr.100 courses that involve elements similar to those of the Engr.100 sections. Such redundancy is necessary for the students who miss the chance to take the corresponding Engr.100 section but still hope to make up specific technical skills. I think most students can benefit from such settings. However, this is just my opinion. We could bring this topic to the CCC meeting for a more extensive discussion.

This response was confirmed by the URO that this could work, and the Enforced Prerequisite should read, "ENGR 100 and no credit in ENGR 100, Topic 29 'Robotics Mechanisms'".



Betsy requested that the ROB department provide their approval and have their CARF created in order for these changes to be implemented for the course.

ROB approved to proceed forward with the changes as requested and forwarded to the URO to immediately implement in the system. CARF to follow from the department for reference to the URO that all changes were made and documented.

- Last update found to change language to “ENGR 100; (C- or better, no OPF) and No Credit in ENGR 100 Topic 29 Robotics Mechanisms)
- CARF signatures are being obtained by the department
- CoE CC Member for ENGR and UG Education confirms that UG Office that ROB 203 counts as a General Elective

Due to the changes that were needed, there has been an update to the CCC Process for Course-Related Requests Document on the CoE CC website for Credit Exclusions for Special Course topics and listing multiple courses.

## HLC Annual Audit Questions for the CoE Curriculum Committee

1. **How should departments handle courses that are taught in combination with other institutions? How should departments handle it when the course is taught at UM? "Do courses that are a teaching collaborative need to follow CoE Policy for the Assignment of Credit Hours? Examples: ROB 498 and 599 (Robotics)**

*ROB 498.004/ROB 599.010 was offered as part of our distributed teaching collaborative and was a course offered between U-M and FAMU. We aligned our course to the FAMU scheduling as their instructor was teaching the course. They plan to run this in WN 24 ROB 498.015/ROB 599.015*

2. **When was the lab policy established? (IOE)**

*The Current CoE Policy for the Assignment of Credit Hours was approved October 13, 2020.*

3. **Do labs need to be scheduled in a formal CoE Computer Lab space when the work can be done online? (EECS)**

*Atul Prakash: I do think the definition of a lab course and the way contact hours are measured is not ideal for software courses in which the lab work can be done virtually at any time and any place by the students. I don't think it is ideal even for hardware courses in which students are able to use a virtual or portable kit and thus a physical lab is less critical. I would recommend another way to designate a course as a lab in CoE so that the courses can get sufficient SCH credit for supporting the teaching staff. A possible way to think about it is if the students are building real or virtual artifacts with software and hardware that requires technical support.*

**Question for CCC discussion:** Does the CCC agree with the use of virtual technical support? How would that look, would students receive immediate feedback?

4. **Are the activities associated with the online, self-paced, asynchronous Canvas modules used in ENGR 101 and 110 and other departmental courses acceptable as CoE contact hours?** These course use contact hours as follows, per feedback from Rachael Schmedlen

The following feedback to define contact hours was gathered from Christne Gerdes, one of the Office of the Provost's identified curriculum specialists for HLC project, regarding guidance on using online self-paced, asynchronous Canvas modules as contact hours:

**[The Office of the Provost Guidance on Defining the Academic Credit Hour](#) states: Faculty and instructors — with oversight and input from faculty-led curriculum committees — should determine the activities that would appropriately be viewed as faculty-led engagement within the context of a course and academic program.**

**Contact hours are defined as time spent by students engaged with the course instructor. This is academic engagement. Hybrid and online courses require an equivalent amount of instruction and student work as required by in-person courses.**

**Engagement with the course instructor/academic engagement is defined by federal guidance, and to be considered a contact hour, the activity in question must follow the regulations under letter (a):**

**(a) Participation in an interactive tutorial, webinar, or other interactive computer-assisted instruction**

**If it meets the standard of (a), as per CoE Curriculum Committee determination, then it is a contact hour. Letter (b) listed below would not follow the determination for the CoE contact hour:**

**(b) Logging into an online class or tutorial without any further participation**

## **Updates for the CoE Curriculum Committee on the HLC Annual Audit Process**

1. Update on the work behind the scenes to better the process. Some members of the HLC Working Group met with Andrea Bolash (CoE Director of Institutional Research RPM) and her team to determine if we can modify the current Annual HLC Audit Report provided by the URO/ITS to make the information more digestible and accurate vs. creating one on our own. The decision was made to try to work with URO/ITS to include new data fields for the individual course components of each course with their associated credit hours to make the data accurate.

**\*Further update from ITS/University Office of the Registrar** regarding HLC Annual Audit Report request by CoE to create data fields in MPathways that would create correct data in the report and an error message when Curriculum Maintainers add course components for a course that doesn't have them. Betsy also asked how the college should be storing information that shows a course was in compliance and what level of detail is needed.

### **Brad Maki, Senior Associate Registrar, responded with the following:**

I think the first thing that needs to be stated is that this report is not an "audit" in that CoE is expected to react in some specified manner and provide an answer back to an auditor. This report is informational in nature, intended for the college to help identify courses that may not be meeting your credit hour policy specifications. From what we are hearing around campus, it is working well for most. We suspect that there are some CoE courses with an uncommon setup that are causing a plethora of rows on the report.

The way the report is built, not all of the data about the courses and classes is available in the way needed to make it more accurate. Rebuilding the report would require significant effort that the Technology team in Enrollment Management just doesn't have the capacity to do.

After discussions with several people in the RO and Enrollment Management, I think the best course of action for you if you are seeing the same issues is:

1. Make brief notes about why some courses are "false positives" in the report. Keep those notes available should the HLC ever revisit this topic with the university. While we were required to give a report for the 2020 site visit, there was no requirement from the HLC that we are expected to give further reports in the future. We were to create policies about credit hours (which we did) and ensure that there was a mechanism in place for schools and colleges to review compliance with their policies (which we did with these reports). It seems you have already done a great job identifying examples of courses (based on your list), and that you need to make short notes about each course (or grouping of courses) and then keep that handy somewhere.
2. Consult with someone on the Curriculum team to talk through some of the scenarios you listed (such as study abroad placeholders). Lisa suggested this might be helpful in reducing the number of courses flagged in the report. Of course, that would require some work on the college's part in working with all of your departmental curriculum

**maintainers**, and that may not be an effort you want to undertake. But if you'd like to have the conversation, you can reach out to Lisa.

**\*\*Further update from LSA:**

Betsy reached out to LSA to determine how they handle data entry of each term's sections and if the issues that CoE are experiencing is an issue for them. **Here is Ashlee Wolfe's response:**

Time schedules are handled by our units; each unit has a curriculum maintainer that is either solely theirs or shared between two units dependent on the administrative structure. (The sharing usually only happens in smaller units.) MPathways does have certain "flexibilities" where it can allow departments to completely disregard how the course is set up on a CARF. The main thing is with components. It is true that departments can adjust which components are on their courses without ever coming to us for approval. I have to say that the **majority of units are pretty good about asking when they want to change up the structure of a class regardless of this and will seek College approval.** There is just the one-off issue here and there. **Our Course Maintenance system does have reporting capabilities to catch some errors-- like when meets-togethers are set up and the credit load or distribution doesn't match up. Also, it catches when the MPathways entry doesn't match what is on the CARF at the time of processing either modifications or new course CARFs.** But we haven't bothered to create a report for components that don't match their CARF to run against the time schedule. I think we could if we needed to, but College culture hasn't made this a problem, at least at the point of my measly 1-year in the job.

2. The HLC Working Group is awaiting spreadsheet responses from MATSCIE and UARTS.
3. A new possible workaround was discovered for departments/units that need to create courses in which the classroom location is to be determined or not needed. This is included below:

Possible workaround for the Curriculum Maintainer in departments/units:

- Set up course in *Maintain Schedule of Classes* each term with Instruction Mode: P (In Person) in Basic Data Tab
- In the Meetings Tab, make sure to set up days and times consistent with CoE's Standard Meeting Pattern, then under Facility ID, put ARR (Arrange) instead of NEEDENG (used to assign classrooms in 25Live scheduling software) so a room is not assigned.
- Students would need to [Override the Time Conflict](#) if one exists, but it will show in HLC Annual Audit Report as meeting the required minimum of contact hours.

This isn't ideal due to the extra step for students (overriding the time conflict) and possibly needing to be explained further by the department if audited.

## **Proposal to Update CoE Minors Policy - Double Counting Policy Discrepancy**

Two Policies in conflict (conflicting information in **yellow highlight**):

### **1. CoE Policy for Engineering Minors (approved by the CoE Curriculum Committee on February 22, 2022)**

#### COLLEGE OF ENGINEERING POLICY FOR ENGINEERING MINORS

Approved by the College of Engineering Curriculum Committee February 22, 2022

Undergraduate students enrolled in a College of Engineering degree program can often benefit from study and practice at some depth outside of their major. An engineering minor is a coherent program of study, but with requirements far less comprehensive than those of a BS or BSE degree. Engineering minors can be sponsored by CoE departments, programs, or, for the purpose of supporting cross-departmental programs, sponsored by the Office of the Associate Dean for Undergraduate Education. An engineering minor is not intended to provide specialization within a student's major field.

An engineering minor will require:

1. At least 15 credit hours, including some upper division courses.
2. A minor must contain some structure and coherence and cannot simply be a requirement for a number of credit hours. A minor can contain grouping of courses and provide students with approved menus of courses within these groups.
3. Courses used to satisfy BSE or BE requirements can also be used to satisfy minor requirements.
4. Courses taken to meet the requirements of a minor must be taken for a grade. However a minor may include pass/fail courses if justified by the sponsoring department
5. An engineering minor can require prerequisites.
6. To create a minor the sponsoring CoE unit will submit a curricular plan for the minor to the College of Engineering Curriculum Committee for consideration and approval/disapproval. If approved the engineering minor proposal will be forwarded to the CoE faculty for final approval/disapproval. The CoE Curriculum Committee must approve all subsequent modifications to the minor requirements (including adding or removing courses from a menu of courses).
7. If a department wishes to allow transfer credit to count toward the minor this must be specified and justified as part of the proposal for the minor.
8. The sponsoring unit is responsible for administering the minor, and must submit with the curricular plan an administrative plan that: identifies faculty/staff positions responsible for providing advising and timely auditing of the minor

requirements during the student's final term. The advisor for an engineering minor will be responsible for approving variances to minor requirements for individual students.

9. For minors sponsored by the Office of the Associate Dean for Undergraduate Education, the ADUE will identify faculty to provide advising and staff to assist with advising and auditing.
10. Students taking an engineering minor will have a notation on their CoE audit. When the student applies for her diploma the CoE registrar will contact the unit(s) sponsoring her minor(s) to request an audit of the minor requirements.
11. The program advisors for a minor, and their designated staff, are responsible for responding to the CoE Registrar's request for the audit of the minor requirements.

Rules:

- A. Completion of an academic minor is optional; no student can be required to complete an engineering minor.
- B. A student can complete one or more engineering minors, along with one or more LSA minors. However, a minor is not intended to provide specialization within a student's major field. Therefore, the posted rules for each minor will outline any restrictions on the availability of a minor for students in a particular program, e.g. "A student seeking a BSE in NERS cannot earn the minor in NERS.
- C. Advanced placement credits may not be used to meet the requirements of an academic minor, but may be used to meet the prerequisites to a minor. D. Transfer credit may not be used to fulfill the requirements of a minor unless specifically stated in the minor requirements.
- E. Courses taken to satisfy the requirements of a minor must be taken for a grade, unless the course was specifically approved as Pass/Fail within the requirements of that minor.
- F. A student must earn an overall GPA of at least 2.0 in courses taken to meet the requirements of an academic minor.
- G. Students are responsible for notifying both the sponsoring program and their major department of their intention to pursue a minor. Such notification should take place prior to enrollment in the upper division courses for the minor.
- H. The advisor for each minor is responsible for approving any variance in course requirements for a minor.
- I. Responsibility for auditing completion of requirements for a minor lies within the sponsoring unit.
- J. Students will always be allowed to count courses towards an approved minor retroactively, as long as the minor is approved before the date of their graduation.
- K. Engineering minors will be noted on a student's academic transcript, but not on her diploma.
- L. Students may not add, complete, or declare a minor after graduation.

Notes on significant differences from LSA minors

- I. LSA minors require 10 credits to be taken in residence (we require all courses

- to be taken in residence unless specifically allowed by rules of the minor).
- II. LSA minors and concentrations (majors) allow double counting only one course (we allow any number of courses to be used for both major and minor; this is consistent with current CoE policy regarding LSA minors).
- III. LSA minors do not allow any double counting of courses between minors (we allow any number of courses to be double counted between minors).
- IV. LSA minors do not allow any Pass/Fail grading in courses used for a minor (we allow some Pass/Fail graded courses in a minor, if specifically approved in the design of the minor).

## 2. [Double Counting Courses - Degree Requirements page of CoE Bulletin](#)

### Double Counting Courses

For Engineering Undergraduate students:

1. No course can count towards more than two degrees, such as two bachelor's degrees or one bachelor's degree and one master's degree.
2. No course can count towards more than two majors, and no more than one major and one minor. A course can only count toward two minors if the credits received for that course are beyond the 128 credits required for the student's major.
3. No course can count towards more than one requirement within a BSE program. Double counting a course for credit towards more than one requirement is not allowed. A single course can qualify to meet the requirements of multiple sections of the BSE, but must formally count towards one requirement.
4. A course can count toward Supplemental Studies, Concentrations/Sub-Plans or Honors Programs regardless of how many majors or minors it is already counting towards.
5. SUGS (Sequential Undergraduate Graduate Studies) students who are pursuing a Rackham or College of Engineering master's degree can only double count courses that are considered general electives or technical electives and are not part of the core requirement. The number of credits allowed to double count is determined by the individual departments, with 9 credits being the maximum allowed by Rackham. MDDP (Multiple Dependent Degree Program, typically referred to as Dual Degree) students are not eligible for the SUGS program.



6. Non-SUGS students pursuing a master's degree from Rackham or the College of Engineering, cannot transfer any credits used for their bachelor's degree, even free electives or tech electives, toward their master's degree. They may only transfer credits from courses that were not used to fulfill requirements for a degree or certificate (verified by an Undergraduate Program Advisor). Furthermore, the transferred credits must be from graduate-level courses and Rackham must receive confirmation (from their Graduate Coordinator) that these courses were at the graduate level and required graduate-level effort.

### **Summary**

The two policies highlighted in yellow are in conflict with one another. "The CoE Policy for Engineering Minors" is missing the details regarding double counting of credit between two CoE minors that is included in the CoE Bulletin "Double Counting Courses" section, which indicates the following:

A course can only count toward two minors if the credits received for that course are beyond the 128 credits required for the student's major.

### **Vote**

To help rectify the discrepancy between the two policies, the CoE Curriculum Committee is being asked to vote on the approval of the following policy update, which will update point III of the "Notes on significant differences from LSA Minors" in the [CoE Policy for Engineering Minors](#) to the following verbiage:

**III. LSA minors do not allow any double counting of courses between minors. For CoE minors, a course can only count toward two minors if the credits received for that course are beyond the 128 credits required for the student's major.**

Prior language for reference:

III. LSA minors do not allow any double counting of courses between minors.(we allow any number of courses to be double counted between minors).

COLLEGE OF ENGINEERING  
POLICY FOR ENGINEERING MINORS

Approved by the College of Engineering Curriculum Committee February 22, 2022

Undergraduate students enrolled in a College of Engineering degree program can often benefit from study and practice at some depth outside of their major. An engineering minor is a coherent program of study, but with requirements far less comprehensive than those of a BS or BSE degree. Engineering minors can be sponsored by CoE departments, programs, or, for the purpose of supporting cross-departmental programs, sponsored by the Office of the Associate Dean for Undergraduate Education. An engineering minor is not intended to provide specialization within a student's major field.

An engineering minor will require:

1. At least 15 credit hours, including some upper division courses.
2. A minor must contain some structure and coherence and cannot simply be a requirement for a number of credit hours. A minor can contain grouping of courses and provide students with approved menus of courses within these groups.
3. Courses used to satisfy BSE or BE requirements can also be used to satisfy minor requirements.
4. Courses taken to meet the requirements of a minor must be taken for a grade. However a minor may include pass/fail courses if justified by the sponsoring department
5. An engineering minor can require prerequisites.
6. To create a minor the sponsoring CoE unit will submit a curricular plan for the minor to the College of Engineering Curriculum Committee for consideration and approval/disapproval. If approved the engineering minor proposal will be forwarded to the CoE faculty for final approval/disapproval. The CoE Curriculum Committee must approve all subsequent modifications to the minor requirements (including adding or removing courses from a menu of courses).
7. If a department wishes to allow transfer credit to count toward the minor this must be specified and justified as part of the proposal for the minor.
8. The sponsoring unit is responsible for administering the minor, and must submit with the curricular plan an administrative plan that: identifies faculty/staff positions responsible for providing advising and timely auditing of the minor requirements during the student's final term. The advisor for an engineering minor will be responsible for approving variances to minor requirements for individual students.
9. For minors sponsored by the Office of the Associate Dean for Undergraduate Education, the ADUE will identify faculty to provide advising and staff to assist with advising and auditing.
10. Students taking an engineering minor will have a notation on their CoE audit. When the student applies for her diploma the CoE registrar will contact the unit(s) sponsoring her minor(s) to request an audit of the minor requirements.

11. The program advisors for a minor, and their designated staff, are responsible for responding to the CoE Registrar's request for the audit of the minor requirements.

Rules:

- A. Completion of an academic minor is optional; no student can be required to complete an engineering minor.
- B. A student can complete one or more engineering minors, along with one or more LSA minors. However, a minor is not intended to provide specialization within a student's major field. Therefore, the posted rules for each minor will outline any restrictions on the availability of a minor for students in a particular program, e.g. "A student seeking a BSE in NERS cannot earn the minor in NERS.
- C. Advanced placement credits may not be used to meet the requirements of an academic minor, but may be used to meet the prerequisites to a minor. D. Transfer credit may not be used to fulfill the requirements of a minor unless specifically stated in the minor requirements.
- E. Courses taken to satisfy the requirements of a minor must be taken for a grade, unless the course was specifically approved as Pass/Fail within the requirements of that minor.
- F. A student must earn an overall GPA of at least 2.0 in courses taken to meet the requirements of an academic minor.
- G. Students are responsible for notifying both the sponsoring program and their major department of their intention to pursue a minor. Such notification should take place prior to enrollment in the upper division courses for the minor.
- H. The advisor for each minor is responsible for approving any variance in course requirements for a minor.
- I. Responsibility for auditing completion of requirements for a minor lies within the sponsoring unit.
- J. Students will always be allowed to count courses towards an approved minor retroactively, as long as the minor is approved before the date of their graduation.
- K. Engineering minors will be noted on a student's academic transcript, but not on her diploma.
- L. Students may not add, complete, or declare a minor after graduation.

Notes on significant differences from LSA minors

- I. LSA minors require 10 credits to be taken in residence (we require all courses to be taken in residence unless specifically allowed by rules of the minor).
- II. LSA minors and concentrations (majors) allow double counting only one course (we allow any number of courses to be used for both major and minor; this is consistent with current CoE policy regarding LSAMinors).
- III. LSA minors do not allow any double counting of courses between minors (we allow any number of courses to be double counted between minors).
- IV. LSA minors do not allow any Pass/Fail grading in courses used for a minor (we allow some Pass/Fail graded courses in a minor, if specifically approved in

the design of the minor).



## Course Approval Request Form

Office of the Registrar, University of Michigan

1210 LSA Building  
500 S. State Street  
Ann Arbor, MI 48109-1382  
Phone: 734.763.2113  
Fax: 734.936.3148  
ro.curriculum@umich.edu  
ro.umich.edu

CHECK APPROPRIATE BOXES FOR ALL CHANGES

### Action Requested

- New Course  
 Modification of Existing Course

Date of Submission: 2023-11-03

Effective Term: Fall 2024

- Deletion of Existing Course

<input checked="" type="checkbox"/>	Course Offered <input checked="" type="checkbox"/> Indefinitely <input type="checkbox"/> One term only	<b>RO USE ONLY</b> Date Received: Date Completed: Completed By:
-------------------------------------	--	--

### CURRENT LISTING

### REQUESTED LISTING


<input checked="" type="checkbox"/>	Dept (Home): Subject: Catalog:	Dept (Home): Elec Engin & Computer Sci Subject: EECS Catalog: 440												
<input type="checkbox"/>	<input type="checkbox"/> Course is Cross-Listed with Other Departments	<input type="checkbox"/> Course is Cross-Listed with Other Departments												
<input type="checkbox"/>	<table border="1" style="width: 100%;"> <thead> <tr> <th style="width: 25%;">Department</th> <th style="width: 25%;">Subject</th> <th style="width: 50%;">Catalog Number</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table>	Department	Subject	Catalog Number				<table border="1" style="width: 100%;"> <thead> <tr> <th style="width: 25%;">Department</th> <th style="width: 25%;">Subject</th> <th style="width: 50%;">Catalog Number</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table>	Department	Subject	Catalog Number			
Department	Subject	Catalog Number												
Department	Subject	Catalog Number												
<input checked="" type="checkbox"/>	Course Title (full title)	Course Title (full title) Extended Reality for Social Impact												
<input checked="" type="checkbox"/>	Abbreviated Title (20 char)	Abbreviated Title (20 char) Extended Reality												
<input checked="" type="checkbox"/>	Course Description (Please limit to 80 words and attach separate sheet if necessary) Design, development, and application of virtual and augmented reality software for social impact. Topics include: virtual reality, augmented reality, game engines, ethics / accessibility, interaction design patterns, agile project management, stakeholder outreach, XR history / culture, and portfolio construction. Student teams develop and exhibit socially impactful new VR / AR applications.													
<input checked="" type="checkbox"/>	Full Term Credit Hours Undergraduate Min: 4      Graduate Min: Undergraduate Max: 4      Graduate Max:	Half Term Credit Hours Undergraduate Min:      Graduate Min: Undergraduate Max:      Graduate Max:												
<input checked="" type="checkbox"/>	Course Credit Type Undergraduate Student													
<input type="checkbox"/>	Repeatability <input type="checkbox"/> Course is Repeatable for Credit <input type="checkbox"/> Course is Y graded Maximum number of repeatable credits: <input type="checkbox"/> Can be taken more than once in the same term													

	Subject:	Catalog:		
<input checked="" type="checkbox"/>	<b>Grading Basis</b> <input checked="" type="checkbox"/> Graded (A – E) <input type="checkbox"/> Credit/No Credit <input type="checkbox"/> Satisfactory/Unsatisfactory <input type="checkbox"/> Pass/Fail <input type="checkbox"/> Business Administration <b>Grading</b> <input type="checkbox"/> Not for Credit <input type="checkbox"/> Not for Degree Credit <input type="checkbox"/> Degree Credit Only			
	<b>Add Consent</b> <input type="checkbox"/> Department Consent <input type="checkbox"/> Instructor Consent <input checked="" type="checkbox"/> No Consent		<b>Drop Consent</b> <input type="checkbox"/> Department Consent <input type="checkbox"/> Instructor Consent <input checked="" type="checkbox"/> No Consent	

	CURRENT LISTING	REQUESTED LISTING
<input type="checkbox"/>	Advisory Prerequisite (254 char)	Advisory Prerequisite (254 char)
<input checked="" type="checkbox"/>	Enforced Prerequisite (254 char) Minimum grade requirement:	Enforced Prerequisite (254 char) EECS 281; (C or better, No OP/F) Minimum grade requirement: C
<input type="checkbox"/>	Credit Exclusions	Credit Exclusions
<input checked="" type="checkbox"/>	<b>Course Components</b> <input checked="" type="checkbox"/> Lecture <input type="checkbox"/> Seminar <input type="checkbox"/> Recitation <input type="checkbox"/> Lab <input type="checkbox"/> Discussion <input type="checkbox"/> Independent Study	<b>Graded Component</b> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
	<b>Terms Typically Offered</b> <input checked="" type="checkbox"/> Fall <input checked="" type="checkbox"/> Winter <input type="checkbox"/> Spring <input type="checkbox"/> Summer <input type="checkbox"/> Spring/Summer	
Cognizant Faculty Member Name: Austin Yarger		Cognizant Faculty Member Title:

**SIGNATURES ARE REQUIRED FROM ALL DEPARTMENTS INVOLVED (Please Print AND Sign Name)**

Contact Person: Punam Vyas                      Email: vyas@umich.edu                      Phone: 647-1754

CoE Curriculum Committee Representative:                       Print: Amir Kamil                      Date: 11/07/23

CoE Curriculum Committee Chair: \_\_\_\_\_                      Print: \_\_\_\_\_                      Date: \_\_\_\_\_

Home Department Chair:                       Print: Andrew DeOrio                      Date: 11/07/2023

Cross-Listed Department Chair: \_\_\_\_\_                      Print: \_\_\_\_\_                      Date: \_\_\_\_\_

Cross-Listed Department Chair: \_\_\_\_\_                      Print: \_\_\_\_\_                      Date: \_\_\_\_\_

Cross-Listed Department Chair: \_\_\_\_\_                      Print: \_\_\_\_\_                      Date: \_\_\_\_\_

**DEPARTMENTAL/COLLEGE USE ONLY**

**Current:****Requested:**Course DescriptionCourse Description

Design, development, and application of virtual and augmented reality software for social impact. Topics include: virtual reality, augmented reality, game engines, ethics / accessibility, interaction design patterns, agile project management, stakeholder outreach, XR history / culture, and portfolio construction. Student teams develop and exhibit socially impactful new VR / AR applications.

Class LengthClass Length

Full term

Contact hours (lecture):Contact hours (lecture):

4

Contact hours (recitation)Contact hours (recitation)Contact hours (lab)Contact hours (lab)**Additional Info:**Submitted by:

Home dept

Describe how this course fits with the degree requirements:

Capstone/MDE elective for CS majors.

Special resources of facilities required for this course:Supporting statement:

The introduction of a permanent new capstone / MDE course will relieve some MDE waitlist pressure within the department (and in particular, on EECS 497, which has been prompted to scale beyond what a design-oriented MDE can reasonably be asked to support).

The highly-multidisciplinary nature of the subject matter (XR development often requiring significant collaboration with and understanding of artists, musicians, etc) will advance CSE's collaboratory mission by promoting a more inclusive, diverse, and collaborative mindset among our junior and senior students (especially when interacting with peers from other departments, such as SI). The course's focus on social impact, as opposed to the more traditional recreational / entertainment uses of VR and AR, will contribute positively to department culture as well.

This capstone course has a uniquely intense hardware requirement (VR headsets, gaming-grade computing technology). These requirements deeply embed the course with campus emerging technology groups and facilities (such as the Duderstadt Visualization Studio), increasing CSE's presence in, usage of, and influence with these impactful groups and resources.

Past offerings:

Fall 2022: 35 students  
Winter 2023: 50 students  
Fall 2023: 37 students



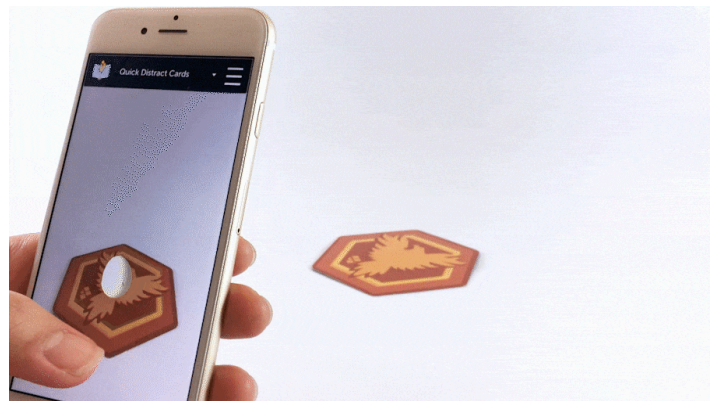


# EECS 498 : Extended Reality and Society

## Syllabus

Instructor : Austin Yarger ([ayarger@umich.edu](mailto:ayarger@umich.edu) - [ayarger.com](http://ayarger.com))

*Note : This syllabus is for a developing course, and is subject to change. Visit [bit.ly/umich\\_xr](http://bit.ly/umich_xr) for latest*



The *emerging*, diverse field of XR holds tremendous potential for positive social impact as seen in local startups like [Spellbound AR](#) (right), which uses augmented reality to improve the pediatric medical experience for children. Our semester concludes with an XR exhibition (left) in which community members partake in our XR experiences (including [Urban Rush](#), the VR fitness adventure you see above on the left).

Motivation / Course Contents	3
<b>Approximate Schedule / Grading</b>	<b>4</b>
Assignments	4
Assignments / Lesson Plans / Lecture Slides (TBD)	5
Grading Scale	6
<b>Prerequisites</b>	<b>6</b>
Required Courses	6
Book Requirement	6
Computation Requirement	6
Local Storage Resources	7
Equipment Lottery	7
Workload	7

Graduate Student Advisory	8
<b>Resources</b>	<b>8</b>
Community	8
Health	8
<b>Policies</b>	<b>8</b>
Health and Safety	9
Lecture Recording	9
Submission	9
Late Assignment Submission	9
Incorrect Submission Format Penalty	10
Penalty Stacking	10
Standard Grading Environment	10
Imbalanced Team Contribution Penalty	10
Attendance / Late Arrival	11
Regrades / Grading Disputes	11
Office Hours	11
Extra Credit	11
Respect and Maturity	11
Collaboration / Cheating / Honor Code	12
<b>Staff AutoBios</b>	<b>12</b>
Austin Yarger	12
Nithisha Kumar	13
Rahmy Salman	13
<b>Discussion</b>	<b>13</b>
Relation to EECS 494 : Introduction to Game Development	14
I have Extra Time– How May I Prepare for the Course?	14

# Motivation / Course Contents

From [pediatric medical care](#), [advanced manufacturing](#), and [commerce](#) to [film analysis](#), [first-responder training](#), and [unconscious bias training](#), the fledgling, immersive field of extended reality may take us far beyond the realm of traditional video games and entertainment, and into the realm of diverse social impact.

“EECS 498 : Extended Reality and Society” is a programming-intensive senior capstone / MDE course that empowers students with the knowledge and experience to...

- Implement medium-sized virtual and augmented reality experiences using industry-standard techniques and technologies.
  - Game Engines (Unreal Engine / Unity), Design Patterns, Basic Graphics Programming, etc.
- Design socially-conscious, empowering user experiences that engage diverse audiences.
- Contribute to cultural discourse on the hopes, concerns, and implications of an XR-oriented future.
  - Privacy / security concerns, XR film review (The Matrix, Black Mirror, etc)
- Carry out user testing and employ feedback after analysis.
  - Requirements + Customer Analysis, Iterative Design Process, Weekly Testing, Analytics, etc.
- Work efficiently in teams of 2-4 using agile production methods and software.
  - Project Management Software (Jira), Version Control (Git), Burndown Charting and Resource Allocation, Sprints, etc.

Students will conclude the course with at least three significant, socially-focused XR projects in their public portfolios.

**Note :** This is not a crypto or metaverse course, though both will be discussed during a lecture on ethics and the technology hype cycle.

## Approximate Schedule / Grading

### Assignments

The approximate value of EECS 498’s assignments are listed below. Each project is represented by a (often substantial) set of sub-assignments (see further below).

Assignment	Est. Grading Weight
<b>Exploration Assignments (week 1)</b> <a href="#">(exploration_vr)</a> <a href="#">(exploration_ar)</a>  Exploratory assignments. Hands-on with modern VR, AR, and campus dev resources. Team size : 1, Duration : 1 week	10%
<b>Tutorial Assignments (week 2)</b> <a href="#">(tutorial_vr)</a> <a href="#">(tutorial_ar)</a>  Heavily-guided development assignments. Intro to Unreal Engine and Unity. Team size : 1, Duration : 1 weeks	10%
<b>Project 1 - VR : CSE Simulator (weeks 3-5)</b> <a href="#">(p1_milestone)</a> <a href="#">(p1_alpha)</a> <a href="#">(p1_gold)</a> <a href="#">(p1_pm)</a> <a href="#">(p1_postmortem)</a>	15%

<p>Exercise virtual reality design techniques and technologies in the creation of a <a href="#">Job Simulator</a> variant titled "CSE Simulator". Team size : 2, Duration : 3 weeks</p>	
<p><b>Project 2 - AR : A2-Go! (weeks 7-9)</b> (<a href="#">p2_milestone</a>) (<a href="#">p2_alpha</a>) (<a href="#">p2_gold</a>) (<a href="#">p2_pm</a>) (<a href="#">p2_postmortem</a>)</p> <p>Exercise augmented reality design techniques and technologies in the creation of a <a href="#">Pokemon Go</a> variant titled A2-Go! Team size : 2, Duration : 3 weeks</p>	15%
<p><b>Project 3: Showcase Impact Project (weeks 11-15)</b> (<a href="#">p3_milestone</a>) (<a href="#">p3_milestone_2</a>) (<a href="#">p3_alpha</a>) (<a href="#">p3_gold</a>) (<a href="#">p3_pm</a>) (<a href="#">p3_postmortem</a>) (<a href="#">p3_research</a>) (<a href="#">p3_marketing</a>)</p> <p>Exercise design, implementation, and project management knowledge in the creation of a substantial, custom, socially-impactful VR or AR project. Team size : 4-5, Duration : 5 weeks (with some preparation in earlier weeks)</p>	37.5%
<p><b>Culture</b></p> <p>Film study. Cultural / artistic explorations of XR's potential impact on the future of society.</p>	10%
<p><b>Participation</b></p> <p>Engage in lectures, user testing sessions, and team projects with congeniality. May go negative and have impact beyond 2.5% of overall grade.</p>	2.5%

## Assignments / Lesson Plans / Lecture Slides (TBD)

The course schedule is subject to industry interviews, field trips, and vacation days that may be difficult to predict--causing small variations from semester to semester.



- [Past : F22 Schedule](#)
- [Past : W23 Schedule](#)

Date	Lecture Plan	A-s-s-i-g-n-m-e-n-t-s				
<u>Wed, 01/04/23</u>	<a href="#">Lecture 1</a>	<a href="#">exploration_vr</a> (7 days)	<a href="#">exploration_ar</a> (7 days)	<a href="#">Equipment Lottery</a>		
<u>Mon, 01/09/23</u>	<a href="#">Lecture 2</a>					
<u>Wed, 01/11/23</u>	<a href="#">Lecture 3</a>				<a href="#">tutorial_vr</a> (7 days)	<a href="#">tutorial_ar</a> (7 days)
<u>Mon, 01/16/23</u>	<b>MLK Day</b> (no lecture)		<a href="#">Join P1 Team</a> (size:2)			
<u>Wed, 01/18/23</u>	<a href="#">Lecture 4</a>	<a href="#">p1_milestone</a> (7 days)		<a href="#">p1_pm</a> (21 days)		
<u>Mon, 01/23/23</u>	Evening Film (no lecture)					7pm film 1670 BBB
<u>Wed, 01/25/23</u>	<a href="#">Lecture 5</a>		<a href="#">p1_alpha</a> (7 days)		62 69 74 2e 6c 79 2f 79 78 33 39 64 6f	
<u>Mon, 01/30/23</u>	<a href="#">Lecture 6</a>					
<u>Wed, 02/01/23</u>	<a href="#">Lecture 7</a>	<a href="#">p1_gold</a> (7 days)				<a href="#">culture_2</a>
<u>Mon, 02/06/23</u>	Evening Film (no lecture)					7pm film 1014 DOW
<u>Wed, 02/08/23</u>	<a href="#">Lecture 8</a>				<a href="#">Join P2 Team</a> (size:2)	<a href="#">p1_postmortem</a> (5 days)

A portion of [Winter 23's EECS 498 schedule document](#). Read top-to-bottom, green means an assignment is launching and red means it is due. Find today's date in the leftmost column, then move your eyes right to see what you should be working on. Lecture plans, including links to slides, announcements, etc, are available in the second-from-right column.

## Grading Scale

EECS 498 employs the standard grading scale below in converting scores to letter grades.

View/Edit Grading Scheme <span style="float: right;">✕</span>		
	Default Grading Scheme	<a href="#">Select Another Scheme</a>  
Name:	Range:	
A+	100%	to 97%
A	< 97%	to 93%
A-	< 93%	to 90%
B+	< 90%	to 87%
B	< 87%	to 84%
B-	< 84%	to 80%
C+	< 80%	to 77%
C	< 77%	to 74%
C-	< 74%	to 70%
D+	< 70%	to 67%
D	< 67%	to 64%
D-	< 64%	to 61%
F	< 61%	to 0%

[manage grading schemes](#) Done

Final grade projections are released on Canvas in the final several weeks of the course. Students are granted a period of at least 24 hours to review and dispute final grades before their submission to Wolverine Access.

## Prerequisites

### Required Courses

Students are required to have successfully passed **EECS 281** before taking this course. Non-CSE undergraduates / graduate students with ample programming experience may request permission without EECS 281, though CSE undergraduates must be prioritized (this course may be necessary for them to graduate).

### Book Requirement

There is no book requirement.

### Computation Requirement

EECS 498 assignments require significant use of modern VR hardware, graphics-intensive [3D software](#) (Unity, Unreal), and access to an AR-capable mobile device ([android phone list](#), [apple device list](#)). Students are expected to make ample, recurring use of the [Duderstadt Center Visualization Studio](#) which contains gaming-grade computers and specialized VR hardware.

Students with personal access to this kind of hardware may use it outside of the Visualization Studio to progress on course assignments and projects. In the first week, students may enter an equipment lottery for a chance to acquire temporary, take-home VR headsets.

Students without access to an AR-capable mobile device are encouraged to obtain one (making a purchase, request one temporarily from a friend, etc) or consider a different course.

Some combinations of equipment / devices will produce more friction than others.

[Check the platform pain matrix to learn more.](#)

## Local Storage Resources

Beyond the initial Unreal Engine installation (~50GB) and Unity installation (20GB), XR projects are notorious for large storage footprints and fast growth (especially when employing high-quality asset packs). If you plan to use a personal device for development during this course, we recommend allocating ~100GB of local hard drive space if possible. [WinDirStat](#) may help Windows users find this space, while [Disk Inventory X](#) may serve Mac users.

## Equipment Lottery

Students will have the opportunity to apply for check-out, take-home VR headsets for use on assignments and projects. Please keep an eye out for the optional “equipment lottery” assignment in the first week of the course. All checked-out equipment must be returned in its original packaging, and in functioning order, before a student’s final course grade will be released. All equipment-lottery VR headsets have a replacement cost of \$299.

## Workload

EECS 498 is a challenging course that requires, and rewards, a significant investment of time.

- Assignments and projects are designed to require up to 9 hours of outside work per week, in line with CSE guidelines.
- Comparisons to heavy courses such as EECS 485 are apt. The course is designed to be noticeably lighter than “Very Heavy” courses such as EECS 494 and EECS 482.

Please reconsider taking EECS 498 if you...

- Are also taking EECS 482, 470, or another high-workload course.
- Are also taking several medium-workload EECS courses.

The course will target an average grade of “B+ / A-”. This course does not employ exams.

## Graduate Student Advisory

Graduate students are advised against taking EECS 498 (XR) due to its substantial time requirements and undergraduate-style grading structure. Please discuss with your advisor and reach out to the course staff if considering registration. Note that undergraduate CS students get priority seating, as they need a course of this type to graduate (graduate students may be required to wait until a week or two after the semester begins to complete registration– reach out to the instructional staff to get access to the course canvas / piazza site in the meantime).

Non-CSE Graduate Students seeking to take the class should fill out [this form](#).

## Resources

A number of resources exist to improve one's EECS 498 experience--

## Community

Check the course canvas site for links to...

- The course Discord server : Instant-chat, virtual office hours, project-progress, announcements, etc.
- The course Piazza site : Q&A with peers and faculty, announcements, etc.
- [Alternate Reality Initiative](#) : Student-run club focused on XR.
- [Wolverine Soft](#) : Student-run club focused on game development.
- [IGDA Ann Arbor](#) : International Game Developers Association, Ann Arbor Chapter.

## Health

Should you feel overwhelmed, or find yourself spending more than 12 hours on assignments per week, please reach out to the course staff at [xrcoursestaff@umich.edu](mailto:xrcoursestaff@umich.edu)-- We will be happy to discuss your progress, techniques for time-efficiency / assignment planning, and ways to get back on track.

Should you find yourself unable to make progress due to factors beyond your control (sickness, family emergency, etc), please reach out immediately for extension consideration. Extensions will not be granted in situations where a project teammate may carry the burden temporarily.

- [Counseling and Psychological Services \(CAPS\)](#) is available to help.

## Policies

### Health and Safety

Course participants must follow all relevant safety protocols and guidelines when interacting with course check-out hardware or the Visualization Laboratory. A sanitization procedure must be completed before and after usage of VR equipment. Please visit the VR laboratory or consult the course Canvas site to learn more.



## Lecture Recording

Lecture recordings will be made available an hour or so after a given lecture concludes (check canvas for a link to the recording repository).

You may find lecture recordings from previous semesters, in addition to other gamedev content, on [the course youtube channel](#).

## Submission

All deliverables required of an assignment must be submitted to the appropriate course canvas site. The student is responsible for meeting every requirement of the assignment specification, and following submission procedures including deliverable naming, timeliness, etc.

Students are responsible for understanding the submission technology and processes used by canvas, and will be held responsible for the team submissions executed by their assignment teammates. Students will not be held responsible for submission difficulties outside of their control, such as widespread canvas server outages. For game-based deliverables, only the executables will be graded. If the submitted executables cannot be graded, the submitted source will be graded (but only the game beginning with the main scene, and only if no changes are necessary to run the game. The Incorrect Submission Format Penalty will also be applied).

## Late Assignment Submission

Assignments submitted late will be subject to the following policy, exhibited in the Python programming language (unless otherwise specified in the assignment specification)--

```
if hours_late < 0.0:
    return graded_assignment_score
elif hours_late < 24.0:
    Return graded_assignment_score * 0.8
return 0.0
```

Note that Canvas' assignment submission timestamp will be the ultimate authority of when something was turned in. Begin your upload early to account for networking or technology-related delays. Assignment extensions will only be granted for exceptional, extenuating circumstances beyond the student's control, and not in situations where a teammate may carry the burden temporarily.

Teams may request the course staff grade an earlier submission if the request is made before the staff begins grading.

## Incorrect Submission Format Penalty

Submitted assignments that do not meet the submission requirements outlined in the “Deliverables” section of its assignment specification will...

- Have the full 20% late penalty automatically applied.

Missing files, broken builds, etc are typical reasons for application of this penalty.

## Penalty Stacking

The “Late Assignment Submission” and “Incorrect Submission Format” penalties described above may stack to form a single 40% penalty.

## Standard Grading Environment

VR projects will see the grading staff utilize HTC Vive and Oculus Quest devices for grading.

## Imbalanced Team Contribution Penalty

Team members that provide substantially less investment than the average of their project peers will have a penalty applied to their project score (via their participation grade). Evidence will be collected from peer evals and the course MS Project time management system, and presented to the student in question.

The penalty calculation adheres to the following formula--

- $Total\ Penalty\ Amount = sum\_of\_technical\_project\_team\_related\_points\_earned * investment\_ratio$
- Where  $sum\_of\_project\_team\_related\_points\_earned$  is the total points earned by the student on technical-project-related assignments that involved a team.
- Where  $investment\_ratio$  is 25%, 50%, or 75% depending on the severity of the lack of investment.

As a result of this policy, team members who contribute substantially less than the average of their peers on a project should expect to receive substantially fewer points on those projects. It is possible for participation grades to go negative, allowing the penalty to have its full impact in all cases.

## Attendance / Late Arrival

There is no explicit penalty for truancy, though much is likely to be missed without timely attendance. Those who cannot attend lectures in-person may watch recordings made available several hours later. Note that several assignments / projects require an on-campus presence.

## Regrades / Grading Disputes

Any released grade may be disputed, either at office hours or via email ([xrcoursestaff@umich.edu](mailto:xrcoursestaff@umich.edu)), so long as the dispute is raised within one week of the released grade.

Grades will only be considered for adjustment in situations where the instructor's grading process is "cut-short", or rendered incomplete, due to the instructor failing to notice features / content of the deliverable, or being physically unable to do so due to a high difficulty or by demanding an unreasonably long time investment of the grading staff.

To clarify, the following are not valid reasons to request a regrade--

- The deliverable was fully evaluated / experienced but the student feels the content of their deliverable deserves additional credit.
- The student's development process was impeded by a reasonably foreseeable / preventable issue.
  - The student failing to read and abide by submission guidelines, for example.
  - The student slept in and missed a deadline without making an earlier "safety submit".

## Office Hours

Office hours will be held weekly, in both in-person and virtual modes, with each instructional staff member contributing at least one hour per week. Check the course Canvas site and calendar for office hour dates, times, and locations.

## Extra Credit

EECS 498 (XR) provides an abundance of extra credit opportunities in an effort to...

- Provide a remedy for student mistakes incurred naturally through the learning process.
- Encourage actions that boost game development knowledge and enrich students' EECS 498 (XR) experience.

Points earned via extra credit opportunities will be applied following any curve when final course grades are tallied. Students need not fear point inflation caused by classmates earning extra credit points.

## Respect and Maturity

EECS 498 (XR) is a highly collaborative course in which students engage often with one another and the course staff. Students will...

- Act respectfully towards fellow students, staff, and guests at all times.
- Act in a mature, thoughtful fashion at all times.
- Refrain from disrupting or egregiously delaying the course, or the delivery of a lecture.

Students who fail to meet these expectations risk reduction of their participation score, or in egregious cases, referral to the Dean's office.

## Collaboration / Cheating / Honor Code

In their EECS 498 (XR) collaboration, students must respect the following policies...

- External assets / code / material not produced by the team *always* requires citation. These may either be included as a comment in the code, or in a text file submitted to canvas (credits.txt).
- External asset usage may be restricted on a per-assignment basis (check the assignment specification).
- Heavy external asset usage *must* be approved by course staff (make contact at [xrcoursestaff@umich.edu](mailto:xrcoursestaff@umich.edu)).
- Students must obtain a permissible license for all external assets in use within coursework (Public domain, [CC-BY](#), etc).
- The sharing of code / assets between teams is forbidden unless stated otherwise.

It is generally acceptable to discuss project goals and technical / design concepts at a high-level. Refrain from discussing exact solutions or viewing another team's assignment code. **If in doubt, ask the course staff.**

Failure to comply with these policies may result in a reduction of the assignment grade and / or referral to the honor council.

## Staff AutoBios

### Austin Yarger

Austin is an international educator and Lecturer of Game Development in the University of Michigan's College of Engineering, co-founder of the International Game Developers Association (Ann Arbor Chapter - <https://igda2.org>), and President of Arbor Interactive, a local game and software development firm ([www.arborinteractive.com](http://www.arborinteractive.com)).

A long-time hobbyist game developer, Austin got his taste of professional game development in the summer of 2014 with an internship at Maxis (Electronic Arts) where he helped engineer 2015's top selling computer game, The Sims 4. Austin has taught EECS 494 ([www.eecs494.com](http://www.eecs494.com)) for 10+ semesters and EECS 498.003 for two. His students have achieved positions at prestigious game developers such as Naughty Dog, Niantic, Sony Santa Monica, 343 Industries, Volition, Zynga, Google, Gaudium, NetEase, Jackbox Games, and Amazon Game Studios among many others.



In addition to stops at EA Mobile and Facebook, Austin served as President of the Wolverine Soft game development organization from 2011 – 2014. He co-founded the Ann Arbor chapter of the International Game Developers Association ([www.igda2.org](http://www.igda2.org)), curates the MichiGames Arcade Cabinet, mentors the Huron High School Game Development Team, consults with technology startups in downtown Ann Arbor, and organizes multi-university exhibitions with Eastern Michigan University. He acquired a Bachelors and Masters degree

from the University of Michigan in 2015 and 2018, respectively. His research interests include non-gaming applications of game development tools, technologies, and techniques. Learn more at [www.ayarger.com](http://www.ayarger.com).

## Nithisha Kumar

Nithisha is a junior studying computer science. She has previously worked at the University of Michigan Transportation Research Institute to develop an AR app that uses accurate human models to test vehicle designs for accommodation of various body types. As a student of EECS 498 during Fall 2022, she worked on developing FaceAR – an AR app that helps facial disorder patients perform facial exercises. In her free time, Nithisha enjoys clicking photos and filming/editing videos.



## Rahmy Salman

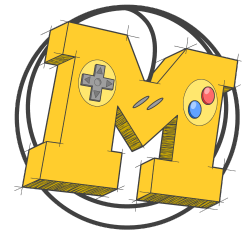
Rahmy is a senior studying Computer Science with a healthy interest in math and machine learning. As a student in EECS 498 in Fall 2022, he contributed to the development of Urban Rush, a virtual reality fitness app designed to promote physical activity. In the past, he contributed to research in the Department of Computational Medicine and Bioinformatics by helping to develop HAT, the Hypergraph Analysis Toolbox, a MATLAB package for visualization and analysis of hypergraph data. Rahmy is a self-described gamer and has recently begun speedrunning Super Mario 64.



## Discussion

## Relation to EECS 494 : Introduction to Game Development

[EECS 494](#) is a CSE capstone course that introduces students to the technologies and techniques behind modern recreational video games. The course focuses on game development in the traditional context of entertainment, only briefly touching upon the unique hardware, challenges, and opportunities afforded by extended reality and non-entertainment / socially-impactful pursuits. These courses have fairly different topics and vastly different goals– students who are seeking to work in the traditional video game industry are recommended to take EECS 494, while those seeking to work in the emerging field of VR, AR, and Visualization applications are encouraged to take EECS 498 (XR). Should you take both courses, know that the second capstone / MDE course that you take will not count as ULCS credit, but likely will count as FlexTech.



### I have Extra Time– How May I Prepare for the Course?

The course is designed such that one does not need to do any pre-semester preparation. If you find yourself with extra time, you might consider...

- Use a program such as WinDirStat (pc) or Disk Inventory X (mac) to free up ~100GB for the course.
  - This should be enough space for Unreal Engine, Unity Engine, and your various projects.
- Downloading and experimenting with Unreal Engine.
  - Complete [a video tutorial](#) or two online. ([one beginner's first six months with Unreal](#)).
- Downloading and experimenting with Unity Engine.
  - Complete [a video tutorial](#) or two online.
- Experiment a bit with [Amazon Web Services \(AWS\)](#), a popular cloud computing provider.
  - [Try creating a \(essentially-free\) static website.](#)
  - [Try launching your very own ubuntu server in the cloud.](#)

Date	Lecture Plan	A s s i g n m e n t s				
Wed, 01/04/23	<a href="#">Lecture 1</a>	exploration_vr (7 days)	exploration_ar (7 days)	Equipment Lottery		
Mon, 01/09/23	<a href="#">Lecture 2</a>					
Wed, 01/11/23	<a href="#">Lecture 3</a>				tutorial_vr (7 days)	tutorial_ar (7 days)
Mon, 01/16/23	<b>MLK Day (no lecture)</b>		Join P1 Team (size:2)			
Wed, 01/18/23	<a href="#">Lecture 4</a>	p1_milestone (7 days)		p1_pm (21 days)		
Mon, 01/23/23	Evening Film (no lecture)					7pm film 1670 BBB
Wed, 01/25/23	<a href="#">Lecture 5</a>		p1_alpha (7 days)		62 69 74 2a 6c 79 2f 79 78 33 39 64 6f	
Mon, 01/30/23	<a href="#">Lecture 6</a>					
Wed, 02/01/23	<a href="#">Lecture 7</a>	p1_gold (7 days)				culture_2
Mon, 02/06/23	Evening Film (no lecture)					7pm film 1014 DOW
Wed, 02/08/23	<a href="#">Lecture 8</a>				Join P2 Team (size:2)	p1_postmortem (5 days)
Mon, 02/13/23	<a href="#">Lecture 9</a>		p2_milestone (7 days)	p2_pm (21 days)		
Wed, 02/15/23	<a href="#">Lecture 10</a>					
Mon, 02/20/23	<a href="#">Lecture 11</a>	p2_alpha (5 days)				
Wed, 02/22/23	<a href="#">Lecture 12</a>					culture_3
Sat, 02/25/23	<b>Spring Break (begins @ noon)</b>	(Due Sat. at noon)				
Wed, 03/01/23	<b>Spring Break</b>					
Mon, 03/06/23	Evening Film (no lecture)		p2_gold (7 days)			7pm Film 1014 DOW
Wed, 03/08/23	<a href="#">Lecture 13</a>					
Mon, 03/13/23	<a href="#">Lecture 14</a>				p2_postmortem (7 days)	

<b>Wed, 03/15/23</b>	<a href="#">Lecture 15</a>		Join P3 Team (size:4-5)				culture_4
<b>Mon, 03/20/23</b>	<a href="#">Lecture 16</a>	p3_milestone (7 days)		p3_research (21 days)	p3_pm		
<b>Wed, 03/22/23</b>	<a href="#">Lecture 17</a>						
<b>Mon, 03/27/23</b>	<a href="#">Playtest</a> 1+2, 3+4,5+6 7+8, 9+12+18		p3_milestone_2 (7 days)				7pm Film 1014 DOW
<b>Wed, 03/29/23</b>	<a href="#">Lecture 18</a>						
<b>Mon, 04/03/23</b>	<a href="#">Playtest</a> 1+4, 5+8+18 3+6, 7+12, 9+2	p3_alpha (7 days)					
<b>Wed, 04/05/23</b>	<a href="#">Lecture 19</a>					p3_marketing (7 days)	culture_5
<b>Mon, 04/10/23</b>	<a href="#">Playtest</a> 1+6+18, 3+8 5+12, 7+2, 9+4		p3_gold (7 days)				7pm Film 1014 DOW
<b>Wed, 04/12/23</b>	<a href="#">Lecture 20</a>						
<b>Mon, 04/17/23</b>	Showcase (6-10pm)		(6pm EST)	p3_postmortem (7 days)	(6pm EST)		
<b>Tue, 04/18/23</b>	494 Showcase (7-10pm) (for fun)						
<b>Mon, 04/24/23</b>				(6pm EST)			







## Project 2 Milestone : A2-GO!

Getting out and about with ARFoundation and MapBox

Estimated Time Investment	Typical Time Allotment	Team Size
10 hours	7 days (check schedule)	2



AR's undisputed killer-app, *Pokemon Go*, was perhaps the first video game to make people go outside and explore.

But should GPS and map usage be considered augmented reality? ([image top-right : TechCrunch](#))

## Pitch

*P2 Milestone* is a sub-assignment that tasks students with showing substantial progress on the *P2 : A2 Go!* project. Students will implement tasks ordered by priority to meet an aggressive early deadline.

## Purpose

Large software projects (such as A2-Go!) are risky and expensive endeavors. Clients / stakeholders funding the development of such projects typically demand a [vertical slice](#) of the final deliverable, so they may be confident a team has the ability to deliver. This is often called a *milestone*, and teams that don't meet it often lose their funding. Student teams will...

- Prove they can complete the entire A2 Go! project by completing a small subset of it quickly.
- Gain an appreciation for prioritization of tasks (by implementing one collectible type, one gains confidence that the other types may eventually be implemented in similar fashion).
- Gain confidence in their ability to triage (sacrifice) tasks when time is running low.
- Gain an appreciation for the project management techniques required to prevent crunch.
- Significantly develop their experience and understanding of Unity Engine, Mapbox, and AR Interactions.

## Tasks

### Assess Available Resources

1. Determine if you may complete this assignment using your personal devices (faster), or if you will need to employ campus resources (slower). Consult [the Platform Pain Matrix](#) to decide.

### Make Progress

1. Make significant progress in all "p2\_milestone" tasks in [the p2 task spreadsheet](#). **Note** that the tasks need not be finished or bug free until p2\_gold– their implementation must simply be non-trivial.

## Deliverables

Submit to canvas and the course discord (exhibition channel) the following–

1. One video recording (.mp4 format with audio) clearly demonstrating progress on each of the p2\_milestone tasks.

(extra credit) Provide Feedback on this Assignment



Type	Task	Description	Hints	Story Points	Milestone
Story	(optional) Review Map Box	Obtain a quick, corporate, slightly-awkward understanding of MapBox (a key component in this project)--  [] Skim <a href="#">the wikipedia article</a> . [] Watch <a href="#">an awkward corporate talk</a> from a head honcho. [] Read up on <a href="#">the limits for free accounts</a> . You want "Maps SDK for Unity".	- Pokemon Go appears to have created their own mapping service that pulls from <a href="#">OpenStreetMaps</a> . While cool, we need something more off-the-shelf for such a short project.	1	p2_milestone
Story	Study project concepts / terminology	Before we begin prototyping, it behooves us to take a 1000-ft view of the project and what we'll need to accomplish.  [] <a href="#">Read this document</a> describing some of the essential systems / concepts of A2-Gor [] <a href="#">Skim this document</a> describing the transition from Unreal to Unity.		3	p2_milestone
Story	Prepare Project	In order to truly begin development, we need a basic unity project with two things-- (a) Your tutorial_ar scene and components. (b) The Mapbox SDK.  You may obtain both of these on your own (read the mapbox import tips below), or... [] Download and open <a href="#">this pre-built project</a> in Unity (use latest LTS of unity to open it). [] (if using the VisStudio, make sure to place the project on the C:\ drive - not the networked drives, or crashes may occur). [] <a href="#">If building for IOS</a> , your builds may fail. Try the changes described in the latter part of <a href="#">this thread</a> or <a href="#">this thread</a> and maybe <a href="#">this note</a> . If that isn't enough, you might need to configure some IOS-related player settings (Edit -> Project Settings -> Player). <a href="#">Your xcode might also hang</a> . [] <a href="#">Configure mapbox with a new token</a> (create an account on MapBox.com to obtain one). [] Verify you may build and deploy the project to your mobile device (trying both the exploration_scene and interaction_scene). IOS / XCode users might wish to return to <a href="#">the build hints in exploration_ar assignment</a> , or examine the hints <a href="#">on this page</a> .	- Failing to see a map appear when you build your app? You may need to enable location services before restarting the app (a "grant permission" prompt should appear asking for permission). Still not appearing? Be sure to fill out the "location usage description" field in the player settings of the unity project.	3	p2_milestone
Story	(Optional) Set Up Version Control	Required for your p2_pm assignment, establishing version control is time well-invested.  [] Consider what version control you might want to use for your projects (git, svn, perforce, thumbdrive, etc). [] If choosing git, consider completing the "Establish Version Control" section in your p2_pm assignment.	- You may transfer content between unity projects (or between partners) via Asset Packages (potentially useful to import your previous unity work). - You may create a new scene via File -> New Scene. You may include that scene in your builds (critical) by visiting File -> Build Settings -> Add Open Scenes. - If "gameobjects" / "actors" are just a collection of components in unity / unreal, then "scenes" / "maps" are a collection of "gameobjects" and "actors". Multiple scenes are often used to represent multiple "levels" in a video game, but they can also represent multiple "modes" or even multiple "menus" in a unity app.	3	p2_milestone
Story	(Optional) Set Up Visual Studio Auto-Completion	Every semester, students lose copious amounts of time to misunderstood functions, return types, and getting lost knowing what is available to them. In Unreal, a node-search tool will help you find useful nodes. In Unity, Visual Studio's autocomplete feature meets this need. Without it, you are flying blind and will move at 50% speed (if you're lucky).  [] Verify you have installed and are using Visual Studio (the <a href="#">version that comes with Unity</a> - not visual studio code). [] Verify you understand <a href="#">what code completion is</a> , and <a href="#">what it looks like in Unity</a> . [] If it isn't working, <a href="#">follow this video to set it up</a> . [] If it still isn't working, <a href="#">check all of these boxes, then click "regenerate project files"</a> . [] If it still isn't working, attend office hours ASAP.		2	p2_milestone
Story	Exploration Development Cheats	<b>Note : Partial walkthrough in-class (check recording)</b>  Having a tight, efficient, iterative development process is key to a successful project (and product). Walking around to test our application will take too much time and energy. While we will do that eventually, it's best if we create a more convenient way to walk around for testing purposes.  [] When playing the game in the Unity editor, if the player uses the arrow keys, <a href="#">our positioning coordinates should move accordingly, allowing us to fake our location</a> .	<a href="#">Hint document here.</a>	3	p2_milestone
Story	Orbit Camera : Follow Behavior	<b>Note : Partial walkthrough in-class (check recording)</b> When our character moves too far, <a href="#">they leave the screen!</a> Our camera gameobject should follow our character so we may always see them.  [] The primary camera gameobject should follow and focus the player gameobject at all times, providing a nice overhead view as we explore. [] When the player moves around (camera following it), the map should update with new tiles to show the new area we are exploring.	<a href="#">Hint document here.</a>	3	p2_milestone



Type	Task	Description	Hints	Story Points	Milestone
Story	Gameobject : Trees 1-6	<p>The player will be planting various types of trees in this game. The trees will act very similarly, but should look different from one another, and will generate different levels of tourism revenue once fully grown.</p> <p><b>BOTH MODES</b></p> <ul style="list-style-type: none"> <li>[] Create 6 different trees, corresponding to the 6 different kinds of seeds available in the game.</li> <li>[] Each tree should have a different model / different aesthetics.</li> <li>[] Each tree should generate different amounts of money, over time, once fully grown.</li> <li>[] The tree should be on the ground (no flying / floating trees) (consider using an <a href="#">ARRaycast</a> downward to find ground).</li> <li>[] The tree's scale (transform.localScale) should be determined by how close it is to being fully grown. A new tree will be small. A grown tree will be large.</li> <li>[] You may choose how trees grow over time (so long as it is gradual) and when (or if) growth ever pauses.</li> </ul> <p><b>EXPLORATION MODE</b></p> <ul style="list-style-type: none"> <li>[] If multiple trees are grow during interaction mode, when you switch to exploration mode, put them nearby but not exactly on top of each other (consider achieving this by adding a <a href="#">random offset</a> to the tree's lat/long location when switching from interaction to exploration mode).</li> </ul> <p><b>INTERACTION MODE</b></p> <ul style="list-style-type: none"> <li>[] When the player switches from exploration mode to interaction mode while standing near several trees, only the single nearest tree need appear in interaction mode.</li> <li>[] If the player walks a lengthy distance while in interaction mode, you need not spawn in or visualize trees that would be nearby if the player were in exploration mode.</li> </ul>	<p>- Confused about a variable's value or the output of a function? Print the variable to your console via <code>Debug.Log()</code>;</p> <p>- Interaction mode and exploration mode exist in two very different coordinate systems / spaces. In order to "bring trees over" from interaction mode to exploration mode (and vice-versa), we recommend re-instantiating a bunch of tree gameobjects to the correct locations by using static data (perhaps a list of tree structs). Avoid <code> DontDestroyOnLoad()</code> here- it's best used for UI gameobjects (which have the same coordinate system in both scenes- that of the screen).</p> <p>- How to find the lat/long coordinates of trees while in interaction mode? You can't easily, so consider using the lat/long of the player instead (this is data you should have from exploration mode).</p> <p>- How to convert between lat/long and unity scene coordinates? The <code>AbstractMap</code> data type (your "Map" gameobject in the <code>exploration_mode</code> scene) has some very useful functions.</p> <p>- <b>WARNING</b> : the <code>AbstractMap</code>.<code>GeoToWorldPosition()</code> function will not work properly until the <code>AbstractMap</code> has finished loading (happens every time a scene changes). This may take several seconds. If your code only runs in <code>Start()</code>, your gameobjects may never be placed into the correct position. Consider running this "positioning" logic every frame (instead in <code>Update()</code>), or using a <a href="#">callback</a> to run it when the <code>AbstractMap</code> is ready. Register a function for this callback via <code>callback += MyFunction,</code></p>	3	p2_alpha
Story	UI : Top Bar	<p>The trees in our game will begin generating currency when they are implemented and fully grown. How is the player going to understand how much currency they have?</p> <p><b>BOTH MODES</b></p> <ul style="list-style-type: none"> <li>[] Decide what the "currency" theming is. Is it dollars? Leaves? <a href="#">Anchovies</a>?</li> <li>[] Add a UI element that informs the player of their current currency count.</li> <li>[] Sometimes background colors can <a href="#">make our UI elements difficult to read</a>. Add an <a href="#">outline</a> to improve accessibility.</li> </ul>	<p>- "currency" is a piece of data you'll likely need to access and adjust from several different parts of your codebase (Trees increase your currency. The seed shop decreases it. Your UI needs to read it and show it to the player). Because of this fact, you want this variable to be easy-to-access. Consider using "public static int" instead of just "int". Do you remember <a href="#">what "static" means?</a></p>	2	p2_alpha
Story	UI : Landmark	<p>When the player discovers a new landmark while walking around Ann Arbor, we want to reward such behavior. We'll create a pop-up to provide them a bit of history, as well as unlock a new seed type for them!</p> <p><b>Exploration Mode</b></p> <ul style="list-style-type: none"> <li>[] When a landmark is tapped (F23+ : and the player is somewhat close to it), a piece of UI appears (likely a UI Panel)</li> <li>[] The user sees some text providing trivia on the landmark, in addition to notifying the user that a new seed type has been unlocked (you'll unlock the actual seed in the UI : Seed Shop task).</li> <li>[] The user sees an image of the real-life landmark.</li> <li>[] When the user taps / clicks a second time, the UI element closes and they're on their way again.</li> </ul>	<p>- Detecting an element in the UI is easy, but how can we detect tapping on a 3D gameobject in the scene? We recommend searching "Unity raycast from mouse to world". We say "mouse" because when you build for a mobile device, your mouse-related code will work for taps as well as clicks.</p> <p>- Landmarks do not need to be visible in interaction mode- only exploration mode.</p> <p>- Consider taking some inspiration from <a href="#">the InfoCanvas prefab</a> the template project provides for you.</p>	3	p2_alpha
Story	UI : Seed Shop	<p>Players need the ability to shop for more seeds, such that they can place down more (and more diverse) trees.</p> <p><b>Both Modes</b></p> <ul style="list-style-type: none"> <li>[] An ever-present button in the UI should exist, opening the "seed shop" when it is pressed.</li> <li>[] The seed shop should take the form of a list or grid of buttons- each button corresponding to a tree type.</li> <li>[] Text labels should be present to indicate cost of each seed type. You may choose the costs, so long as it is above zero.</li> <li>[] Tapping a button should, if the player has enough currency, add a seed to the inventory bar and deduct from the player's currency counter.</li> <li>[] Some buttons should be disabled / non-present if the player has not unlocked that seed type via a landmark.</li> </ul>	<p>- Reuse your knowledge from the UI : Landmark task!</p> <p>- During class, we demonstrated the power of "horizontal layout groups". You may want to investigate "vertical" or "grid" layout groups for this task.</p>	3	p2_alpha
Story	Orbit Camera : Orbit Controls	<p>The user may wish to <a href="#">look around this new reality</a>- we must give them the controls to do so.</p> <p><b>Exploration Mode :</b></p> <ul style="list-style-type: none"> <li>[] No matter where the camera is, it should always angle towards the player avatar.</li> <li>[] When the user drags their finger vertically on screen, the camera's orbit should change vertically, taking the camera between a side-view of the avatar and an overhead view.</li> <li>[] When the player drags their finger left or right, the camera should orbit around the player avatar (around the vertical axis), letting the player see the world from a different angle.</li> </ul>	<p><a href="#">Hint document here.</a></p> <p>- Is your app crashing or slowing down? If you angle your camera too low, and it can see far into the distance, <code>Mapbox</code> may try to load a lot more map data than it needs to. Restrict the camera from performing low angles to prevent this (or re-configure <code>mapbox</code> and <a href="#">maybe use fog</a>).</p> <p>- We recommend combining your "Orbit Controls" logic with the "Follow Behavior" logic into one "Orbit Camera" component, rather than splitting it into two.</p>	3	p2_alpha



University of Michigan  
Fall 2022 Instructor Report  
EECS 498-003: Special Topics  
Austin Yarger

21 out of 35 students responded to this evaluation.

**Responses to University-wide questions about the course:**

	SA	A	N	D	SD	N/A	Your Median	Univ-wide Median	School/College Median
This course advanced my understanding of the subject matter. (Q1631)	19	1	0	0	0	0	5.0	4.5	4.5
My interest in the subject has increased because of this course. (Q1632)	18	2	0	0	0	0	4.9	4.2	4.2
I knew what was expected of me in this course. (Q1633)	14	4	2	0	0	0	4.8	4.6	4.4
I had a strong desire to take this course. (Q4)	14	4	1	1	0	0	4.8	4.0	4.1
As compared with other courses of equal credit, the workload for this course was (SA=Much Lighter, A=Lighter, N=Typical, D=Heavier, SD=Much Heavier). (Q891)	1	1	6	5	7	0	2.1	3.0	2.8

**Responses to University-wide questions about the instructor:**

	SA	A	N	D	SD	N/A	Your Median	Univ-wide Median	School/College Median
Austin Yarger seemed well prepared for class meetings. (Q230)	17	1	0	1	0	1	4.9	4.8	4.8
Austin Yarger explained material clearly. (Q199)	15	4	0	0	0	1	4.9	4.7	4.7
Austin Yarger treated students with respect. (Q217)	18	2	0	0	0	0	4.9	4.8	4.8

**Responses to questions about the course:**

	SA	A	N	D	SD	N/A	Your Median
Overall, this was an excellent course. (Q1)	14	5	1	0	0	0	4.8
The textbook made a valuable contribution to the course. (Q64)	2	0	3	0	1	14	3.2
Prerequisites provided adequate preparation for this course. (Q61)	14	1	5	0	0	0	4.8
I felt comfortable asking questions in class. (Q521)	16	3	1	0	0	0	4.9
I developed confidence in my abilities as an engineer. (Q1769)	18	1	1	0	0	0	4.9
I developed the ability to solve real world engineering problems. (Q1770)	17	2	1	0	0	0	4.9
I felt included and valued when working with other students. (Q253)	18	1	1	0	0	0	4.9

**Responses to questions about the instructor:**

	SA	A	N	D	SD	N/A	Your Median
Overall, Austin Yarger was an excellent teacher. (Q2)	19	1	0	0	0	0	5.0

University of Michigan  
Winter 2023 Instructor Report  
EECS 498-003: Special Topics  
Austin Yarger

25 out of 50 students responded to this evaluation.

**Responses to University-wide questions about the course:**

	SA	A	N	D	SD	N/A	Your Median	School/College Median	Univ-Wide Median
This course advanced my understanding of the subject matter. (Q1631)	18	7	0	0	0	0	4.8	4.4	4.5
My interest in the subject has increased because of this course. (Q1632)	13	9	2	1	0	0	4.5	4.1	4.2
I knew what was expected of me in this course.(Q1633)	15	9	1	0	0	0	4.7	4.3	4.6
I had a strong desire to take this course.(Q4)	14	11	0	0	0	0	4.6	4.0	4.1
As compared with other courses of equal credit, the workload for this course was (SA=Much Lighter, A=Lighter, N=Typical, D=Heavier, SD=Much Heavier). (Q891)	2	2	5	13	3	0	2.2	2.8	3.0

**Responses to University-wide questions about the instructor:**

	SA	A	N	D	SD	N/A	Your Median	School/College Median	Univ-Wide Median
Austin Yarger seemed well prepared for class meetings.(Q230)	21	3	1	0	0	0	4.9	4.7	4.8
Austin Yarger explained material clearly.(Q199)	21	4	0	0	0	0	4.9	4.6	4.7
Austin Yarger treated students with respect.(Q217)	19	4	2	0	0	0	4.8	4.8	4.8

**Responses to questions about the course:**

	SA	A	N	D	SD	N/A	Your Median
Overall, this was an excellent course. (Q1)	14	10	1	0	0	0	4.6
The textbook made a valuable contribution to the course. (Q64)	6	1	2	2	2	12	4.0
Prerequisites provided adequate preparation for this course. (Q61)	13	7	2	0	0	3	4.7
I felt comfortable asking questions in class. (Q521)	13	8	2	0	0	2	4.6
I developed confidence in my abilities as an engineer. (Q1769)	12	13	0	0	0	0	4.5
I developed the ability to solve real world engineering problems. (Q1770)	13	12	0	0	0	0	4.5
I felt included and valued when working with other students. (Q253)	17	7	0	1	0	0	4.8

**Responses to questions about the instructor:**

	SA	A	N	D	SD	N/A	Your Median
Overall, Austin Yarger was an excellent teacher. (Q2)	20	4	1	0	0	0	4.9