



Catalog Description

This class is a continuation of ETGG 1801 and is intended to further develop the students' understanding of the simulation/gaming production and implementation process. Class activities are focused upon understanding of more advanced concepts and implementation techniques central to the game and simulation development process. Lab activities are focused upon the writing of simple, yet complete, interactive programs in a high-level programming language.

Textbooks (optional)

- A good Python reference (textbook, online tutorials, etc.)

Student Learning Outcomes. By the end of this course, students will be able to...

- *Contribute* to a large group software project.
- *Become* intermediate-level Python programmers.
- *Appreciate* the role of versioning and team-integration tools in software engineering.
- *Effectively critique* fellow group-member's contributions to a software project.

ADA Statement

Any student who believes s/he may need an accommodation based on the impact of a documented disability should first contact a Coordinator in the Office of Disability Services, Student Success Center, Massie Hall, 740-351-3276 to schedule a meeting to identify potential reasonable accommodation(s). Students are strongly encouraged to initiate the accommodation process in the early part of the semester or as soon as the need is recognized. After meeting with the Coordinator, students are then required to meet with their instructors to discuss the student's specific needs related to their disability. If a student does not make a timely request for disability accommodations and/or fails to meet with the Coordinator of Disability Services and the instructor, a reasonable accommodation might not be able to be provided.

Grading Policy

In this course, your entire grade will be based on lab assignments. Most of these lab assignments will 1 to 2 weeks in length. The first few weeks will involve (very) small groups. After the first few weeks, we'll transition to a larger group project (~10 people per team). You will be evaluated approximately as follows:

- 33%: How well did the group do in meeting their interval goals (which you'll design with Jason)?
- 34%: How well did you do individually in contributing to the project?
- 33%: Peer-evaluation

Quizzes

Pop-quizzes could be given at any time, without prior notice. These will generally be counted as a low-point lab and are meant to encourage participation in discussions and lectures. They will be counted as a small-point lab.

Attendance

Attendance will be taken. It doesn't directly affect your grade, but the instructor will consider attendance when assigning grades for intervals (as will most people in the peer evaluation)

Grading Scale

0-60	60-64	64-67	67-70	70-74	74-77	77-80	80-84	84-87	87-90	90-94	94+
F	D-	D	D+	C-	C	C+	B-	B	B+	A-	A

Succeeding in a group environment.

This can be a bit intimidating the first time you are a part of a group. But in the "real world", you'll rarely be working in isolation, which is why many of our courses in the degree contain a group project component. Here are some tips for success:

- **Follow directions.** The group will help me develop deliverables / milestones at the beginning of each interval. Make sure you understand these completely – if not, ask Jason or a team-mate.
- **Stay busy.** This will primarily be in-class, but out-of-class work is expected as well. If you can't find anything to do:
 - Look on the interval document (or job board, if we have one) for open jobs.
 - Ask the teacher or a team-mate for a job.
 - If all else fails, do paired programming¹.
- **Communicate with others.** Stay away from these two extremes:
 - The "hero": Tries to do the project herself.
 - The "hermit": Withdraws and is afraid to touch the code.
- **Pay attention during "stand-up's".** These are meant to be short surveys of where you are and what you're working on. I want you to listen during these times, not work on the project.
- **Work towards the milestones.** Generally everyone will share the same grade on the "group effectiveness" portion of your interval evaluation. Make sure your deliverables are always being worked towards.

Semester Schedule

- [M] January 9, 2017: Classes begin
- [M] January 13, 2017: Last day to add a class
- [M] January 16, 2017: **Martin Luther King Jr Day** (NO CLASS)
- [F-Su] January 20 – 22, 2017: **Global Game Jam** (Bonus points!)
- [Su] January 29, 2017: Last day to drop for 90% refund
- [M – F+] February 13 – 24: Student Progress Reporting
- [Sa] February 25, 2017: Student Progress Report available on MySSU
- [M – F] February 27, 2017 – March 3, 2017: **GDC** (Bonus points)
- [M – Su] March 6 – 12, 2017: **Spring Break** (NO CLASS)
- [W] March 22, 2017: **Last day to drop** an individual class on MySSU
- [M – W] April 3 – 6, 2017: Priority **Registration** for Summer / Fall Semester
- [T – R] April 18 – 20, 2017: **East Coast Gaming Conference (ECGC)** (Bonus points!)
- [F] April 28, 2017: Last day of classes
- [Sa – F] April 29, 2017 – May 5, 2017: Final Exams
 - **ETGG3802 "exam" is [T] May 2, 2017 12:00 – 1:50pm**
- [Sa] May 6, 2016: Spring Commencement
- [T] May 9, 2017: Grades due by noon
- [W] May 10, 2017: Grades available on MySSU

Resources if you need help

- The class SI (if there is one available)...although in this class, I don't know that they'd be much help.
- Individual tutors (available through the Success Center)
- Lindsay Monihen (MAS 132), CPS Advisor: academic crises, financial aid questions, transferring, etc.
- Accessibility Services (Success Center): disability issues / accommodations.
- Dean of Students Office (UC 222): resolution of academic and non-academic resources.
- Student Ombudsperson, Linda Hunt (ADM 140): help with appeals, complaints

Let's make something awesome this semester!

¹ One student is the "driver" for ~15 minutes. The driver types on the keyboard. The other student, the "observer" reviews the code being typed by the other student. After the 15 minutes are up, swap places.