

Introduction to Game Development – Fall, 2014 Syllabus

USC School of Cinematic Arts, CTIN 483 (Section 18354)

Instructor

Margaret Moser

Student Assistants

Steve Cha and Zachary Davis

Contacting Us

All communications for this course will occur through your @usc.edu email. This includes emails you send. **Emails from other domains will be ignored.** This is because it is good professional practice to separate your personal and professional communications.

Please use **ctin483@gmail.com** for questions about class content, such as:

- “I need help with this bug”
- “I can’t log in to the class repository”
- “I don’t understand the assignment”

This email is checked frequently by several people, and it will get you the fastest response.

You may contact me at **mmoser@cinema.usc.edu** for administrative issues, such as:

- “I’m sick and will miss class”
- “I don’t understand my grade”
- “I want to ask for accommodations due to a documented condition”

I get a lot of email. I will try to get back to you within 24 hours. If I receive an email from you, I will verify that I received it, even if I can’t respond to the substance right away. If you don’t receive confirmation, I didn’t get your email.

Course Rationale

All designers need the ability to communicate their ideas to others. Because games are interactive, a functioning prototype has much greater fidelity (i.e. resembles the desired experience) and communicates the designer’s ideas more clearly than any static description can. A game designer’s ability to prototype is thus equivalent to a cinematographer’s ability to sketch – while the skill does not directly appear in the final product, it allows the designer to refine his or her ideas and communicate them in a direct way to both team members and test audiences.

A good prototype is literally an experiment; it asks questions about game design. More often than not, prototypes prove that a given design direction is not viable or worth further pursuit, and such an outcome is a valuable failure. The ultimate goal for students in this program — this semester and after — is to develop innovative and compelling games, and prototyping is the fastest, surest way to achieve that goal.

This course will teach you the basic knowledge you need to be able to create digital prototypes of your own ideas. While you should not expect to come out of this course a great programmer, you will come out a better designer, equipped to explore and test your ideas without needing help.

Students of game design benefit in several ways from understanding code.

1. In cinema, the screenplay is the conceptual document for a film, allowing all team members to begin in preproduction with a shared understanding of the goals and potential pitfalls of the final piece. The central form for preproduction in games is the digital prototype. A concise and clear prototype provides an excellent template from which to engage collaborators, to discover unexpected play patterns, and to receive constructive criticism. If a core game idea proves to be

engaging and interesting, the designer can move forward with confidence when forming a team and spending resources to develop it.

2. Designers of digital games benefit from understanding, even in broad terms, what the programmers on a team do on a day-to-day basis. This is essential in order for everyone on the team to work together from a shared vision of the game.
3. The rule sets, patterns, and behaviors that form the game experience are a direct reflection of the underlying code. Code is the raw material with which interactive experiences are built, as pottery is made of clay and paintings are made of paint. The designer who understands code therefore has a much deeper understanding of games as a medium.

Course Description

This production class is focused on rapidly developing game prototypes. In this core course for the Interactive Media Division, students will learn the art of creating digital game prototypes, a practice they will hone throughout their time in the program. Although students are encouraged to have previously taken CTIN 101: Fundamentals of Procedural Media (or CSCI 101), this semester begins with the fundamentals of procedural programming before introducing object oriented programming and is therefore still open to programming novices.

This class is taught in the Unity game development environment using C# scripting. From extensive experience with this class, we have found that this combination of tools provides students with the easiest path to making game prototypes while also teaching them the fundamentals of game programming and giving them a platform which is actually being used to make award-winning, high-profile games (including Kentucky Route Zero, Deus Ex: The Fall, Thomas Was Alone, and many more) across multiple platforms (Mac, PC, web, iOS, Android, Wii U, PS4, etc.)

We will also introduce basics of Agile, the industry-standard production methodology, and the use of version control systems in developing code.

Class Meetings

When: Mondays and Wednesdays 3pm – 4:50pm

Where: XML – SCI L113

Office Hours

You may schedule appointments outside of regular hours by emailing ctin483@gmail.com.

Margaret Moser Tuesday afternoons, 12-4pm or by appointment

Steve Cha By Appointment

Zach Davis By Appointment

Course Pre-requisites

None, though either CTIN 101 or CSCI 101 is recommended. If you are working with code for the first time, you should commit extra time to classwork each week.

Materials

No textbook is required, but Jeremy Gibson's book, *Introduction to Game Design, Prototyping, and Development*, is **very strongly** recommended for all students. It is available via Amazon in both print and electronic forms. Although it is not required, I will refer to it and suggest readings from it related to whatever we are covering in class.

We will use Unity 4.5.2f1 for this semester's class. You can download the free version of Unity from <http://unity3d.com/unity/download>

There are also a few different Pro versions of the software. These include better lighting and visual effects, an AI pathfinding system, animation tools, and improved team workflow. Unity offers a 1-year educational Pro license for students (currently \$99) at

<http://www.studica.com/unity>

This includes iOS Pro, Android Pro and free upgrade to 5.0 when it is released.

Evaluation of student performance

Homework	40%
Classic Game Project	20%
Final Game Project	30%
Participation	10%
Total:	100%

In this class, good work and satisfying all of the requirements of the assignments will earn a student a B. To receive an A, a student must show creative, exceptional work which goes beyond the basic requirements of each assignment and brings something creative or otherwise impressive to the work. While graphical art can contribute to this transition from B to A, graphical ability is not otherwise graded in this class. CTIN 483 is about learning to develop games, and students will not earn credit for art unless all of the programming requirements for the assignment or project are already met.

During the tutorial phase of the class, homework will be assigned in almost every class. These assignments will be pass/fail based on whether the student followed instructions, demonstrated an effort to complete the assignment, and turned the work in on time. "On time" means at or before exactly 3pm. You are responsible for understanding assignments. It is best to ask for clarification in class, but you may also send questions to ctin483@gmail.com. Remember that we check that address frequently, but not constantly, and you probably will not receive an immediate response.

Some homework will take the form of simple quizzes to reinforce the material; some will require you to make things in Unity. When you make something in Unity, you must turn in a **build** with functioning code. (We will go over what a build is, how to make one, and how to turn it in.) If it doesn't do everything it's supposed to, but demonstrates effort, you will still receive credit. If it doesn't run at all, you will not receive credit.

Participation consists of participating in class discussions and exercises, and coming to office hours, in whatever combination you like. If you do not raise your hand all semester and never come to office hours, you will lose points here. Participation is also affected by attendance (see below for attendance policy).

Course Outline

This material is subject to change. This course is designed to be adapted to the skills and goals of its students, and this outline reflects that.

Week 1 – Week 6: Introduction to Unity and C#

Structure: Throughout this part of the semester, students will be instructed in various aspects of game prototyping using C# and Unity. We will go over general syntax and code structures in C#, how to use the Unity editor, and how to work with Unity objects through code.

Assignments: Individual assignments each week. There will be frequent, small "quizzes" reviewing concepts and syntax, as well as exercises in Unity itself. All assignments are completion.

Week 7 – Week 10: Classic Game Project

Structure: Students will work in pairs to create a game prototype which mimics the mechanics and "game feel" of a classic game from the 8-bit era.

Assignments: Pair assignment due Week 10. Students will recreate a game from the 8-bit era. For this assignment, it is not at all necessary that the graphics look like the original game; rather it is much more important that the game mimic the mechanics of the original as exactly as possible. For instance, if a pair were to recreate Super Mario Bros. for the NES, it would be fine if Mario were visually just a box or capsule as long as the way in which Mario jumped felt exactly like the jump in the original game (i.e. pressing the jump button for the same amount of time produces the same upward acceleration, jump height, and downward deceleration as seen in the original game).

Week 11 – Week 15: Final Game Project

Structure: Students will work in pairs to create an original game prototype.

Assignments: Pair assignment due during the final exam period. Students will create a new, unique game prototype. This will be based on their work throughout the semester and should both showcase all of the skills that they've learned throughout the semester and express a unique game design vision.

Final Exam – Monday, December 15 @ 2-4pm

FINAL GAME PROJECT DUE

Students will present their final games to each other and the faculty.

Course Communications

Most assignments and materials will be available through the Perforce version control system, which will be explained and documented separately. Some review quizzes will use an online quiz system. All email announcements will be from ctin483@gmail.com.

Absence Policy

Students are expected to attend every class. This is for your own sake – we will move quickly, and it will be easy to fall behind. Unexcused absences will affect your participation grade. You will also lose points on the classic game and final game projects if you are absent without excuse on a day when your team is presenting. The only excused absences are for illness, family emergencies, and (with advance notice) commitments related to a scholarship you are receiving, e.g. for a varsity sport. You must contact me as soon as possible regarding your absence. Generally I will expect to hear from you before class; in exigent circumstances I would expect to hear from you within 24 hours.

All that said:

1. **If you are sick, stay home.** You need to be healthy to learn, and so do your classmates (and instructors). Remember that this is a very large community and many people within it are immune-compromised, even if they look well. What is an annoying cough for you can put someone else in the hospital.

If you are well enough to go to the Student Health Center, you can use class time to see a doctor or counselor and get a note. If not, stay home and worry about the note later.

2. I do not distinguish between mental health and physical health. If you cannot complete an assignment on time or come to class because of mental health issues, you must contact me and provide a note, just as with physical health issues. You are not obliged to get into detail in either case; your doctor or counselor will provide a generic note saying that you had a health problem and were unable to attend class.

Incompletes

The only acceptable reasons for taking an incomplete in the course are personal illness or a family emergency. Students who wish to take incompletes must present documentation of the problem to the instructor before final grades are due. Incompletes are not available before the Week 12 withdrawal deadline.

Behavior in Class

Part of the purpose of this class is to understand and practice professional behavior. This includes many areas, from email communications to working in pairs, and it would be impossible to list them all. However, these are the general expectations:

1. Behave respectfully towards everyone.
2. Put forth your best effort.
3. Follow through on commitments, including communicating when you can't meet them.

Content Warnings

This course is intended to support your creative explorations in code. For the original game you make as a final project, you are encouraged to make something that interests you. Generally this takes the form of lighthearted arcade-style games. However, if you include content in the work that you produce which may cause distress to your fellow students, please make a verbal 'content warning' immediately before you present the work in class, and include a written content warning, either at the beginning of a piece of written work, or in the readme file of a project, when you submit the work for grading.

Students who ever feel the need to step outside class during the presentation or discussion of work that warrants a content warning may always do so without academic penalty. (You will, however, be responsible for any material you miss. If you do leave the room for a significant time, please make arrangements to get notes from another student or see me individually.)

Content which requires a content warning includes graphic depictions or descriptions of violence, sexual acts, abuse (especially sexual abuse or torture), self-harming behavior such as suicide, self-inflicted injuries or disordered eating, eating-disordered behavior or body shaming, and depictions, especially lengthy or psychologically realistic ones, of the mental state of someone suffering abuse or engaging in self-harming behavior.

If you have any questions about what warrants a content warning, including visual, auditory or tactile depictions, textual or verbal descriptions, and meaning embodied in game mechanics and interaction patterns, please let me (the class instructor) know.

If you ever wish to discuss your personal reactions to material presented in class, either with the class or with me afterwards, I welcome such discussion as an appropriate part of our coursework.

Note for Students with Disabilities

Any student requesting academic accommodations based on a disability is required to register with Disability Services and Programs (DSP) each semester. A letter of verification for approved accommodations can be obtained from DSP. Please be sure the letter is delivered to me (or to an SA) as early in the semester as possible. DSP is located in STU 301 and is open 8:30 a.m.–5:00 p.m., Monday through Friday. The phone number for DSP is (213) 740-0776.

Academic Integrity

USC seeks to maintain an optimal learning environment. General principles of academic honesty include the concept of respect for the intellectual property of others, the expectation that individual work will be submitted unless otherwise allowed by an instructor, and the obligations both to protect one's own academic

work from misuse by others as well as to avoid using another's work as one's own. All students are expected to understand and abide by these principles. Scampus, the Student Guidebook, contains the Student Conduct Code in Section 11.00, while the recommended sanctions are located in Appendix A: <http://www.usc.edu/dept/publications/SCAMPUS/gov/>. Students will be referred to the Office of Student Judicial Affairs and Community Standards for further review, should there be any suspicion of academic dishonesty. The Review process can be found at <http://www.usc.edu/student-affairs/SJACS/>.

For this class, you are encouraged to copy and modify code from online sources and from class demonstration projects. You are also welcome to work together. If you use more than 3 lines of code from an external source without modifying it, you must provide a link to the source as a comment next to the copied code. You may use any code presented in class without attribution.

If you use any assets (images, textures, sounds, etc.) that are not your own work, you must name and link to the source in a separate credits text file.

Instructor Bio

Margaret Moser is an Assistant Professor of Practice at the USC School of Cinematic Arts, where she teaches game design, digital prototyping, design of mobile experiences, and experimental interface design.

Margaret has been designing and making interactive systems since 1998. Her work has been shown at Come Out & Play, Games4Change, and the Babycastles guerrilla game gallery in Brooklyn. She has built web-based games at MTV Networks, lectured on Agile production methods for small teams, and served as lead producer of two commercial iPad applications.

Recent projects include *1968*, an experimental narrative game being developed by the Leisure Society; *Accused*, a persuasive game sponsored by the Innocence Project; and *Trollympics*, and experimental HTML5 game recognized as a top entry in the 2013 Mozilla Game On challenge.

Margaret holds an MFA in Design and Technology from Parsons. Her research interests include persuasive games related to environmental issues, alternative and non-screen interfaces, and formal structures for interactive narratives. She splits her spare time between finding weird new games to play and the quest for the perfect taco.