For more info, watch this video: https://bit.ly/38wb169

# ITP 499 – Game Programming in Unreal

**Units: 4**

**Fall 2021—M/W—Time: 2-3:50PM**

**Location:** OHE 540

## Remote Attendance:

The general expectation of the University is that students will attend in-person unless they have an approved accommodation, are stuck overseas, or are currently experiencing symptoms and should isolate. This course will accommodate remote students. Contact VASE (viterbi.studentservices@usc.edu) for any further questions.

**Instructor:** Sanjay Madhav
**Office:** RRB 216
**Office Hours:** W/Th 4:30-6:30PM
**Contact Info:** madhav@usc.edu

**Teaching Assistant:** TBD
**Office:** TBD
**Office Hours:** TBD
**Contact Info:** TBD

## Course Description

An exploration of intermediate game programming techniques through the prism of Unreal Engine. This course is designed for students who want to expand on their existing game programming knowledge to become comfortable with programming C++ code for a game using Unreal.

## Learning Objectives

- Understand the fundamentals of how Unreal's game systems function including the game framework, characters, collisions, behavior trees, animations, user interfaces, and networking
- Implement and augment several systems in Unreal using C++
- Evaluate the quality and correctness of written code
- Create major new gameplay systems in Unreal

**Prerequisite(s):** As this is a 499 course, there is officially no prerequisite. However, the material in this course will be nearly impossible to follow if you have not previously taken ITP 380 (Video Game Programming) or otherwise have sufficient experience with C++ and game programming.

## Course Notes

This course is graded for a letter grade. Lecture slides and assignments will be posted on Blackboard. Outside of class, course discussion will take place on Piazza and Discord.

Twelve class meetings are "flipped classroom" meetings where students are expected to watch assigned videos prior to class. Students will be assigned work to complete during these meetings and must complete a Gradescope quiz each week to receive credit for the flipped classroom.

## Exams

There is a midterm and a cumulative final exam.

## Required and Supplemental Readings

The books for this class are available for free online through the USC library.

**Required**: Fozi, et.al. *Game Development Projects with Unreal Engine*. Packt Publishing. 2020. ISBN: 9781800209220. https://www.oreilly.com/library/view/game-development-projects/9781800209220/?ar

A handful of assigned readings are standalone articles which will be made available on Blackboard.

**Supplemental**: While there are no specific assigned readings from this book, students may find it helpful as a reference for programming assignments:
Doran, et.al. *Unreal Engine 4.x Scripting with C++ Cookbook*. Packt Publishing. 2019. ISBN: 9781789809503. https://www.oreilly.com/library/view/unreal-engine-4x/9781789809503/?ar

## Technological Proficiency and Hardware/Software Required

Unreal Engine requires a relatively strong PC or Mac to run well (recommended specifications). Each student will be provided a personal Virtual Machine to use during the course of the semester.Instructions on how to configure your computer to work with C++ and Unreal will be given in class. All software used in the class is free to download and use.

## Description of Assignments

Most flipped classroom activities will involve some amount of coding work which, depending on the activity, may involve working under the direction of the instructor, working individually, or working in small groups. This is to reinforce and expand on concepts from the lecture and required reading for that week.

The programming assignments will be individual assignments that expand on the materials covered in class meetings, and students should expect to spend 4-6 hours/week working on the programming assignments.

## Grading and Assessment of Assignments

Flipped classroom work will be graded Credit (CR)/No Credit (NC) based on whether the required activity items were completed in a satisfactory manner.

Exams are graded on a standard points scale from 0 to 100.

Programming assignments are graded using specifications grading[1,2] which lists specific requirements for each assignment. Students then recieve an initial grade based on their adherence to the specifications:

A – All specifications are met, there are no significant errors or bugs, and the code quality meets or exceed code review expectations
B – All specifications are met but there are bugs and/or code quality issues to address
C – Not all specifications are met, but at least 50% of them are
F – Less than 50% of the specifications are met

Students will receive their initial grade for each assignment with a list of requirements that should be satisfied in order to improve their grade. Students will have up to four days to make a regrade submission, upon which their assignment will be regraded. The grade can improve up to one letter grade (eg. B can

[1] Nilson, Linda B. *Specifications grading: Restoring rigor, motivating students, and saving faculty time*. Stylus Publishing, LLC, 2015.

[2] Roberson, Christian. "Techniques for using specifications grading in computer science." *Journal of Computing Sciences in Colleges* 33.6 (2018): 192-193.

improve to A, C can improve to B, and F can improve to C). The purpose of this grading system is to become comfortable with the standard software engineering industry practice of code reviews, as well as to allow room to learn from mistakes.

Final course grades are tabulated according to the table on the next page. For the "programming assignment" grades, the grade after any regrades is what counts towards the final grade criteria. Students must satisfy all criterion listed for a grade in order to receive that grade.

## Assignment Submission Policy

All assignments must be submitted on GitHub in order to be graded. Instructions will be provided in class and on Blackboard.

## Late Policy for Assignments

We may make exceptions on the "at most one letter on a regrade" rule on a case by case basis, if you provide a documented and sufficient reason to the instructor. Beyond this, the regrade policy allows students to resubmit each assignment to improve their grade by one letter grade.

## Make-up Policy for Exams

To make up for a missed exam, the student must provide a satisfactory reason (as determined by the instructor) along with documentation. Make-up exams are only allowed under extraordinary circumstances.

## Plagiarism and Individual Work Policy

In this class, programming assignments are expected to represent the individual effort of each student. All programming assignment submissions will be compared with current, previous, and future students' submissions using MOSS, which is a code plagiarism identification program. If your code significantly matches another student's submission, you will be referred to SJACS with a recommended penalty of an F in the course.

It is okay to discuss solutions to specific problems with other students, but it is not okay to look through another student's code. It does not matter if this code is online or from a student you know, it is cheating. Do not share your code with anyone else in this or a future section of the course, as allowing someone else to copy your code carries the same penalty as copying the code yourself.

## Course Material Policy

Do not reproduce, distribute, or post any assignment solutions or exams publicly without written consent of the instructor, as this will be considered an academic integrity violation. Except where otherwise noted, the lecture and assignment materials for this course are © 2021 Sanjay Madhav, and are licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

| Grade | Criteria |
|---|---|
| A | • Credit (CR) for at least 9/12 flipped classrooms<br>• Get a B or higher on all 12 programming assignments<br>• Get an A on at least 8/12 programming assignments<br>• Average exam grade of at least 85% |
| A- | • Credit (CR) for at least 9/12 flipped classrooms<br>• Get a B or higher on at least 11/12 programming assignments<br>• Get an A on at least 6/12 programming assignments<br>• Average exam grade of at least 80% |
| B+ | • Credit (CR) for at least 8/12 flipped classrooms<br>• Get a B or higher on at least 11/12 programming assignments<br>• Get an A on at least 4/12 programming assignments<br>• No Fs on any lab assignments<br>• Average exam grade of at least 75% |
| B | • Credit (CR) for at least 7/12 flipped classrooms<br>• Get a B or higher on at least 10/12 programming assignments<br>• Get an A on at least 3/12 programming assignments<br>• No Fs on any lab assignments<br>• Average exam grade of at least 75% |
| B- | • Credit (CR) for at least 6/12 flipped classrooms<br>• Get a B or higher on at least 9/12 programming assignments<br>• No more than 1 grade of F on any programming assignments<br>• Average exam grade of at least 70% |
| C+ | • Credit (CR) for at least 6/12 flipped classrooms<br>• Get a B or higher on at least 8/12 programming assignments<br>• No more than 2 grades of F on any programming assignments<br>• Average exam grade of at least 70% |
| C | • Credit (CR) for at least 5/12 flipped classrooms<br>• Get a B or higher on at least 7/12 programming assignments<br>• No more than 2 grades of F on any programming assignments<br>• Average exam grade of at least 65% |
| C- | • Credit (CR) for at least 4/12 flipped classrooms<br>• Get a B or higher on at least 6/12 programming assignments<br>• No more than 3 grades of F on any programming assignments<br>• Average exam grade of at least 60% |
| D+ | • Credit (CR) for at least 3/12 flipped classrooms<br>• Get a B or higher on at least 5/12 programming assignments<br>• No more than 4 grades of F on any programming assignments<br>• Average exam grade of at least 60% |
| D | • Credit (CR) for at least 3/12 flipped classrooms<br>• Get a B or higher on at least 5/12 programming assignments<br>• No more than 4 grades of F on any programming assignments<br>• Average exam grade of at least 55% |
| D- | • Credit (CR) for at least 3/12 flipped classrooms<br>• Get a B or higher on at least 5/12 programming assignments<br>• No more than 4 grades of F on any programming assignments<br>• Average exam grade of at least 50% |
| F | • Fail to meet criteria for a D- |

## Course Schedule

| Date | Lectures/Flipped Classrooms | Readings | Due Dates |
|---|---|---|---|
| 8/23 | Intro; Unreal Basics; Blueprints | | |
| 8/25 | FC1: Basic Game Prototype | Fozi – Ch. 1 | |
| 8/30 | Working with C++ in Unreal; Game Feature Plugins | Olano – "Navigating a Large Codebase" | **PA1: 8/31 @ 11:59PM** |
| 9/1 | FC2: Adding C++ Classes in Unreal | Fozi – Ch. 2 & 3 | |
| 9/6 | ***Labor Day (No class)*** | | **PA2: 9/7 @ 11:59PM** |
| 9/8 | FC3: Physics and Collisions | Fozi – Ch. 5 & 6 | |
| 9/13 | Extending the Game Framework | | **PA3: 9/14 @ 11:59PM** |
| 9/15 | FC4: Extending ActorComponent | Fozi – Ch. 9 | |
| 9/20 | User Interface Systems | | **PA4: 9/21 @ 11:59PM** |
| 9/22 | FC5: Custom UserWidgets | Fozi – Ch. 8 | |
| 9/27 | Audio Systems | | **PA5: 9/28 @ 11:59PM** |
| 9/29 | FC6: Audio / Particle Effects | Fozi – Ch. 7 | |
| 10/4 | Animations, Blending, Montages | | **PA6: 10/5 @ 11:59PM** |
| 10/6 | Midterm Exam Review | | |
| 10/11 | **Midterm Exam** | | |
| 10/13 | FC7: Improved Animations | Fozi – Ch. 11 | |
| 10/18 | Behavior Trees | | **PA7: 10/19 @ 11:59PM** |
| 10/20 | FC8: Basic BTNodes | Fozi – Ch. 13 | |
| 10/25 | Networked Multiplayer Basics | | **PA8: 10/26 @ 11:59PM** |
| 10/27 | FC9: Setting up Server/Client | Fozi – Ch. 16 | |
| 11/1 | RPCS; Networking Character Movement | | **PA9: 11/2 @ 11:59PM** |
| 11/3 | FC10: Networked Player Teleport | Fozi – Ch. 17 & 18 | |
| 11/8 | Plugins and Modifying the Engine | | **PA10: 11/9 @ 11:59PM** |
| 11/10 | FC11: Input Interface Plugin | Olano – "Basic Plugins" | |
| 11/15 | Working with Different Platforms | | **PA11: 11/16 @ 11:59PM** |
| 11/17 | FC12: Coding for IOS or Android | "Mobile Game Development" | |
| 11/22 | Gameplay Ability System | Krestanek – "GAS Documentation" | |
| 11/24 | ***Thanksgiving Holiday (No class)*** | | |
| 11/29 | Future of Unreal | | |
| 12/1 | Final Exam Review | | **PA12: 12/3 @ 11:59PM** |
| **FINAL** | **Final Exam** – Refer to the final exam schedule in the USC Schedule of Classes at classes.usc.edu | | |

*Notes*:
FC = Flipped Classroom. Students will be expected to have completed the assigned reading prior to these sessions, and students must attend them to complete the in-class activities. The flipped classroom meetings are highlighted in the schedule for clarity.

PA = Programming Assignment, the (almost weekly) homework assignments.

Readings in quotation marks will be made available on Blackboard.

## Statement on Academic Conduct and Support Systems

### Academic Conduct:

Plagiarism – presenting someone else's ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in SCampus in Part B, Section 11, "Behavior Violating University Standards" policy.usc.edu/scampus-part-b. Other forms of academic dishonesty are equally unacceptable. See additional information in SCampus and university policies on scientific misconduct, policy.usc.edu/scientific-misconduct.

### Support Systems:

*Counseling and Mental Health - (213) 740-9355 – 24/7 on call*
studenthealth.usc.edu/counseling
Free and confidential mental health treatment for students, including short-term psychotherapy, group counseling, stress fitness workshops, and crisis intervention.

*National Suicide Prevention Lifeline - 1 (800) 273-8255 – 24/7 on call*
suicidepreventionlifeline.org
Free and confidential emotional support to people in suicidal crisis or emotional distress 24 hours a day, 7 days a week.

*Relationship and Sexual Violence Prevention Services (RSVP) - (213) 740-9355(WELL), press "0" after hours – 24/7 on call*
studenthealth.usc.edu/sexual-assault
Free and confidential therapy services, workshops, and training for situations related to gender-based harm.

*Office of Equity and Diversity (OED) - (213) 740-5086 | Title IX – (213) 821-8298*
equity.usc.edu, titleix.usc.edu
Information about how to get help or help someone affected by harassment or discrimination, rights of protected classes, reporting options, and additional resources for students, faculty, staff, visitors, and applicants.

*Reporting Incidents of Bias or Harassment - (213) 740-5086 or (213) 821-8298*
usc-advocate.symplicity.com/care_report
Avenue to report incidents of bias, hate crimes, and microaggressions to the Office of Equity and Diversity |Title IX for appropriate investigation, supportive measures, and response.

*The Office of Disability Services and Programs - (213) 740-0776*
dsp.usc.edu
Support and accommodations for students with disabilities. Services include assistance in providing readers/notetakers/interpreters, special accommodations for test taking needs, assistance with architectural barriers, assistive technology, and support for individual needs.

*USC Campus Support and Intervention - (213) 821-4710*
campussupport.usc.edu
Assists students and families in resolving complex personal, financial, and academic issues adversely affecting their success as a student.

*Diversity at USC - (213) 740-2101*
diversity.usc.edu
Information on events, programs and training, the Provost's Diversity and Inclusion Council, Diversity Liaisons for each academic school, chronology, participation, and various resources for students.

*USC Emergency - UPC: (213) 740-4321, HSC: (323) 442-1000 – 24/7 on call*
dps.usc.edu, emergency.usc.edu
Emergency assistance and avenue to report a crime. Latest updates regarding safety, including ways in which instruction will be continued if an officially declared emergency makes travel to campus infeasible.

*USC Department of Public Safety - UPC: (213) 740-6000, HSC: (323) 442-120 – 24/7 on call*
dps.usc.edu
Non-emergency assistance or information.

*Office of the Ombuds - (213) 821-9556 (UPC) / (323-442-0382 (HSC)*
ombuds.usc.edu
A safe and confidential place to share your USC-related issues with a University Ombuds who will work with you to explore options or paths to manage your concern.