San Jose State University
Department of Computer Science
CS134, Computer Game DESIGN

Spring Semester 2022

Course and Contact Information

Instructor:	Kevin Smith
Office Location:	Online
Email:	kevin.smith@sjsu.edu
Office Hours:	Online (See Canvas Meeting Times)
Class Days/Time:	Tuesday and Thursdays – 1:30 AM – 2:45 AM PST
Classroom:	Science Building 311
Prerequisites:	Prerequisite: CS 146 and either CS 151 or CMPE 135 (with a grade of "C-" or better in each); or instructor consent.

Catalogue Description

Architectures and object-oriented patterns for computer game design. Animation, simulation, user interfaces, graphics, and intelligent behaviors. Team projects using an existing game engine framework.

Course Description

In this course, you will learn the critical elements in the design and implementation of a computer games from the ground up. This will include some of core components required to implement a modern high-performance game engine. The course will initially focus on 2D games and then we will extend our knowledge to include 3D. You will implement required functionality in your own game engine to support navigation, animation, physics, audio and user-input through designing and building an actual game. We will augment our knowledge with case studies of existing games and current commercially available game engines.

Course Learning Outcomes (CLO)

Upon successful completion of this course, students will be able to:

1: Understand how modern video games (2D and 3D) are designed and implemented.

2: Design and create a game from scratch in C++ (using a C++ graphics).

3: Learn the techniques and design patterns used to develop high-performance game engines.

4: Develop a physics-based animation system for player motion and visual effects

5: Gain an understanding of the current state-of-the art in game technology through use-cases.

Required Texts/Readings

Textbook

There is no textbook required for CS134. The following books are recommended reading:

Jason Gregory, *Game Engine Architecture (Second Edition).*
David H. Eberly, *3D Game Engine Design*
Robert Nystrom, *Game Programming Patterns*

Software and Computer

Students will be required to have access to a modern capable laptop or desktop computer running recent version of Windows or MacOS. It is preferable to have a machine with a GPU. In addition to a computer, a three-button mouse is required for the programming assignments. The development projects for this class will be done in C++. Students will be required to download and install a development framework for their particular operating system including Visual Studio (Windows) or Xcode (macOS) and a C++ graphics development library (instructions will be provided on first day of class).

Software Packages

Students are required to use the following software packages for this course:

1. Visual Studio 2017 Free Community Version (PC) or Xcode (MAC)
2. Adobe Photoshop CC or equivalent open software package such as GIMP (for sprites)
3. Camtasia or SnagIt Video Capture Software (or equivalent)
4. Autodesk Maya (free student version available)
5. OpenFrameworks C++ Library (Open Source – current version)

Adobe Photoshop will be used in the class for creating game content, such as sprites, background images and textures.

Autodesk Maya will be used for generating 3D content.

Camtasia or SnagIt will be used for creating videos of your assignments and projects.

Course Requirements and Assignments

It is expected that students will spend a minimum of forty-five hours for each unit of credit (normally three hours per unit per week), including preparing for class, participating in course activities, completing assignments, and so on. More details about student workload can be found in University Policy S12-3 at http://www.sjsu.edu/senate/docs/S12-3.pdf.

1. Development Projects (50%)
Students will complete a series of development projects involving the use of C++ and/or production tools covered in the class. The projects will be specified on Canvas.

2. In Class Exercises (5%)
Students are expected to attend every class and be engaged. This component of the grade will be determined by graded short in-class exercises. These are normally due on the same day of class. Longer labs may be given additional time.

3. Mid-Term Exam (10%)
The student will be required to take a closed book mid-term exam or a take-home exam which will cover material presented in class and the reading material assigned. The mid-term may also include problems to be solved.

4. Final Project (35%)
The student project will be comprehensive game prototype that will leverage concepts learned and components from pervious assignments.

Projects

For "Development Projects"" specified in (1) above, students will complete a series of sequential programming projects that will be assigned during the semester. Most of the projects will be dependent on the previous and the final project will be the culmination of the previous projects, therefore, it is required that all projects be completed to be successful in the course. Some of the projects may be a "team" project where students can work together and present their results. Students will post either a still frame or video of the project on the Google Class Community Page. Instructions for posting will be provided on Canvas.

Final Project

In lieu of a final exam, the students will be required to submit a final project which demonstrates comprehensive knowledge learned in the class.

Gallery

A Google Community will be provided for the course where students will be required post a movie of their assignments in a Gallery.

Academic Honesty

Students are required to produce their own work either individually or with a team member if the assignment is a team-based assignment. Each code source file, whether you work in a team or individually, must contain a header comment at the top of the file that includes the author's (the student's) name. It is recommended that when teams work together, the work is divided so that each team member works on different non-conflicting

source files. If a source file is shared, then each team member must create signed comment block for the section of the code within that source file (ex: method or class) identifying their own work.

If you use any code or algorithms from sources outside of the class, you must include a citation for it in your work. It is expected that the vast majority of the work is your own original work and you will be graded on your contribution to the project not on cited code from external sources.

Zero Tolerance Plagiarism Policy

If student work is determined to be copied or derived from another source, the work assignment or project will receive an automatic grade of "0" and an Academic Integrity Report will be filed.

Course Materials and Copyright Information

The course materials including slides, notes, example code and videos are all copyright by the author (Kevin M. Smith). Copying these materials or work derived from it without permission from the author is prohibited by law. This includes copying to other third-party websites or services.

Grading Policy

No make-up tests (exams and quizzes) will be given and ***no*** **late work will be accepted**. This includes: homework, projects, videos, in-class exercises or any other work related to the class. If an exam or work is missed or late, it will be graded as a "0". If you are in doubt about the submission time for an assignment, it is better to submit it early.

At least	Grade
97%	A+
93%	A
90%	A-
87%	B+
83%	B
80%	B-
77%	C+
72%	C
70%	C-
67%	D+
62%	D
60%	D-
<60%	F

Note that "All students have the right, within a reasonable time, to know their academic scores, to review their grade-dependent work, and to be provided with explanations for the determination of their course grades." See University Policy F13-1 at http://www.sjsu.edu/senate/docs/F13-1.pdf for more details.

NOTE that University policy F69-24 at http://www.sjsu.edu/senate/docs/F69-24.pdf states that "Students should attend all meetings of their classes, not only because they are responsible for material discussed therein, but because active participation is frequently essential to insure maximum benefit for all members of the class. Attendance per se shall not be used as a criterion for grading."

Classroom Protocol

Class attendance is required to gain maximum benefit from the presented materials, presentations and discussion.

Laptop are used for the in-class assignment/exercises and not permitted (including tablets) during the lecture part of the class.

Cell phones are not permitted to be used in class.

Please be courteous and minimize any classroom distractions which may affect the learning environment including conversation, eating, taking unnecessary breaks and coming to class late.

Students with special requirements should notify the instructor or contact the Accessible Education Department.

Since the material presented in class is copyrighted, there is no photography allowed (including mobile phone cameras).

University Policies

Per University Policy S16-9, university-wide policy information relevant to all courses, such as academic integrity, accommodations, etc. will be available on Office of Graduate and Undergraduate Programs' Syllabus Information web page at http://www.sjsu.edu/gup/syllabusinfo/

CS134, Computer Game Design, Course Schedule

This schedule is tentative and is subject to change. Due dates for assignments will be posted in Canvas and are generally due the following week after are assigned.

Course Schedule

Week	Date	Topics, Readings, Assignments, Deadlines
1	1/26	Introduction and Development Environment, Game Engine Dev
2	1/31, 2/2	Vector Mathematics Review Basic 2D Vintage Arcade Game Project
3	2/7, 2/9	**Feb 7 – Last Day to Drop** Basic 2D game interactivity - Rendering/Drawing and Input
4	2/14, 2/16	Basic 2D game interactivity – Sprites and Animation
5	2/21,2/23	Physics – Physics Engine Basics
6	2/28,3/2	Physics – Trajectory, Ballistic Motion and Integrators Introduction Exercise in 3D
7	3/7.3/9	Physics – Particles Systems Architecture and Forces
8	3/14, 3/16	Physics – Collision Detection
9	3/21 3/23	**Review** **Midterm Exam – take home**
10	3/28-3/30	**Spring Recess**
11	4/4, 4/6	3D Computer Graphics for Games - I
12	4/11, 4/13	3D Computer Graphics for Games - II
13	4/18, 4/20	3D Game Design
14	4/25, 4/27	3D Game Design
15	5/2. 5/4	Game Engine Case Study
16	5/9, 5/11	Open Topic
17	5/16	**Final Project Due**