San José State University
Department of Computer Science
CS134, Computer Game DESIGN, Section 1

Fall Semester, 2018

Course and Contact Information

Instructor:	Kevin Smith
Office Location:	DH 282
Email:	kevin.smith@sjsu.edu
Office Hours:	Tues 1300-1400 or by appointment
Class Days/Time:	MW 1200-1315
Classroom:	MH 222
Prerequisites:	CS 130 or CS 116A, 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.

Prerequisite: CS 146 and either CS 151 or CMPE 135 (with a grade of "C-" or better in each); or instructor consent.

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, path-finding, 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++ wrapper library).

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

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

Required Texts/Readings

Textbook

The following textbook is required for the class:

Jason Gregory, *Game Engine Architecture (Second Edition).*

Recommended Books

Although not required, the following books are recommended:

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. Unity or Unreal Engine 4 – Game Engine (Downloadable Free Version Available)
2. Godot Game Engine (Open Source)
3. Visual Studio 2017 Free Community Version (PC) or Xcode (MAC)
4. Adobe Photoshop CC (or equivalent)
5. Camtasia or SnagIt Video Capture Software (or equivalent)
6. Autodesk Maya (free student version available)
7. OpenFrameworks 0.10.0 C++ Library (Open Source)

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 (70%)
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. Engagement (5%)
Students are expected to attend every class and be engaged. This component of the grade will be determined by quiz results and graded short in-class exercises.

3. Mid-Term Exam (10%)
The student will be required to take a closed book mid-term 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 (15%)
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 approximately five (5) 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 student 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 have to 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.

Grading Policy

***No* make-up tests will be given and *no* late homework (or other work) will be accepted. 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 and it part of the *engagement* component of the grade.

Laptop or tablet use is encouraged for taking notes during the class. Students should practice common courtesy and refrain from using laptops for email, messaging or social media during class.

Cell phones are generally not permitted to be used in class (including text messaging).

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	8/22	Introduction and Development Environment
2	8/27, 8/29	Parts of a Game Engine, Vector Mathematics Basic 2D Vintage Arcade Game Project
3	9/5	Basic 2D game interactivity - Rendering/Drawing
4	9/17, 9/19	Basic 2D game interactivity – Sprites and Animation
5	9/24. 9/26	Physics – Physics Engine Basics
6	10/1, 10/3	Physics – Trajectory, Ballistic Motion and Integrators Introduction Exercise in 3D
7	10/8, 10/10	Physics – Particles Systems Architecture and Forces
8	10/15, 10/17	Physics – Collision Detection Review
9	10/29 10/31	**Midterm Exam (Monday, October 29)** Putting it all together in your 2D Arcade Game Project
10	11/5, 11/7	3D Computer Graphics for Games
12	11/14	Extending your game project to 3D
13	11/19	Commercial Game Engines – Introduction and Case Study
14	11/26, 11/28	Game Engine Character Animation
15	12/3. 12/5	Game Engine Lighting/Rendering
15	12/10	Open Topic
16/17	12/12-12/18	**Final Project Presentations**