CS 381: Game Engine Architecture

Sushil J. Louis
http://www.cse.unr.edu/~sushil/class/381
Classroom: Wednesdays 1pm: SEM 261
Laboratory: Mondays 1pm: SEM 231B and SEM 231C
Office: SEM 237/203

Catalog Description

CS 381, Game Engine Architecture
Credit(s): 3

Introduction to the technical elements of modern videogame and the pipeline for assembling
them, plus issues of interface design, quality assurance, and business practice.

Prereq(s): CS202 or consent of instructor

Course Objectives

This course introduces students to game engine architecture and the game development pipeline
by teaching students how to design and implement a 3D game engine based (this semester) on the
open source Ogre graphics engine. The course will cover the following topics.

• The Ogre graphics engine
• Game engine architecture
• Adding 3D models to a Scene
• Keyboard interaction
• Game entity management
• Introductory game physics
• Introductory AI
• Interaction design
• If time permits
  – Sound
  – Multiplayer networking
Office hours

- Sushil
  - MTW from 11:00 a.m. until noon
  - By appointment (sushil@cse.unr.edu). Please check my calendar [https://www.cse.unr.edu/~sushil/calendar/](https://www.cse.unr.edu/~sushil/calendar/), find an open spot, and send me a calendar invite or an email for that time. Typically, 15 - 30 minutes.

- Nathan
  - TR from 1:30 p.m. until 4:30 pm in the ECC labs
  - Fridays from 12:30 - 1:30 p.m. in the ECC labs
  - By appointment email: navarrogriffin@gmail.com

- Nick
  - MW from 4:00 - 6:00 p.m. in the ECC labs
  - Fridays from 1:30 - 2:30 p.m. in the ECC labs
  - By appointment email: nicholasharris42@gmail.com

Course Arrangement

The course has two components: Lecture and Laboratory. Both lecture and lab components are mandatory. Note specifically that the laboratory component is mandatory. All assignments, homework, quizzes, and exams are required.

Textbook

- **Required**: Blender Wiki/Manual, [https://www.blender.org/support/](https://www.blender.org/support/)

You will have to use the information above, class notes posted on the class web page, and your own notes to study for tests and to do the assignments.

Required Materials

- ECC computer accounts. You should automatically have one, if you are enrolled in the class.
Unique Class Procedures/Structures

- One lecture on Wednesdays (in SEM 261) and one lab on Mondays (in JTB 125) per week
- Lab portion requires coding in lab
- Required in class demonstrations of student assignments
- Peer review of final projects
- No textbook
- Expected independent learning from online tutorials

Important Dates

- Exam 1: Tentative: March 13, 2019
- Exam 2: Tentative: April 10, 2019
- Final Exam and Project Presentations: Monday May 13, 2019, 9:50 a.m. - 11:50 p.m. If you cannot make this time, please drop this course.

Assignments

Assignments require writing C/C++ programs to integrate and “glue” existing and newly created components into a game engine. You will be using the linux boxes in the engineering computer labs in SEM and JTB. Each programming assignment will be demonstrated in class/lab and graded on the demonstration.

The set of assignments will culminate in a 3D game engine that you will use to brainstorm, design, and build an exemplar 3D computer game.

Course Rules

- Late programming assignments or exercises will not be accepted

- Exams/Tests/Quizzes (if any) are individual efforts. A severe penalty will be given for collusion or other form of academic dishonesty. The usual penalty for academic dishonesty on assignments or an exam is failure in the course.

- Each assignment will specify whether the assignment is an individual effort or a group assignment. Maximum group size is two (2).

- For individual assignments, do not show, exchange, or copy code. Using another person’s code or having another person ”ghost write” a lab will be considered academic dishonesty. For group assignments, if you are in a group do not show, exchange, or copy code outside your group.

- The final project is a group project.
Preliminary Information on Grading and Assessment

- Students will be assigned letter grades. Your grade will be one of A, B, C, D, or F. We will use the +/- grading system.

- There will be a number of programming assignments. No late assignments will be accepted.

- There will be one or two midterm exams. Midterms take place in your regular classroom.

- The **Final Exam: is on Monday May 13, 2019 from 9:50 - 11:50.** If you cannot make this date, please drop the course.

- There will be announced and unannounced quizzes.

Tentatively, your final grade will be based on:

<table>
<thead>
<tr>
<th>Item</th>
<th>Final Grade Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Midterm exams (10% each)</td>
<td>20%</td>
</tr>
<tr>
<td>Assignments</td>
<td>50%</td>
</tr>
<tr>
<td>Final Project &amp; Exam</td>
<td>25%</td>
</tr>
<tr>
<td>Quizzes and Exercises</td>
<td>5%</td>
</tr>
</tbody>
</table>

Preliminary Course Outline

This outline is approximately in order.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Textbooks</th>
<th>Lectures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>Syllabus</td>
<td>1</td>
</tr>
<tr>
<td>GFX: Setting up Ogre</td>
<td>Tutorial 1</td>
<td>1</td>
</tr>
<tr>
<td>GFX: Coordinate Systems, Scenes, Entities, lights</td>
<td>Tutorial 1, Tutorial 2, Tutorial 3</td>
<td>4</td>
</tr>
<tr>
<td>GFX/UI: Input/Output Systems</td>
<td>Tutorial 4, Tutorial 5,</td>
<td>1</td>
</tr>
<tr>
<td>GFX/UI: Cameras, mice, keyboards</td>
<td>Tutorial 4, Tutorial 5,</td>
<td>1</td>
</tr>
<tr>
<td>GFX: Ogre architecture</td>
<td>Tutorial 6</td>
<td>1</td>
</tr>
<tr>
<td>Game Engine Architecture</td>
<td>Notes</td>
<td>1</td>
</tr>
<tr>
<td>Entity Management: Creation</td>
<td>Notes</td>
<td>1</td>
</tr>
<tr>
<td>PHX: Entity movement: Physics</td>
<td>Notes</td>
<td>2</td>
</tr>
<tr>
<td>AI: Entity smarts: Unit AI</td>
<td>Notes</td>
<td>1</td>
</tr>
<tr>
<td>AI: Entity Group smarts</td>
<td>Notes</td>
<td>1</td>
</tr>
<tr>
<td>UI: Game Interaction (UI)</td>
<td>Notes</td>
<td>2</td>
</tr>
<tr>
<td>Net: Game Networking</td>
<td>Notes</td>
<td>2</td>
</tr>
<tr>
<td>Brainstorming a game</td>
<td>None</td>
<td>1</td>
</tr>
<tr>
<td>Designing and finishing a complete game</td>
<td>None</td>
<td>8+</td>
</tr>
</tbody>
</table>


Student Learning Outcomes

- Students will have an ability to identify, formulate, analyze, and solve complex computing or engineering problems by applying principles of computing, engineering, science, and mathematics.

  - Students will be able to demonstrate they can integrate existing components to build a 3D game engine

  - Strategies and Actions:
    * Lectures, assignments, and projects covering game engine architecture and existing game engine components in particular, Ogre.

  - ABET Criteria covered:
    * (1) an ability to select and apply the knowledge, techniques, skills, and modern tools of the discipline to broadly-defined engineering technology activities
    * (6) an ability to identify, analyze, and solve broadly-defined engineering technology problems

  - Program Objectives covered:
    * Demonstrate strong analytic, design, and implementation skills required to formulate and solve computer science or computer engineering problems in a professional or research environment

  - Assessment:
    * Programming assignments and projects
    * Test/Exams

- Students will have an ability to apply computer science theory and software development fundamentals to produce computing-based solutions.

  - Students demonstrate they can learn and use computing tools in the pursuit of a broader project goal.

  - Strategies and Actions:
    * Lectures, assignments, and projects covering game engine architecture and existing game engine components and in integrating them into a game engine

  - ABET Criteria covered:
    * (1) an ability to select and apply the knowledge, techniques, skills, and modern tools of the discipline to broadly-defined engineering technology activities
    * (6) an ability to identify, analyze, and solve broadly-defined engineering technology problems

  - Program Objectives covered:
    * Demonstrate strong analytic, design, and implementation skills required to formulate and solve computer science or computer engineering problems in a professional or research environment
– Assessment:
  * Programming assignments that build a game engine from components and use that game engine to build a playable game

**Statement on Academic Dishonesty**

The University Academic Standards Policy defines academic dishonesty, and mandates specific sanctions for violations. See the University Academic Standards policy: UAM 6,502

**Statement of Disability Services**

Any student with a disability needing academic adjustments or accommodations is requested to speak with me or the Disability Resource Center as soon as possible to arrange for appropriate accommodations. This course may leverage 3rd party web/multimedia content, if you experience any issues accessing this content, please notify your instructor.

**Statement for Academic Success Services**

Your student fees cover usage of the Math Center (784-443 or [http://www.unr.edu/mathcenter/](http://www.unr.edu/mathcenter/)), Tutoring Center (784-6801 or [www.unr.edu/tutoring-center](http://www.unr.edu/tutoring-center)), and University Writing Center (784-6030 or [http://www.unr.edu/writing-center](http://www.unr.edu/writing-center)). These centers support your classroom learning; it is your responsibility to take advantage of their services. Keep in mind that seeking help outside of class is the sign of a responsible and successful student.

Note that the University Math Center (UMC) is focused on helping students with mathematical and statistical concepts. While mathematics is used extensively in engineering, the UMC does not have the resources to help students with engineering courses. Engineering students are encouraged to use the UMC for help in their math classes, and they are welcome to use its computer lab and study area any time regardless of course. However, UMC tutors cannot answer questions regarding engineering courses.

**Engineering Tutoring**

The Engineering Tutoring Center offers drop-in peer tutoring, specifically for engineering students. They are located in Edmund J. Cain Hall 108H.

**Statement on Audio and Video Recording**

Surreptitious or covert video-taping of class or unauthorized audio recording of class is prohibited by law and by Board of Regents policy. This class may be videotaped or audio recorded only with the written permission of the instructor. In order to accommodate students with disabilities, some students may be given permission to record class lectures and discussions. Therefore, students should understand that their comments during class may be recorded.
Communications

If I need to communicate with the class as group I will post a message on our class web page. You are required to check the class web page and your email every day.

The University of Nevada, Reno is committed to providing a safe learning and work environment for all. If you believe you have experienced discrimination, sexual harassment, sexual assault, domestic/dating violence, or stalking, whether on or off campus, or need information related to immigration concerns, please contact the University’s Equal Opportunity & Title IX office at 775-784-1547. Resources and interim measures are available to assist you. For more information, please visit the Equal Opportunity and Title IX page.