COM 299, Game Development

American University of Central Asia
Software Engineering Department

1 Course Information

Course ID
COM 299

Course ID
3956

Prerequisite
COM-119, Object Oriented Programming

Credits
6

Professors, TAs, Time, Place
Lecture (Dmitrii Toksaitov): Tuesday 10:50–12:05, Online
Wednesday (Dmitrii Toksaitov): Thursday 10:50–12:05, 433

Course Repository
`https://github.com/auca/com.299`

2 Contact Information

Instructor
Toksaitov Dmitrii Alexandrovich
toksaitov_d@auca.kg

Office
AUCA, room 315

Office Hours
By appointment throughout the work week (write to me to make an appointment)

3 Course Overview

The course introduces students to a topic of game development. It covers theory and practice of video game production. It delves into such fields as computer graphics, computational physics, artificial intelligence, and game-play design. During the course students will get an opportunity to build two market-ready games for desktop, web, or mobile platforms. They will not only learn on how to create their own lightweight graphics, physics and game-play engines, but also how to use heavyweight 3-rd solutions such as Unity 3D or Unreal Engine.

4 Topics Covered

- Week 1: Introduction, History, Industry Overview (3 hours)
- Week 1–2: The Unity Engine (6 hours)
- Week 2: Vectors (3 hours)
- Week 3: Matrices (3 hours)
- Week 4: Space Transformation (3 hours)
- Week 1–7: C# Scripting (20 hours)
- Week 2–9: The Unity OOP Model (20 hours)
- Week 10–11: The Graphics, Physics and UI Subsystems (9 hours)
- Week 12: The AI Subsystems (3 hours)
- Week 13–14: Game Engine Internals (6 hours)

5 Course Projects

Throughout the course, students will have to create two games on their own. The first game will be a recreation of a classical game to study the development environments. The second game will be a clone of some current popular product on the market.

6 Presentation

Students will get a chance to make one presentation about any market game of their choice. The presentation should be focused on the game's internals, its development or production process, and tools or techniques used to create it

7 Course Materials, Recordings and Screencasts

Students will find all the course materials on GitHub. We hope that by working with GitHub, students will become familiar with the Git version control system and the popular (among developers) GitHub service. Though version control is not the focus of the course, some course tasks may have to be submitted through it on the GitHub Classroom service.

Every class is screencasted online and recorded to YouTube for students convenience. An ability to watch a class remotely at any time MUST NOT be a reason not to attend the online Zoom session. Active class participation is necessary to succeed in this course.

8 Software

Students are recommended to install the following software on their machines.

- Unity: `https://unity.com`
- Visual Studio: `https://visualstudio.microsoft.com`

It is required to register accounts to start using the Unity Engine. It is recommended to install Unity first, install Visual Studio through it next.

9 Reading

1. 3D Math Primer for Graphics and Game Development, Second Edition by Fletcher Done and Ian Parberry (ISBN: 978-1-4398-6981-9)

9.1 Supplemental Reading

1. Game Development Essentials: An Introduction 3rd Edition by Jeannie Novak (ISBN: 978-1111307653)
2. Game Coding Complete, Fourth Edition by McShaffry and David Graham (ISBN: 978-1133776574)
3. Game Engine Architecture, Second Edition by Jason Gregory (ISBN: 978-1568814131)
4. Game Programming Patterns by Robert Nystrom (ISBN: 978-0990582908)
5. Mathematics for 3D Game Programming and Computer Graphics, Third Edition by Eric Lengyel (ISBN: 978-1435458864)

10 Grading

- Midterm Practice Defence (Project #1): 35%
- Final Practice Defence (Project #2): 50%
- Presentation: 10%
- Participation (through the course Discussion Board): 5%

- 90%–100%: A
- 80%–89%: A-
- 70%–79%: B+
- 65%–69%: B
- 60%–64%: B-
- 56%–59%: C+
- 53%–55%: C
- 50%–52%: C-
- 46%–49%: D+
- 43%–45%: D
- 40%–42%: D-
- Less than 39%: F

11 Rules

Students are required to follow the rules of conduct of the Software Engineering Department and the American University of Central Asia.

11.1 Participation

Active work during the class may be awarded with up to 5 extra points at the instructors discretion.

Poor student performance during a class can lead to up to 5 points being deducted from the final grade.

Instructors may conduct pop-checks during classes at random without prior notice. Students MUST be ready for every class in order not to lose points.

11.2 Questions

We believe that a question from one student is most likely a question that other students are also interested in. That is why we encourage students to use the course Discussion Board to ask questions in public that other students can see and answer and NOT ask them through E-mail in private UNLESS the question itself is about private matters to discuss with the professor.

11.3 Late Policy

Late submissions and late exams are not allowed. Exceptions may be made at the discretion of the professor only in force-majeure circumstances.

11.4 Exam Ceremonies

Students MUST follow exam ceremonies. It means they MUST prepare task list forms with all points appropriately calculated. They MUST submit them correctly. They must bring task list forms to the exam. Failure to do so will result in lost points. Throughout your career, you will have to work with various supporting documents (contracts, timesheets, etc.). It is a good idea to start learning to work with such documents accurately early. We will remove points for not following these rules or even refuse to accept your exam defense.

11.5 Incomplete

As with late exams, the grade I may be awarded only in exceptional circumstances. The student must start a discussion on getting the grade I with the instructors in advance and not during the last week before the final exams.

11.6 Academic Honesty

Plagiarism can be defined as an act or an example of copying or stealing someone elses words or ideas and appropriating them as ones own. The concept of plagiarism applies to all tasks and their components, including program code, abstracts, reports, graphs, statistical tables, etc.

In addition to being unethical, this indicates that the student has not studied the given material. Tasks written from somewhere for 10% or less will be assessed accordingly or will receive a 0 at the discretion of the teacher. If plagiarism is more than 10%, the case will be transferred to the AUCA Disciplinary Committee.

Students are not recommended to memorize before exams, as this is a difficult and inefficient way to learn; and since practice exams consist of open questions designed to test a students analytical skills, memorization invariably leads to the fact that the answers are inappropriate and of poor quality.

On this course teamwork is NOT encouraged. The same blocks of code or similar structural pieces in separate submissions will be considered as academic dishonesty, and all parties will get zero for the task.

The following are examples of some common acts of plagiarism:

1. Representing the work of others as their own
2. Using other people's ideas or phrases without specifying the author
3. Copying code snippets, sentences, phrases, paragraphs or ideas from other people's works,published or unpublished, without referring to the author
4. Replacing selected words from a passage and using them as your own

5. Copying from any type of multimedia (graphics, audio, video, Internet streams), computer programs, graphs or diagrams from other people's works without representation of authorship

6. Buying work from a website or from another source and presenting it as your own work