NEW YORK UNIVERSITY — TISCH SCHOOL OF THE ARTS

INTRODUCTION TO GAME DEVELOPMENT

GAMES 121-001

SPRING 2019

COURSE SYLLABUS

Professor:
Technical Instructors:

Course Description

Introduction to Game Development is a practical course that introduces students to the methods, tools and principles used in developing digital games. Over the course of the semester, students will work alone to create three ‘clones’ of simple digital game prototypes, then elaborating them creatively to produce original work. This is a hands-on, primarily lab-based course, and so the focus is on learning-by-doing rather than on reading and discussion.

Course Objectives

At the completion of this course, the student will be able to:

1. Practice the fundamentally integrated technical processes of digital game development, by rolling together elements of visual art and design, sound design, systems design, interaction design and code.
2. Learn to implement game assets and code in an established digital game engine (Unity).
3. Identify major principles of implementation-level game design, and learn the ‘tricks of the trade’ that serve to engage a user and provide subconscious-level enjoyment of a game.
4. Analyze and articulate strengths and weaknesses in the student’s and classmates’ work.
5. Present their work to a group, highlighting its functionality and strengths.
6. Develop a personal creative process, allowing the student to translate ideas into the form of a digital game.

Course Format

Introduction to Game Development is primarily a project-based course. Weekly meetings consist of a lecture class, which will contain skill-building exercises and critical feedback sessions, and a lab, in which students can work with assistance and hands-on technical instruction from the instructor and the TA. Students are expected to be ready to present their work for feedback in every lecture class.

In weeks where students have completed creative designs, every student will present their work for feedback. In other weeks, only a random subset may be asked to present.

Assignments and Readings

The assignments in this class are aimed at teaching you some of the most popular game development patterns and the most useful features of Unity. To do that,

Assignment 0: Warmup Modification Project
Start by modifying a small sample project creatively for one week. Use this time to refresh your memory of Unity and C#.
Due: Lecture class, Week 2
Graded Pass/Fail

Assignment 1: Action Game Clone
Create an exact copy of a sample action game, given in class. This assignment requires you to use your knowledge of sprites, instantiation, input, vectors and timers.
Due: In class, Week 4
Graded A-F

Assignment 2: Action Game Elaboration
Creatively change the design and assets of your clone to turn it into an original game.
Due: In class, Week 5
Graded A-F

Assignment 3: Narrative Game Clone
Create an exact copy of a sample visual novel game, given in class. For this final clone, you will need to load text files, manage scenes and states, and use a singleton pattern
Due: In class, Week 7
Graded A-F

Assignment 4: Narrative Game Elaboration
Creatively change the design and assets of your clone to turn it into an original game.
Due: In class, Week 8
Graded A-F

Assignment 5: Grid Game Clone
Create an exact copy of a sample grid-based game, given in class. This assignment will require you to exercise your skills in arrays, functions, classes, and raycasts.
Due: In class, Week 10
Grade A-F

Assignment 6: Grid Game Elaboration
Creatively change the design and assets of your clone to turn it into an original game.
Due: In class, Week 11
Graded A-F

Assignment 7: Group Project
After forming into groups, you will select one of the earlier elaborated projects to turn into a more polished, ambitious game. For this final project you will learn about version control and basic project management.
Due: In class, Week 14

Readings

As this is a practical course there is no set text. However, we will be discussing short web-based readings and recorded lectures, including the following:

- Michael Brough, How To Do A Game Jam.
  http://mightyvision.blogspot.co.uk/2013/04/how-to-do-game-jam.html
- Chris De Leon: Game Programming fundamentals
  http://www.hobbygamedev.com/articles/vol5/game-programming-fundamentals/
- Petri Purho and Martin Jonasson: Juice it or Lose it (video).
  http://www.youtube.com/watch?v=Fy0aCDmgnxg
- Jonathan Blow: Push and Convey (audio + slides):
  http://braid-game.com/news/2008/02/another-lecture-this-time-from-denmark/

Prerequisites

Games 180 - Intro to Game Programming

Credits Allocated

4 Credits

Grading

Game assignments must be presented in class AND submitted via NYU Classes to be graded complete.

Since each phase of the class is divided by focus, the grading on each project will be different.

Clone projects will be graded on:

- **Functionality**. Does the game run, and is it free of bugs?
- **Completeness**. Has the student implemented all the main functions of the original game, as outlined by the instructor?
- **Fidelity.** Do the specified features function the same way in the clone as in the original?
- **Improvement**. Once the functions of the original game have been implemented, was the student able to improve on them at all?

Elaborations will be graded on

- **Feel**. Has the student created a strong sense of engagement in the player and harnessed established rules-of-thumb to make a game that feels smooth and seamless?
- **Creativity**. Does the project show innovation and uniqueness? Does it show a creative imagination that does not solve the given design problem in an ordinary way?
- **Scope**. Did the student constrain the limits of the project in such a way that it can be considered a ‘finished’ prototype?

Final group projects will be graded on:

- **Polish**. Does the game show attention to detail in eliminating audiovisual flaws, areas of confusion for the player, and a general sense of being ‘finished’.
- **Usability**. Has the student communicated the rules and procedures of the game clearly to the player. Is it painless and enjoyable to begin the game and play it?
- **Functionality**. Is the game bug-free and free of performance issues?
- **Progress**. These final games should build and improve on the gameplay prototype and the audiovisual prototype, to create a game that is better than the preceding versions. (Note: this project is NOT graded on whether or not a game is ‘fun’ to play)

In addition, students are graded on participation in class critique sessions, and given pass/fail grades for updating their development backlog every week, and for participating in the end-of-term game-jam. Finally, if a student maintains a weekly *public* development log, (on Tumblr for example), they can earn up to 5% extra credit.

Assignment Requirements

In addition to the oral presentation of the game to the class, each completed assignment must be comprised of the following, submitted to the appropriate assignment listing on NYU Albert:

- A working executable version of your game, for Windows or Mac.
- A zip file containing your Unity project folder (including your source code & all assets).

Your creative, elaborated projects should also include:

- 3 x screenshots

- A one-page document detailing your game description and play instructions.
- A statement of self-evaluation on the game's merit.

Grade calculation

Students will be given grades based on a 100-point scale. Each assignment will be graded on a point scale, and these points will be added up to determine the final grade, according to the following:

92-100	A
90-91	A-
88-89	B+
82-87	B
etc.	

For the clone projects, generally a game that is a perfect recreation of the original game will be given a B or B+, a game that improves on the original and *also* implements all the original features will receive an A or A-. A game that is missing one or two functions of the original game will get a B- or a C, and if many functions are missing, inaccurate or incomplete, the project will receive a failing grade.

The following are the components of the grade:

Participation	20
Warmup Project	10 (Pass/Fail)
Project 1: Action Game Clone	10
Project 2: Action Game Elaboration	10
Project 3: Narrative Game Clone	10
Project 4: Narrative Game Elaboration	10
Project 5: Grid Game Clone	10
Project 6: Grid Game Elaboration	10
Project 7: Final Group Project	10
TOTAL	**100**

Attendance: Attendance and arriving on time to all class sessions is required and expected, too many unexcused absences will lower your final grade. Three unexcused absences lower your final grade by a letter. Each subsequent unexcused absence will lower another letter grade. Two tardies will count as one absence. Arriving more than 15 minutes late will also count as an absence. If you will be missing a class due to illness, or unavoidable personal circumstances, you must notify your professor in advance via email for the absence to be eligible to be excused.

Late policy: projects submitted after the due date will be graded **zero** unless permission to submit late is sought from the professor in advance.

For your game projects to be graded complete, they **must** be submitted to NYU Classes under the appropriate assignment listing.

Statement Of Academic Integrity:

Plagiarism is presenting someone else's work as though it were your own. More specifically, plagiarism is to present as your own: A sequence of words quoted without quotation marks from another writer or a paraphrased passage from another writer's work or facts, ideas or images composed by someone else.

Importantly however, Intro to Game Development is not a course in programming, so while students are expected to produce original and unique gameplay mechanics, art and sound, they may borrow code liberally from their classmates or from online sources.

Accessibility:

Academic accommodations are available for students with documented disabilities. Please contact the Moses Center for Students with Disabilities at 212 998-4980 for further information.

Wellness:

Your health and safety are a priority at NYU. If you experience any health or mental health issues during this course, we encourage you to utilize the support services of the 24/7 NYU Wellness Exchange 212-443-9999. Also, all students who may require an academic accommodation due to a qualified disability, physical or mental, please register with the Moses Center 212-998-4980. Please let your instructor know if you need help connecting to these resources.

Title IX:

Tisch School of the Arts to dedicated to providing its students with a learning environment that is rigorous, respectful, supportive and nurturing so that they can engage in the free exchange of ideas and commit themselves fully to the study of their discipline. To that end Tisch is committed to enforcing University policies prohibiting all forms of sexual misconduct as well as discrimination on the basis of sex and gender. Detailed information regarding these policies and the resources that are available to students through the Title IX office can be found by using the this link.
https://www.nyu.edu/about/policies-guidelines-compliance/equal-opportunity/title9.html

Schedule

Week 1 – Recap of Unity, Getting Started

A reintroduction to working in Unity and C# and project ideation

Introductions.
A recap of Unity and C#.
Thinking about scope.
Beginning work on modifying the sample warmup project.

Tutorial and in-class exercise: Unity scene organization and workflow. The input manager. The camera. Going over the modification project structure.

Tutorial 2: Developing a backlog. Time boxing and project management.

Reading: Michael Brough, How To Do A Game Jam.
http://mightyvision.blogspot.co.uk/2013/04/how-to-do-game-jam.html

Homework: Modify a simple game.

Week 2 – Overview of Game Feel, Pong Tutorial

Present your modifications to the class.

Develop ability to analyse and talk about games in terms of their feel.

In-class exercise: Game Feel: group analysis of a simple game in terms of feel, discuss principles of game feel.

Tutorial: Pong tutorial - covering the features and functions you'll need to make your clone.

Homework: Choose 1 of 2 action games to clone. Begin your clone project. Install Photoshop.

Week 3 – Basic Sprite Workflow using Photoshop

Understand the basics of using Photoshop to prepare visual assets for Unity, and of the Unity 2D sprite rendering system.

Discussion of the development of a visual style for your gameplay prototype.

Tutorial: Basics of photoshop for Unity's sprite system. Produce a simple sprite and import into Unity.

Homework: Continue your action game clone project.

Week 4 – Playtesting Methods

Presenting first clone projects.

Tutorial: methods for testing games and documenting tests. Setting up a digital game for ease of testing and collection of data.

Homework: Creative elaboration of the clone to turn it into a new, original game of around the same scope.

Reading: Game Design Workshop, Tracey Fullerton - playtesting chapter

Week 5 – Setting Up A Narrative Game

Presentation of elaborated action games.

Tutorial: Scene management, state machines, the singleton pattern, and other skills related to final clone project.

Homework: Begin work on the narrative clone project.

Week 6 – Tuning

Learn methods for setting up your game for ease of tuning.

Tutorial: interactive tunable project: take a broken game and improve it using exposed variables.

Homework: Finish the narrative clones.

Week 7 – Intro to Sound Design

Presentation of clone projects.

Understand the basics of sound design, and how sound is used to convey information and mood in a game. Import and play sounds in Unity.

Tutorial: Basics of sound design. Recording sounds, and adapting them from third-party sources. Understand how sounds are ‘mixed’ by Unity’s audio engine. Trigger a sound from an ‘AudioSource’ component.

In-class exercise: Add a sound effect to your game.

Homework: Creatively elaborate your narrative game.

Week 8 – Setting Up A Grid-based Game

Presentation and critical feedback on the first creative projects.

Tutorial: Covering arrays, functions, and the features of Unity you need to set up your clone project.

Homework: Begin work on the grid-based clone project

Week 9 - Animations and Effects Unity

Understanding the technical and artistic underpinnings of animations, using Photoshop and in Unity. Learn to use Unity's particle system for a variety of effects.

Tutorial: Animations in Photoshop and Unity. Produce a simple 'straight ahead' animation in Photoshop and a keyframe animation in Unity

Tutorial: Particle systems.

Video: "Juice it or Lose It"

Homework: Finish work on the grid-based clone project

Week 10 - Teaching the player

Present finished clone projects

Tutorial: analysis of the ways information about game state and systems are conveyed to the player in a game, using layout, structured information and juicy feedback.

Discuss methods for making games more usable, focusing on the features that enable a player to understand how to play the game, and how to understand the game's state. Discussion of tutorials and information conveyance in games. Understand some approaches to teaching players how to play a game. Discuss pros and cons of each method.

Homework: Elaborate the grid-based game clones creatively
Get a Github account, install Git and Sublime Merge.

Week 11 – Source control

Present the elaborations of the final clone projects.

In Class: Form into groups and choose a project to work on.

Tutorial: Asset control: create a repository, make local commits, push to the remote and revert changes. Set up a Unity project for use with asset control software.

Homework: Begin work on the group project.

Week 12 — Working on final projects

Present progress on final project

Tutorial: Improving a pre-built game project solely through the modification of tunable variables, comparison in class.

Homework: Finish game jam games for final presentations.

Week 13 – Beta test

In-class beta test of final projects.

In-class exercise: Problem-solving exercise for addressing last-minute issues

Homework: Bring final game project to class for presentation, in a compiled 'player build' format.

Week 14 – Final presentations

Present final group games games in class. Critique session.