

Welcome to Game Design and Development

This Today's Agenda

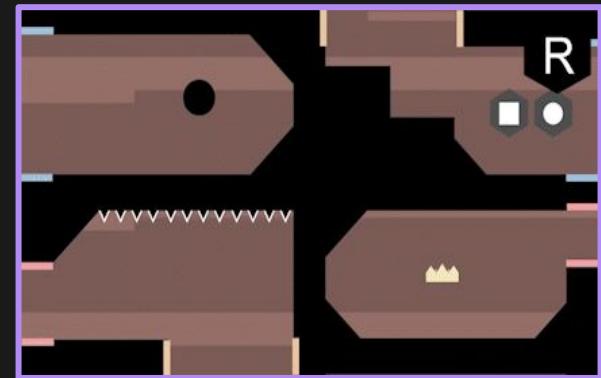
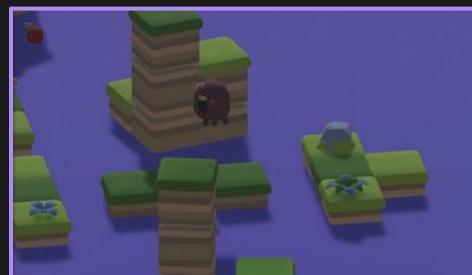
- Introductions.
- Go over the curriculum and grading for the semester.
- Discuss the Video Game Industry and roles of Game Developers.

Introductions

Who are we?

My name is Sebastian Grygorczuk, and I'm currently working on a game called Claw Arena with Mocha Chili. This is my third year teaching this class.

Outside of gaming, my interests include robotics, hiking, and history.



Instruction for the kids

1. Tell us your name,
2. Your favorite game,
3. Rank or swap crewmates in *Among Us*.

S	
A	
B	
C	
D	



Class Content

What Is The Goal Of This Class?

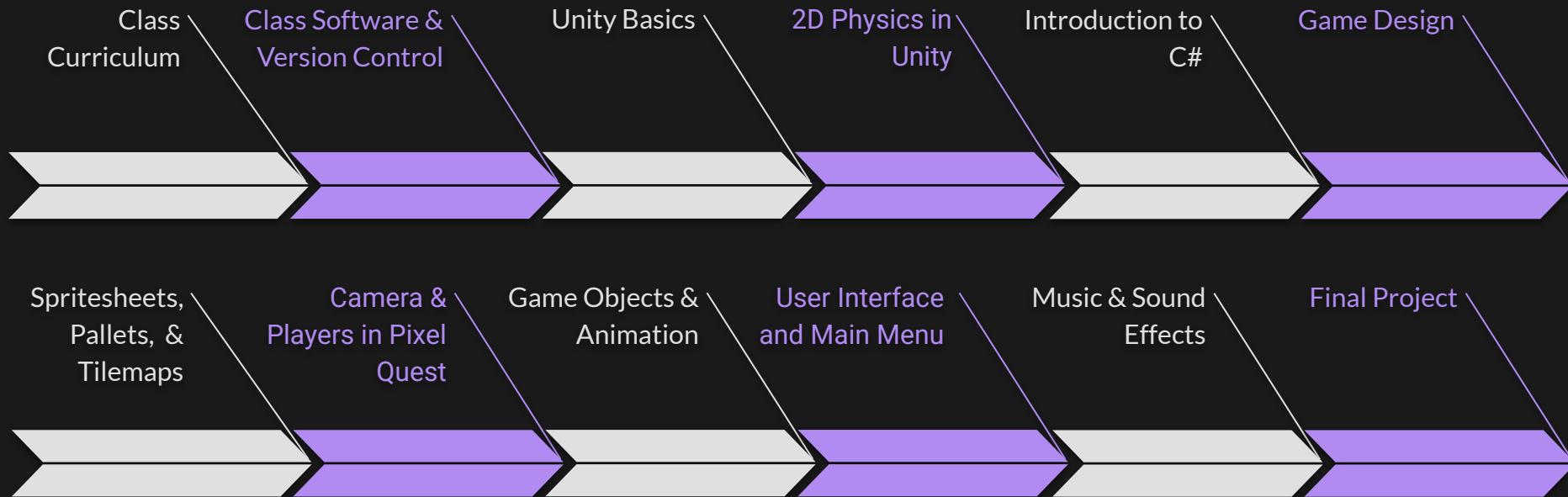
The purpose of this class is threefold:

1. To give you the opportunity to create a game and understand the basics of the game industry.
2. To help you develop critical thinking skills to solve problems.
3. To build friendships and have fun.

This is an extracurricular class, and you should be rewarded for going the extra mile to learn and achieve.



Class Road Map



Class Road Map

Class Curriculum

We'll discuss class expectations, homework, and the final project. We'll also explore how the skills you gain can lead to careers in the video game industry.

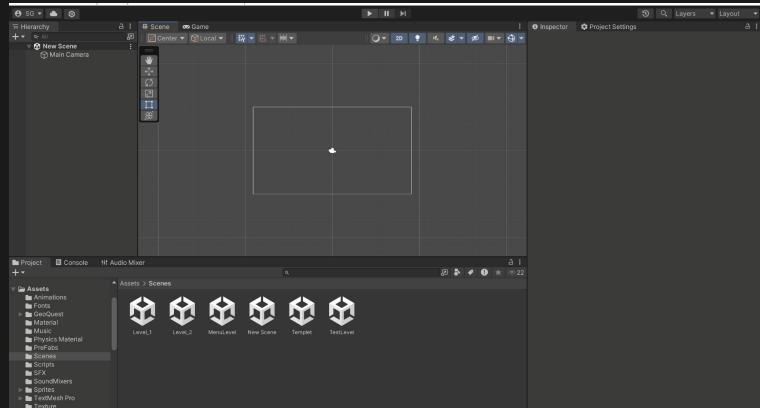
Class Software & Version Control

Setting up Unity, Visual Studio, and GitHub.



Unity Basics

Navigating the Unity Editor and implementing game elements.



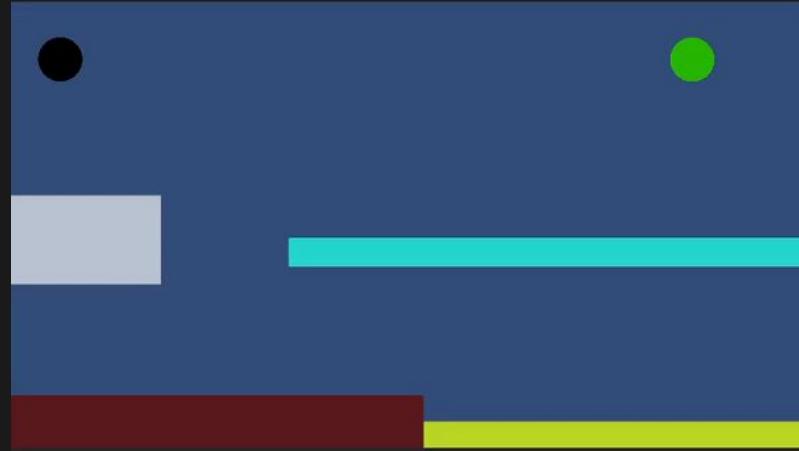
Class Road Map

Unity 2D Physics

Understanding Unity's 2D physics, including Colliders, Rigidbodies, and Effectors.

Introduction to C#

The basics of C# scripting, such as classes, variables, and functions.



```
public class GeoController : MonoBehaviour
{
    private Rigidbody2D _rigidbody2D;
    public int speed = 4;
    public string nextLevel = "GeoLevel_2";

    // Start is called before the first frame update
    void Start()
    {
        _rigidbody2D= GetComponent<Rigidbody2D>();
    }

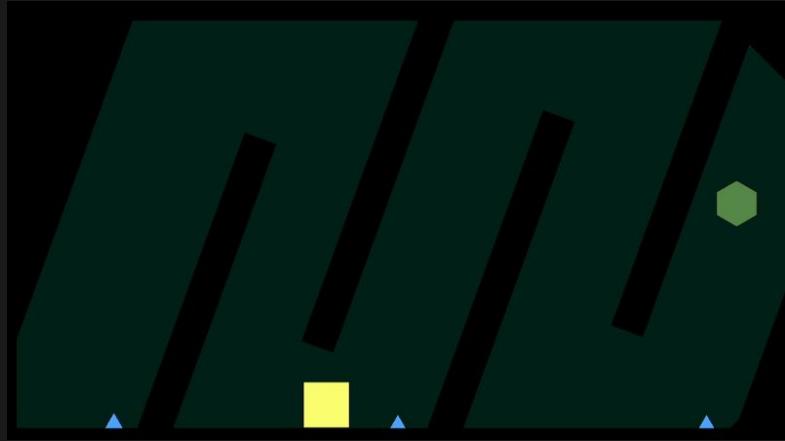
    // Update is called once per frame
    void Update()
    {
        float horizontal = Input.GetAxis("Horizontal");

        _rigidbody2D.velocity = new Vector2 (horizontal * speed, _rigidbody2D.velocity.y);
    }
}
```

Class Road Map

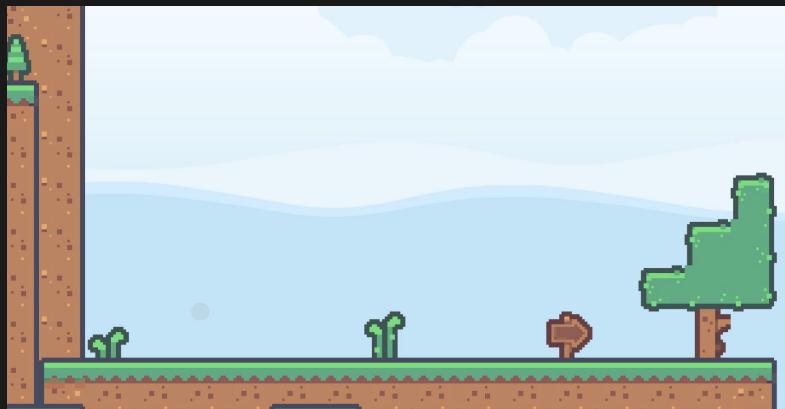
Game Design

Designing our first game, *Geo Quest*, using basic player controls and 2D physics objects.



Spritesheets, Pallets, & Tilemaps

Improving our game's presentation by breaking down spritesheets and using the Tilemap system for large levels.



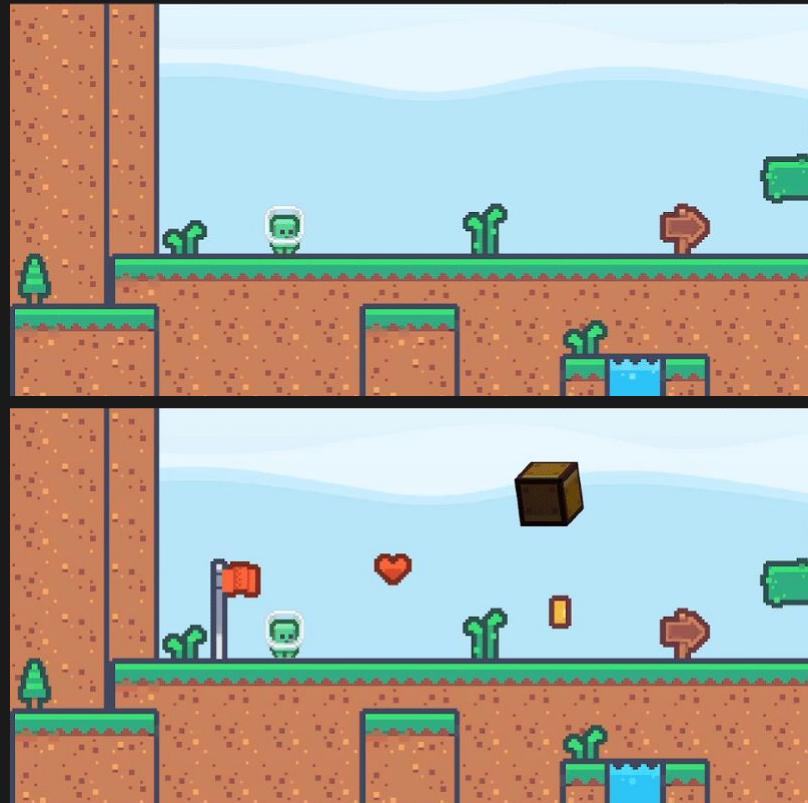
Class Road Map

Start Pixel Quest:
Player & Camera

Enhancing the player controller with features like jumping and camera tracking.

Game Objects &
Animation

Adding interactive game objects like coins, hearts, checkpoints, and level transitions, then animating them.



Class Road Map

User Interface and Main Menu

Building a user interface (UI) to display the player's stats and a main menu.

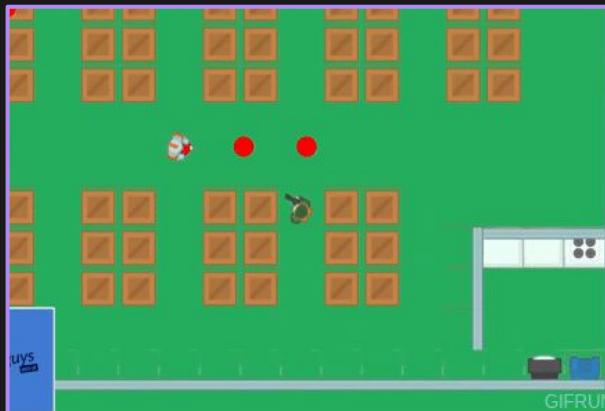


Music and Sound Effects

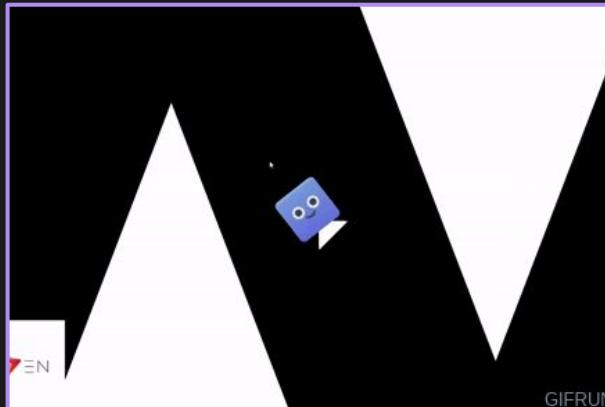
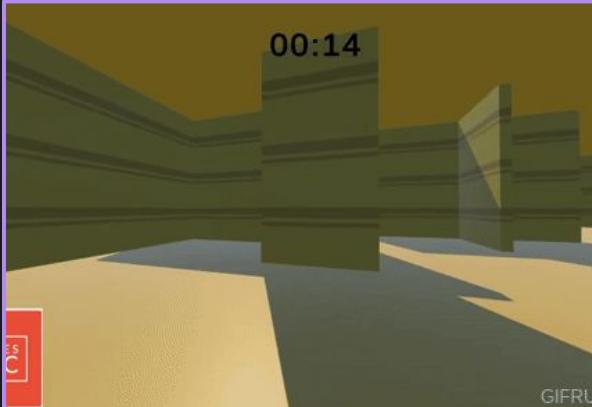
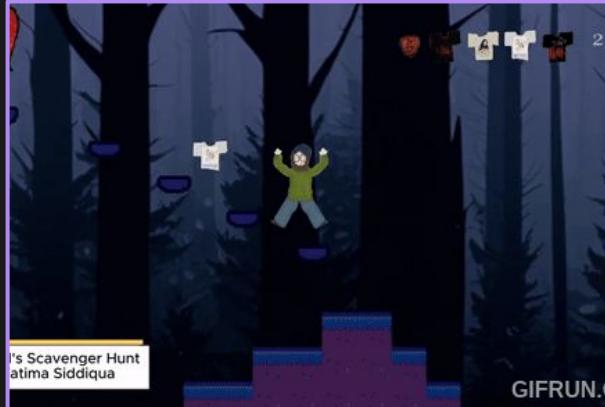
Adding music and sound effects to enhance the game's atmosphere and provide feedback.



Previous Classes Projects



Previous Classes Projects



Grading

Class Resources



No Textbook Required:
All course materials are provided through slides. Additionally, a video series covering the same topics is available for those who want to get ahead.

Attendance

Attendance

Attendance is crucial to stay up to date with the class. We'll take attendance at the start of each session, so please notify the STEM office if you have an unexpected absence.

Absences

You are allowed up to X unexcused absences. The Xth unexcused absence will result in withdrawal from the course.

You can have up to X excused absences, but they must be documented within a week of the absence.

Exceeding the total allowed absences, whether excused or unexcused, may result in withdrawal.

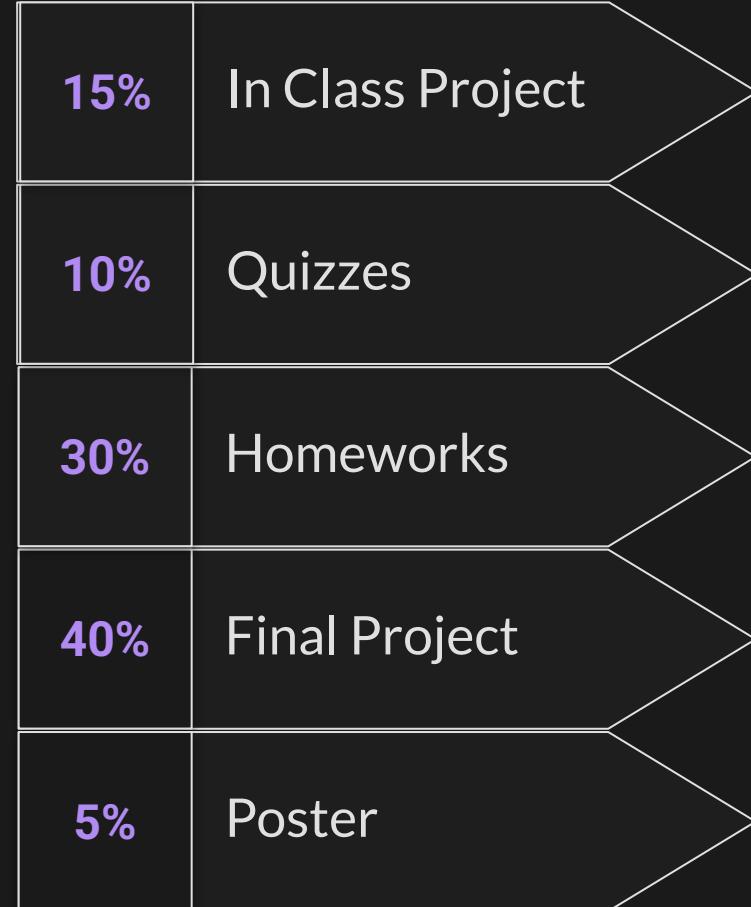
Withdrawal

The withdrawal deadline is *Month Day*. If you decide to withdraw, visit DreamClass to submit the withdrawal form. It will appear on your transcript as "Not Complete."

Grading

Grading System: STEM Institute uses a Pass/Fail grading system. Your grade will depend on homework and project completion:

- **Fail:** Below 65%
- **Pass:** 65% - 89%
- **Numerical:** 90% or above If you prefer not to receive a numerical grade, let me know.



Grading College Now

College Now Option: Eligible

10th-12th grade students can earn both high school and college credit through College Now.

You should have already signed up during registration but if you'd like to participate, please contact the admin team after class hours.

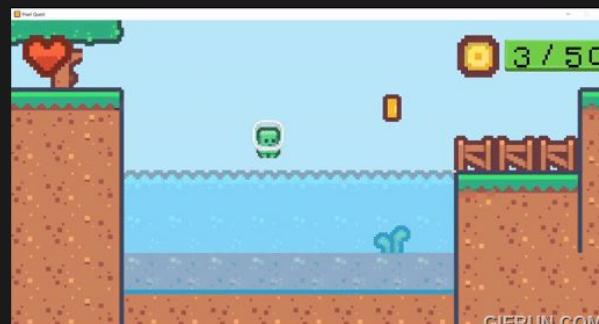
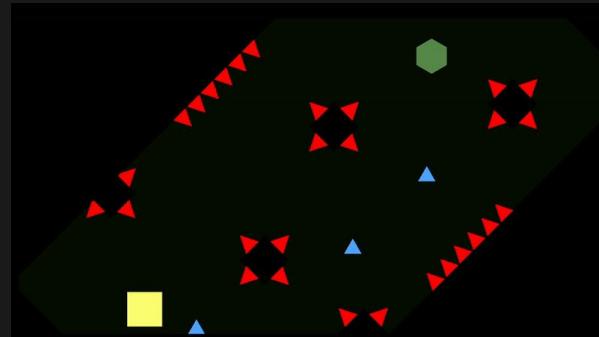
97-100	A+
93-96	A
90-92	A-
87-89	B+
83-86	B
80-82	B-
77-79	C+
73-76	C
70-72	C-
67-69	D+
60-66	D
0-59	F

In Class Project

This class is centered on creating 2D platformer games. You'll submit a video of your game for both *Geo Quest* and *Pixel Quest*, which will be worth 15% of your grade (7.5% for each).

15%

In Class Project



Quizzes

Quizzes: After completing each project section, there will be quizzes to assess your understanding. Quizzes will cover:

1. Class software and version control
2. Unity basics
3. Unity 2D physics
4. Introduction to C#
5. Game design
6. Spritesheets, Palettes, & Tilemaps
7. Package manager and cameras
8. Animation
9. UI
10. Music and sound effects

10%

Quizzes

Each quiz will include:

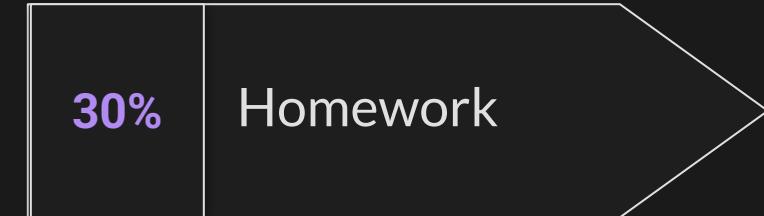
- Multiple-choice questions
- A "freebie" question at the end
- A 40-minute duration with a review afterward
- A cheat sheet (small index card) is allowed

Homeworks

Homework Assignments: You'll complete three homework assignments, each exploring different game development styles.

Each assignment includes a video tutorial with clear instructions.

Following the tutorial accounts for **70%** of your homework grade, while the remaining **30%** will be earned by you solving additional challenges presented in the homeworks.



- **Unity Access Required:** You'll need Unity to complete the homework. If you don't have access at home, school computers will be available during office hours which are XXX.
- Feel free to contact me via email (sgrygorczuk@gmail.com) or DreamClass messaging if you need assistance.

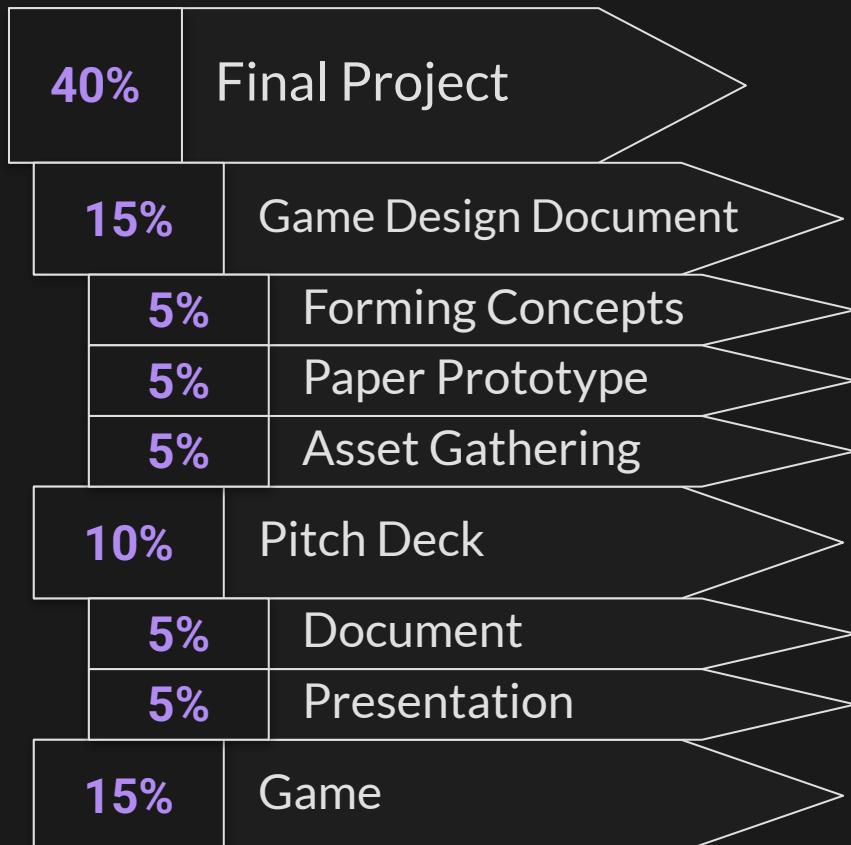
Final Project

Final projects will involve applying all the skills you've learned throughout the course to create your own game. You'll work in groups of up to 2 people, as game development is highly collaborative.

Once we finish the regular homework, you will have 3 new assignments to start building your game design document. A Game Design Document (GDD) outlines what the game is and how it works, and it will be the foundation of your project.

After we complete the in-class project, we'll work on a pitch deck, which is a presentation of your game idea—just like how studios pitch for funding.

Finally, you'll create the game itself, applying your concept and executing it to the best of your abilities.



Posters

End-of-Semester Posters:
Each group will create a poster to showcase your game. Each group will get a third of a poster board to highlight your game's features and present your idea.



Middle School Fall 2024 Game Development
Sebastian Grygorczuk, Steve Guan, Benito Ladrido

Pixel Quest
Made By Team Stem
Jane Doe & John Doe

Pixel Quest is a 2D platforming game designed for a younger audience, serving as an entry-level experience that introduces players to the gameplay loop while offering small yet enjoyable challenges. Set in a vibrant and tranquil Digital Forest, players follow Pixel the Adventurer as he navigates through robotic hordes of enemies, collecting coins and protecting his home.

For this game, we developed a complex player controller that allows users to walk, jump, swim, collect coins and hearts, and take damage. Additionally, players can transition between different levels, offering a progressively challenging experience. We designed multiple levels to ensure a ramped-up sense of difficulty, making the gameplay engaging and rewarding for players.

CELESTIAL BREAKER
Made By Team Stem
Jane Doe & John Doe

Celestial Breaker is an action-packed game where you play as an astronaut tasked with saving the world by breaking apart incoming celestial bodies threatening to destroy Earth!

For this game, we created custom scripts that allowed the player to move, shoot bullets, and respond to collisions with incoming asteroids. We used Visual Studio and programmed the game in C# to bring these mechanics to life.

REALM CHATTER
Made By Team Stem
Jane Doe & John Doe

Realm Chatter is a narrative adventure game that follows the journey of a noble knight on a quest to save their kingdom. Join the knight in their epic mission to recover the legendary Apple of York!

For this project, we created a world using tilemaps. To achieve this, we first cut up a sprite sheet into individual images, which we then used to draw the world map. We designed multiple tilemaps, with some serving as background elements and others functioning as physical obstacles to prevent the player from moving freely across certain areas.

DreamClass

Dreamclass

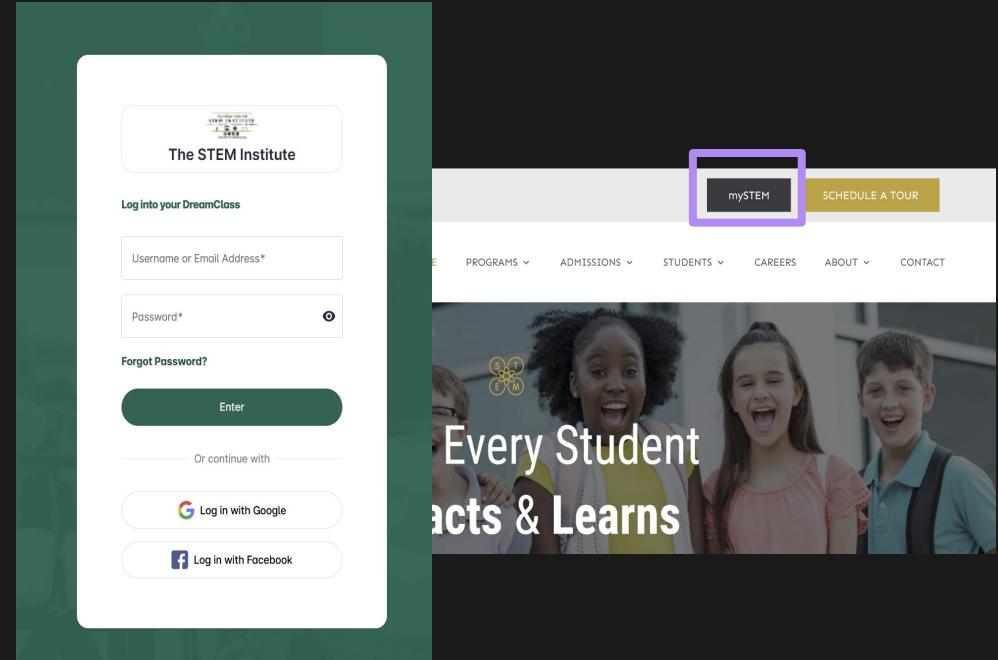
Dreamclass is where all the class resources will be stored. You can access it by either going to

<https://steminstituteny.org> and clicking on "mySTEM," or you can go directly to

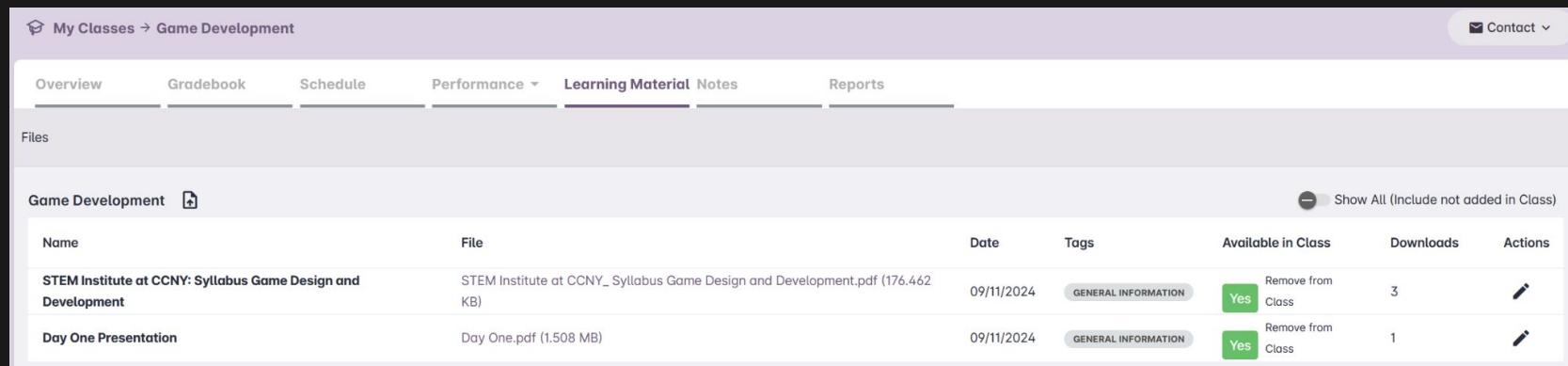
<https://stem-institute.dreamclass.io>

From there, you'll be able to view your grades, homework, and class resources.

If anyone doesn't have access to Dreamclass, please let us know now!



Learning Material & Assignments



My Classes → Game Development

Overview Gradebook Schedule Performance Learning Material Notes Reports

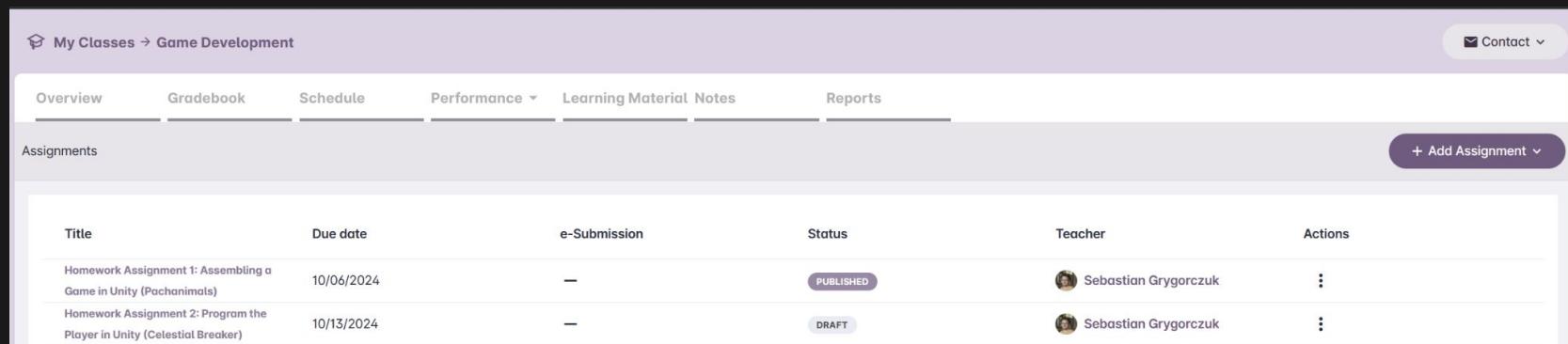
Files

Game Development 

Show All (Include not added in Class)

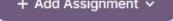
Name	File	Date	Tags	Available in Class	Downloads	Actions
STEM Institute at CCNY: Syllabus Game Design and Development	STEM Institute at CCNY_Syllabus Game Design and Development.pdf (176.462 KB)	09/11/2024	GENERAL INFORMATION	 Remove from Class	3	
Day One Presentation	Day One.pdf (1.508 MB)	09/11/2024	GENERAL INFORMATION	 Remove from Class	1	

In **Learning Materials**, you'll find the syllabus, this presentation, and any other resources needed for the class.



My Classes → Game Development

Overview Gradebook Schedule Performance Learning Material Notes Reports

Assignments 

Title	Due date	e-Submission	Status	Teacher	Actions
Homework Assignment 1: Assembling a Game in Unity (Pachanimals)	10/06/2024	—	PUBLISHED	 Sebastian Grygorczuk	
Homework Assignment 2: Program the Player in Unity (Celestial Breaker)	10/13/2024	—	DRAFT	 Sebastian Grygorczuk	

Under Content→Assignments, you'll find your homework.

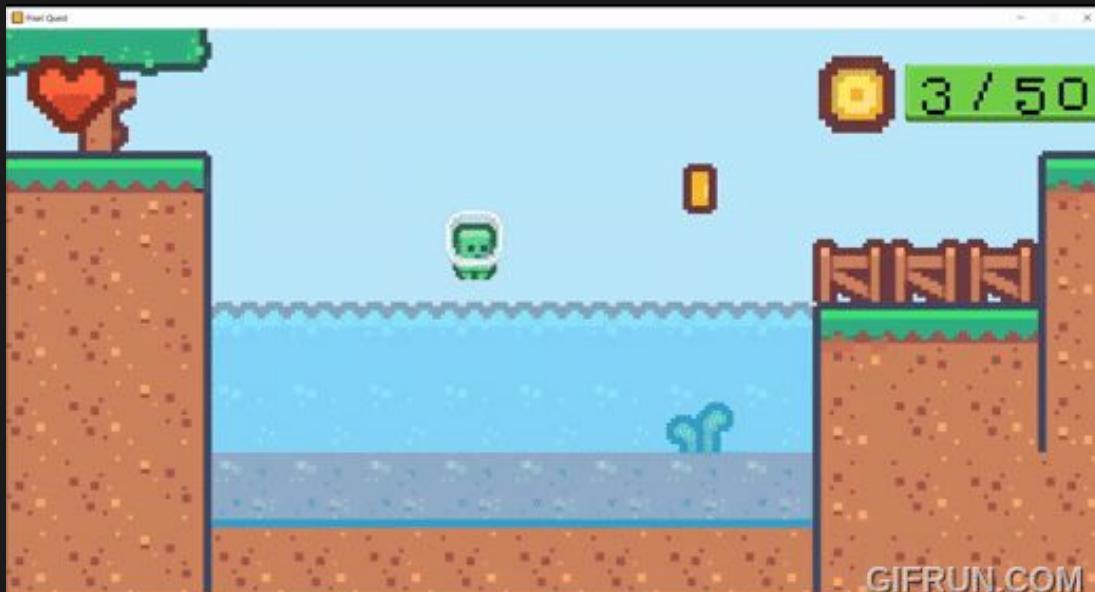
Game Projects

Geo Quest - In Class Project



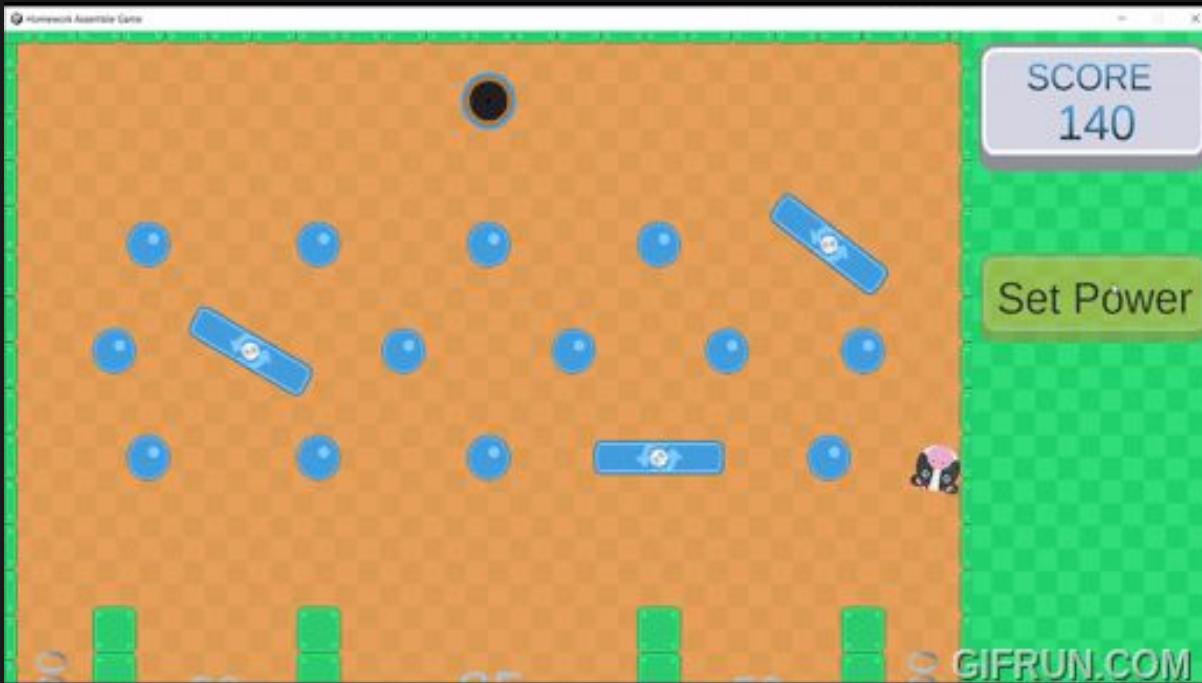
Geo Quest will be our first step into game development. We'll create a series of small, single-screen levels using basic 2D physics. We'll also implement a simple C# script for functionality.

Pixel Quest - In Class Project



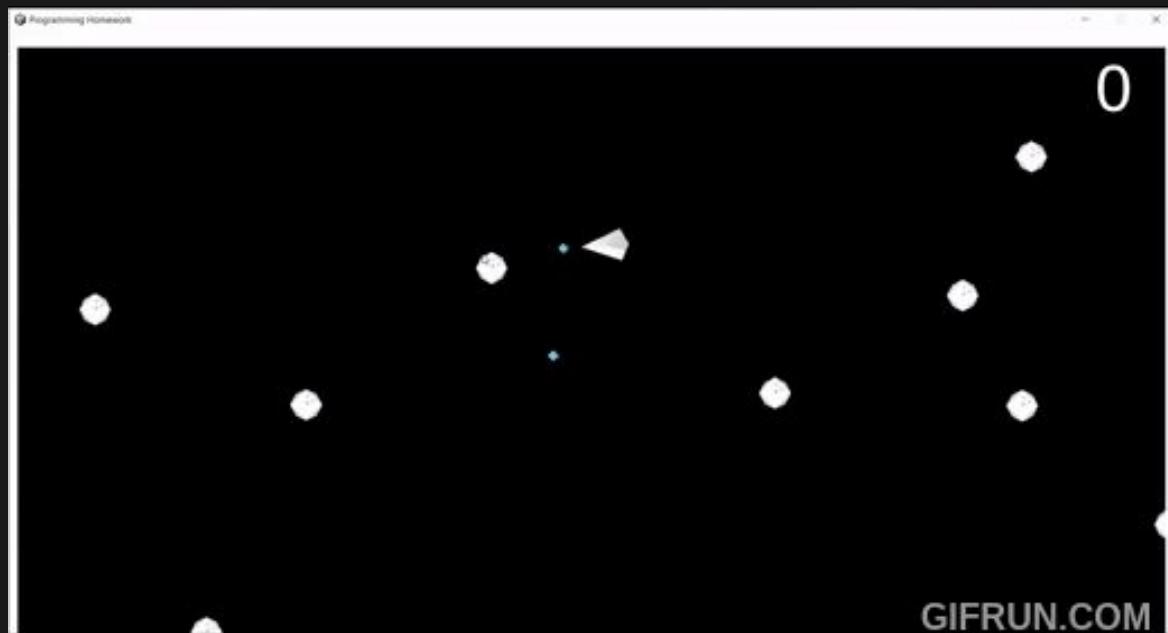
Pixel Quest will build on what we've learned, improving the presentation and interactivity. We'll create enhanced tilemaps for better visuals, add collectible objects to the game, implement a user interface to show player stats, and include music and sound effects to provide feedback on player actions.

Pachanimals - Homework #1



The first homework will test your ability to modify a premade environment in the Unity Editor. You'll be required to move and clone objects in the scene and change values in the components.

Celestial Breaker - Homework #2



The second homework will focus on coding practice. You'll follow an example to implement top-down movement and game object spawning.

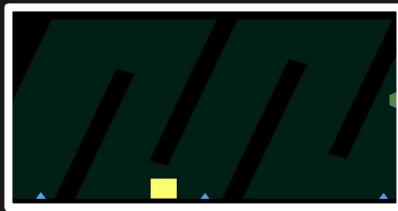
Realm Chatter - Homework #3



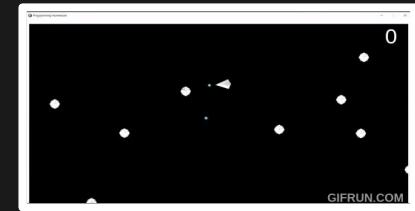
The third homework will focus on creating a world for the player to explore. You'll use the provided tile palette to design a level, or you can import your own tiles if you prefer. You'll also design NPCs with different dialogue options to make the world more immersive for the player.

Try Out The Projects

Please take about 20 minutes to try out the games you'll be making during this semester.



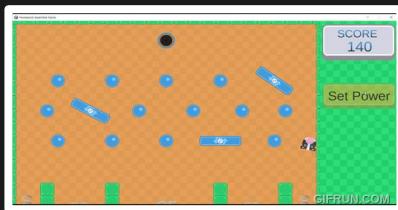
[Link](#)



[Link](#)



[Link](#)



[Link](#)



[Link](#)

Games Industry

Video Game Industry

The games industry is huge, generating nearly \$200 billion a year across all platforms. It makes more money than movies and music combined.

There are hundreds of studios and teams actively making new games for you to play.

With this growth, there are also hundreds of different roles you can fulfill at one of these studios.

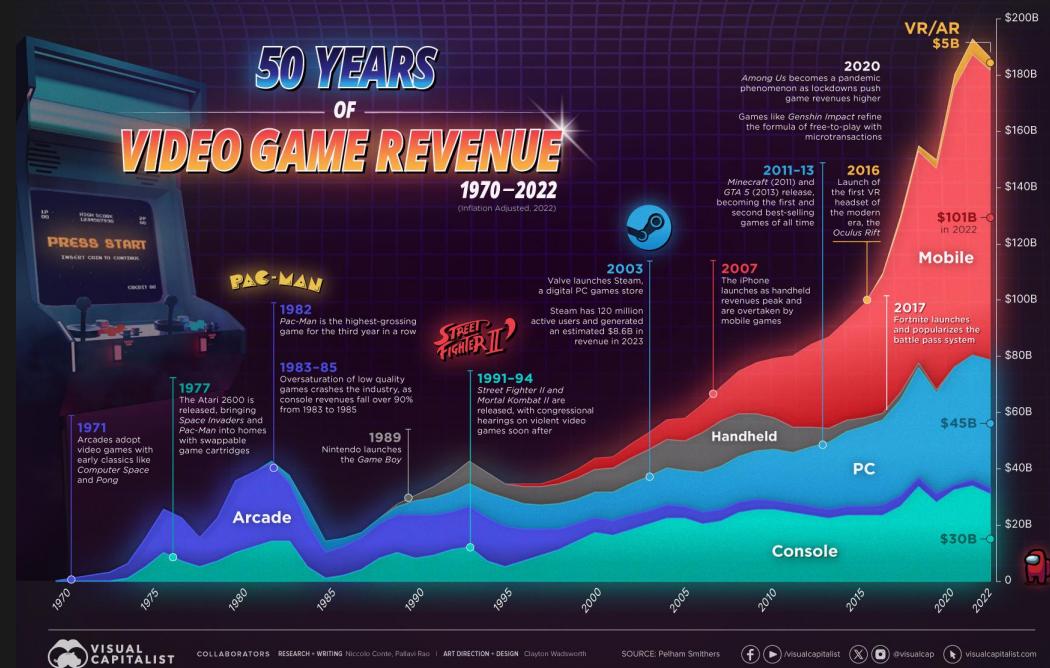


Chart from [Visual Capitalist](#)

Stages of Game Development

Video games go through many different phases, from the initial idea to what you play on your computer. Many games don't even make it past some of the stages and are never heard of. The chart shows all the stages a game can go through, but we will focus on the first four.

1. **Planning:** We'll plan our final project game by creating a Game Design Document and answering key questions about the game.
2. **Pre-production:** We'll gather the assets we need and create a pitch deck with ideas.
3. **Production:** We'll create game objects and systems, similar to what we did for Pixel Quest.
4. **Playtesting:** We'll test our games to fix bugs, improve unfun sections, and enhance the overall experience.



Additional Resource: [The 7 Stages of Game Development](#)



What is a Game Developer?

Game development is not just about programming. Since games consist of many assets like code, art, animations, and sound, a game developer is responsible for creating these assets. This includes roles like Game Designers, Programmers, 2D and 3D Artists, Animators, Sound Engineers, and Testers.

Throughout this class, you'll take on each of these roles. Depending on the size of the team, each person might focus on only one role or work on many.



Additional Resource: [The Big List of: Video Game Development Team](#)

Game Designer

Game design involves many specialized roles, including:

- **Game Designer (General):** Develops creative vision, gameplay mechanics, and integrates game elements.
- **Level Designer:** Designs levels and creates challenges based on the story.
- **Systems Designer:** Balances game mechanics, player progression, and core systems.
- **UX/UI Designer:** Designs the user interface (UI) and experience (UX) to ensure intuitive navigation and accessibility.
- **Gameplay Designer:** Refines gameplay mechanics and improves the player experience.
- **Narrative Designer:** Integrates the story, dialogue, and choices that engage players.
- **Combat Designer:** Balances combat systems and enemy behavior.
- **AI Designer:** Develops artificial intelligence for NPCs (non-playable characters).
- **Economy/Balance Designer:** Designs in-game economies and progression systems.
- **Quest/Content Designer:** Creates quests and narrative-driven challenges.

Examples of Game Designers

Shigeru Miyamoto is a legendary game designer, creator of iconic franchises like Zelda, Donkey Kong, and Super Mario Bros. He currently serves as the General Manager of Nintendo.



Hidetaka Miyazaki is a renowned game director and creator of the challenging Dark Souls series, Bloodborne, and Elden Ring. He is the President and Director of FromSoftware.



Will Wright is a pioneering game designer behind simulation games like SimCity and The Sims. He is the co-founder and Chief Design Officer of Gallium Studios.



Todd Howard is a celebrated game director and producer known for his work on Skyrim, Fallout 3, and Starfield. He is the Director and Executive Producer at Bethesda Game Studios.



Game Programmer

Game programmers are essential for making games functional and interactive.

- **Gameplay Programmer:** Implements game mechanics and player controls.
- **Backend/Server/Online Programmer:** Manages networking and online features.
- **AI Programmer:** Designs NPC behavior and decision-making.
- **Graphics Programmer:** Implements rendering techniques and visual effects.
- **UI Programmer:** Develops user interfaces and interactive elements.
- **Animation Programmer:** Implements animation systems for characters and objects.
- **Physics Programmer:** Handles collision detection and physics simulations.
- **VFX Programmer:** Creates visual effects like particles and lighting.
- **Audio Programmer:** Implements sound systems, including music and sound effects.
- **Tools Programmer:** Develops tools and editors for game development.

Examples of Game Programmers

Markus “Notch” Persson is the creator of Minecraft and founder of Mojang. He is now an independent game developer after selling Mojang to Microsoft in 2014.

John Carmack is a groundbreaking programmer known for Wolfenstein 3D, DOOM, and Quake. He currently serves as the Chief Technology Officer of Oculus VR.

Tim Sweeney is the creator of Unreal Engine and CEO of Epic Games, overseeing Unreal Engine development and Fortnite.

Ed Boon is the co-creator of Mortal Kombat and serves as the creative director at NetherRealm Studios, the studio behind Mortal Kombat and Injustice



Game Artist

Game artists bring games to life by creating visual assets. Some of the key roles are:

- **Concept Artist:** Designs characters, environments, and game assets.
- **Character Artist:** Creates 3D models and textures for characters.
- **Environment Artist:** Designs 3D environments and landscapes.
- **Animator:** Develops animations for characters and objects.
- **UI/UX Artist:** Designs the user interface and experience.
- **Texture Artist:** Creates textures for 3D models.
- **VFX Artist:** Creates special effects like explosions and magic spells.
- **Lighting Artist:** Sets up lighting and shading for atmosphere.
- **Technical Artist:** Bridges art and technology, optimizing workflows.

Examples of Game Artists

Tetsuya Nomura is an artist and game director known for his work on Final Fantasy and Kingdom Hearts. He is currently a Game Director at Square Enix.

Keijiro Inoue is a Visual Effects Artist at Nintendo, contributing to titles like Pikmin, Super Mario Odyssey, and Breath of the Wild.

Josh Scherr is a cinematic director and writer at Naughty Dog, known for his work on Uncharted and The Last of Us, shaping iconic storytelling and action.

Jamie McNulty is known for his work on BioShock, BioShock Infinite, and Gears of War 4 and 5. He currently applies his expertise as an Environment Artist at Deviation Games.



Sound Designers

Sound designers are crucial for creating the game's audio experience. Their responsibilities include:

- **Sound Effects:** Creating and editing sounds for footsteps, weapons, and the environment.
- **Music Composition:** Composing or collaborating on music that fits the game's atmosphere.
- **Voiceover Production:** Casting voice actors and editing recordings.
- **Technical Integration:** Using tools like FMOD or Wwise to implement sound into the game.
- **Quality Assurance:** Testing the audio across platforms and incorporating feedback.

Examples of Sound Designers

Jason Hayes is a veteran composer known for his work on Starcraft, Warcraft, and Diablo. His compositions have defined Blizzard's iconic worlds.

Toby Fox is a composer and developer, best known for his music in Undertale and Deltarune, which defined the games' emotional impact.

Kōji Kondō is a legendary composer known for his iconic scores in Zelda and Mario, enhancing gameplay with unforgettable melodies.

Kenneth C. M. Young is a composer and sound designer praised for his work on the Astro Bot series, enhancing their whimsical and immersive experiences.



Writers

Writers are responsible for the narrative, dialogue, and storytelling in games. Key writing roles include:

- **Game Writer/Lead Writer:** Develops the story and characters.
- **Narrative Designer:** Integrates the story with gameplay mechanics.
- **Dialogue Writer:** Writes character dialogues.
- **Quest/Content Designer:** Designs quests and narrative-driven challenges.
- **World/Setting Writer:** Expands the game's lore and setting.
- **Localization Writer:** Translates the game text into multiple languages.
- **Technical Writer:** Writes manuals, guides, and tutorials.
- **Editor:** Reviews and revises content for clarity.

Examples of Writers

Amy Hennig is a renowned writer and director, known for her work on the Uncharted series, shaping cinematic storytelling in action-adventure games.

Chris Avellone is a celebrated writer for RPGs like Planescape: Torment and Fallout: New Vegas, renowned for his mastery of philosophical themes and non-linear storytelling.

George R. R. Martin is a famed fantasy author who contributed to the lore and world-building of Elden Ring, a dark fantasy RPG developed by FromSoftware.

Chris Metzen is a game designer and writer who contributed to Blizzard's Warcraft, Diablo, and StarCraft universes. He retired as Blizzard's Senior VP of Story and Franchise Development in 2016.



Quality Assurance (QA)

QA testers ensure that games are technically sound and provide a fun, bug-free experience. QA roles include:

- **QA Tester:** Identifies bugs and gameplay issues.
- **Automation Tester:** Develops automated testing scripts.
- **Localization Tester:** Ensures accurate translation.
- **Compatibility Tester:** Tests the game on different hardware.
- **Multiplayer Tester:** Evaluates online gameplay features.
- **Usability Tester:** Tests the game's user-friendliness.
- **Regression Tester:** Verifies bug fixes.
- **Beta Tester:** Provides feedback during pre-release phases.

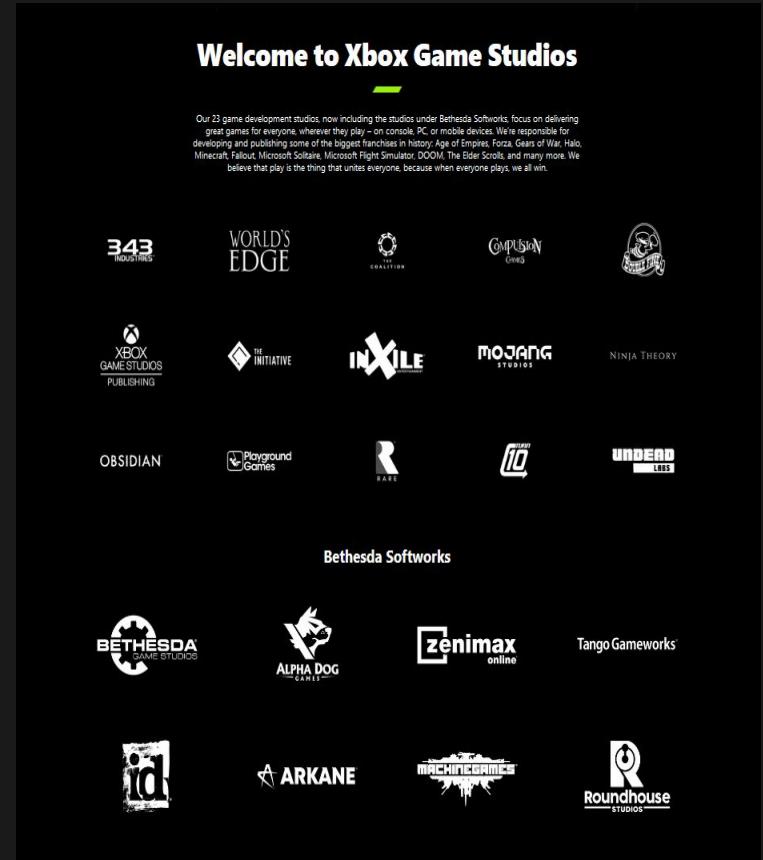
Publisher and Game Development Studio

Game Development Studio:

- **Role:** Develops the game, responsible for design, programming, art, sound, and testing.
- **Responsibilities:** Designing gameplay, writing code, creating art, animating, designing sound, testing, and refining based on feedback.

Publisher:

- **Role:** Manages the business and distribution, often funding and promoting the game.
- **Responsibilities:** Providing funding, marketing the game, handling distribution, negotiating contracts, managing legal issues, and working with external partners.



Indie Developers

Let's give honorable mentions to some inspiring Indie Game Developers that take all the talents of a Game Development Studio and perform them all by themselves.

Toby Fox, is a breakout star as a Game Designer, Programmer, Artist and Sound Design creating *Undertale* as a one man team.

Supergiant Games is a small team that has created hits such as *Bastion*, *Transistor* and *Hades*. Known for beautiful art, music and innovative approaches to storytelling.

Scott Cawthon is the creator of the massive franchise of *Five Nights at Freddy's* which is now a multimedia empire.

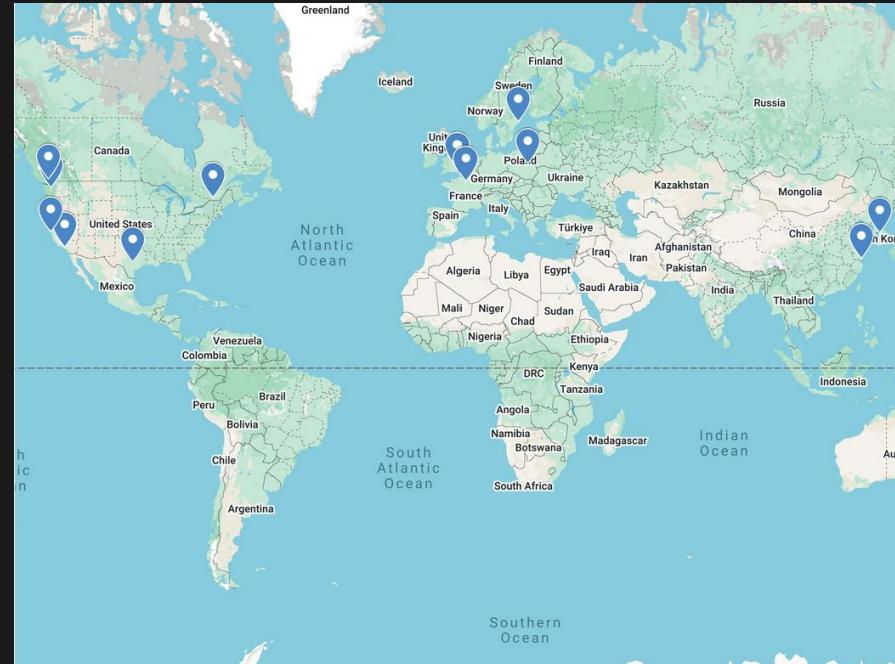
If you'd like to look into what Indie Game Development is like you can watch Developing Hell, a documentary of how Supergiant Games created their latest game *Hades*.



Where Game Studios Exist

Game Development Studios are abound in many places

1. Los Angeles, USA: Riot, Naughty Dog, Santa Monica Studio, and Activision Blizzard.
2. San Francisco, USA: Unity Technologies, Niantic, and Zynga.
3. Tokyo, Japan: Nintendo, Sony Interactive Entertainment, Capcom, Sega, and Square Enix.
4. Seoul, Korea: Nexon, NCsoft, and Netmarbles
5. Paris, France: Ubisoft, Quantic Dream, Dontnod
6. London, UK: Rocksteady, King, and Creative Assembly.
7. Montreal, Canada: Ubisoft Montreal, WB Games Montreal, Eidos Montreal, and GameLoft.
8. Seattle, USA: Microsoft Game Studios (Xbox), and Valve Corporation.
9. Vancouver, Canada: EA, and Relic Entertainment.
10. Stockholm, Sweden: Mojang, Paradox Interactive, and DICE.
11. Austin, USA: Retro Studios and Aspyr Media.
12. Warsaw, Poland: CD Projekt.
13. Shanghai, China: Tencent Games, NetEase Games and Perfect World Entertainment.
14. Helsinki, Finland: Supercell, Remedy Entertainment and Rovio Entertainment.



Future Hub: New York City

Currently, New York City has limited opportunities in game development, with only a few smaller studios. However, this is changing.

Many new programs are emerging to teach game development. Once you master the skills from this class, you'll be ready to create your own games and even start your own studio.

As the number of talented developers in NYC grows, it will attract more funding and create even more opportunities.

You could be part of the next generation of developers who put NYC on the game development map.



Why Unity?

Perks of Unity



- Port it to any platform; Unity allows you to create executable files for your game to pretty much any platform that's currently available. This includes: Windows, Mac, Linux, Web Browser, Android, iOS, AR, VR, Playstation, Xbox, and Switch consoles.
- Is free software for personal use as long as your projects don't exceed \$200,000 revenue in a year.
- Is connected to the Unity Asset Store which host a treasure trove of free and purchasable assets.

Games Created With Unity

Unity is a multi-tool that allows whoever wields it to create anything:

Of course primarily we think of games when it comes to Unity and there are plenty of great examples such as "Cuphead", "Hearthstone", "Rust", "City Skylines", and many more.

All of these games cover wildly different styles of gameplay and complexity.



If you're interested in checking the comprehensive list of [games](#) made with Unity.

Animation Created With Unity

Unity also has an incredibly powerful animation system. It can be used to create full animated shows and movies.

For example, *Mr. Carton* (created in 2017 by Michaël Bolufer) was made with Unity.

Additionally, companies like Disney use Unity to film in VR environments, helping create movies like *The Lion King* (2019).



If you're interested in checking out some other animations made with Unity. "[Lion King](#)" Article.

Animation Created With Unity

"Enemies" is the latest project by Unity's award-winning Demo Team. It showcases Unity's capabilities in powering high-end visuals in 2022.



[Making of Enemies](#)

Non Game Related Software With Unity

Unity's versatility extends to other industries as well. Its main uses outside of gaming include visualization and simulations. Engineers, architects, and medical professionals use Unity to see what they're constructing or working on, while simulating the physics of their prototypes.

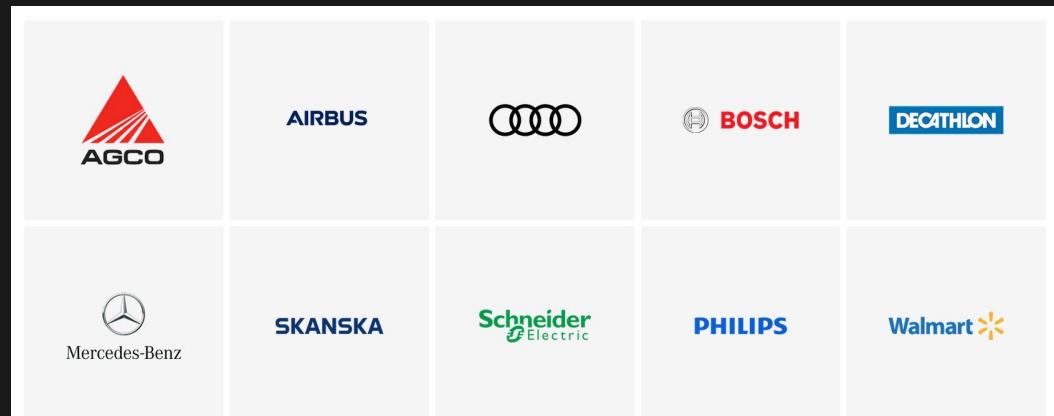


If you're interested in checking out software made for [automobiles](#), [engineering](#) or [aerospace](#) using Unity.

Non Game Related Software With Unity

Here's some uses and companies that use Unity in their productions.

Training and guidance	Increase employee knowledge retention and improve task preparation by delivering personalized and immersive training experiences.
3D design collaboration	Experience designs in a shared visual space to improve collaboration and iteration speed.
Customer experience	Engage buyers with detailed virtual assets, interactive product configurators, and customized shopping experiences.
XR	Deliver interactive experiences that drive sales, provide valuable training, and solve your toughest visualization challenges.
Human-machine interface (HMI)	Seamlessly connect HMI development processes, from design to deployment.



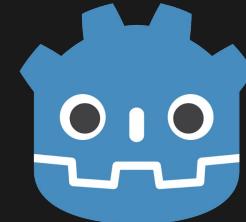
If you're interested check out the [industry uses](#)

Other Game Engines

Unreal: Industry Standard uses 2D/3D game engine using C++ is free to use. Witcher 4 is being developed in it.

GoDot: Develing into creating 2D and 3D Environments and Levels, allows you to Programming with C# and C++ and Visual Scripting. Free to Use.

There dozens of game engines all different from each other however, many use similar standardized tools so that once you are familiar in one you can transfer your skills to another easily.



Additional Resources: [Most used Engines](#) [The Best Game Engines of 2021](#)