Digital Interactive Experience

Lecture 1 & 2

Yuhan Cao

ShanghaiTech University

December 13, 2024

What is a Game?

Snake is a game;

What is a Game?

Snake is a game;

Angry Bird is a game;

What is a Game?

Snake is a game;

Angry Bird is a game;

Plants VS Zombies is a game;

What is a Game?

Snake is a game;

Angry Bird is a game;

Plants VS Zombies is a game;

Turtle Blind Boxes is a game;

What is a Game?

Snake is a game;

Angry Bird is a game;

Plants VS Zombies is a game;

Turtle Blind Boxes is a game;

Identity V~~A Four-Chinese-Character-Cannot-Be-Said Game~~ is a game;

What is a Game?

Snake is a game;

Angry Bird is a game;

Plants VS Zombies is a game;

Turtle Blind Boxes is a game;

Identity V~~A Four-Chinese-Character-Cannot-Be-Said Game~~ is a game;

~~Game Theory is a game...?~~

What is a Game?

Snake is a game;

Angry Bird is a game;

Plants VS Zombies is a game;

Turtle Blind Boxes is a game;

Identity V~~A Four-Chinese-Character-Cannot-Be-Said Game~~ is a game;

~~Game Theory is a game...?~~

~~Genshin Impact is a game~~!

What is a Game?

Of course, there won't be any elements of narrative meta-storytelling involved here related to "what is a game".

An electronic game consists of three primary components: programming, design and art.

Course Schedule

Lecture 0: OOP Review, Git

Lecture 1 & 2: Introduction, PyGame, Scene I (Simple Scene), Character I (Player Basics), Scene II (Tilemap & Obstacles)

Lecture 3: Character II (Player & NPC), Game Development I (Events), Advanced Python, UI I

Lecture 4: Art, Game Development II (Broadcast & Listening, Singleton, Code Design), Turn-based Combat System

Course Schedule

Lecture 5: Turn-based Combat System (cont'd), Character III (Resources), Gameplay

Lecture 6: LLM Agent I (LLM Intro-Without-Any-AI-Knowledge, API)

Lecture 7: LLM Agent II (Prompt Engineering), UI II

Lecture 8: Writing, (Selected) Demo Presentation?

Found Some Problem?

A game consists of three primary components: programming, design and art.

There is NO design requirement in this project.

Scene (15%)

- One large scene with the camera following (10%)
- One interactive object (e.g. obstacle) (5%)
- ≥ Two *different* scenes (optional)

Characters (15%)

- A main character (5%)
- Two *different* friendly NPCs (5%)
- A simple enemy (5%)
- A special enemy (optional)

Game Mechanics (20%)

- Core mechanics (10%)
- Collision system (5%)
- Resource system (5%)

LLM Agent System (20%)

- Dialogue system (10%)
- Decision system (10%)

Gameplay (5%)

- Main menu (3%)
- BGM (2%)

Code (5%)

- Readability (3%)
- Design (2%)

How we check your project

Your submission must include a **playable game** and a **report**. Please note that in the report, you must explain how you fulfilled each of our requirements and detail any innovations you have made. Otherwise, you will not receive scores for the corresponding sections. We will conduct regular checks on your progress in development.

How we check your project

Your submission must include a **playable game** and a **report**. Please note that in the report, you must explain how you fulfilled each of our requirements and detail any innovations you have made. Otherwise, you will not receive scores for the corresponding sections. We will conduct regular checks on your progress in development.

Further details will be discussed later.

Grading for Creativity

This is not a game design project. It's also challenging to incorporate extensive game design into this project. Therefore, what we would prefer to see is your ability to create an excellent game through **programming**.

Grading for Creativity

In this project, to achieve a higher score, you can implement more detailed combat systems, create richer map content, and enhance the overall interactive experience, etc.

You DON'T need to focus on outstanding gameplay design, write deeply engaging storylines, or prioritize 'fun' above all else. While these aspects are certainly important in games, you may soon realize that game development is inherently *painful*. Allocating the enormous effort required to achieve these aspects in a game might leave you with insufficient time for programming.

Git

https://msyamkumar.com/cs320/s23/lec/02-repro2/slides.pdf

https://msyamkumar.com/cs320/s23/lec/03-repro3/slides.pdf

Intro to PyGame

~~(Powered by GPT)~~ Pygame is a cross-platform set of Python modules designed for writing 2D games. It is built on top of the Simple DirectMedia Layer (SDL) and provides modules for handling graphics, sound, events, and more. Pygame makes it relatively easy for developers to create simple games, simulators, and other graphical applications.

Show Me The Code

A Very Simple Map

Today our goal is simply to give your game the appearance of having a background. Therefore, we have provided you with an image. You just need to load it into your game.

Player.png

Today our goal is simply to give your game the appearance of having a character. Therefore, we have provided you with an image. You just need to load it into your game.

Move the Player.png

However, we may have a slightly additional requirement for it. You also need to learn how to manipulate your character.

What is GitHub?

GitHub is a web-based platform for version control using Git. It provides a collaborative environment for software development, enabling teams to work on projects together.

Create Your first Repository!

We now provide an example repository:
github.com/teafrogsf/SI100B_DIE_Fall_2024_Repo

NOTE: Please DO NOT use 'git push -f' if you are unsure about the CONSEQUENCES!

Settings

Firstly, you can write all the variables in one file for ease of management.

Map

Later, we will generate larger and more complex maps. We can write related content in the same file.

Game Manager

To better manage attributes in a game, we can write related content in the same file.

Player

Similar to Map.py, we can also write player-related content in the same file.

Main

Finally, we keep the most basic logic framework for the game running in one file.

Tiles

You may have noticed that in pixel games, the map is often composed of individual small "tiles". We refer to this type of map as a *tilemap*.

pygame.sprite

In Pygame, a sprite is a 2D image or animation that can be manipulated independently of the background or other sprites. The pygame.sprite.Sprite class serves as a base class for creating sprite objects. It includes features to manage the position, movement, and drawing of sprites.

Customize Your Map

From now on, you can try customizing your map on your own!

Create A Bigger Scene

Sometimes, you may feel that your map should be larger than your window. This is quite common because the size of the window is always limited.

Camera Following

However, if the map is larger than the window, it means that if the map doesn't move, your character will walk outside the window. We need to make the camera follow it, in other words, "make the map move".

Obstacles

There might be some impassable areas on the map. We need to ensure that the player cannot move through these areas.