Game Report

SusuCube

Team W4
Team SuS

Team Information

Developers

Tay Hao Cheng	--	183322F	--	Team Leader / ROLE
Ryan Tan	--	182414L	--	Programmer, Scribe
Nam Kai Zhe	--	180716S	--	Programmer
Lorenzo	--	173582Y	--	Programmer

Project Overview

Game Concept

SusuCube is a top down shooter through a procedurally generated world. The world is generated through chunks/rooms. Each chunk is an open exterior area with different biomes. Each chunk can have different weather which affects both visuals, gameplay and enemies. Players combat enemies with customisable guns. Enemies drop gun parts which can be attached to the current weapon during gameplay. The player will be able to interact with objects in the world to create/move/hold cover.

Project Schedule

Scrum Board: https://trello.com/b/ATah22UG

- Week 1
  - Kai Zhe
    - Map Generation
    - Chunk Collider
  - Lorenzo
    - Biome System
  - Ryan
    - Gun Script/Behaviours
    - Grenades & Projectiles
    - Unique Weapon parts
  - Hao Cheng
    - Player movement
    - UI elements
    - Inventory
    - Reticle cast
    - Observer
- Week 2
  - Kai Zhe
    - Biome Randomizer
    - CRT Screen and Damage Shaders
  - Lorenzo
    - Particle System
    - Interactables
  - Hao Cheng
    - Enemy AI
    - Enemy Spawning
    - Fog Shaders
    - Menus
  - Ryan Tan
    - Generic weapon parts (Removed)
    - Weapon Augments (Added)
    - Loot Drop system
    - Chunk Event System

- Week 3
    - Debugging / Game Testing
    - Preparation of documentation
    - Preparation of presentation
    - Optimisation

    - Hao Cheng
        - Player Data
    - Lorenzo
        - Ability System

Learning Experiences

Problems Encountered and Solved:

- Tay Hao Cheng
    - RE Framework
        - As I designed and implemented most of the framework and used it for my assignments, I knew of every issue with it. Even though I made attempts to abstractify classes and make wrapper functions so that teammates would not need to understand backend stuff, teammates still had issues as they were not aware of issues such as material color not affect when light is not enabled. Some issues teammates faced I already knew of, but they wasted time debugging it themselves. Making issue more prominent via error calls, throws and comments would be the solution.
        - Wrapping functionality. The backend was not as well abstracted and wrapped. Thus teammates needed some understanding of unrelated system to implement their feature. A better planning of the framework backend would have been required.
    - Frame buffer crash
        - An issue that plagued the game only after I switch the rendering from a 3D render to a 2D textured quad render through an FBO. This caused a crash when players would walk a distance. This issue only appeared on Friday of week 2. I had spent 3 days looking into the issue. At one point i realised the FBO data was corrupted, and assumed it was due to bad memory allocation. The issue was 2 renderingmanager was initialised even though only 1 was updated and used. The 2nd renman's memory overlapped with the first, and only after more gameobjects were loaded, did it overwrite data. This was due to a careless mistake regarding singletons.
- Nam Kai Zhe
    - Collision issues
        - I had to redo the collision between game object and chunk multiple times. At first I used a force-based system which pushed the player out, and operated on a chunk-by-chunk basis i.e. each chunk would be checked once. This was obviously buggy and had issues on chunk boundaries.
        - I redid the collision multiple times to successively improve the functionality of the collision response and optimize the code to run faster, spending much of the first week upgrading my collision alongside other tasks.

- Ryan Tan Zheng Rong
    - Vector iterators & removal of objects from vectors
        - One of the features of the weapon parts was the removal of weapon parts from their respective vector when their durability was used up. Due to my lack of knowledge of iterators and the std::vector class I ended up causing crashes to the program. My solution to this issue is to read up more on the vector class and iterators, I learnt that vector iterators such as (.end()) gets messed up when I remove an element from the vector if I am iterating through the vector via for loop. I also learnt about other members of the vector class like (.rend()) and how to better manage vectors in the future.
    - Development of programming skills & Work Ethics
        - Prior to this studio project, whenever I was assigned new tasks to do, I would continuously ask others for help over features I was in charge of whenever i ran into an issue due to a lack of planning or I would be easily distracted.
        - Over the course of this studio project, I developed better working/programming habits such as taking the time to plan out how I will be doing my feature via creating class diagrams and asking my fellow teammates about the design along with what I need to be aware of when I am making the feature. Another habit I developed was to note down what was required of me for the day (tasks, bug-fixes, etc.) and would focus on those tasks. To elaborate more on my development as a programmer, I was too focused on the way I designed my code (coding patterns, standardising my code) and would put too little attention into the implementation of the code in relation to the systems I was working with prior to SP3.
        - During my time in SP3, as I did more features I started to design my code in correlation to the other systems my teammates made rather than just design the code on my own and just implementing it while hoping that the rest of the systems will just work together with my code. One example was during the creation of the chunk events feature. That feature relied on systems/features created by Kai Zhe (His ChunkData Class) and Hao Cheng (AI Entities, DataContainer Class), I had to design my code with respect to their systems.
- Lorenzo Yanga
    - Adapting and debugging the framework
        - The first week of SP3 was complicated as I had trouble understanding the framework. Debugging issues in the game were complicated, especially since I was using a framework I was still new to.
        - During the second week, Kai Zhe pushed an update (specifically, changing chunk collision) that fatally slowed down the frame rate of the

game in Debug mode. A poor frame rate messes with the collisions in the game, and graphically it can look jarring. The immediate solution at the time was to run the game in Release mode while developing, which limited the scope of what we could do when debugging the game.

- Work Ethics
  - Towards the first half of SP3, I got too easily distracted, or focused on small tasks for too long. This ended up in me having less features compared to the other 3. Week 2 issues made me have to restructure my to-do list what to do to get sufficient features done.