Team Milestone Report

Milestone 4 

Team Name: **Sangrin**
Game Name: **Manzo**
Milestone: **4**

1) List all team members and their roles. (5 pts)

Gyuwon Na	Tech lead
Won Kim	Art lead
Seokhwa Hong	Test lead / sub art & audio
Seohyeon Min	Producer / sub art & audio
Jiyeop Kang	Level designer

2) This semester, students can create their own team hierarchy and roles and team responsibilities. Explain the roles of each student in adequate detail. (10 pts)

This semester, our team has adopted a flat and equal structure where everyone shares the same level of authority. This approach encourages open and comfortable communication, allowing all members to freely share ideas and collaborate efficiently.

Seo hyun Min – Producer / Technical Artist
Seo-hyun manages the project schedule and task assignments. She also contributes to the technical side by developing shaders and supporting the art pipeline.

Won Kim – Art Lead / Technical Level Designer
Won is responsible for implementing game maps based on the designs. She focuses on integrating map elements into the game and checks technical functionality.

Seok-hwa Hong – QA / Game Designer / Sub Artist
Seok-hwa designs the game maps and plans key gameplay mechanics. He also supports visual design and contributes to quality assurance by testing the game.

Gyu-won Na – Tech Lead / Gameplay Programmer
Gyu-won leads the development of core gameplay systems, including the implementation of various features and fish AI behavior, ensuring smooth functionality and system integration.

Ji-yeop Kang – Gameplay Programmer / UI/UX Programmer
Ji-yeop handles the design and implementation of user interfaces, including both visual components and their connection to underlying game systems.

3) Give a 3~5 sentence description of your game. (5 pts)

- Our game is a rhythm-based exploration set underwater, where players move in sync with the rhythm by clicking in the desired direction. As the player dive deeper, they

follow mysterious Morse code signals that lead them to strange fish. These strange fish seem to be hiding secrets about the missing grandmother. The shallow waters are bright and playful, but the deeper you go, the darker and more unsettling it becomes.

4) List the major components and systems that were designed and created during this milestone for each of the following disciplines:

- Debugging, Testing, Optimization (5 pts)

  Debugging and Validating Fish Spawning Logic
  - To ensure the stability and correctness of fish spawning, we thoroughly debugged position values using runtime logging and visual markers. Testing was done to confirm that fish appear at randomized Y coordinates between -300 and -2000, and only on valid screen sides.

  Optimizing Text Rendering Loop
  - We also optimized the text rendering loop using conditionals and reduced redundancy with a loop-based approach, improving maintainability and reducing CPU overhead in repeated draw calls.

  Frustum Culling for Rendering Optimization
  - To reduce unnecessary rendering and improve frame performance, we implemented frustum culling to ensure that only objects within the camera's view are rendered. By calculating the camera's frustum planes and checking whether fish objects or environmental elements intersect with the visible frustum, we skipped rendering for objects outside the screen. This significantly reduced GPU load, especially when numerous fish entities were present in the background.

  Viewport Based Rock Loading System
  - As the map size increased, loading all 'rock' objects at once began to cause noticeable frame drops. To resolve this issue, we implemented a loading system based on the camera's current viewport. rocks are now only added to the GameObjectManager when their bounding boxes intersect with the camera boundary.

- Art (5 pts)
  - Boss and Monster assets
  - New shaders (Title background shader, Water surface shader for Mode2, Distorted post process shader for water flow effect, background gradient shader along the player's position for main gameplay)

- Gameplay, Game Design (5 pts)
  - Boss and Monster play
  - New map for Level 1

- Music, SFX, Audio (5 pts)

    - Upgraded spatial audio
    - Music for main screen, Mode2, Mode1 and Boss fight.

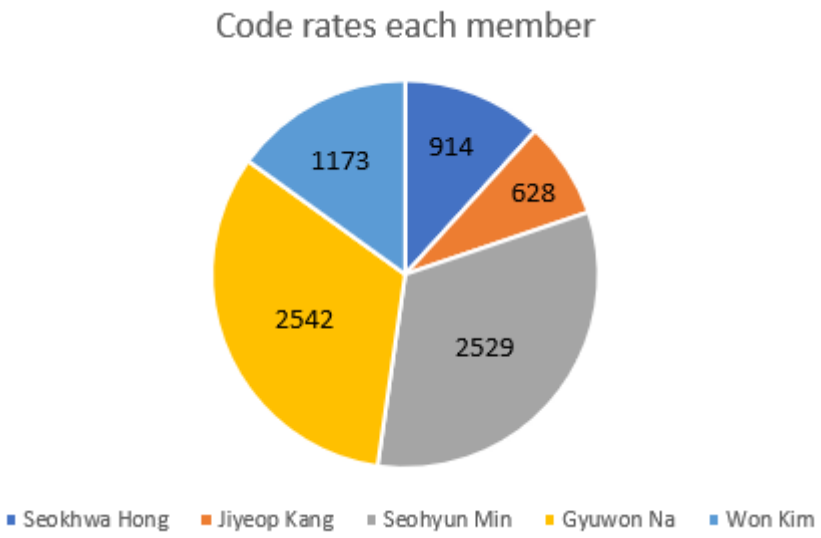
- Physics (5 pts)
    - Boss bullet (Basic kinetic motion, Homing behavior with dynamic acceleration and Directional spread)

- Tool Development, Building Pipeline (5 pts)
    - Extension of SVG Parsing (Now it can read information not only about rocks but map objects, like treasure.)

5) Rate your project in the following categories 1~10: (Your grade will not be changed based off of your ratings.) (5 pts)

- Gameplay 6
- Visuals 8
- Audio 8
- Technical feats 8
- Overall presentation and cohesion 7

6) Give me a rough estimate of how much code each member in your team wrote. (This should be out of 100%. You can even make a pie chart! 😋) (5 pts)



7) Look over this long list of features. For each one, mark if this feature has already been implemented, will be implemented in the future for the final product, or will not be implemented for your project: (10 pts)

Feature	Implemented	Will Implement	Won't Implement
Mouse support	O		
Keyboard support	O		

Gamepad support		O	
Touchscreen support		O	
Motion controls		O	
Voice-over audio			X
In-game SFX	O		
Background music	O		
Spatial audio	O		
UI/menu SFX		O	
Advanced/dynamic audio filters		O	
Dynamic lighting/shading	O		
Vector graphics	O		
Sprite-based animation system	O		
Sprite scaling	O		
Sprite rotation	O		
Particle effects	O		
Image masks		O	
Partial-transparency image blending or alpha blending methods	O		
Parallax backgrounds	O		
Multiple layers of background graphics	O		
Kinematic/skeletal art		O	
Scripted motion using vector paths		O	
Animation tweens		O	
Multiple levels/environments		O	
Asymmetrical gameplay	O		
Advanced physics simulations	O		
Local multiplayer			X
Networked multiplayer			X
Downloadable content or online			X

content fetching			
Any other network features			X
Integration into web technologies (Such as web portals)			X
Component-based architecture	O		
Game objects use C++ interfaces	O		
Game objects are data-driven from factories	O		
Menu systems		O	
HUD		O	
File parsing for gameplay content, such as levels, scenarios, enemies, etc.	O		
In-game level editor		O	
Scripting language integration	O		
Metaclasses, reflection, data-binding	O		
Live object property inspection	O		
Art pipeline tools	O		
Audio pipeline tools		O	
Engine/game build (compilation) tools			X
External gameplay editor			X
In-game gameplay editor			X
Testing tools and advanced debugging features			X
Ability to jump into the game in a given scenario for testing		O O	
Fast-forward and rewinding gameplay		O	
Extensive/complex debug drawing		O	
In-game performance visualization	O		

8) What other advanced features do you plan on implementing that are not listed, above? What other advanced features have you implemented, already, that's not listed, above? (5 pts)

I want to make the game playable with a gamepad. Aside from that, most of the planned features seem to be implemented as intended. The sound effects system is also complete, so I just need to find and add the appropriate sounds.

9) Assess your project's scope based off of questions #7 and #8… How has the project's scope changed since last semester? Did it grow/shrink or stay the same? Why? Justify the change in scope. Do you think your team is on track to complete every feature that is planned by the end of the semester? Why or why not? Please go into detail with these questions. (15 pts)

Our project's scope has significantly shrunk since last semester. Initially, we ambitiously planned to create as many bosses as there are letters in the alphabet, which already felt a bit tight. However, as we progressed through development, we encountered many unexpected bugs and issues that slowed us down, ultimately forcing us to reduce the scope.

Originally, we envisioned an open-world structure with vague boundaries between areas. Now, we've pivoted to a level-based design, where each level is clearly separated and features a unique boss. We are currently planning 3 to 5 levels, but we believe that just 3 levels are enough to deliver the core gameplay experience we intend. As a result, we're prioritizing Level 4 and Level 5, which are set in the deep sea. If time permits, we will add Levels 2 and 3 to complete all five levels.

Despite the reduced scope, we are confident in the overall completeness of the game. All major systems are implemented, except for some bugs, and we are now at the stage of adding final content. We also aim to make the visuals as impactful as possible. While we're not entirely sure if the game will visually match our ideal vision, we believe that from a systems and gameplay perspective, the experience we want to deliver is fully achievable.

10) Generally speaking, what's the next step for your project? What's the direction that this project is moving and how will the future features' work be split between your members? (10 pts)

One of the reasons our team is functioning so well is that each member has clearly defined responsibilities.

Seokhwa is in charge of planning and game design, and is also creating the bosses. Won is focused on map implementation and SVG parsing. Gyu-won has developed core systems and the underwater ecosystem, while also supporting UI development. Ji-yeop, as a gameplay programmer, has been building the necessary systems piece by piece, such as UI logic. Seohyeon has managed our schedule and contributed to shader development and adding juicy visual effects throughout the game.

As long as we continue working the way we have so far, we believe the next milestone will go smoothly. Right now, what we need are more bosses, larger maps, improved visual polish, connecting all systems for smooth gameplay, and integrating UI assets. Overall, the project is moving in the right direction — what we need most now is time.

11) Please list any issues your team has dealt with during this milestone: (This can be technical issues, personal hardships, team disagreements, etc.) What was the plan to overcome these issues? (5 pts, but up to 5 pts of extra credit)

There were times when we fell behind schedule due to procrastination. It made us wonder how to handle it, but we found that staying together in the lab to work as a team greatly improved our efficiency when needed.

12) Extra notes for the instructor! (Completely optional!) As a team, you can use this area to tell me anything else you find important! Does your team think you'll have the best game in the class? Why? Does your team deserve extra credit for something? Tell me! Seriously, anything else you want me to know, put it, here!

Of course!!!!!!! Even though we decided to decrease the volume of our game, we think our game is the best game in the class. We have beautiful assets, shaders and colorful background music, audio systems, boss, monsters, a large maze-like map and other cool things. Also, we aim to exhibit our game at major events such as G-STAR or other indie game festivals, where we can connect with players, developers, and potential publishers.