Implementation report

Throughout this document we have used Geese Lightning's requirements [1] and testing [2] to justify various parts of implementation.

How the Architecture is Implemented

We added new classes that weren't previously in the concrete architecture as they weren't required for the previous assessment. We added a BonusLevel and BonusGoose class which are required for the minigame. These build on the minigame class and obstacle classes from the abstract architecture. We didn't make the same minigame as had been planned in the abstract architecture as we didn't know what was planned for the minigame by the previous group on this game, as a result we have removed the rider class entirely.

In addition, to the entirely new classes mentioned above we have also added classes for each new level, and zombie type. This seems to be how the architecture was intended to be extended as each level and character had their own class already. As a result, we have followed the same format when adding new levels and zombies. We implemented bosses in the same way as our new zombies, but they also have some extra features and abilities than the normal zombies.

Significant new Features

Feature	Justification	Relation to Requirements or Architecture	Side Effects	Comments
Drama player type added. This was added in the same way that other player types were added.	Three player types were required for the assessment and by the requirements.	Requirement F3 required three player types. No significant changes to architecture.	No noticeable side effects.	Abilities still need to be added for all player types.
Player abilities added to every player type. These use same input method, but affect the player temporarily. This done by changing player stats and using a delta time to measure how long the ability is active and how long the cooldown should last.	No players had abilities which were required in the requirements.	Requirement F3.2 required each player to have different abilities.	This could cause certain players to be stronger than others.	May need balancing to prevent certain abilities from being too powerful.
New power up types insta kill and no ability cooldowns. Implemented in a similar way to old power ups, but affect different stats,	This results in a total of five power ups which we require for the assessment and this meets the game requirements.	Requirement F6 requires the game to have 5 power ups. Two new classes have been added for the new power ups. These extend the PowerUp class as previous power ups do.	Causes other power ups to spawn less frequently. This could mean a player never experiences	May need balancing to stop the power ups from being too powerful.

			some power ups.	
Added saves to the game. Buttons were already there, but now they are functional. This is done by reading and writing the level progress to a .txt file.	A save feature was in the requirements.	Requirement F8 required the game to be saveable. This hasn't change the architecture of the game.	No noticeable side effects.	Player may be able to change the text file to cheat. Not sure if it will work when the executable is made.
Minigame was added to the game. This is "Goose" hunt and is accessible from the bonus game button. Buttons were implemented in a the same way as other buttons in the game. Much of the movement and other features are new.	A minigame was required for the assessment and in the requirements.	Requirement F5 required this to be added to the game. This resulted in two new classes being added to the architecture. These were BonusLevel and BonusGoose. Bonus level didn't extend level due to how different the minigame is from the main game.	An extra button has been added to the level select screen to access the minigame.	Using stage buttons and sprites on the same screen seems to cause this to run slowly. This could also be due to certain aspects not being rendered correctly.
Two bosses added to the game, one at the end of the third level and one at the end of the last level. These work in a similar way to other zombies, but also spawn extra zombies when there aren't many zombies left on a level.	Two bosses were required by the assessment and our requirements.	Requirement F7 requires two bosses in the game and F7.1 requires one boss in the third level and one in the sixth. This has resulted in two new classes being added BossCourtyard and BossCentralHall. These extend the character class and allow the different functionality of the different bosses.	Stages with a boss are harder. Spawning of zombies had to be changed to allow bosses to spawn.	Could vary bosses a bit more if there is time.
Added two new zombie types. A fast zombie and a flaming zombie. These work the same way as other zombies, but have different stats.	This adds a extra layer to the game and 3 different zombie types were in the requirements.	This meets the requirement to have 3 zombie types F4 and the requirement F2 for the game to get progressively more difficult. As a result we had to had two new classes one for each new zombie. This fits with how the architecture was already laid out.	Other zombies now spawn less frequently. The spawning method picks a number between 1 and 11 and different numbers spawn different zombies.	Might be better to gradually add different zombies in later levels.
Added 3 new locations. These are the library, Physics and Central hall. Added in the same way as previous levels.	This was required for this assessment and in our requirements.	F1 requires the game to be split in to six locations, with these extra locations there are now six locations in total. This has resulted in 3 new classes being added to the game, one for each location. This seems to fit how the	We had to make buttons smaller on the level select screen to fit all the buttons on.	Some levels are odd shapes causing zombies to get stuck occasionally.

		architecture was already implementing new levels.		

Significant Changes

Change	Justification	Relation to Requirements or Architecture	Side Effects	Comments
Fixed crash after level completion. This was done by stopping files from being used after they were disposed.	This made the game worse to play, stopped the game from running smoothly and made it impossible to test after a few levels had been played.	Requirement P2 required the game to run smoothly so this needed to be fixed. This hasn't changed the Architecture.	No significant side effects.	All changes within level class.
Improved movement. Old movement would cause the player to stop when changing direction sometimes. This was done by changing the key up method in ZeprInputProcessor.	This cause the player to jolt about and made controls difficult to use.	Requirement P2 requires the game to run smoothly. This has made the game more smooth. This hasn't change the architecture.	No noticeable side effects.	All changes in ZeprInputProcessor class.
Changed textures for different player types. This allows the player to identify which type they are easily.	Textures were all very similar. This allows the player to more easily identify which type they are. Also, this allows indicators for some abilities on the player.	Fitting with requirement F3 this makes the player types more varied. This hasn't change the architecture.	No noticeable side effects.	This only required changes to the player class.
Changed power up spawning. Powerups always spawn if one isn't active or on the map at the end of a wave.	Power ups didn't spawn occasionally as they were in a separate if statement in the update method from zombie spawning, but spawned due to the same condition.	Requirement F6.1 requires a power up to spawn at the end of every wave.	No noticeable side effects.	This is all within the update method for level.
Made player information reset on respawn. E.g health, powerups, attacks.	Previously when respawning player attacks and movement would be carried over.	This could have been confusing to the use going against requirement N1. This required the game to be easy to learn to play.	No significant side effects.	May need to do this with other features if any are added.
Added display for power ups. Tells player power ups that are available	The ability to do this was already implemented it just wasn't fully	This fits the requirement E2 to give the user feedback. This hasn't changed the architecture.	No significant side effects.	Side text might be better if it was smaller. It covers a lot of

and how long they have left.	displayed. Gives the player more feedback and makes the game easier to pick up.			the screen.
Removed disposal of zombie texture before it had been removed from the level.	This caused a weird texture glitch which reduced the smoothness of the game.	This wasn't required, but was clearly an issue with the game.	No significant side effects.	Texture should be disposed automatically when the space is cleared by garbage disposal.
Changed combat mechanic to stop player from constantly holding down the attack button.	This fixed the issue in the test 10.1 that prevented the player from turning when attacking and makes the combat more natural.	Requirement F10 required this change for the combat to fully meet the requirement. This hasn't changed the game architecture.	This makes combat harder.	An increased attack duration allows the player to do more damage in a hit.
Improved collisions allowing the player and enemies to move in one direction and collide in the other.	Before the collisions would stop the player from moving up if they were colliding with something on their right. Similar things happened with any combination of x and y movements.	This allows F9 of the zombies seeking out the player to be more fluent. This also improves N1 making the movement easier to understand.	Makes combat harder as zombies can get to the player more easily.	Could maybe improve collisions with characters a bit.

Features that haven't been fully Implemented

We have fully implemented all features in the Scenario for the game to the best of our ability. However, it is hard to judge some of the scenario requirements such as the progressive difficulty increase. Difficulty can vary from player to player and due to having some random elements in the game this could also vary from player to player. Without extensive user testing it is hard to judge this, but our main way of increasing difficulty was increasing the number of zombies that spawn in a level.

In addition, to this the save feature is slightly different than it was required to be by F8. This required the game to be savable at any time. Instead we have allowed saving only on the menu screen. This way the player can save in between levels. This made the implementation of the save feature easier, but also was the only way for saves to work with the previously implemented menu system. This also prevents the game from becoming too easy by allowing the player to save halfway through a level if they are doing well and gradually get through rather than playing levels through as they were intended.

References

[1] Geese Lighting, 'Updated Requirements' [Online], January 2019, Available: https://drive.google.com/file/d/1vr9Qq7EU2rcqsxOB8uAZA9nS6YLWn-rH/view

[2] Geese Lighting, ‘Black Box Testing Evidence’ [Online], January 2019, Available: https://drive.google.com/file/d/1MOTAnexh0i3vHoNs4XxCxbprjVM150Sr/view