

2021-22

UFO RESISTANCE

FINAL REPORT

Liam Rickman – 1902527

Abertay University

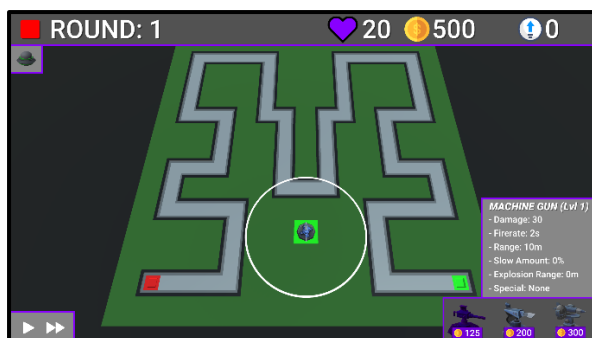
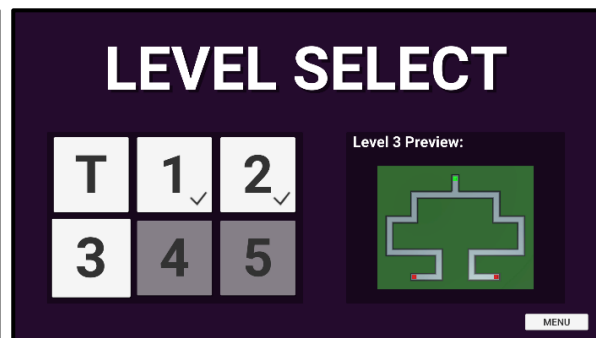
DES308 – Analytics and Data-Driven Game Design

Contents

Design of Prototype.....	1
Market Analysis.....	2
Data Hooks.....	2
Data Collection.....	3
Testing Outcomes.....	4
Data Analysis.....	5
Prototype Iterations.....	16

Design of Prototype

UFO Resistance is a tower defence game where players are tasked with defending their home base against waves of enemies that move along a path. I began the prototype by following a [tutorial](#) to get the basics implement. However, this was heavily expanded on to create a more complete game. Finally, I used an [asset pack](#) to make it UFO themed. I chose to create a tower defence game as I realised there would be a large range of both qualitative and quantitative data that I could collect and analyse to improve the prototype.



Market Analysis

Whilst designing my prototype and planning its features, I conducted market research to see where my game would fit in the current market. I started looking at the statistics of similar games to see the potential of the game.

GAME	Bloons TD 6	Plants vs Zombies	Orcs Must Die 3	Defense Grid 2
UNIT SALES	~3.15 million	~3.85 million	~200 thousand	~200 thousand
DIGITAL ONLY?	Yes	Yes	Yes	Yes
SELF-PUBLISHED	Yes – Ninja Kiwi	Yes – PopCap Games & EA	Yes – Robot Entertainment	No – 505 Games
PRICE (USD)	\$9.99	\$4.99	\$29.99	\$14.99
LENGTH	25-65 hrs	12-25 hrs	15-20 hrs	13-50 hrs

Statistics from [SteamDB](#)

The market already features a large range of tower defence style games with many selling hundreds of thousands of copies. As my prototype is small scale, it would need to be on the cheaper side, possibly even free to play with a microtransaction model. There is the potential to reach a large audience of both casual and committed players. Finally, I also created a table to see the suitability of my project for this module

TITLE	UFO RESISTANCE
GENRE	Real-Time Strategy
PLATFORM	PC
CONTROL	Mouse and Keyboard
CORE MECHANICS	Enemies moving along a path towards a home base.
	Players building turrets that deal damage or apply effects to the enemies.
	Upgrading and selling turrets
SKILL LEVEL	Beginner > Expert (Depending on level difficulty)
GAME LENGTH	~10 minutes > Theoretically infinite depending on number of rounds.
SESSION LENGTH	1-3 games, depending on how players survive each game.
PRICE	Free > \$4.99

Data Hooks

For quantitative data, I focused on collecting data that could be used to improve the balancing and analyse how difficult the game was for both new and returning players. While each level has the same enemies, money, and general difficulty level, I decided to record each level separately to see if different layouts had any

impact. Across the whole game, I recorded a summary of which scenes players accessed and how long for, to see if any levels were particularly time consuming.

In each play test I recorded data on player health to give me a sense of general difficulty. I also recorded how many of each enemy were killed and how many made it through the defences to see if certain enemy types were making to game too hard. I also noted which turrets were being built, to see if players gravitated towards a particular turret which would indicate that turret being too strong. This will also tell me how much players are upgrading turrets as I wanted this to be a prominent feature in my game. Finally, I collected position data on enemy deaths and turret placements to create heatmaps which could highlight problem areas in the level layout.

In the third play tests, I recorded the amount of money and upgrades the player spent, which will tell me if the players are using up the available resources. I recorded which turrets killed enemies to further analyse which turrets were overperforming. I had planned to use this to get an average kill per turret built, however due to an error in the way I recorded turrets being built this was inaccurate.

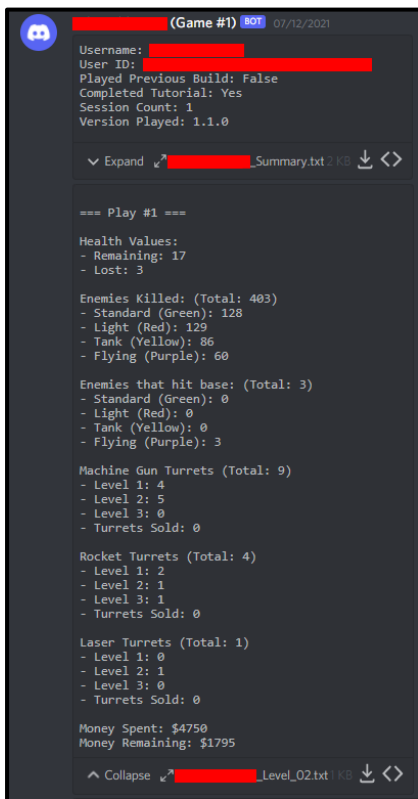
Finally, I collected qualitative player feedback in all play tests. I asked specific questions to narrow down the feedback I received, but also collected any other ideas they had. This will mostly be used to improve the feel and quality of the game.

Data Collection

I planned to use Unity Analytics to collect data as it is built into Unity and has a dashboard that could quickly display data in graphs. Unfortunately, I found the dashboard too unreliable, with it being difficult to distinguish between builds and data often taking up to a week to come through. This wasn't good for my prototypes as it would slow down data analysis and production of any iterations. I decided to find another way to collect data and decided on a webhook based script that could store data in text files and send them to a webhook hosted in a Discord server.

I found a [script](#) online which gave me the functionality to send files to Discord, which I then heavily modified to have the functions I needed. I created functions to create, add to, edit, delete, and send files. Unlike Unity Analytics, this data was sent instantly, which allowed me to analyse the data as soon as I wanted. This didn't have a dashboard or automatic graph creation, so I used Microsoft Excel to compile the data and create graphs to visually analyse the data. This script evolved as the build got more complex and more data was recorded. I added .csv functionality to speed up importing data to Excel. Final Version: <https://pastebin.com/BCTBZajK>

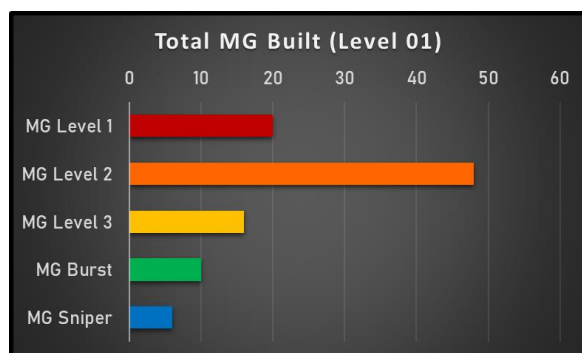
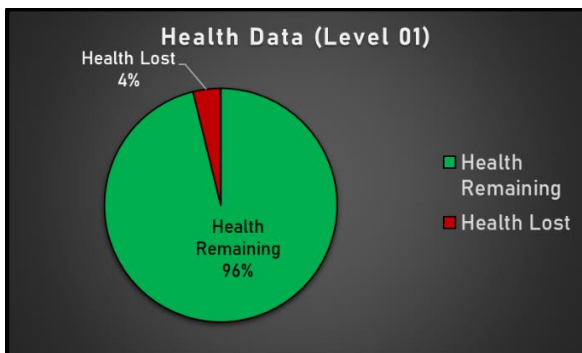
Data in Discord (v1.1.0):



.csv Data (v1.2.2):

Level 01	REDACTED				
Play #1	Round Reached: 10				
Health Stats	Remaining	Lost			
REDACTED	17	3			
Currency Stats	Current Money	Money Spent	Current Upgrades	Upgrades Spent	
REDACTED	1377	1200	0	43	
Enemies Killed	Standard	Light	Tank	Flying	
REDACTED	217	195	130	93	
Enemies At End	Standard	Light	Tank	Flying	
REDACTED	3	0	0	0	
MG Built	MG Level 1	MG Level 2	MG Level 3	MG Burst	MG Sniper
REDACTED	0	1	1	1	1
MG Sold	MG Level 1	MG Level 2	MG Level 3	MG Burst	MG Sniper
REDACTED	0	0	0	0	0
MG Kills	MG Level 1	MG Level 2	MG Level 3	MG Burst	MG Sniper
REDACTED	87	121	137	46	49
Laser Built	Laser Level 1	Laser Level 2	Laser Level 3	Laser AOE	Laser Damage
REDACTED	0	1	0	0	1
Laser Sold	Laser Level 1	Laser Level 2	Laser Level 3	Laser AOE	Laser Damage
REDACTED	0	0	0	0	0
Laser Kills	Laser Level 1	Laser Level 2	Laser Level 3	Laser AOE	Laser Damage
REDACTED	2	15	0	0	3
Rocket Built	Rocket Level 1	Rocket Level 2	Rocket Level 3	Rocket RF	Rocket Nuke
REDACTED	0	0	0	1	0
Rocket Sold	Rocket Level 1	Rocket Level 2	Rocket Level 3	Rocket RF	Rocket Nuke
REDACTED	0	0	0	0	0
Rocket Kills	Rocket Level 1	Rocket Level 2	Rocket Level 3	Rocket RF	Rocket Nuke
REDACTED	0	63	57	55	

Example Excel Graphs:



I also used a Heatmap script provided, which allowed me to find problem areas in each level, this was modified to ensure it worked in build and had the functions needed. I used this to record enemy death positions and turret placements. Finally, to collect qualitative data I took notes while watching players test the game and gave them a [feedback form](#) to collect more specific feedback.

Testing Outcomes

Overall, I believe my testing was a success. I was able to gather a range of qualitative and quantitative data that I used to iterate on my prototype with positive feedback from returning players. I was particularly happy with my Discord webhook script for collecting data as it sent the data instantly for me to analyse, something I couldn't do reliably with Unity Analytics. I did have to create the

graphs manually in Excel; however, this was still quicker than waiting for Unity Analytics to update and wasn't too time consuming at this small scale.

To ensure I had enough testers to gain new players each iteration along with returning players, I aimed for my play tests to gain 5 new players each session, and as many returning players as I could get. This worked for the first two, but unfortunately, I did not receive as many for the final iteration, but I still received a lot of valuable data.

Test Session	New	Returning	Total
1	5	0	5
2	5	5	10
3	4	9	13

As more testers played my game as the project progressed, it got more time consuming to import the data into Excel. This was improved slightly with the .csv functionality of my Discord script, however if I were to continue collecting data this way, I would create a better system to speed up data analysis. If my game testing kept scaling up, it would be beneficial to switch to Unity Analytics or another online dashboard to be more automated. This may increase time between play test and data analysis while waiting for dashboards to update but would be an important trade-off for efficiency.

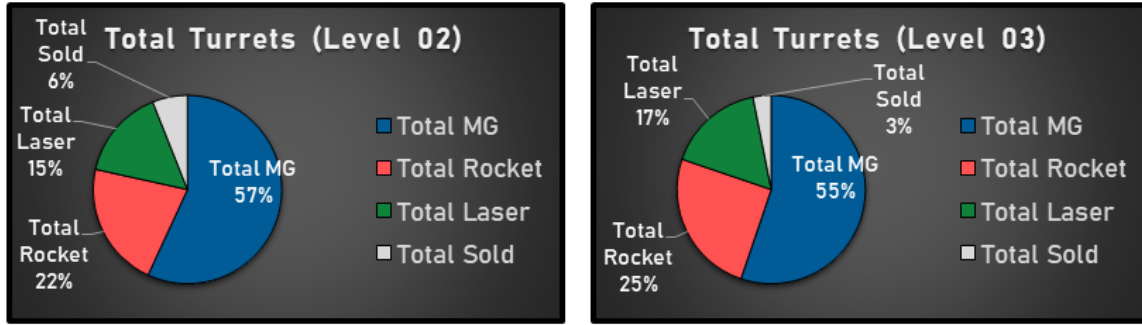
Data Analysis

Session #1



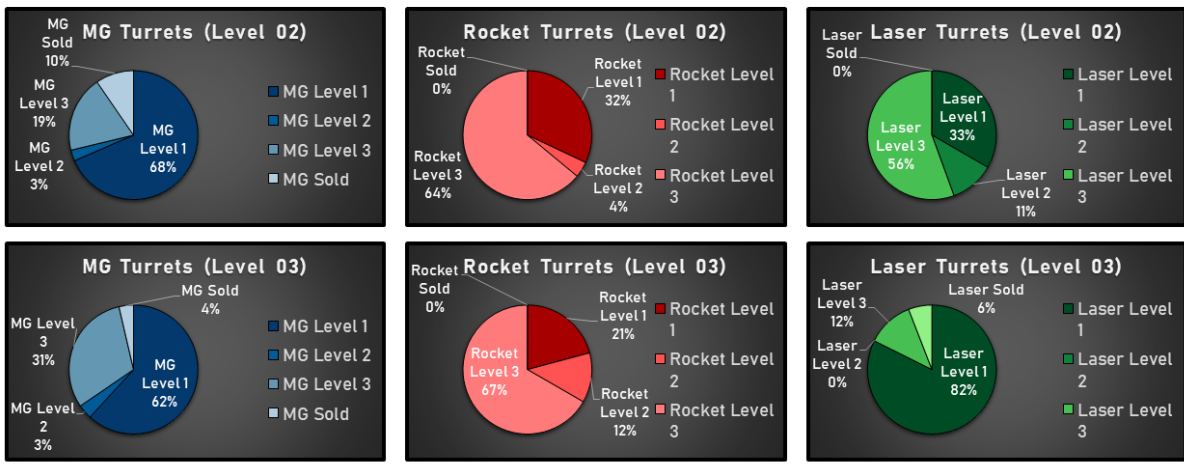
Enemies rarely made it to the end over both levels, which suggests the game is currently too easy with enemies not posing enough of a challenge.

Turrets – General (v1.0.2)



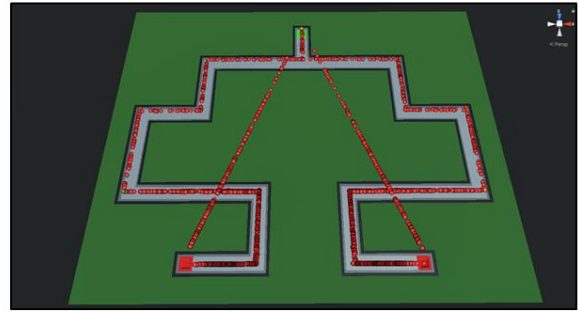
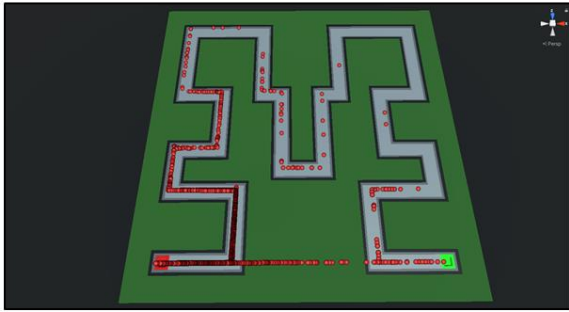
Players are gravitating towards the MG turrets which isn't a huge deal as they are the standard turret, however I would like users to use the other turrets more, as they can change up the gameplay which could help with player engagement. This also suggests the MG turrets are too strong compared to other turrets.

Turrets – Split (v1.0.2)

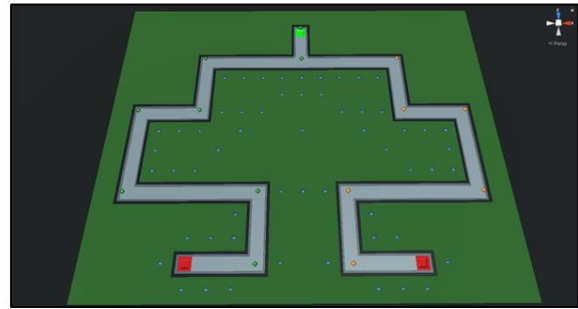
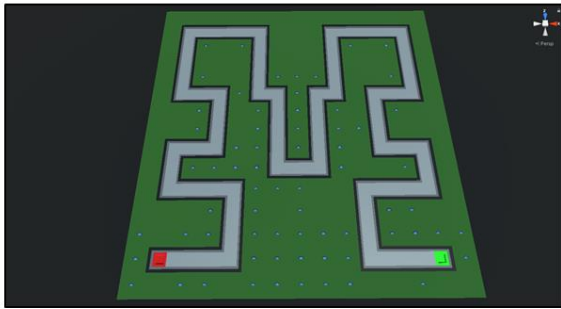


MG turrets were mainly level 1, with only a few being upgraded. This is less than desired, suggesting level 1 turrets are too strong and need balancing. Interestingly, rocket turrets were mostly fully upgraded, suggesting it is too easy to reach this level and are too strong. Finally, laser turrets were built and upgraded less in the second level, suggesting players didn't enjoy using them in the first level, especially for the price.

Enemy Deaths (v1.0.2)



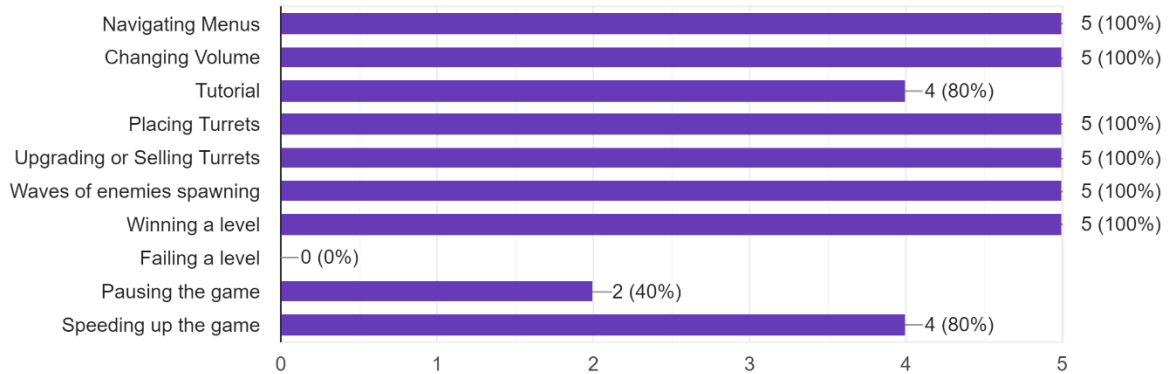
Turret Positions (v1.0.2)



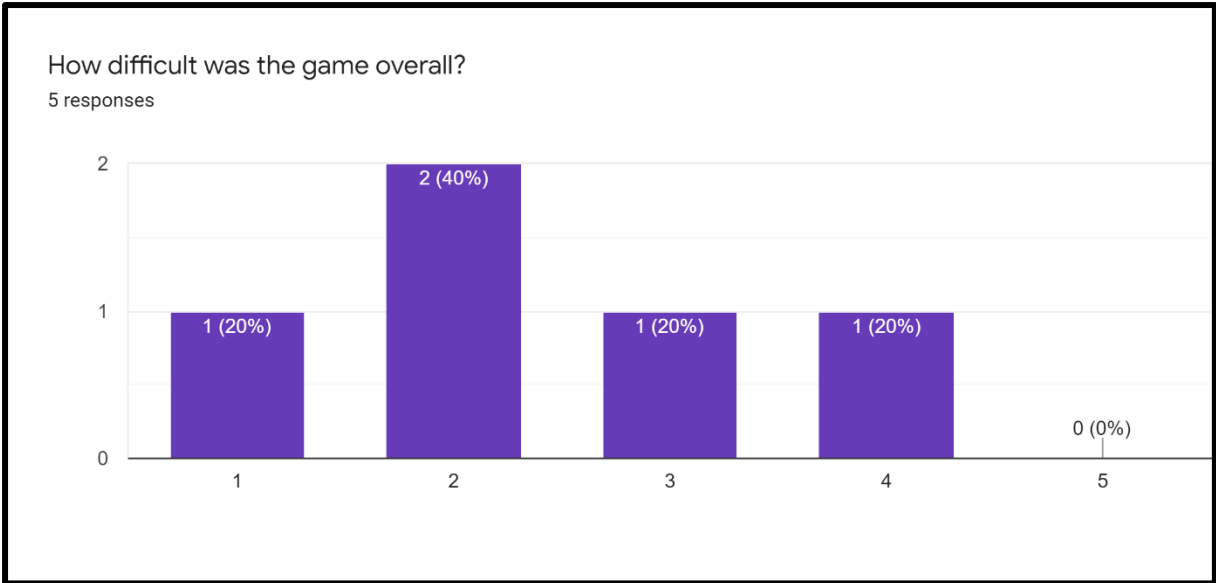
Heatmaps show that players focused turrets at the bottom in level 02 resulting in grouped enemy deaths, but spread turrets out more in level 03, leading to more spread-out enemy deaths. This shows the split spawn and shorter paths of level 03, changes up players strategies.

Which mechanics did you interact with?

5 responses



No users failed a level, hinting that it is too easy. One user didn't even access the tutorial and was later confused by the spawn and base tiles.

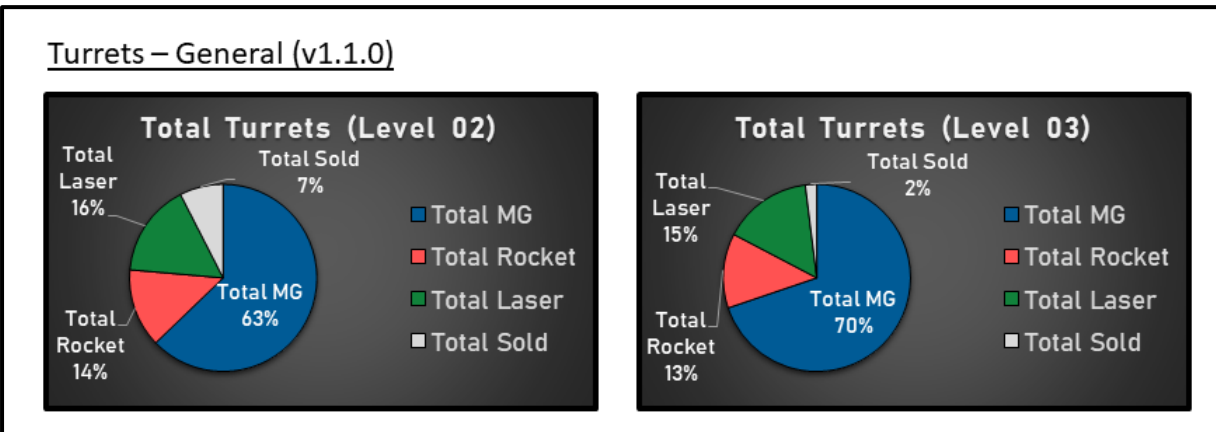


Players agreed that the game was too easy, further backing up the data. Users also suggested some quality-of-life improvements that would improve game quality and feel.

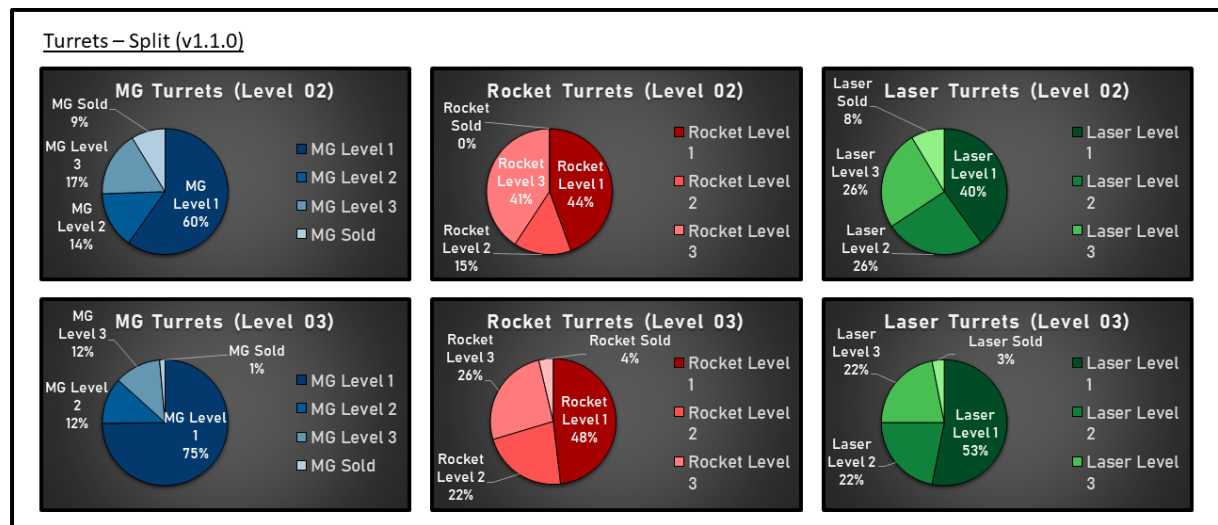
Session #2



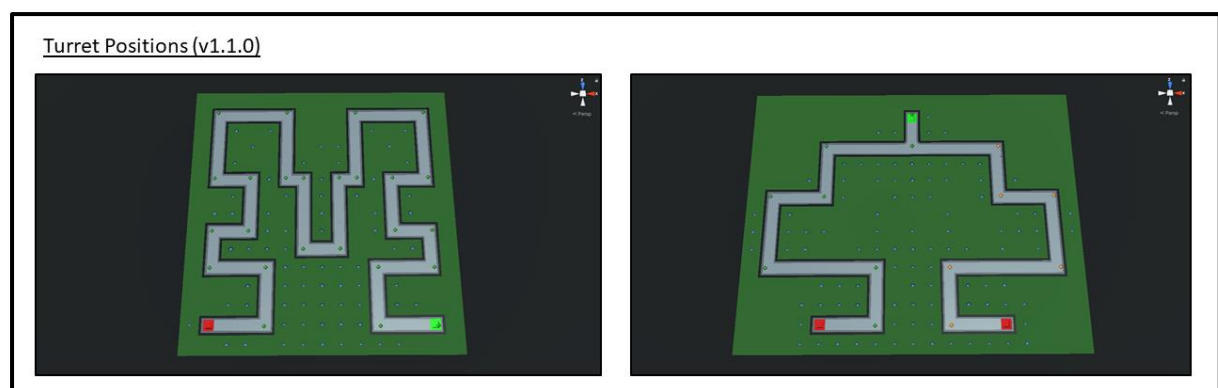
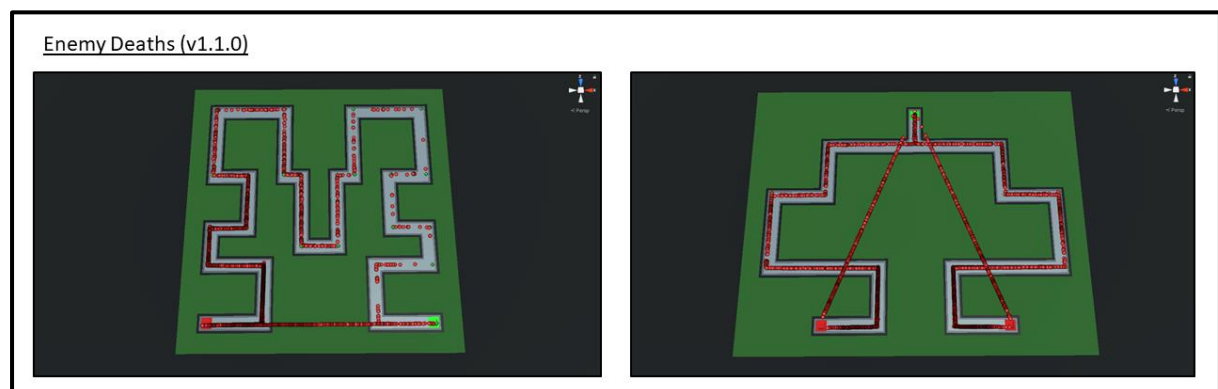
The main problem with level 02, were flying enemies, likely due to the level layout as they move along the bottom which isn't often covered by turrets. Level 03 was more balanced and had no obvious issues.



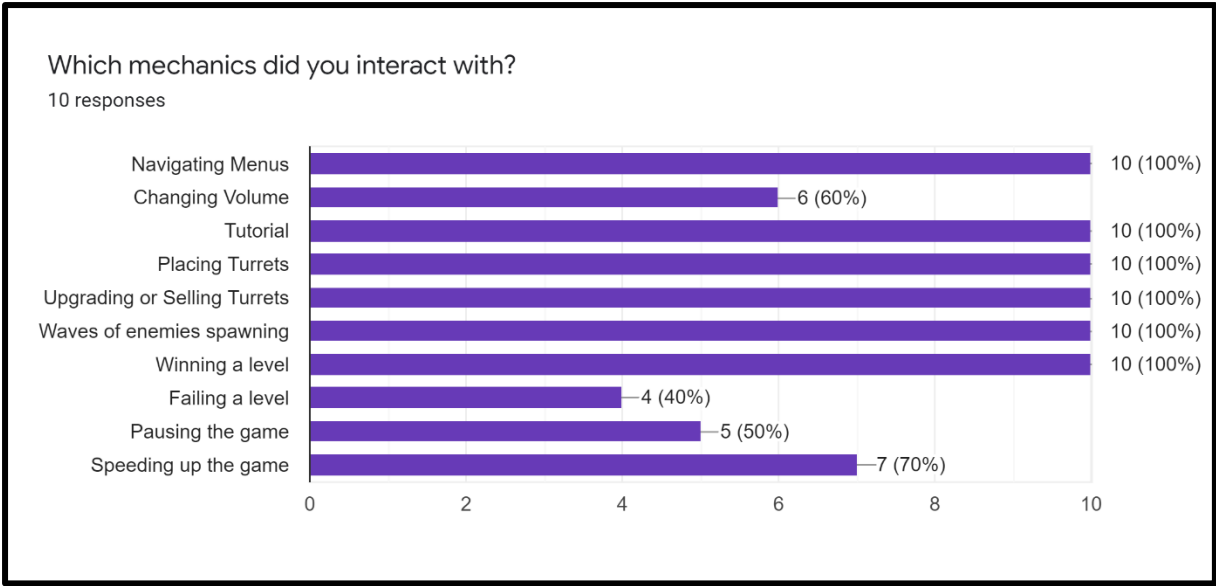
Players were still building mostly MG turrets. Newer players used a larger variety, whereas returning players stuck with MG turrets, suggesting previous nerfs were not enough to discourage their usage.



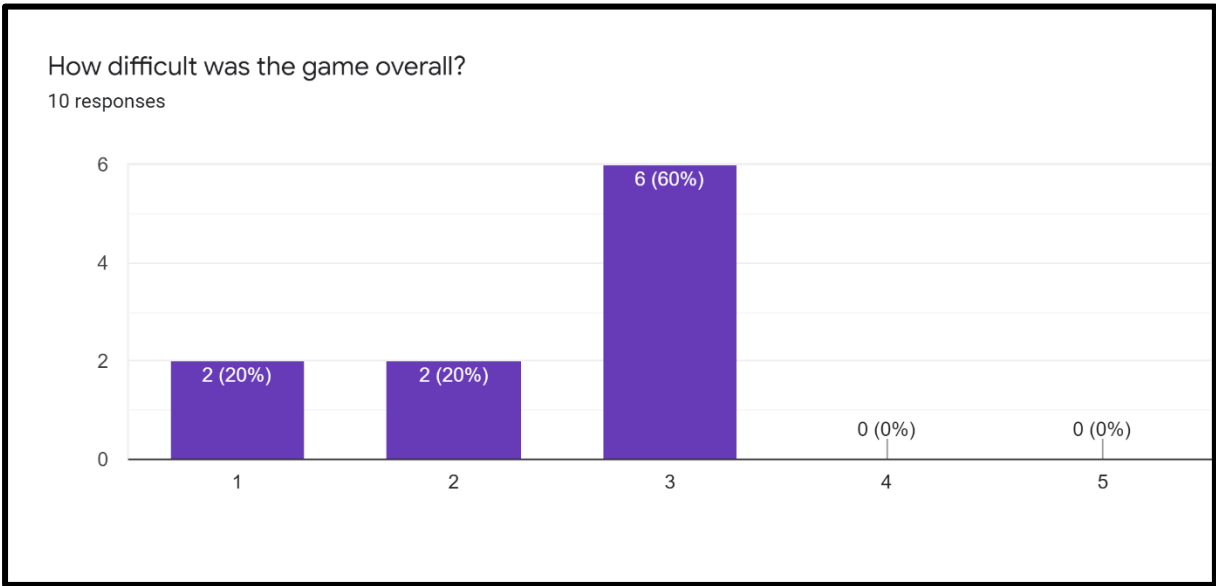
Rocket and laser turrets are upgraded more now, however MG turrets are still not being upgraded, further suggesting that level 1 versions are still too strong.



Heatmaps remained similar to the previous play test.

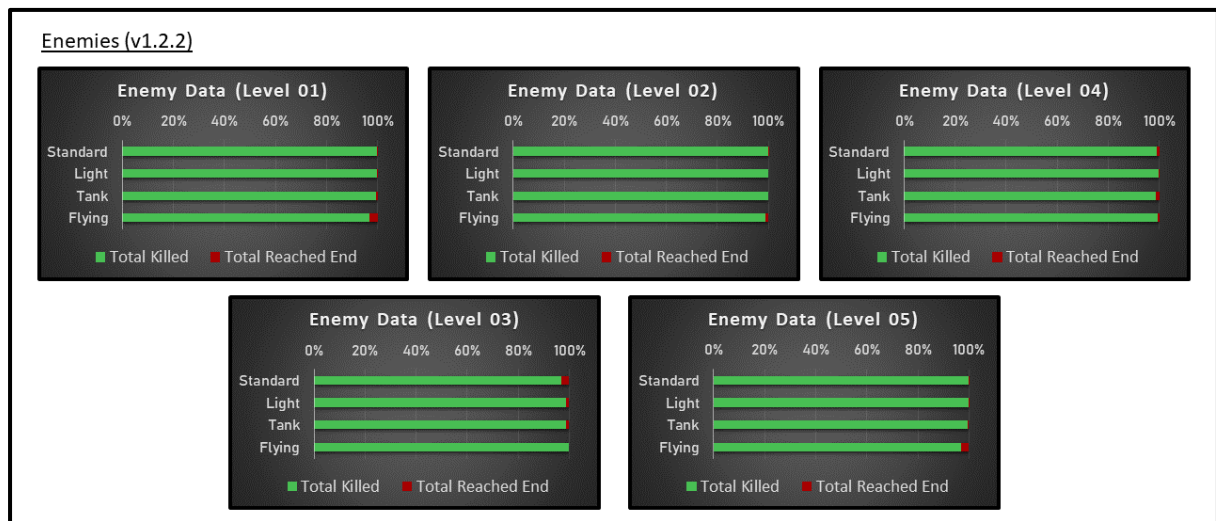


Players are interacting with more mechanics now and the game must have got harder as 40% of players failed a level compared to 0% previously.



Players agree that the game has got harder overall, however still on the easier side, especially for returning players.

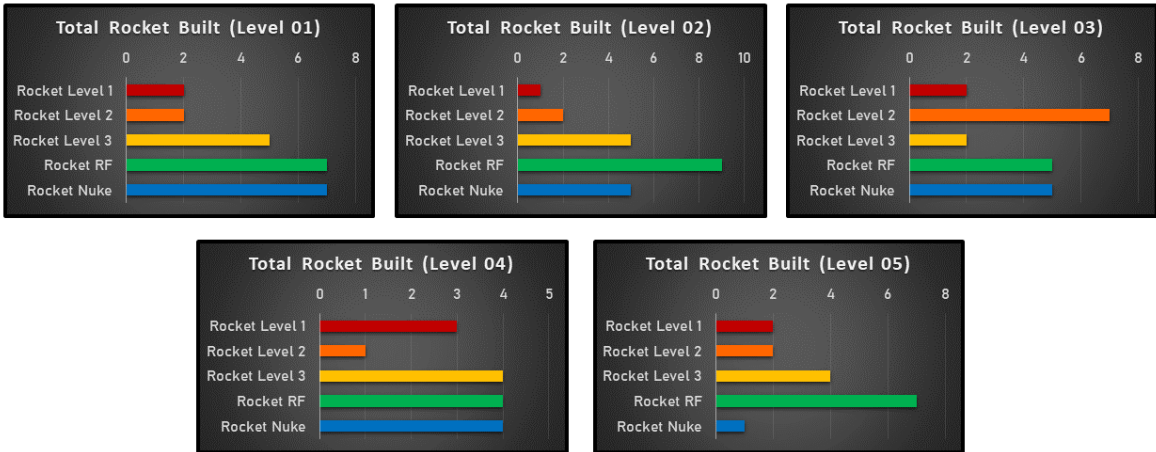
Session #3



No enemies are overly difficult, data shows it is varied per level, likely due to the level layout. For example, flying enemies are the biggest threats on levels where they ignore large portions of the path.

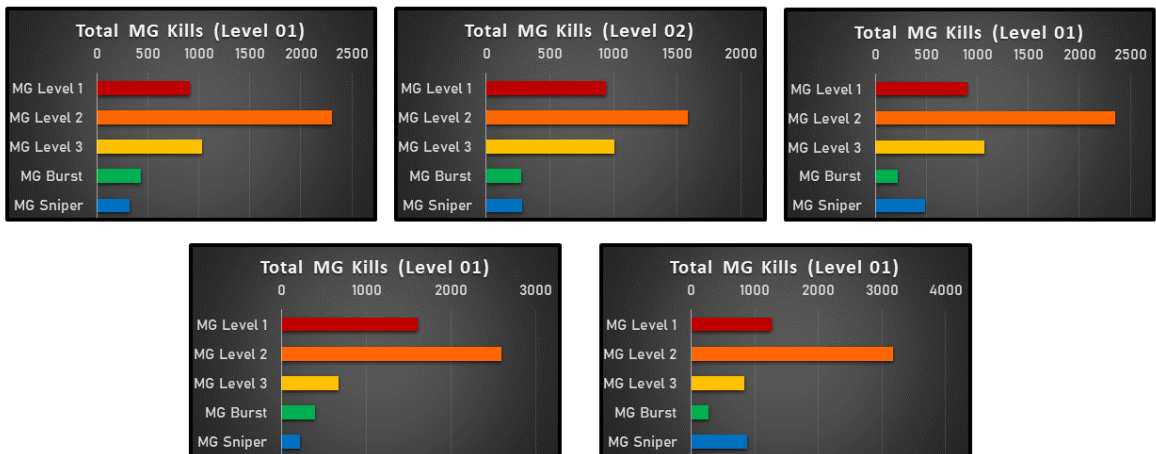


Rocket Turrets Built (v1.2.2)



Players are upgrading turrets much more, especially rocket turrets that were often fully upgraded. MG turrets were usually kept low level still, suggesting differences between each level are not obvious enough.

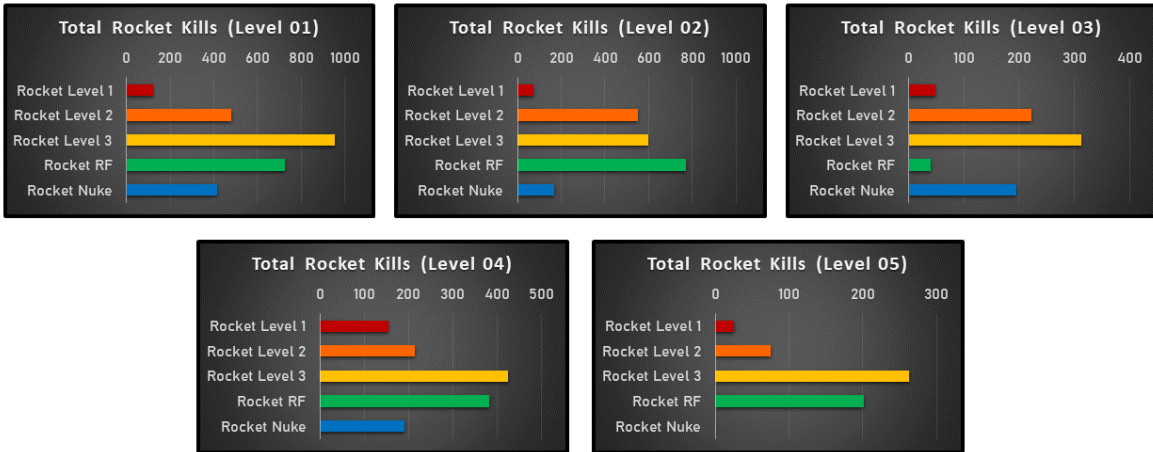
MG Turret Kills (v1.2.2)



Laser Turret Kills (v1.2.2)

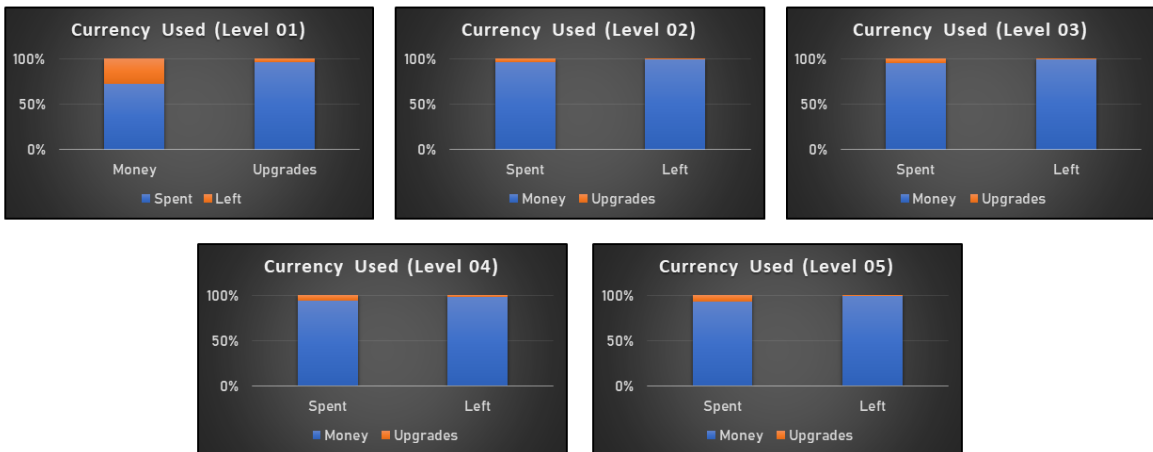


Rocket Turret Kills (v1.2.2)



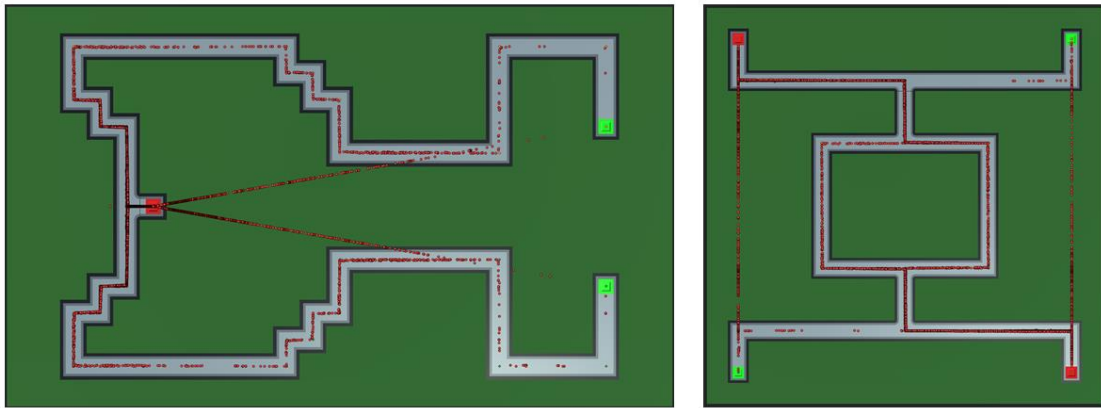
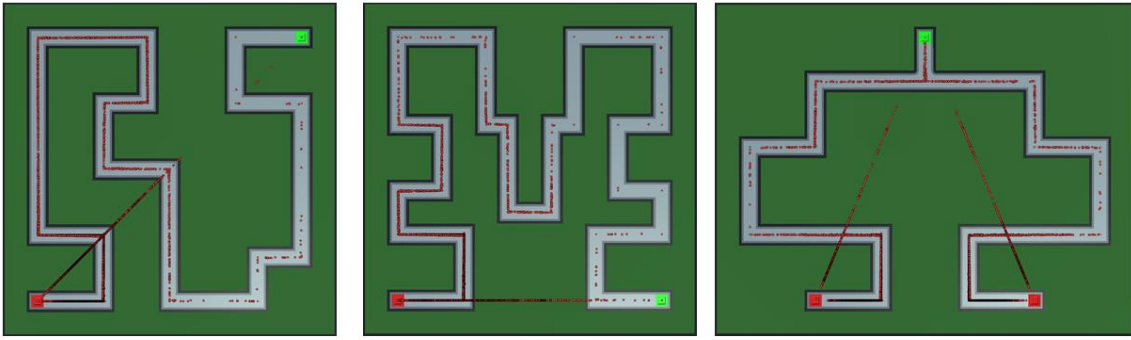
Laser turrets are weaker than other turrets, however they are meant to be utility turrets rather than direct damage. MG turrets make up the bulk of the kills, however this is due to their being more MG turrets than others.

Currency Used (v1.2.2)

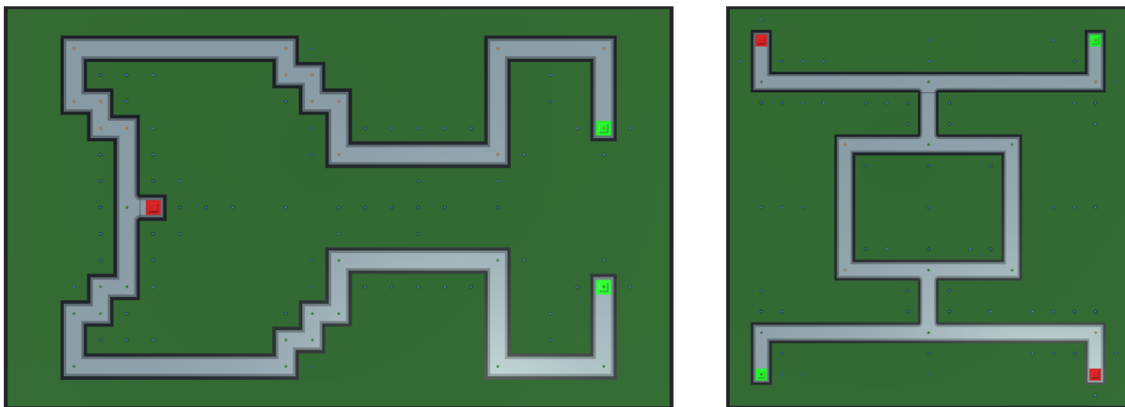
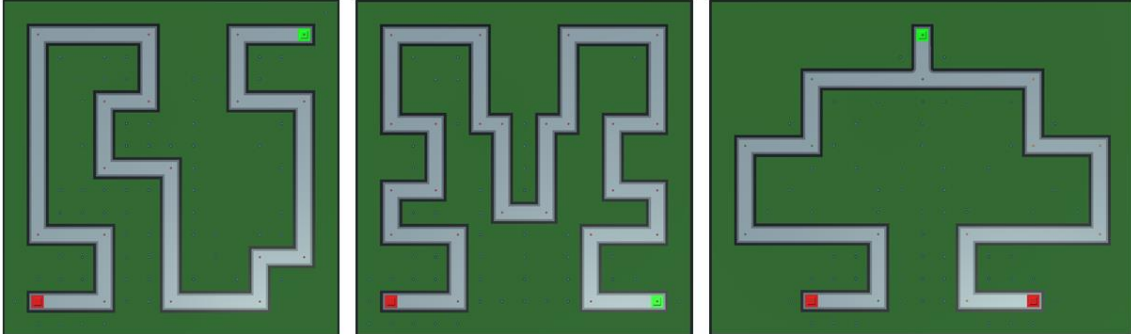


Players are spending the majority of available resources which has made players upgrade turrets much more, which I wanted to be a key feature.

Enemy Deaths (v1.2.2)

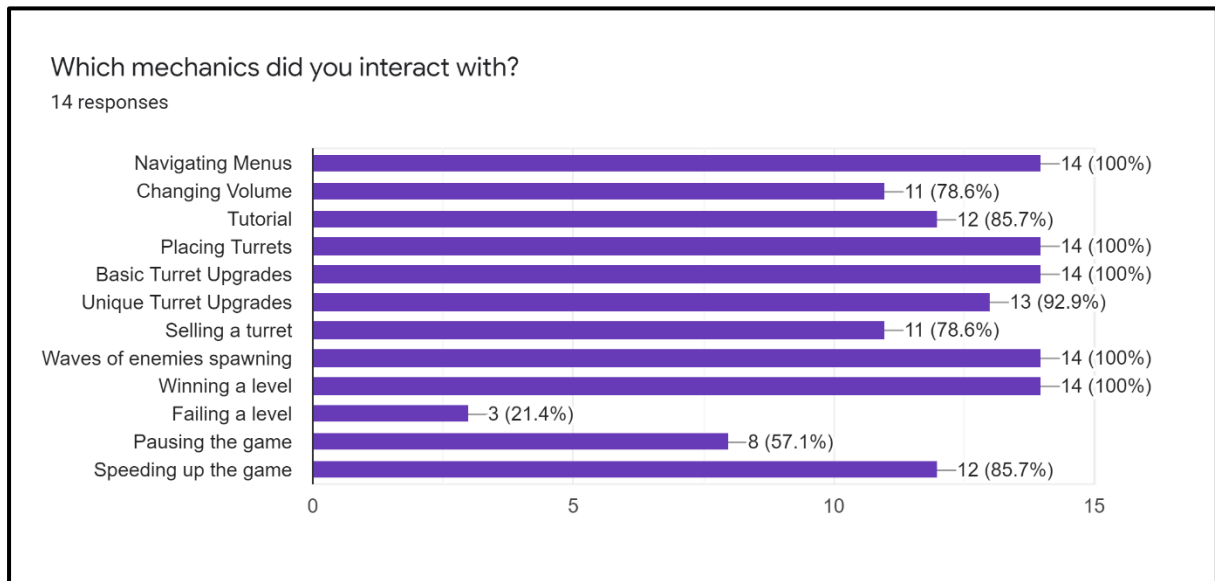


Turret Positions (v1.2.2)

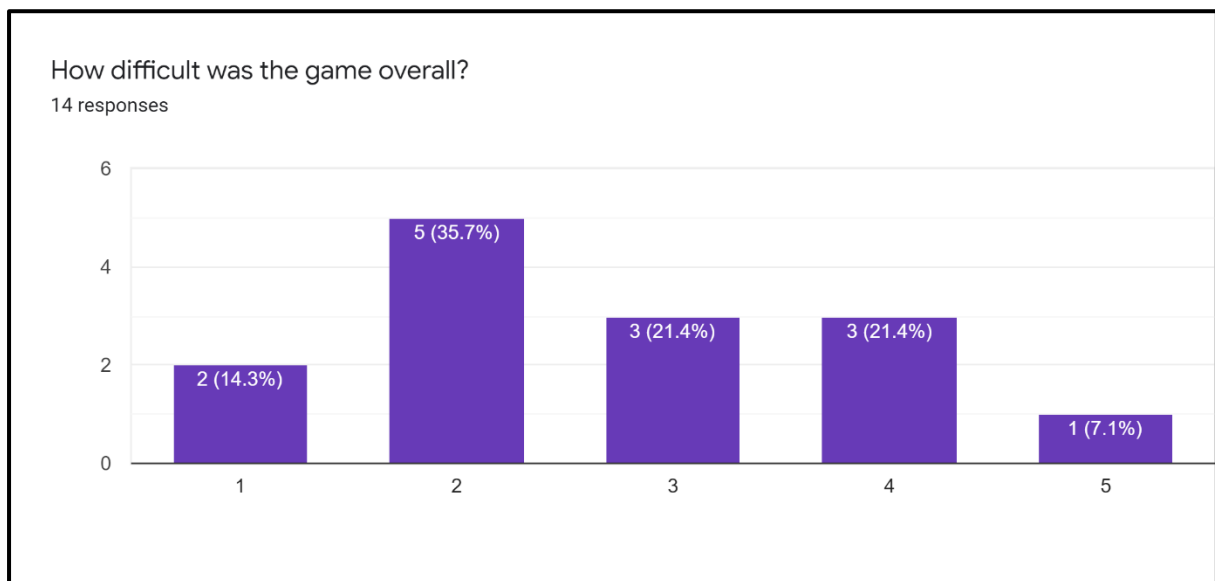


Heatmaps continue to show players focusing turrets near the start of levels, meaning enemies rarely get round the whole level. For levels with split paths that

stay further apart this is less of an issue. On level 04, spawn killing was a huge issue with turrets placed near the single spawn being far too strong.



The majority of mechanics are being interacted with now, with only non-essential mechanics being missed.



Players agree that difficulty has increased again, getting closer to a balanced state. Some suggested that levels should have difficulty ratings and be organised better as some levels are much harder than others.

Players are fans of the new upgrade system and felt themselves upgrading turrets much more as apparent in the data collected.

Prototype Iterations

Iteration #1

Analysing data from the first play test showed the game was too easy, with no players failing the level or losing a large amount of health. The data showed that some turrets were too powerful which could have been causing this. I increased the MG turret cost and swapped the laser and rocket turret costs to try balance them better. I also increased the laser turrets damage, range, and slow rate to make them stronger. To further increase the difficulty, I lowered the amount of money gained when an enemy was killed, to try limit the total turrets players could build.

Players qualitative feedback gave me insight into the games feel and enjoyability. Some players felt rounds were too structured with enemies spawning in blocks. I changed this so enemies now spawn in a random order and each round is more chaotic. Finally, I added some quality-of-life improvements including a selected turret indicator in the shop and visible range indicators when building turrets.

[Video Evidence](#)

Iteration #2

My second iteration had a HUD overhaul and new upgrade system. Data showed that the game had improved, but players were still not upgrading turrets as much as I wanted. I added a different currency to upgrade turrets that should encourage players to upgrade more. I also adjusted the enemy value each round to further limit the number of turrets they could place. Along with the upgrade system I added unique turrets to each turret type where players could choose one of two paths to go down.

Many players highlighted that it was difficult to see the benefit of upgrading turrets, so I added turret statistic to the UI to aid this. I also overhauled the UI, making it more consistent and less floaty. Many new players struggled with flying enemies, particularly on level 02, so to help warn them I added a round preview displaying what enemies were coming up and also sent a smaller number of flying enemies earlier in the level to allow them to prepare defences earlier. Many players requested more content in the game, so I added three new levels. As part of this I restructured the level select menu and made the tutorial more obvious. I created an easier level 01 that encourages turret placement that are better suited to kill flying enemies, to help new players struggling with the enemy type. I tried some new layouts for levels 04 and 05 to see how these layouts would improve or degrade the experience.

[Video Evidence](#)

[Iteration #3](#)

These are hypothetical changes I would make to a further iteration. Despite previous changes, the difficulty is still slightly easy, especially once you get later into the game. Unfortunately, new upgrades have made it too easy again as they are too strong. I would like to implement new UFO types such as a splitting UFO or a boss UFO to create an end wave. To further help increase late game difficulty, I would implement enemy scaling to make it harder as the rounds progress.

A strategy that became clear with data analysis was placing turrets around the spawn areas for the UFOs. In single spawn levels this was particularly strong as all the damage was focused in one area. I'd like to experiment with limiting where players can place turrets to make them think more about placements. I would also like to add a repair mechanic that costs money to force players to upkeep turrets otherwise they would be destroyed. Finally, I would make further UI improvements such as an enemies left counter and improving the turret descriptions, so they are more obvious to players.

[Video Evidence](#)