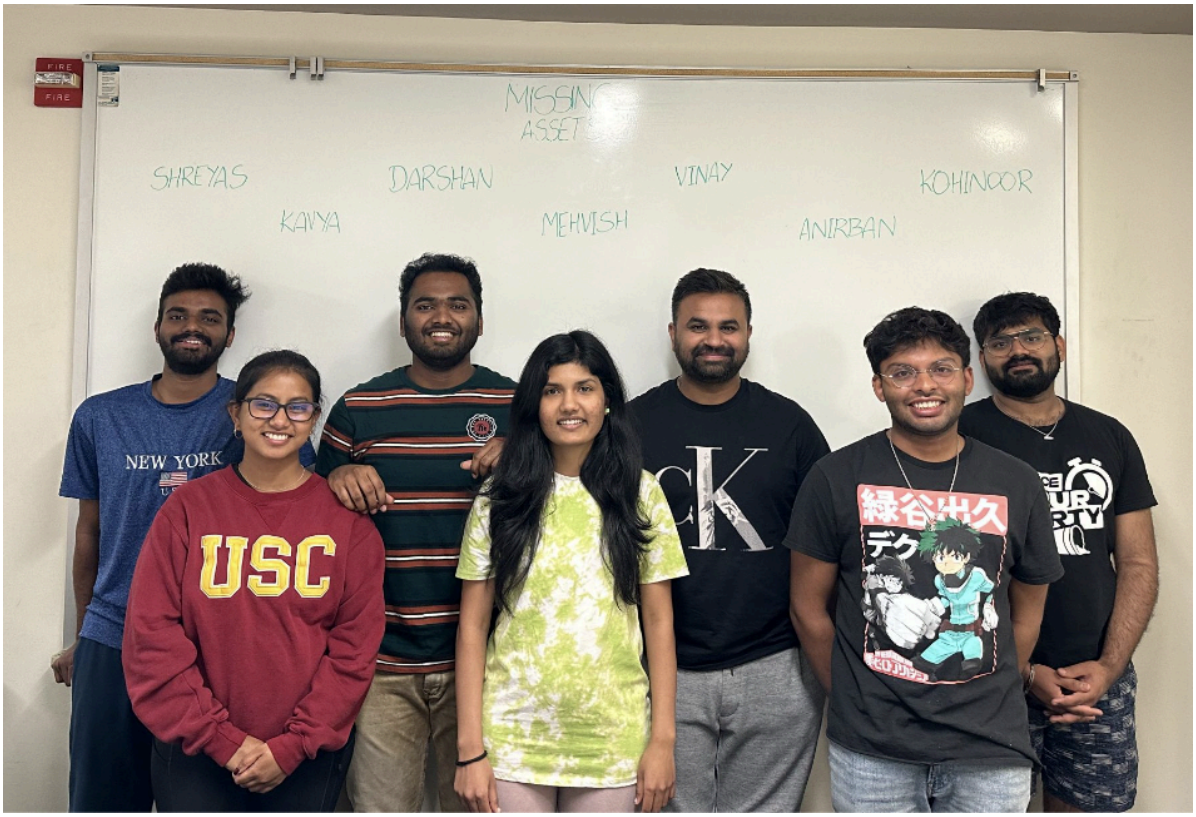
Game Design Document

Team: Missing Assets

ScribbleSpace Shootout

A 2D Space Shooter!

Team



Name	Roles	Email
Darshan Vishwanath	Team Captain, Gravity Engineer	dvishwan@usc.edu
Kavya Ramamoorthy	Gameplay Obstacles Engineer (General)	kramamoo@usc.edu
Anir Mukherjee	Gravity and UI engineer, Notes taker (General)	am28141@usc.edu
Kohinoor Chatterjee	Shooting Mechanics Engineer (General)	kchatter@usc.edu
Vinay Raj	Analytics Engineer (PM)	gowdarar@usc.edu
Mehvish Akhtar	Graphics Engineer (General)	makhtar@usc.edu
Shreyas Seshadri	Enemy Mechanics Engineer (General)	ss05661@usc.edu

Tutorials

Darshan Vishwanath	- Learn Unity - Beginner's G... - How to simulate circular g... - Unity 2D Platformer Tutori...
Shreyas Seshadri	https://learn.unity.com/course/create-with-code How to Simulate Gravity in Unity
Kavya Ramamoorthy	Collision Detection Setting Boundaries Unassigned Reference Exception Debugging
Mehvish AKhtar	The Unity Tutorial For Complete ... Coding Adventure: Solar System Horizontal Movement (Let's Mak...
Vinay Gowdara Raj	How to create Angry Birds in Un... Make your Game Better with Un... How to setup Custom Events for...
Anirban Mukherjee	Introductory Unity Tutorials How to create Angry Birds in Un... Unity Space Tutorial #1 - How to ...
Kohinoor Chatterjee	How to Build a 2D Game in Unity https://www.youtube.com/watch?v=Ii-scMenaOQ&list=PLrnPJCHvNZuCVTz6lvhR81nnaf1a-b67U&ab_channel=CodinginFlow

Important Links

For Class Submission

GitHub Repository	*https://github.com/MissingAssets/MissingAssetsGame*
Prototype WebGL Link	*https://missingassets.github.io/MissingAssetsWebGL/*
Midterm Video	*https://drive.google.com/file/d/15H9RBQWeuKccDPccdkggFzOAN8JAJh1_/view?usp=share_link*
Midterm WebGL Link	*https://missingassets.github.io/MissingAssetsWebGLWeek9/*
Final Video	*https://drive.google.com/file/d/1hMyOgv5iLKrsTAxAM0WzSorNRCnhxRGg/view?usp=sharing*
Final WebGL Link	*https://missingassets.github.io/MissingAssetsWebGLWeek15/*

Game Overview

Logline

A scribbled 2D space shooter with orbital drifting and fun boss battles.

Short Description

This game borrows from traditional arcade style space shooters but adds one fun and interesting core game mechanic: gravity. This game takes place in a scribble sheet of a bored kid in class. The player can orbit around celestial objects and use them to increase speed or to trap the enemies and kill them. The goal of the game is to complete each level by collecting all the collectibles, and most traditionally a deathmatch with the enemy shooters throughout the level. As players progress through the game they will find new abilities and shields with which they can strategize and develop unique play styles. Ultimately, the most fun component aims to be the gravity themselves as the player can make use of them in clever ways to make quicker work of the enemies.

Other Key Items

- ➔ **Target Audience:**
  - Mobile gamers looking for fun action games to pass time

- ➔ **Game Mechanics:**
  - Talk about how the game is different from other games in the genre
  - Significant mechanics needed for each member
  - Mechanics to be explored in multiple levels
  - A mechanic for guidance

- ➔ **Innovation/Unique Selling Points:**
  - Space shootout with actual gravity mechanics
  - Slingshot using gravity
  - Space obstacles
  - Overpowered aliens!
  - Different enemy weapons
  - The premise! Its happening on a notebook of a bored kid

- ➔ **Key Features:**
  - Gravity
  - Different types of enemies

- ➔ **Genre:**
  - Arcade space shootout

- ➔ **Platforms:**
  - Laptop

Obstacles

- ➔ Meteors
- ➔ Gravity
- ➔ Henchmen enemies
- ➔ Boss enemies

Powerups

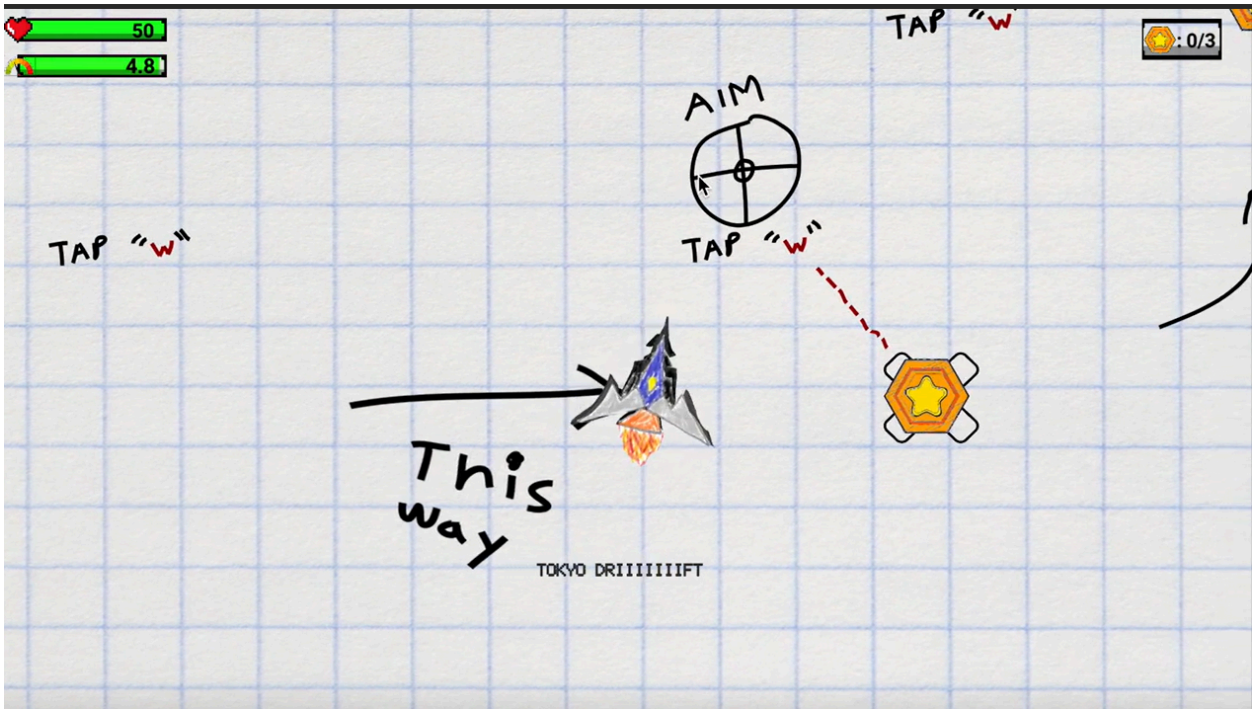
- ➔ Shield

Detailed Design

Game Elements

Thrust and Player Movement

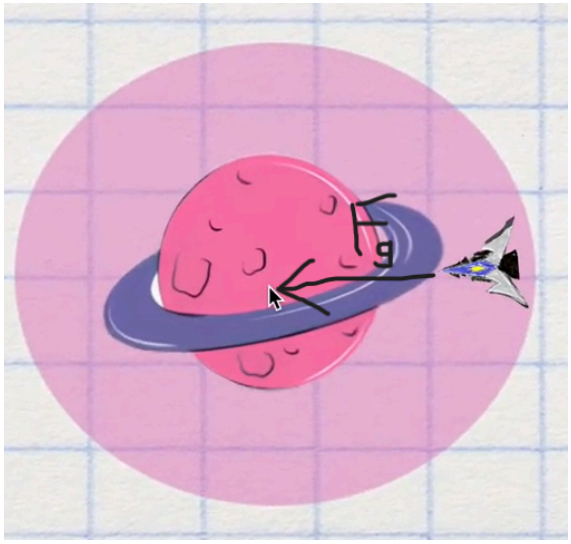
There are no frictional or reactionary forces acting on the player in this game while moving, and the movement itself is based entirely on the forces acting on the player, be it through thrust/collisions/gravity, etc. The player has a limited amount of thrust that they can use in a certain direction. Thrust has a cooldown period and recharges latently. Thrust is applied in the direction opposite of where the player is facing (the direction of the mouse cursor), essentially making the player move toward the mouse.



Gravity Fields

Planets are game objects that exert gravitational force on a sprite that enters a certain radius from the center of gravity. The gravity is not entirely Newtonian, but rather exerts a constant amount of force on the sprite it's acting on towards the center of the planet. This kind of behavior allows players to glide effortlessly across planets and slingshot thereby preserving their thrust.

$$F = \frac{G m_1 m_2}{d}$$



Collectibles

These are the crucial objectives of the game. The player must collect all the collectibles in order to progress to a new level.

Gunplay

The player aims in the direction of the mouse cursor and fires with the left mouse button. The blasters have a cooldown. Enemies can return fire and can be either stationary or moving targets. Guns are projectile weapons that ignore the effect of gravity. Players take damage to their shields first (which recharge) and once shields are depleted they take hull damage (does not recharge).

The player has only one type of gun which shoots a comet-like bullet.

The enemies have 3 types of guns which are

1. Basic gun
2. Multiple shooting guns
3. Missiles

Enemies:

Types of enemies

1. Basic Enemy

The basic enemy shoots a configurable amount of bullets in a given time frame and has a configurable amount of bullets needed to be killed. Each enemy bullet damages the player by a magnitude of 5.

2. Fixed Axis Moving Enemy

They are the same as basic enemies except that they are able to move in a fixed axis either x or y with a configurable speed.

3. Follow Enemy

Same as basic enemies but they follow the player wherever they go given that they are triggered by the player going within a certain configurable distance of the boss enemy.

4. Boss Enemy

The boss enemy is an enemy which the player must defeat in order to progress to the next level. The major configurations that can be changed are the following

- Health
- Missile speed
- Fire rate
- Movement speed of boss
- Missile damage

Powerups

The player can collect power-ups, one of which is the overshield. This gives the player temporary invulnerability. This power-up is activated when the player collides with a capsule sprite, which then activates a shield “halo” around the player. This “halo” indicates that the player is now invulnerable, and will be inactivated after about 7 seconds. During this time, any collision of the player object either with bullets or asteroids will not lower its health value.

Obstacles

Asteroid obstacles were placed in various places of the map inorder to add challenges to the maneuverability of the player.

Gaming Mechanics

Gravity

The gravity mechanic is one of the core mechanics of our game. The idea was that the planet exerts a force on the player towards the center of mass. Initially the governing equation of the force was the exact newtonian gravity

$$F = \frac{G m_1 m_2}{d^2}$$

This was found to be challenging to play as the motion of the player around the planet will be elliptical. We then experimented with different variations of the equations and found the best to be

$$F = \frac{G m_1 m_2}{d}$$

In this the force depends inversely not on distance squared but on distance. We found this equation easier to fine tune to design levels and easier to play without losing the fun of gravity.

Thrust (Control: ‘w’)

Traversing the entire map of the game only with gravity was difficult and hence a thrust mechanic was added. This was another core game mechanic that was added to the game. The thrust works by providing an impulse force on the player in the direction of the mouse. Conditions of long press of thrust control were taken care of by limiting the force given to the player in a specific time frame.

Thrust also recharges when not consumed. This was initially done by giving a ratio of consumption and recharge which is configurable. We configure 4 parameters which are consumption amount, consumption time, recharge amount and recharge time.

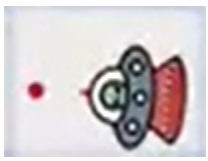
The player can also use reverse thrust by turning the mouse in the opposite direction thereby exerting a force in the opposite direction and reducing the velocity

Shooting (Control: Spacebar, ‘e’, Left mouse click)

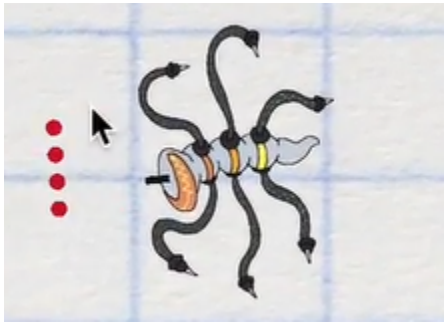
The other core mechanic in our game is shooting and interaction with the enemy. In order to make the gameplay interesting, different types of bullets and enemies were incorporated into the level design each with different characteristics and difficulty levels. All of these types of shooting have a configurable fire rate and a range. Friendly fire between enemies has been disabled, meaning that bullets fired from enemies cannot damage other enemies. All the bullets are instantiated from a “firepoint”, an empty point sprite a certain configurable offset distance from the enemy.

The main types of shooting involved are

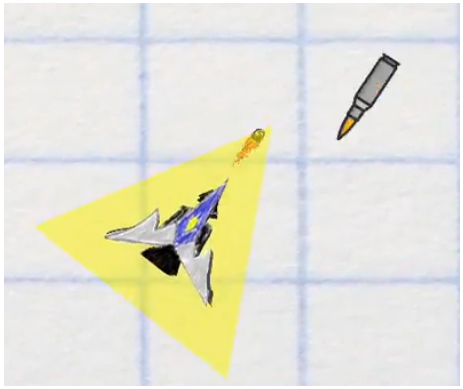
1. Normal Shooting: A bullet comes out at a configured speed and range and destroys itself on collision with the player or a non-enemy object. These bullets do not follow the player and only travel in a straight line from the point of instantiation.



2. Machine gun: The machine gun spawns configurable number of bullets equidistant



3. Missiles: A bullet that spawns at firepoint, gradually increasing speed from its initiation till a particular cap of speed is reached or it collides with an object. It also has an agent subroutine part of it that follows a target given to it.



Enemies

Inorder to make the gameplay more fun different types of enemies were introduced. Our core game design incorporates various choices to the player by either engaging enemies or going through complex maneuvering. The main types of enemies are.

1. Normal enemies: These are the basic enemies which are fixed to their initial location and fire any type of bullet.

2. Fixed axis moving enemies: These types of enemies move in a fixed axis, either x or y (configurable) to a certain distance from the initial point.
3. Follow enemies: These enemies follow the player if they get within a configurable upper bound and lower bound of distance from the enemy.
4. Boss Enemy: This enemy has to be defeated in order to progress to a new level. The boss enemy has very high health compared to normal enemies. They randomly fire missiles, machine guns and normal bullets at the player.

Braking (Control: 's')

After feedback from players, it was found that reverse thrust was not enough to slow down the player leading to out of bounds death and difficulty in maneuvering quite often. This was the reason for adding the breaking mechanic to the player, to increase the control the player had on the movement. Breaking initially worked by adding a configurable amount of force in the opposite direction of the current player velocity direction. When the velocity of the player is close to zero, it is forced to zero. Later on it was changed to Linearly interpolation of velocity from current velocity to zero.

Gravity traps

In Order to make better use of gravity in the game play, a mechanic of trapping following enemies in gravity was added. The gravity trap, when enabled, works on the boss enemy as well as following enemies. Since these enemies follow the player, they can be lured by the player into the gravitational field of one of the planets after which their movement and shooting abilities are disabled, making it easier for the player to kill them.

Asteroid Damage

Inorder to penalize players from avoiding hitting the asteroids, we added an extra mechanic to give damage to the player on collision. The damage given to the player is directly proportional to the velocity of collision with the obstacle multiplied by a configurable factor. We also add a rebound force in the direction opposite to the direction of collision.

Guidance Triggers

Inorder to guide the players at the right time circle colliders were introduced in certain areas of the map which when crossed by the player displays a dialog with appropriate guidance message.

Shield

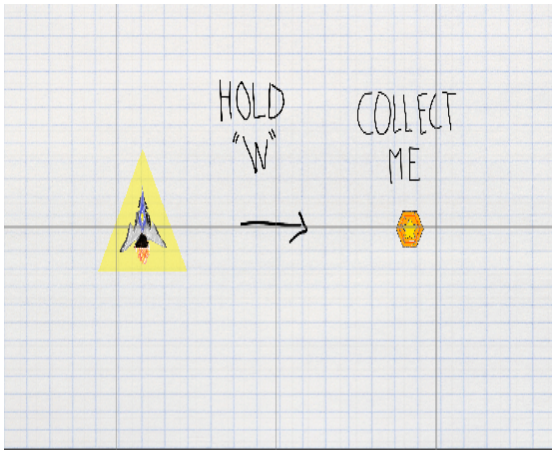
A powerup was added in order to assist players in difficult levels. This powerup makes the player invulnerable to any kinds of damage for a configurable period of time.

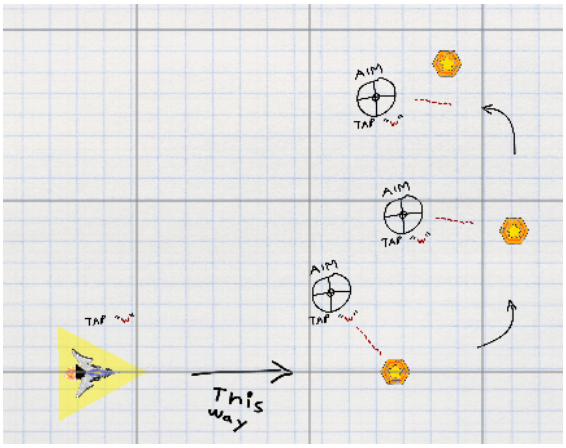
Levels

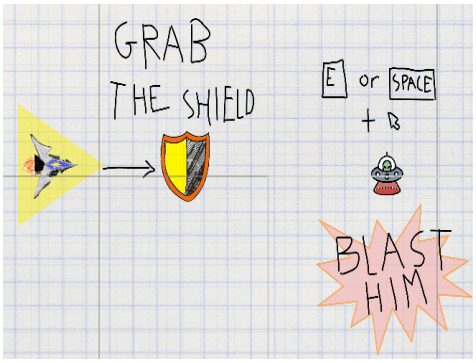
There are totally 13 levels in the game and each level and these levels have been broadly broken down into 4 subcategories based on the objective of its design and the game elements used in designing the level.

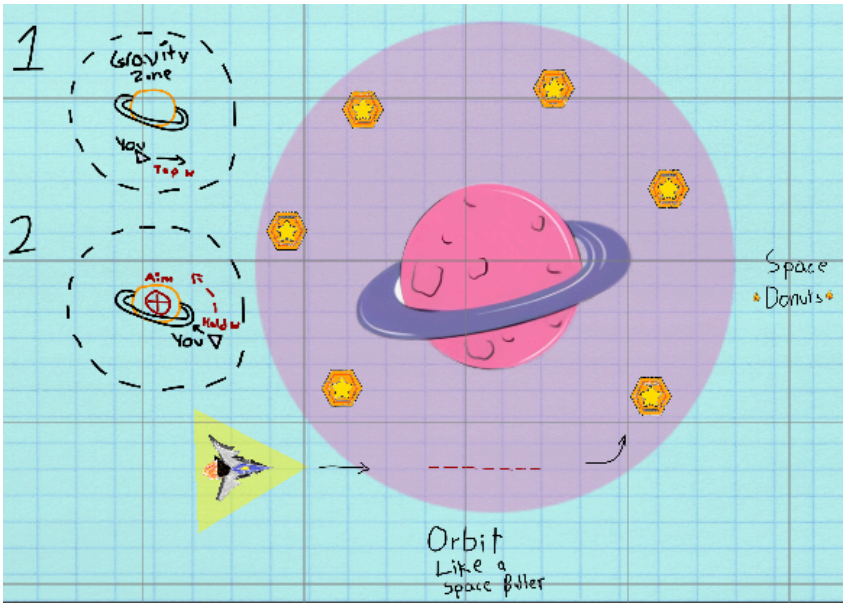
Tutorial Levels

These levels are introduced inorder to train the player with controls and strategies which are used in later levels. They are designed to be simple and only to train the user to use a specific strategy without incorporating multiple game elements unlike a challenging level.


HOLD
"W"
COLLECT
ME

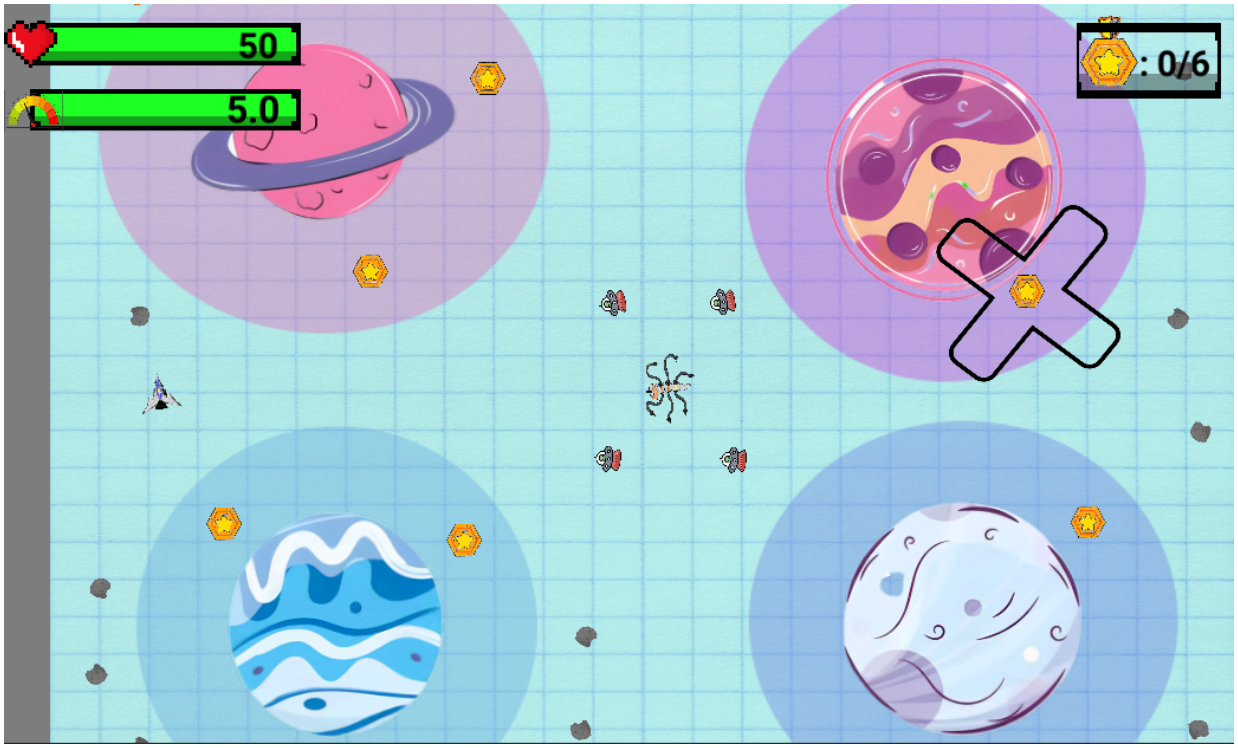

AIM
TAP "W"
AIM
TAP "W"
AIM
TAP "W"
TAP "W"
This
way

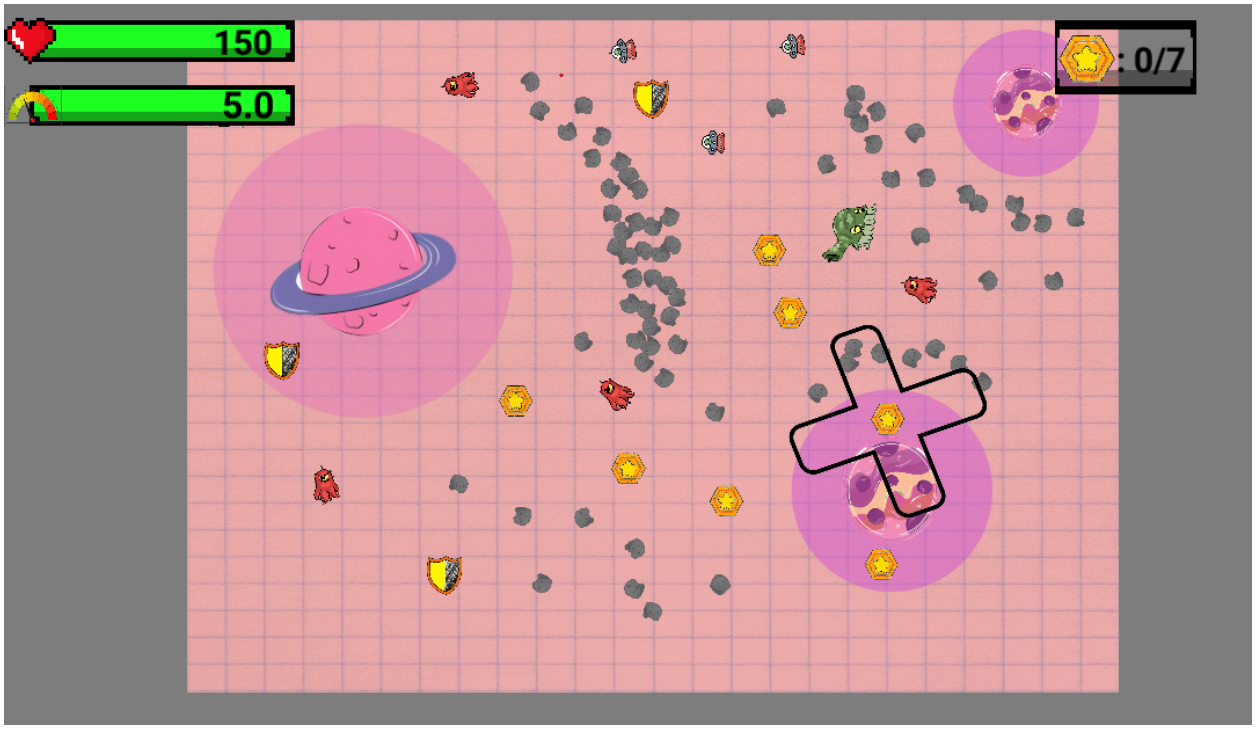

GRAB
THE SHIELD
E or SPACE
+ B
BLAST
HIM


1
Gravity
Zone
You
Top W
2
Aim
Hold W
You
Space
Donuts
Orbit
Like a
Space Ball

Skill Levels

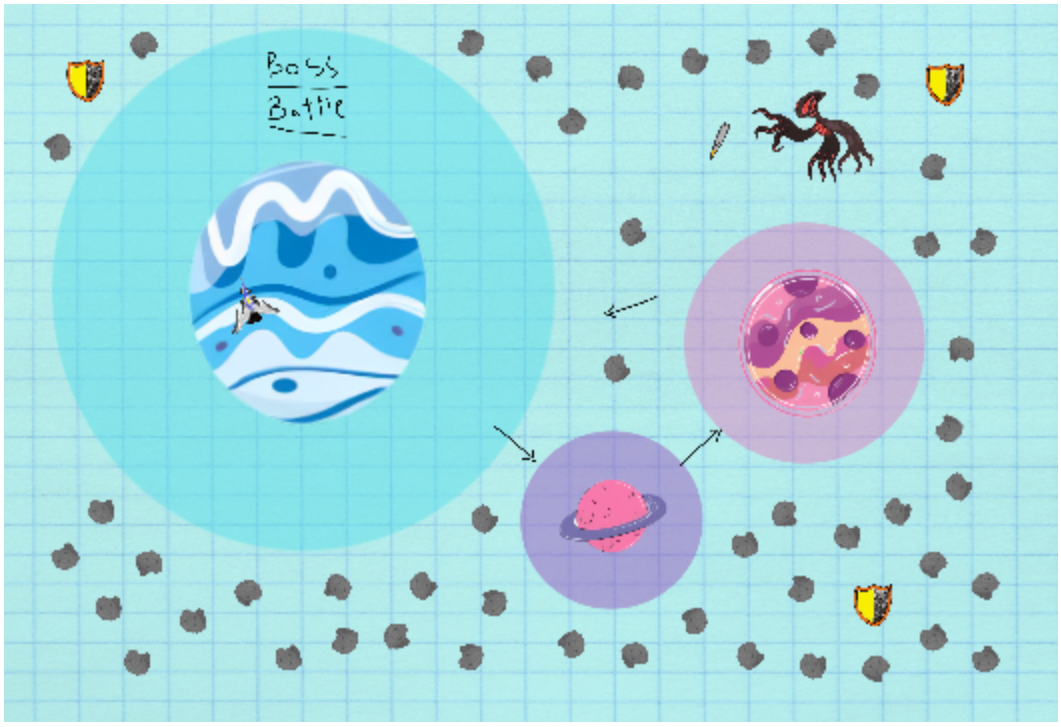
These levels incorporate multiple game elements and test the player on the strategies learnt in the tutorial games. Some of the level design images are given below.

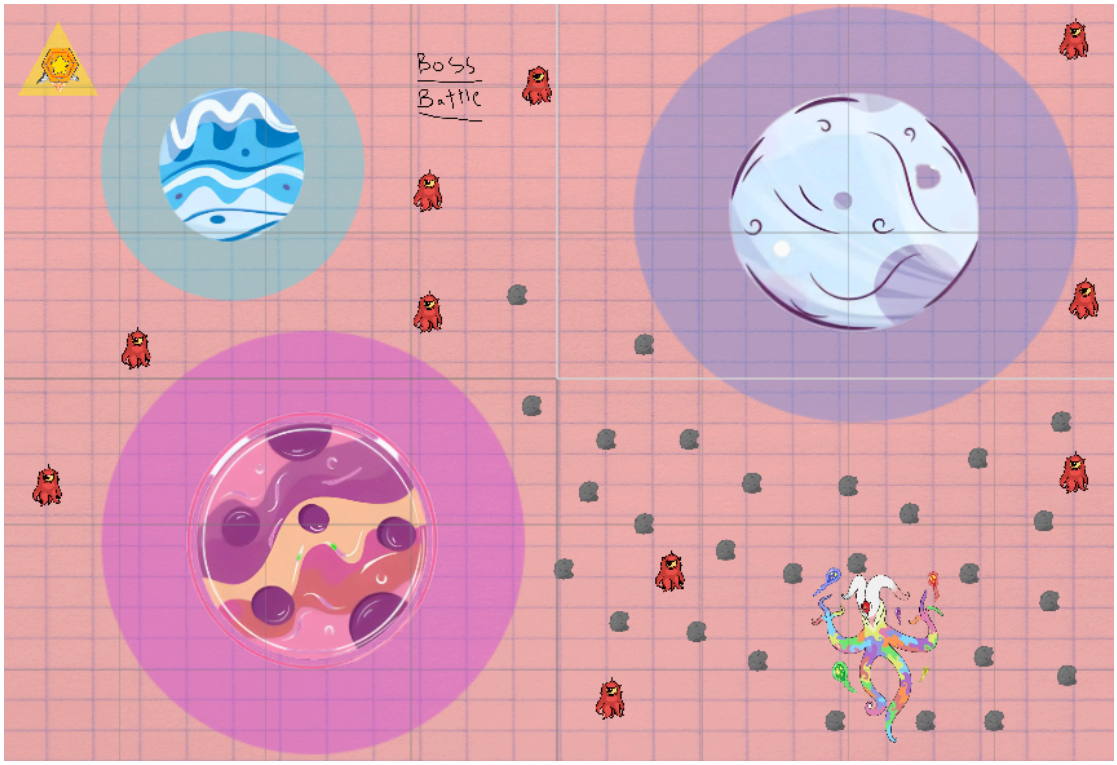

50
5.0
: 0/6


150
5.0
: 0/7

Boss Enemy Levels

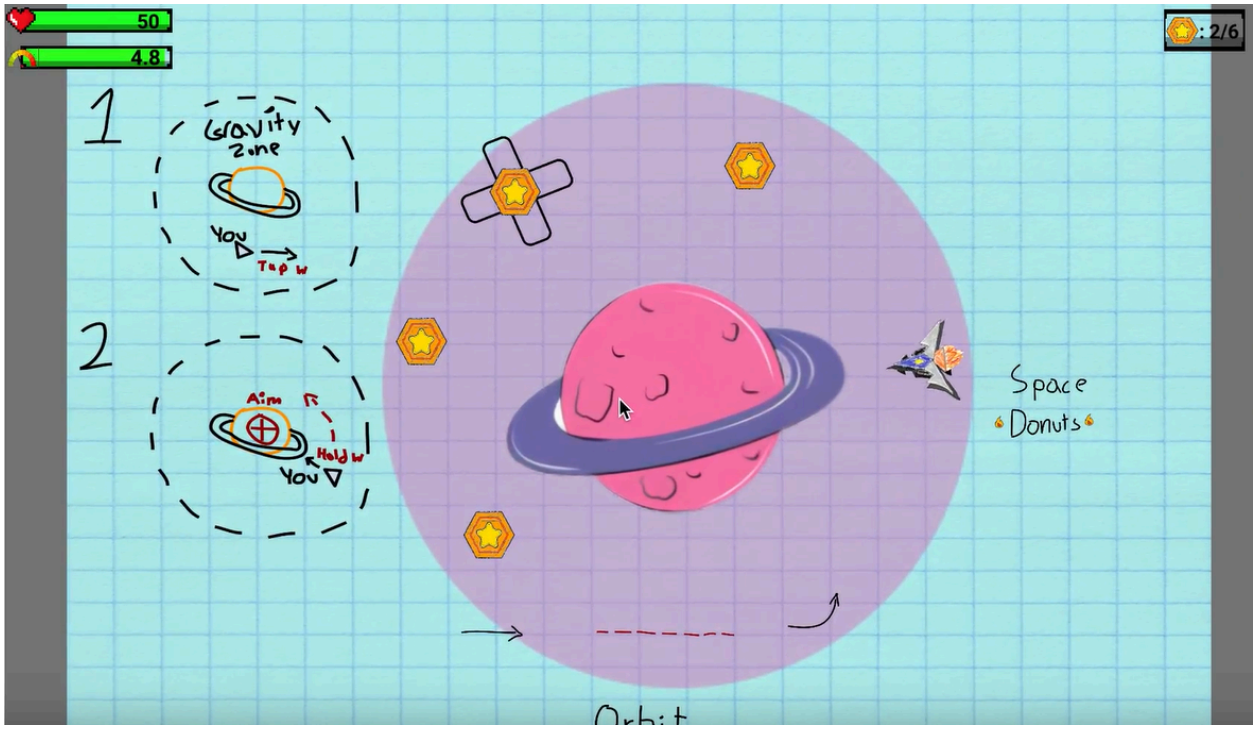
These levels are the levels that involve boss enemies which must be killed in order to progress to the next level. The boss enemy levels have been designed in such a way as to give the player choices on when to start engaging the boss enemy as they do not follow the player unless they get to a certain distance from the boss.


Boss
Battle


Boss
Battle

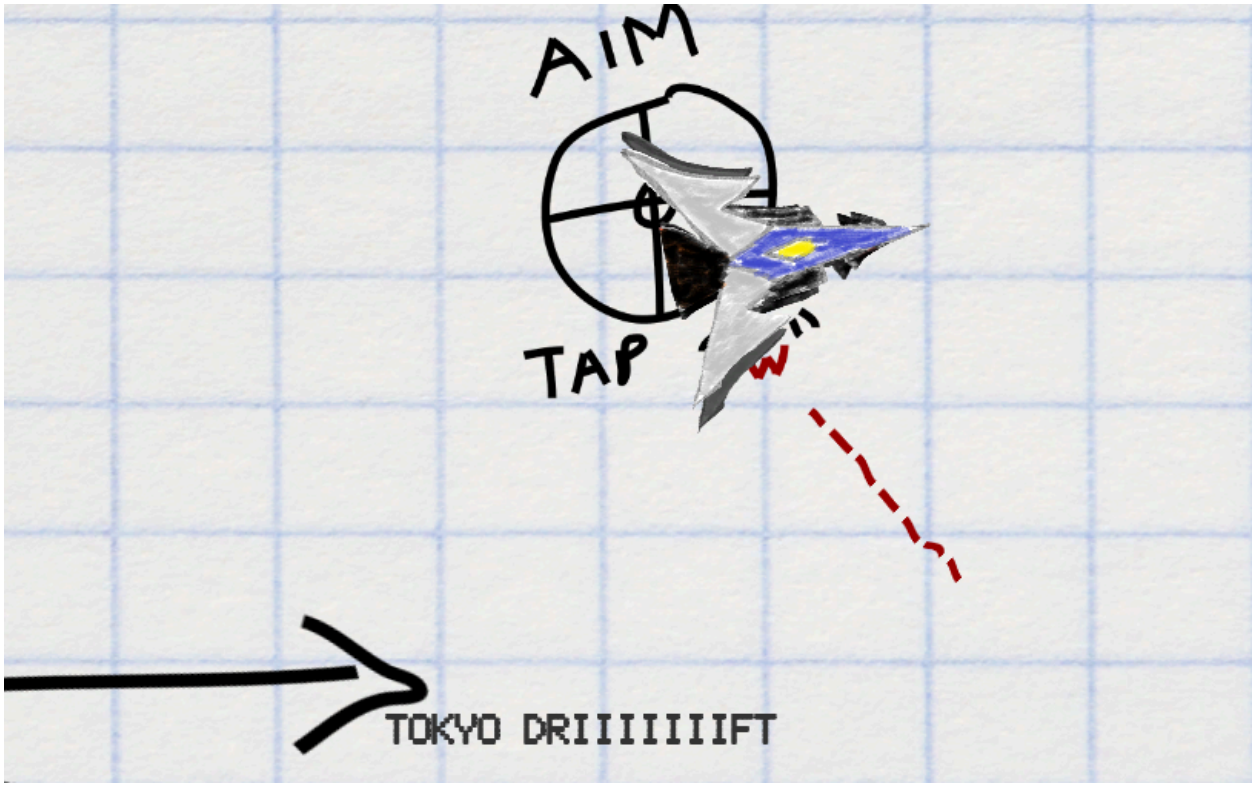
User Interface

General Elements

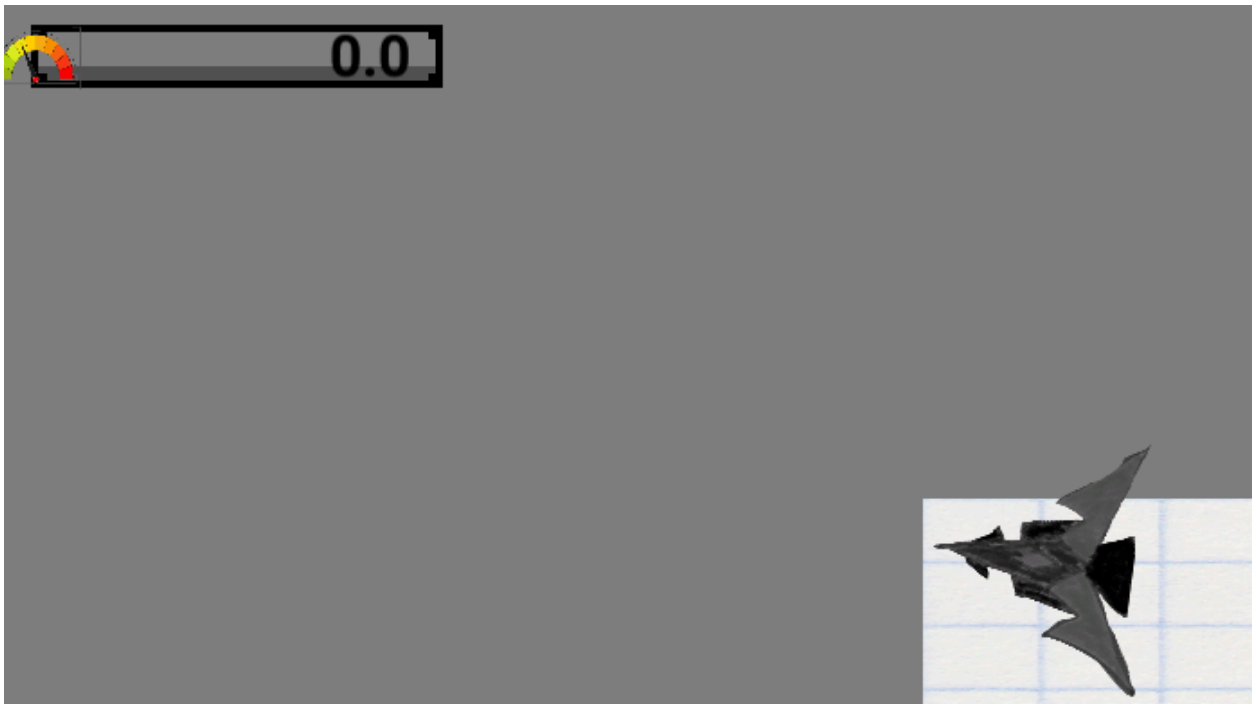


- The top left corner shows the health bar and fuel bar
- The top right corner shows the collectibles collected so far and the total collectibles needed inorder to complete this level
- We have 6 collectibles to be collected on screen and one of the collectibles has a rotating cross which indicates that the player has to collect it.
- We also provide guidance of movement on the screen as shown by the doodles on the left of the planet

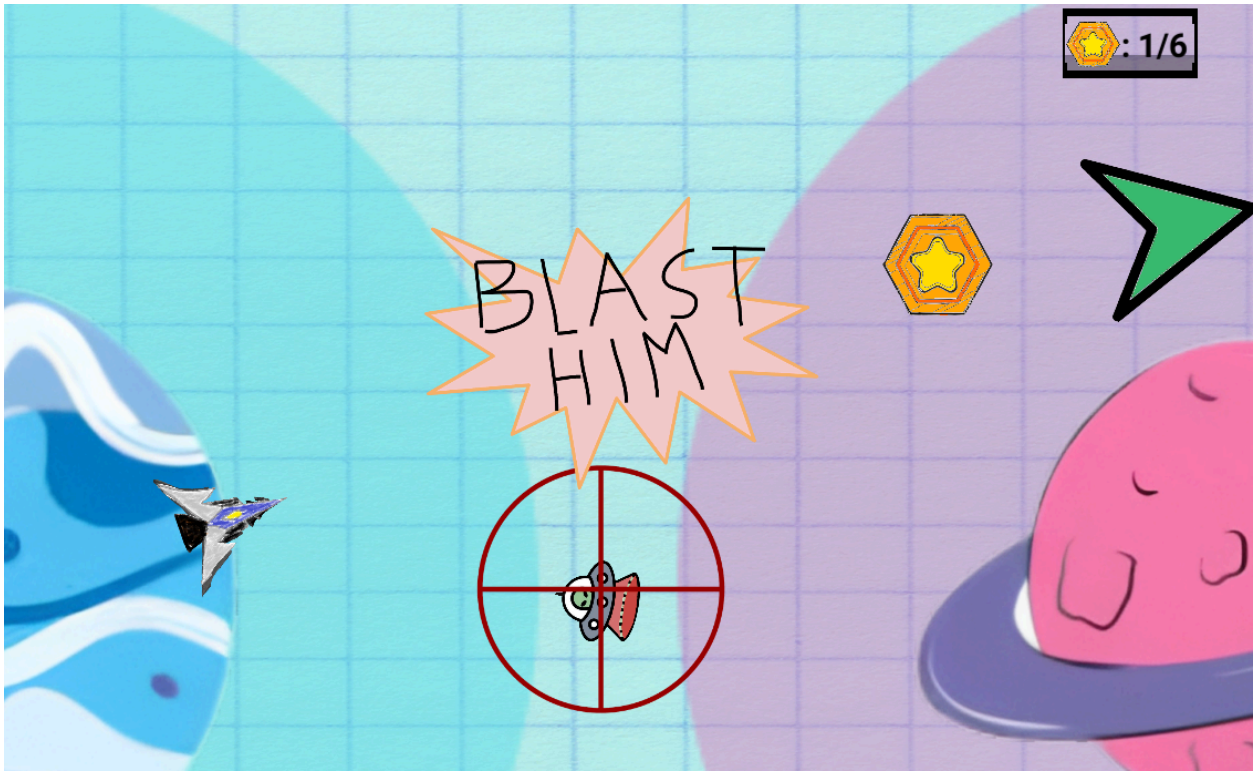
Guidance Elements



We also provide dialogues with useful information at certain points as shown in the bottom of the above image.



Whenever the thrust bar is zero the user is alerted by the fact that the rocket ship is grayed out.

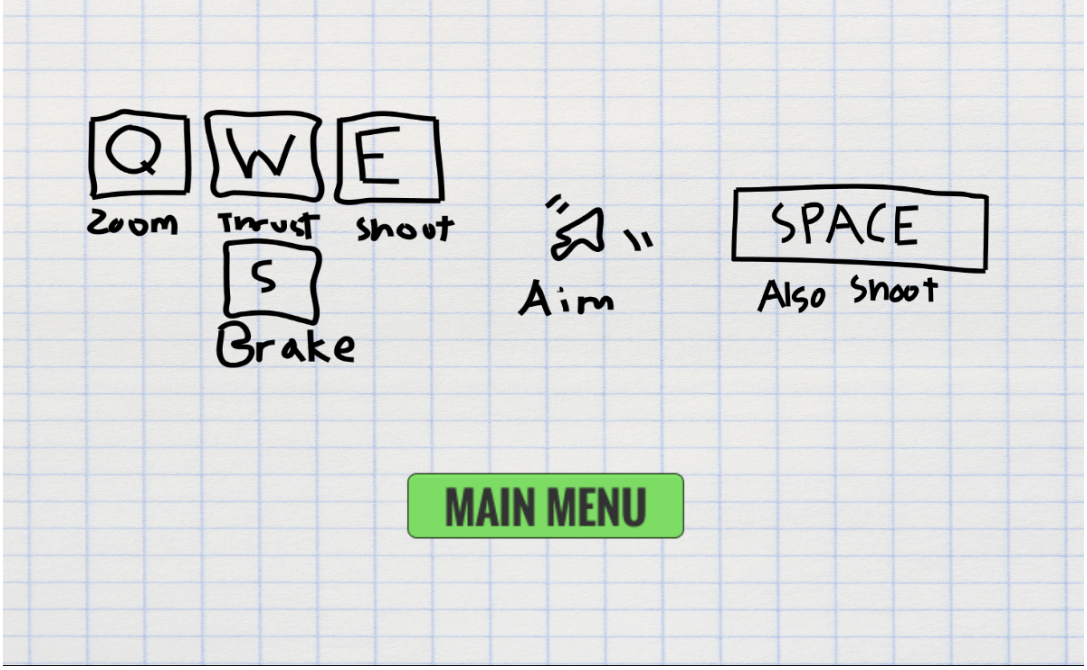


When the collectible is not in sight we guide the player towards the collectible by showing a quest pointer which is a green pointer shown towards the top right corner of the above image.

Menu and controls

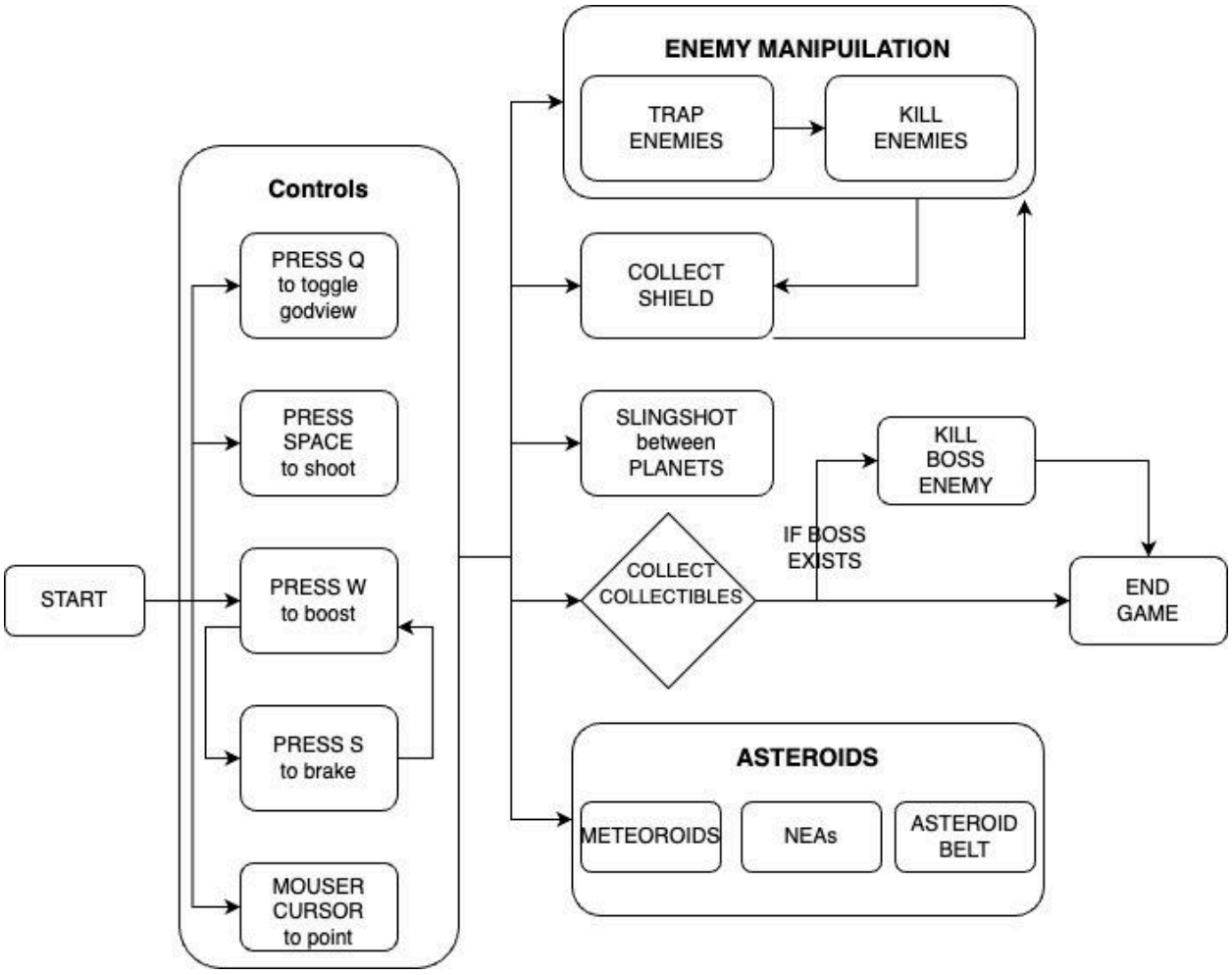
Landing screen and main menu

Controls menu where we show all the controls



Level menu where we show all the levels

Game Loop


ENEMY MANIPUILATION
TRAP ENEMIES
KILL ENEMIES
Controls
PRESS Q to toggle godview
COLLECT SHIELD
PRESS SPACE to shoot
SLINGSHOT between PLANETS
KILL BOSS ENEMY
START
PRESS W to boost
COLLECT COLLECTIBLES
IF BOSS EXISTS
END GAME
PRESS S to brake
ASTEROIDS
METEOROIDS
NEAs
ASTEROID BELT
MOUSER CURSOR to point

Project Progress

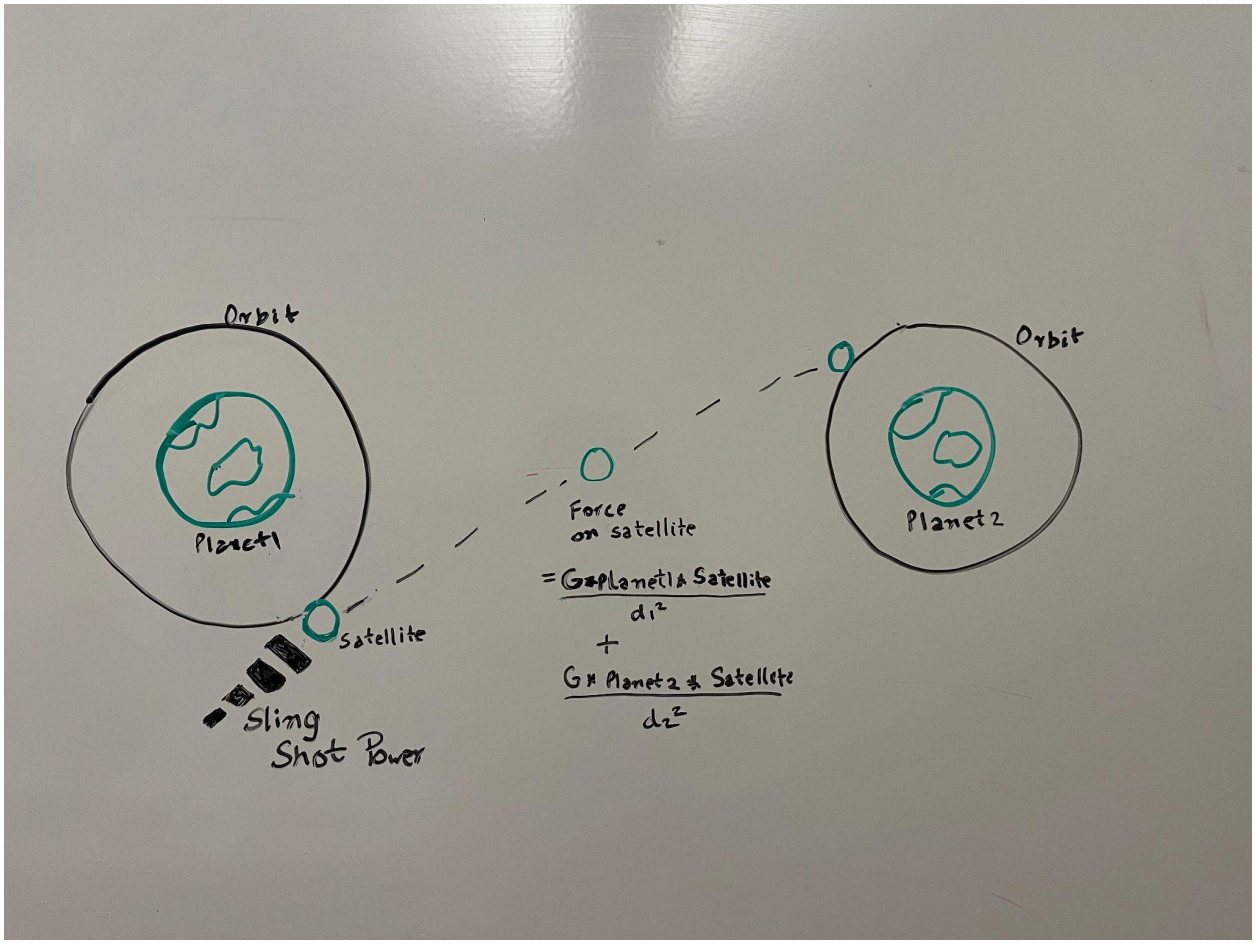
Genre: 2D Space Shooter

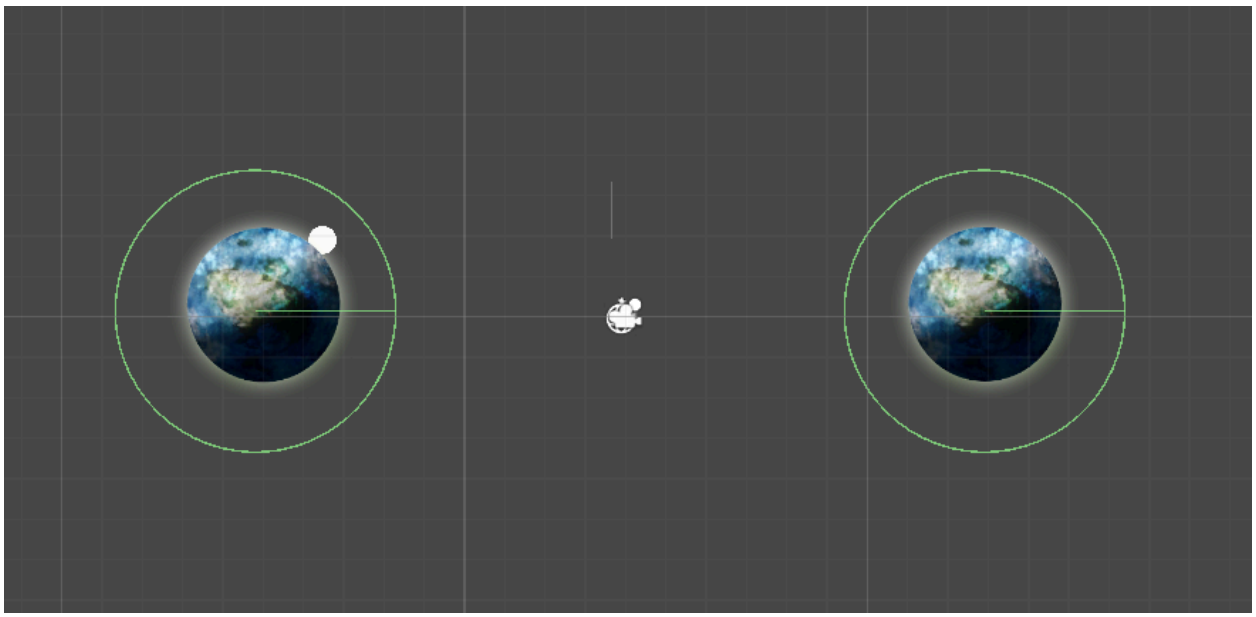
Prototypes

Prototype 1

Description: Prototype 1 verifies the concept of slingshotting under the influence of gravity (governed by the Universal Law of Gravitation). Player sling shots from one planet and reaches the orbit of the other planet. If the slingshot lands the player very close to the orbit of the planet, the player enters the orbit and revolves around the planet, or else the player escapes

Drawing:





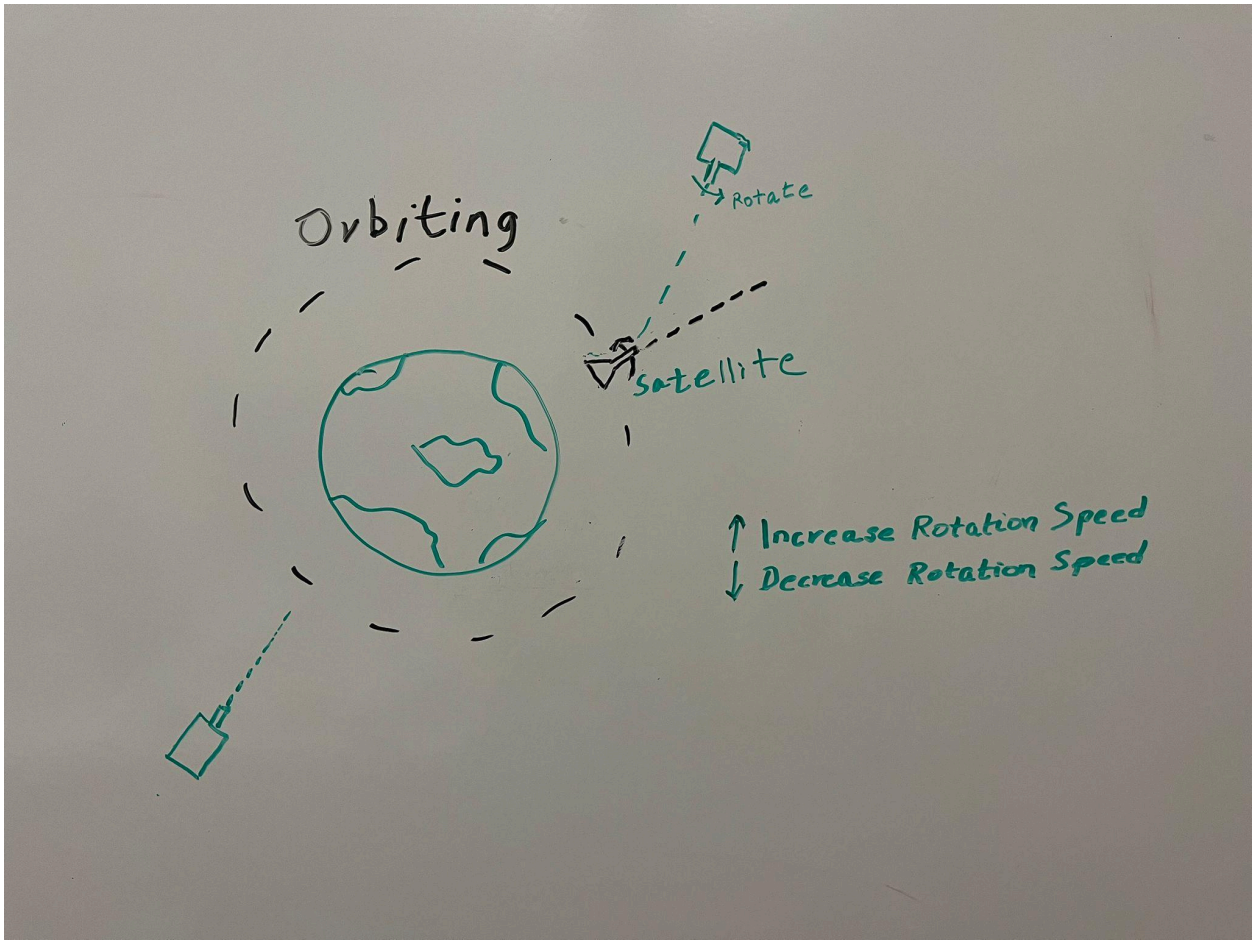
The player, if slingshots using the right angle, will revolve around the predefined orbit.

Link: https://kohinoor98.github.io/MissingAssets-P1/

Prototype 2

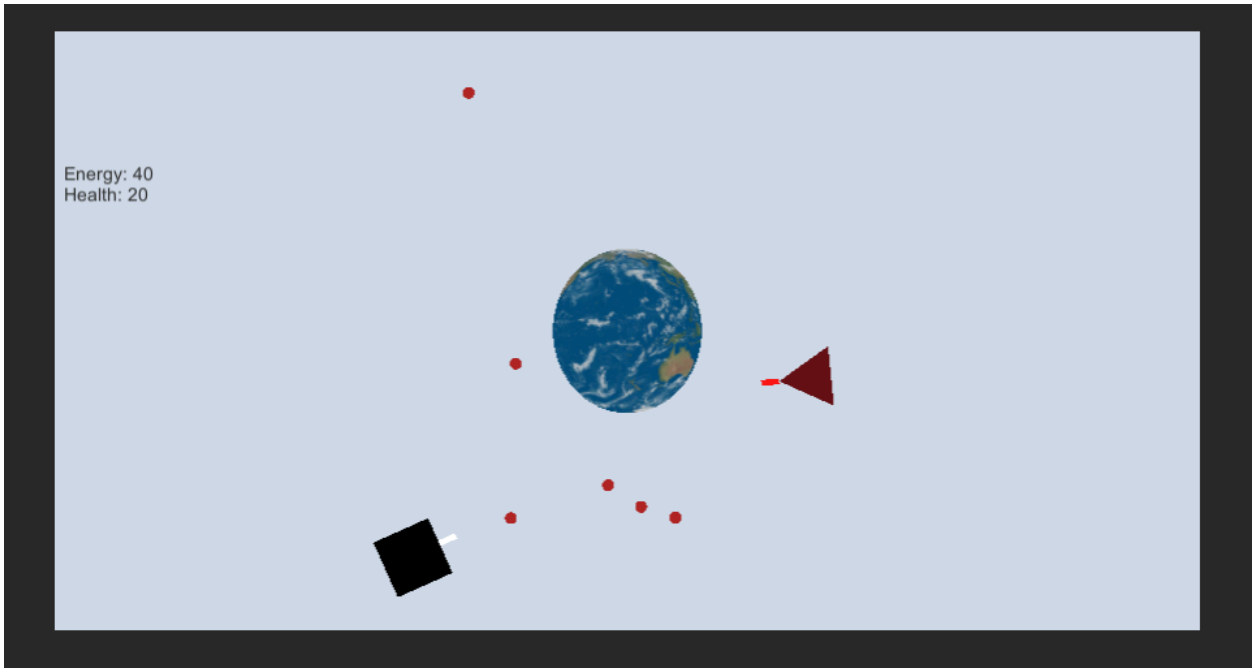
Description: Prototype 2 tests the concept of shooting mechanisms while revolving around the planet. Once the player is in orbit of the planet, the player fights the enemies by revolving around the planet. The player can vary his orbiting speed and shoot at the enemy shooters using the mouse position. Player has a limited number of hits they can take before running out of lives.

Drawing:



In the above image, the team decided to create an automatic movement mechanism - the player orbiting around a planet instead of the conventional input. We reached the conclusion of creating enemies which would be static, pointing to the player, and shooting at different rates at the player.
We also wanted to give the player the power to increase and decrease their speed to dodge high fire rate enemies.

The player would point with the mouse and click to fire at the enemy.



We accomplished all the targets set by us. The image above is a snapshot of the game in action. The player is a triangle that orbits around the planet. The black objects are enemies and the pellets are bullets.

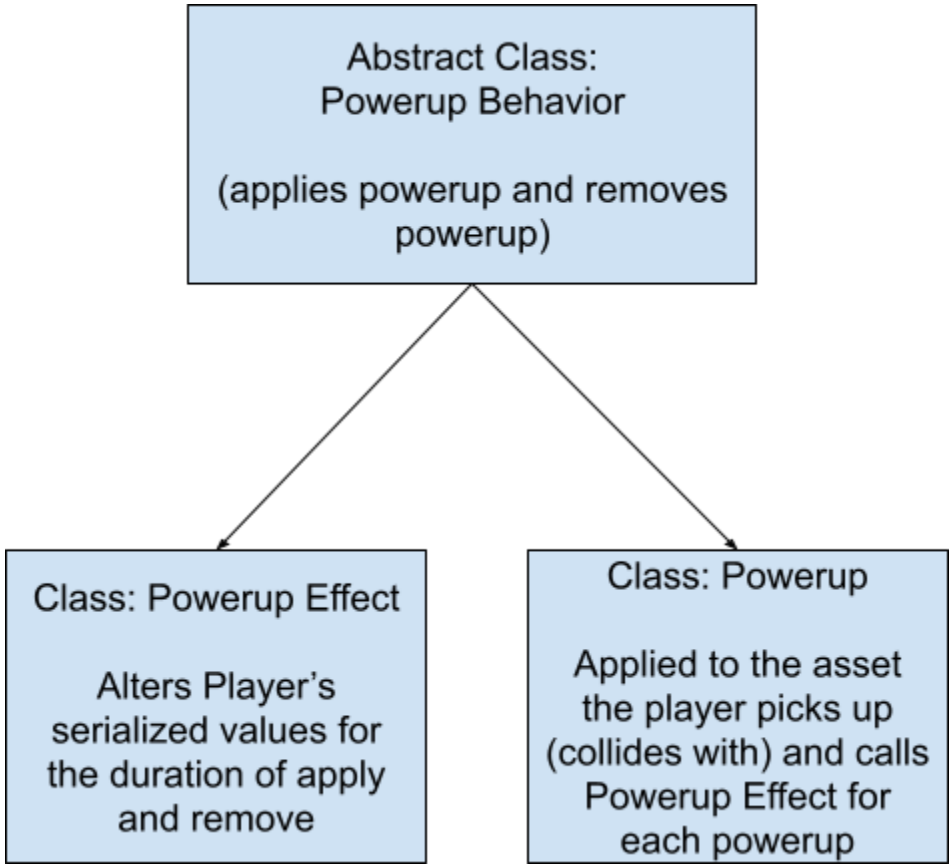
Every bullet hit to the player reduces the health of the player. The energy decreases with an increase in speed. Every bullet shot from the player to the enemy will kill the enemy.

Link: https://kohinoor98.github.io/MissingAssets-P2/

Prototype 3

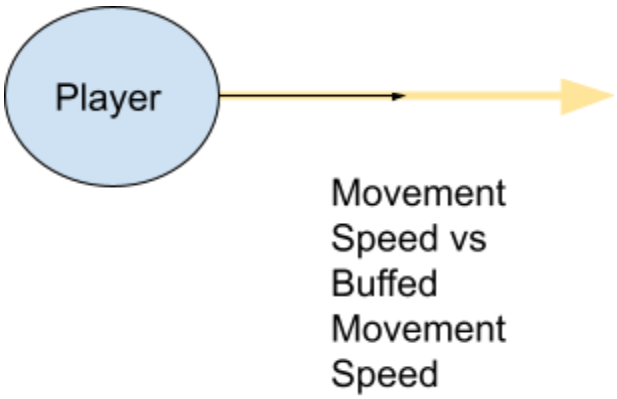
Description: Prototype 3 focuses on possible powerups in the game. We develop an abstract class powerup that is inherited by each powerup's effects and behavior.

A powerup alters the player's sprite behavior and how the player assets interact with enemies and the game environment.



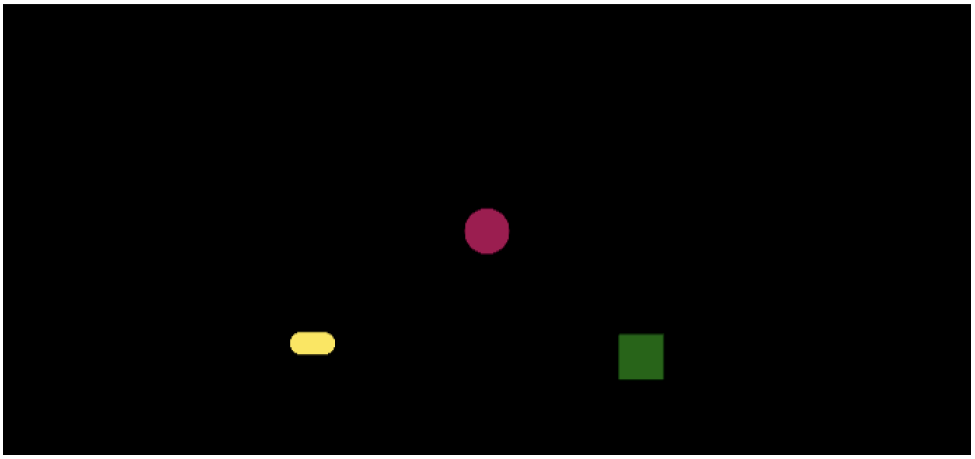
Powerup 1: Speed Boost

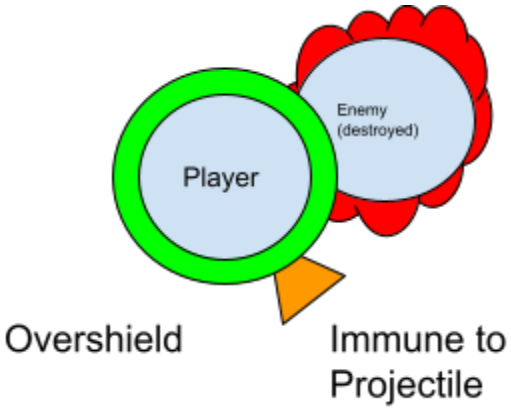
Player Picks up speed boost and instantly gets buff to base movement speed



Powerup 2: Overshield

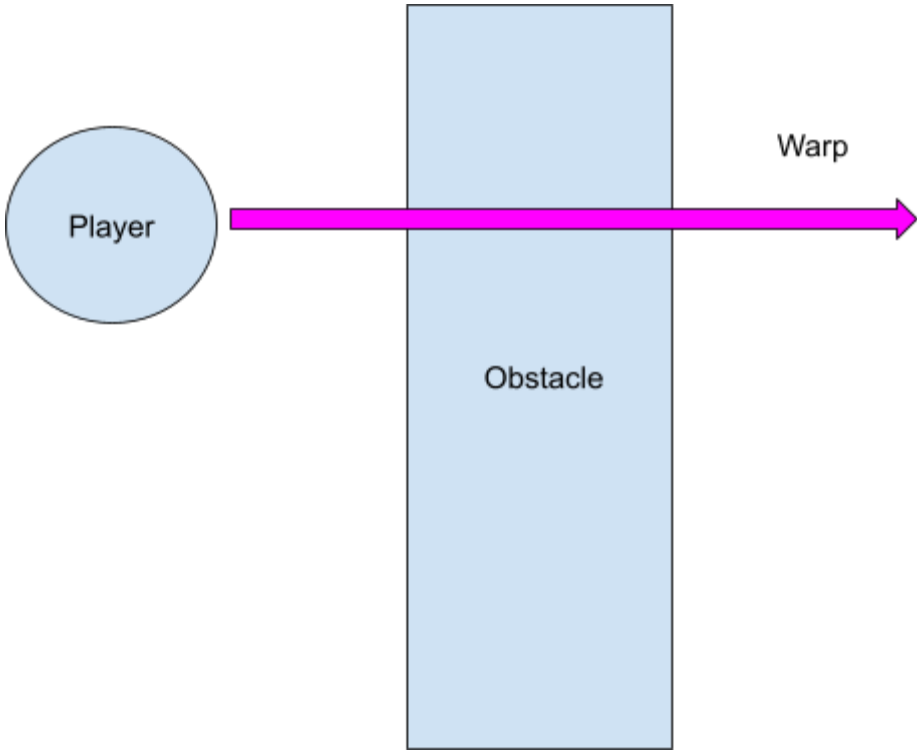
Player picks up the overshield (yellow capsule) and gains invulnerability for a short amount of time. During this time, any enemies or obstacles (green square) the player collides with are instantly destroyed.





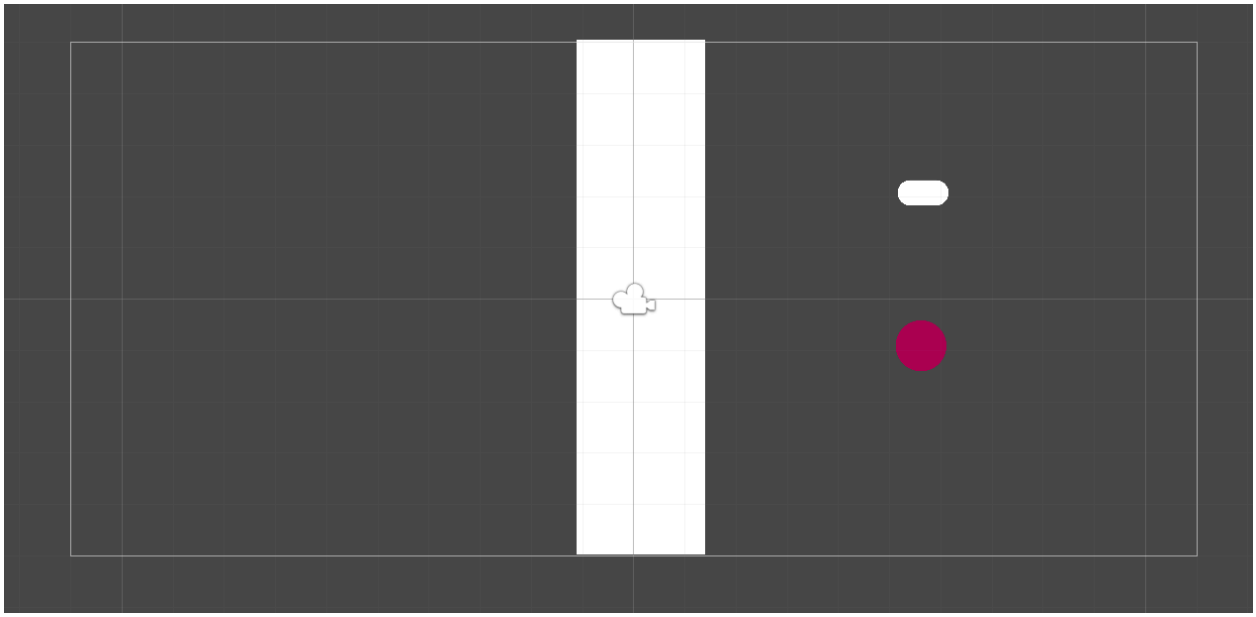
Powerup 3: Warp Drive

Player picks up the warp drive, moves in a straight line with a speed buff through any enemies or obstacles, and doesn't take damage or inflict any damage while doing so.

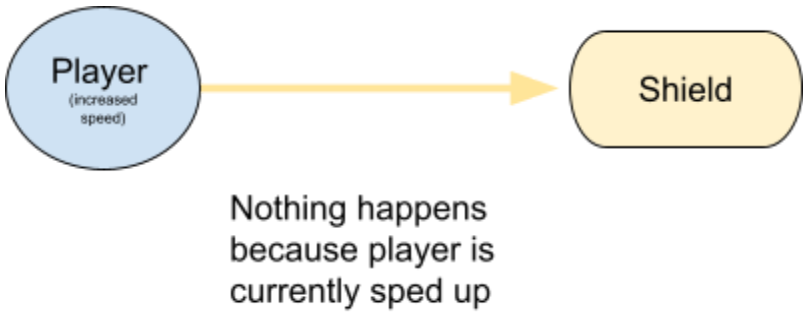


We set up the scene similarly to our warp diagram where the player picks up the powerup and travels through the obstacle.

Drawing:



We also tried combining powerups to see how they would interact with each other. We limited the player to one powerup at a time. For instance, if the player currently has a speed (white capsule) powerup and tries to collect an overshield (yellow capsule), they are unable to.

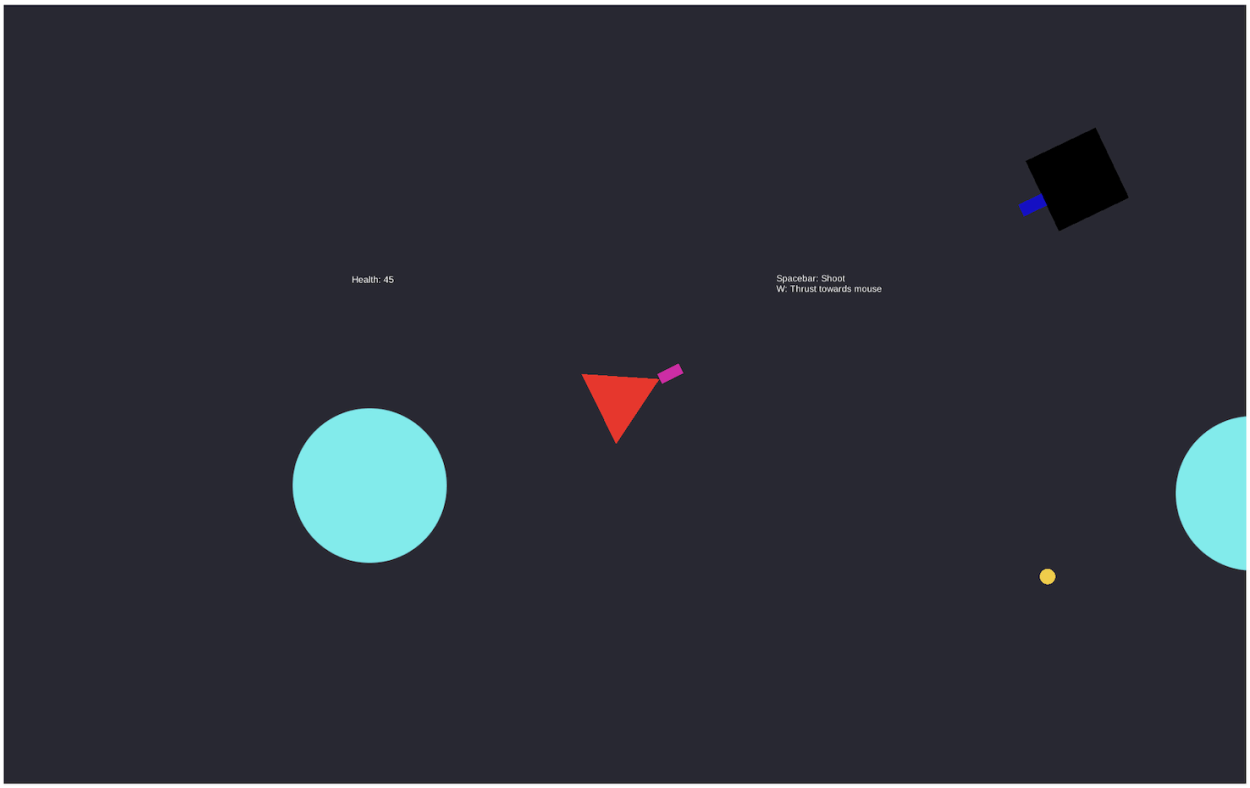


Link: https://kohinoor98.github.io/MissingAssets-P3/

GrayBox Prototype

Description: This prototype is a combination of prototype 1, 2 and 3. We incorporate 3 different game mechanics into one prototype to check how the mechanics feel in the game. We test how seamless it is to go from one mechanic to another.

Drawing:



Link: https://kohinoor98.github.io/SpaceShooter/

Weekly Milestones

Week	**Description of new features, changes, and other noteworthy events**
Week 1	Introduction Week
Week 2	Team Formation and Coming up with Gaming ideas
Week 3	Working on Protypes 1, 2 and 3. Started research on Analytics.
Week 4	Camera movement, powerups & obstacles, player & enemy mechanics, gravitational force, player firing mechanics & health, Analytics - firebase db and graph plotting using python script
Week 5	Update on Graviation interactions, introducing shields, Thrust & Game termination conditions, Analytics pipeline.
Week 6	Storyboarding & level scene creation, Landing screen, Minimap, Bullet range and come up with 2 Analytics metrics.
Week 7	Art assets, level designs, Menu items, Quest pointer & Complete analytics metrics.
Week 8	Play test for all levels, Instruction, restart and pause menu actions, Missle control, Script to auto plot analytics data.
Week 9	Midterm presentation
Week 10	Sumarize given respones and formulate into issues, work on trackpad
Week 11	Arrow, Thrust & Shooting button art, Show controls on Esc, Reverse thrust, Out of bound Bouncing back, Level selection Option, Level re-design 1,2 and 3.
Week 12	Guiding assets, re-design all levels, Boss enemy game loop, Health bar fixes, Bullet mechanics and art redesign.
Week 13	Title screen redesign, Asteroids, Boss enemy collision fix, Enemies following and gravity traps, Level 4 and 5 redesign.
Week 14	Updating GDD, adding more game loops, Fixing minor bug issues, More on Boss enemy and additional levels.

Analytics

Methodology

1. Database : **Firebase** [cloud-based noSql database] (Link to Test DB) (Link to Prod DB)

2. Graph Plotting **:** Using **pandas + matplotlib**

3. Datatype **: JSON** (find the sample json data below)

Code snippet:

```
StatisticsManager.buildAnaltyicsDataObjAndPush(0,1,"DeathByEnemy","0%",0,"player_termination","enemy");
StatisticsManager.buildAnaltyicsDataObjAndPush(numberOfEnemiesKilled,1,"NumEnemiesKilled","0%",0,"numEnemiesKilled","enemy");
```

How are we collecting the analytics data?

Since Firebase doesn't have integration support with Unity WebGL, we're using RestClient library in our Unity app to make a http POST call to firebase with analytics data payload.

Sample Request - Response :
POST https://missing-assets-default-rtdb.firebaseio.com/analytics_data.json

Request Headers:
Accept : application/json

Request Body:

```
{
   "timestamp": "01/29/2023 01:22:45 PM",
   "level": 4,
   "type": "completion",
   "reason": "level_up",
   "remEnergy" : "30%",
   "enemiesKilledTotalCount" : 6,
   "totalSuccessfulJumpsCount" : 4,
   "sourceOfDeath" : null
}
```

Response: 200 OK
Response body :

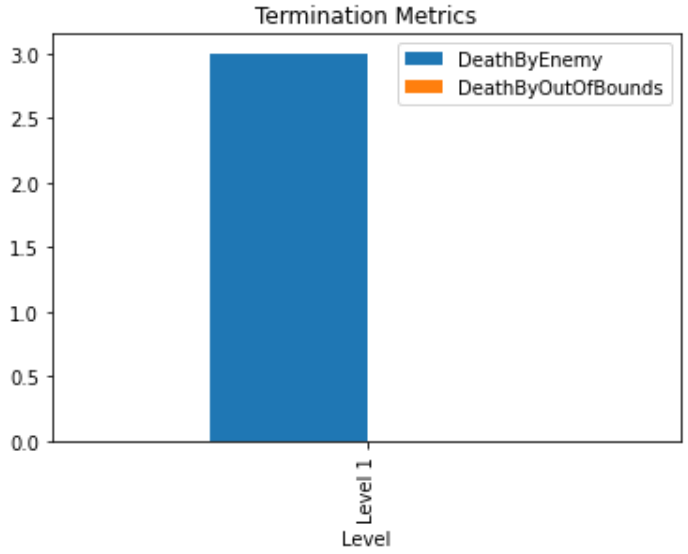
```
{
  "name": "-NMzPsx7ORgz5ronwa7H"
}
```

Metrics and Graphs

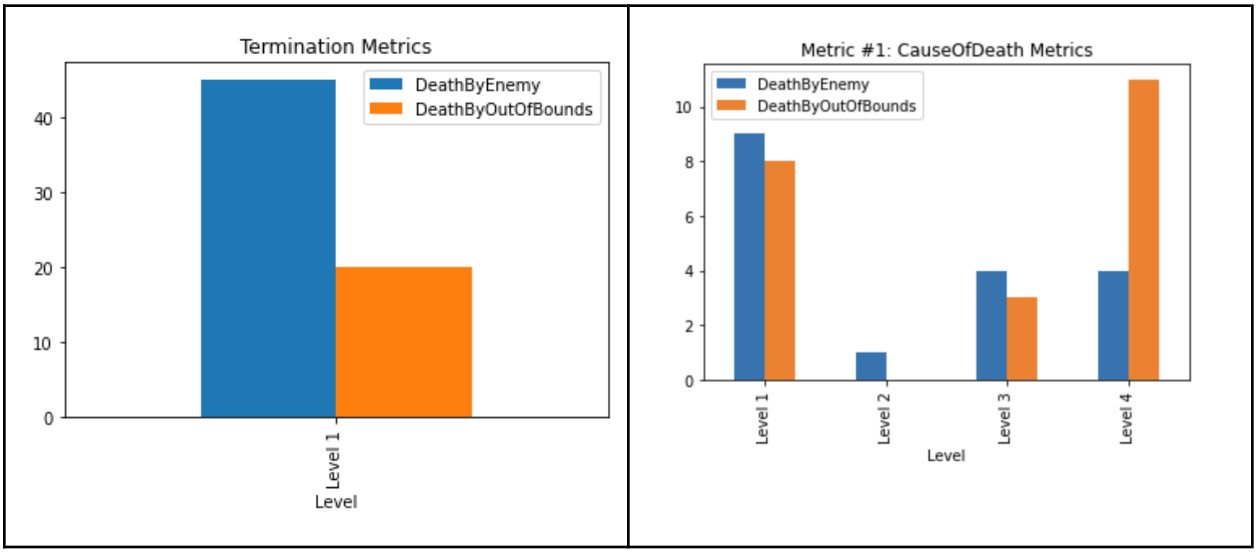
Metric #1: Player Termination metrics

Description: *These metrics are used to understand the different ways the games terminate because of player dying. To start with, there are mainly two ways in which this could happen. The player can die by attack from the enemy, Or can be killed because the player went out of the allowed game zone.*

Initial Sketches: (Image)



Pre-Midterm Graphs: (Image)



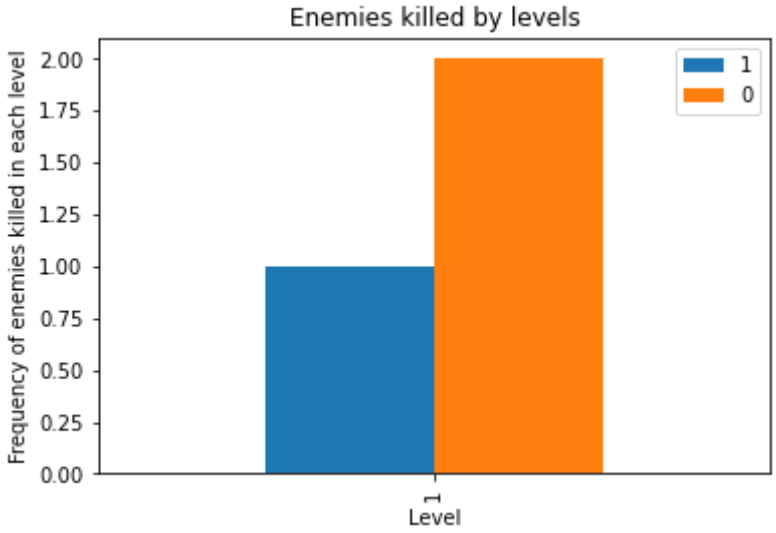
Explanation: *These metrics help us understand how the game is being terminated in each level. It helps us to know if a player is dying by the enemy*

or by being out of bounds. Analyzing these metrics will help us to understand the complexity of the game and decide on out of bounds conditions and number of enemies to deploy.

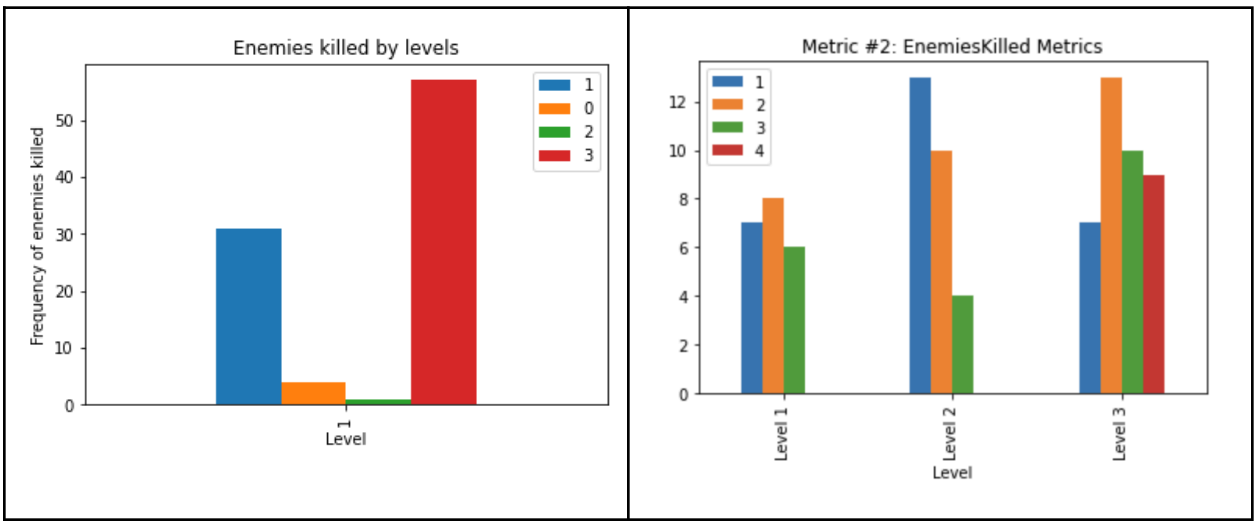
Metric #2: Enemies Killed Metrics

Description: These metrics contain data wrt the total number of enemies killed from a player on each level. On every level complete, the data will be pushed to Firebase db. The graph represents the data of frequency(average) enemies killed by the players, on each level.

Initial Sketches: (Image)



Pre-Midterm Graphs: (Image)



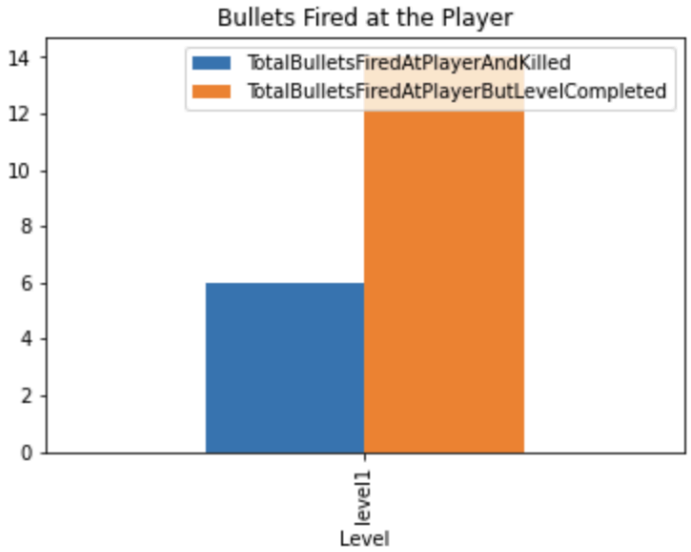
Explanation:

These metrics help us understand how many enemies the players are usually able to kill in each level. Having data of the frequencies of enemies killed at each level helps us to understand how difficult the game is. If the distribution is skewed where everyone is able to kill all the enemies, then that means the level is too easy. This graph is vital to know whether to increase the difficulty level in terms of enemies.

Metric #3: Accuracy of bullets fired by the player towards enemies

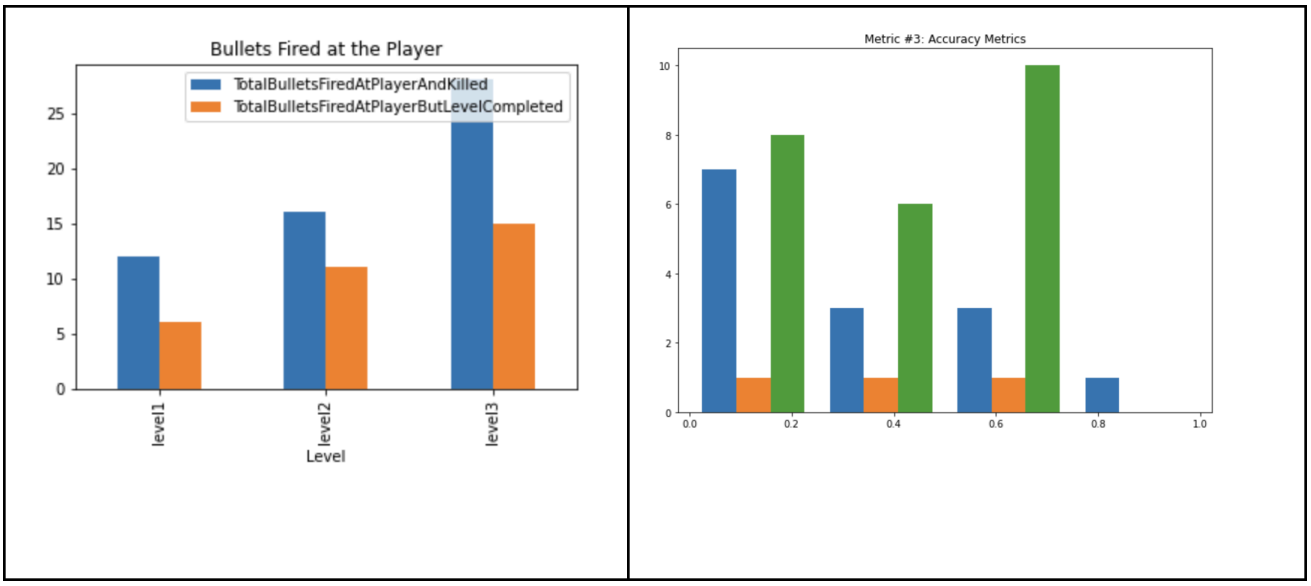
Description: *This metric is used to understand how good the players are shooting bullets at the enemy troops. This is to understand the ease at which players are getting used to killing the enemies. If the accuracy level is too low, that means the enemies are moving too fast or the game level is tough to kill the enemies, and the player would eventually die from the enemy bullets, thus can't proceed to the next level.*

Initial Sketches: (Image)



Note : The above metric has been changed to Accuracy which is more critical for game advancement.

Pre-Midterm Graphs: (Image)

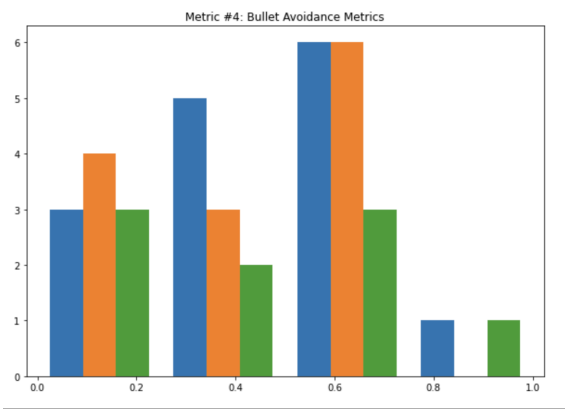
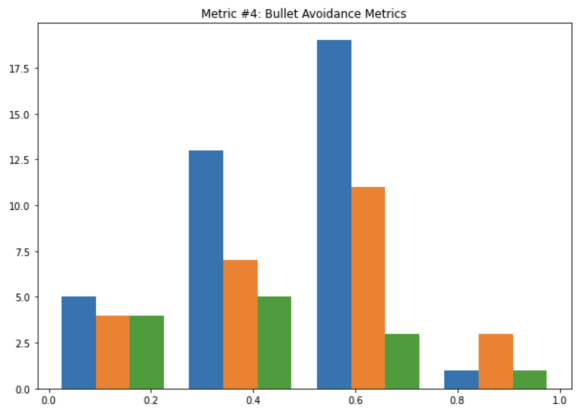


Explanation:

This graph helps us understand the enemy density in a given level and the toughness of that specific level. This also explains how well the players are handling the enemy attack before they could die or complete the specific level. If we see that the accuracy of the players bullets fired is less, we would know right away that a particular level is tougher and the players are having issues killing the enemy troops. Also, if the accuracy is too high, then we could increase the level toughness if required, by placing tougher (fast-moving) enemies in that level.

Metric #4: Bullet Avoidance by the player from enemy troops

Description: This metric explains how difficult it is to play against the enemy troops while achieving the goal under the influence of gravity.

Initial Sketches: (Image)	Pre-Midterm Graphs: (Image)
	

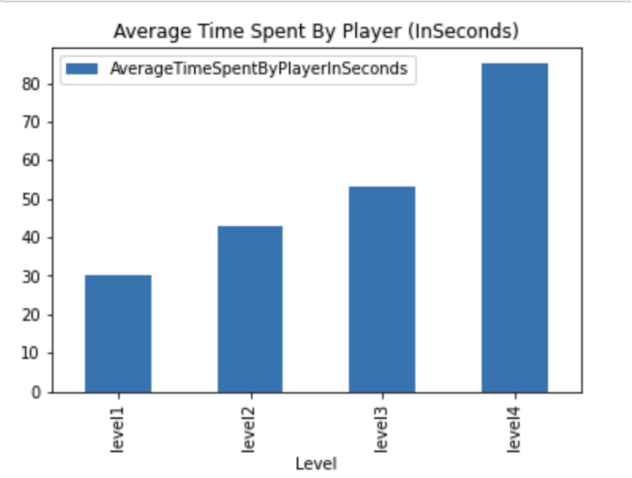
Explanation:

If the bullet avoidance is high, then we can conclude that the players are comfortable playing that level against the enemy troops by avoiding their bullets and end up finishing the level. If the bullet avoidance is extremely low, then the difficulty level at which enemy troops are attacking the player is high and might end up in killing all the players and not letting anyone finish a specific level.

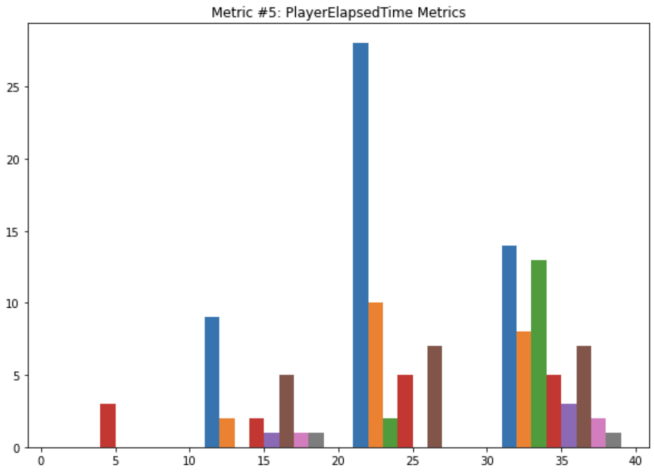
Metric #5: Time spent by a player on each level

Description: This is the most crucial metric of our game. The amount of time every player spends on a specific level will give insights into the difficulty of the level. If a specific level has been completed by most of the players within a given time threshold, then we can consider that as a very easy level and work on placing more difficulties to complete the level. This metric is being pushed from the game on every level completion, as well as player termination.

Initial Sketches: (Image)



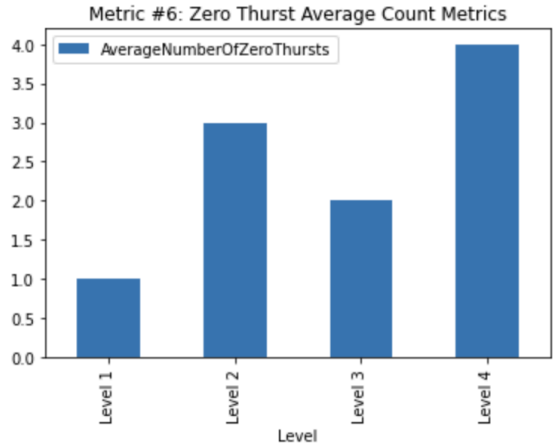
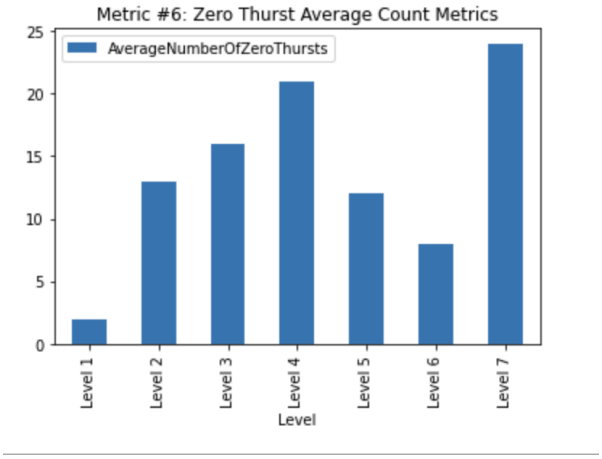
Pre-Midterm Graphs: (Image)



Explanation: Just by looking at the above graph, it is clear that the toughness of level keeps increasing on every level since the average time spent for any player keeps increasing as the level increases. This is very crucial in designing each level and understanding the toughness associated with it. If the players are not reaching to further levels of the game, this graph helps us determine where it's going wrong.

Metric #6 : Average number of times thrust reached zero

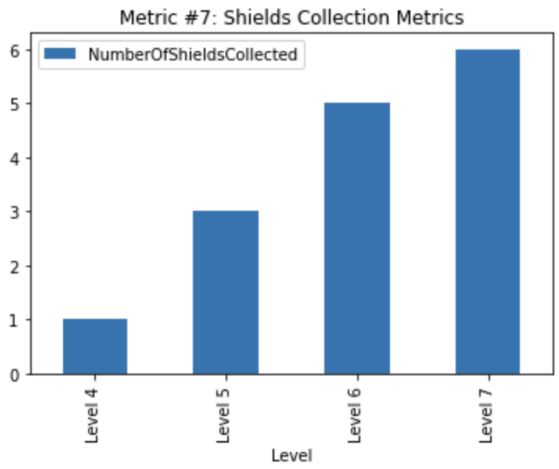
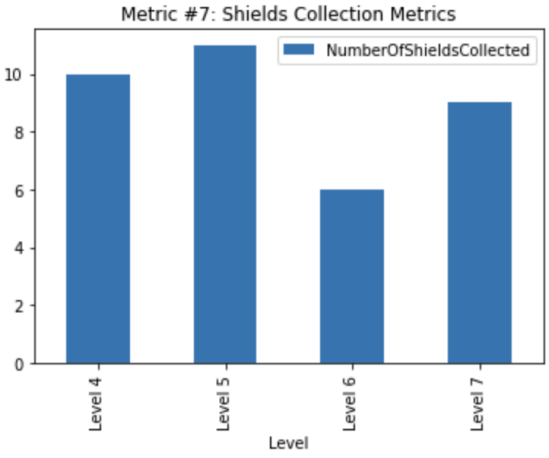
Description: This metric is used to understand the average no. of times the thrust reached zero for a player in a specific level. Thrust is used to maneuver the player in the space in between the planets and around their gravitation fields to collect the collectibles and kill the enemies.

Initial Sketches: (Image)	Pre-Midterm Graphs: (Image)
	

Explanation: For a given level, if the average number of times the thrust reaches zero is more, then the specified thrust level is not sufficient enough for the players to complete the level and that's why we'll have to increase the thrust level. If the average number of times the thrust level goes to zero is minimal, then the pre-set thrust level for the game in a specific level is sufficient. This is very helpful from the developer point of view to understand the pre-set thrust level and adjust accordingly to ease the player to complete the level and increase the player user experience.

Metric #7 : Number of shields collected at a specific level.

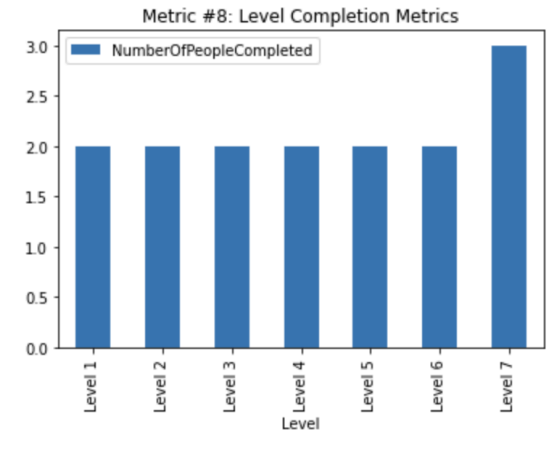
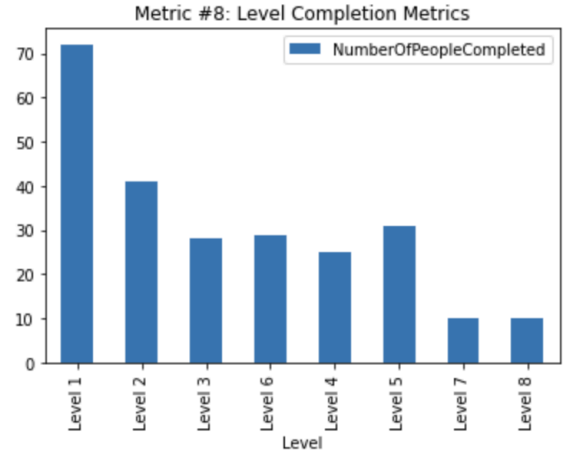
Description: This metric is used to understand the no. of shields collected at a specific level. Shields are used to save against an enemy shooting. This graph along with Metric #2 will help us decide to further help players to complete a tougher level by placing one or more shields in a specific level, thus allowing the player a chance to fight against the tough enemy line and still complete the level.

Initial Sketches: (Image)	Pre-Midterm Graphs: (Image)
	

Explanation: From the above graph, we'll know how many people are actually using the shield to fight against the enemy, and how the shield is making a specific level easy or difficult to complete.

Metric #8 : Number of players that complete a specific level

Description: This is the most crucial metric of our game. This graph will let us know how many players completed a specific level. Drop in numbers in any specific level shows that a particular level is tougher and can re-design the level based on that.

Initial Sketches: (Image)	Pre-Midterm Graphs: (Image)
	

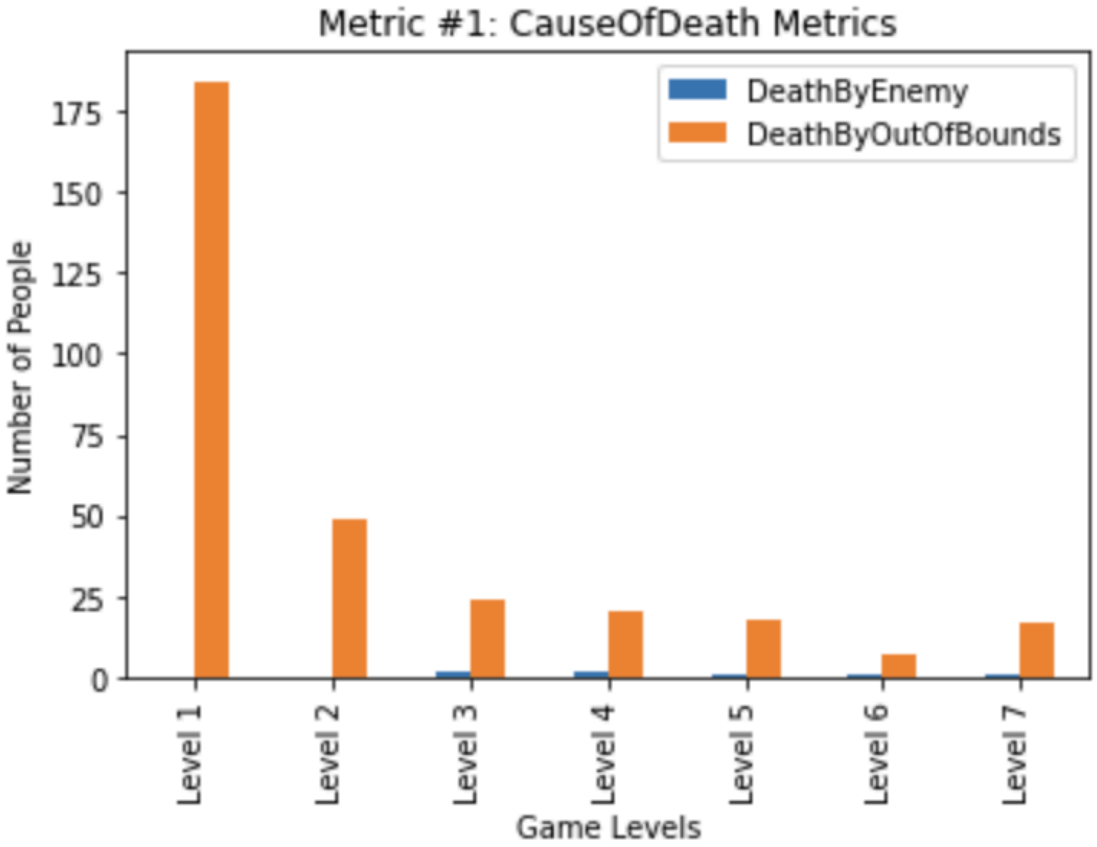
Explanation: Just by looking at the above graph, it is clear that the toughness of level keeps increasing on every level since the number of players completing the level decreases. This is very crucial in designing each level and understanding the toughness associated with it. If the players are not reaching further levels of the game, this graph helps us determine where it’s going wrong.

Analytics - Midterm

Metric #1: Player Termination metrics

Description: *These metrics are used to understand the different ways the games terminate because of player dying. To start with, there are mainly two ways in which this could happen. The player can die by attack from the enemy, Or can be killed because the player went out of the allowed game zone.*

Midterm Graphs: (Image)

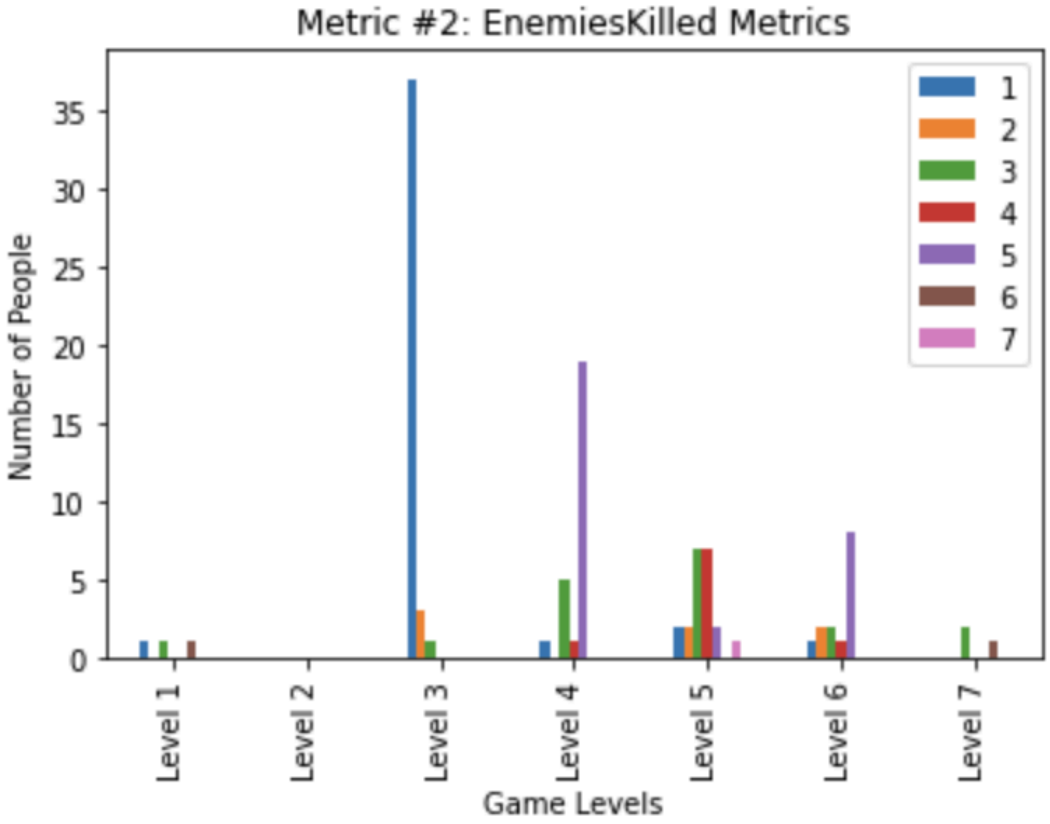


Analysis:
From Metrics 1 graph, we can see that the players are finding it difficult to clear the first round, as they are being killed because of Out of Bounds Metrics. The number of deaths due to enemies bullets are very less. This gave us a great insight of where the people are stuck and the exact reason for it.

Metric #2: Enemies Killed Metrics

Description: These metrics contain data wrt the total number of enemies killed from a player on each level. On every level complete, the data will be pushed to Firebase db. The graph represents the data of frequency(average) enemies killed by the players, on each level.

Midterm Graphs: (Image)

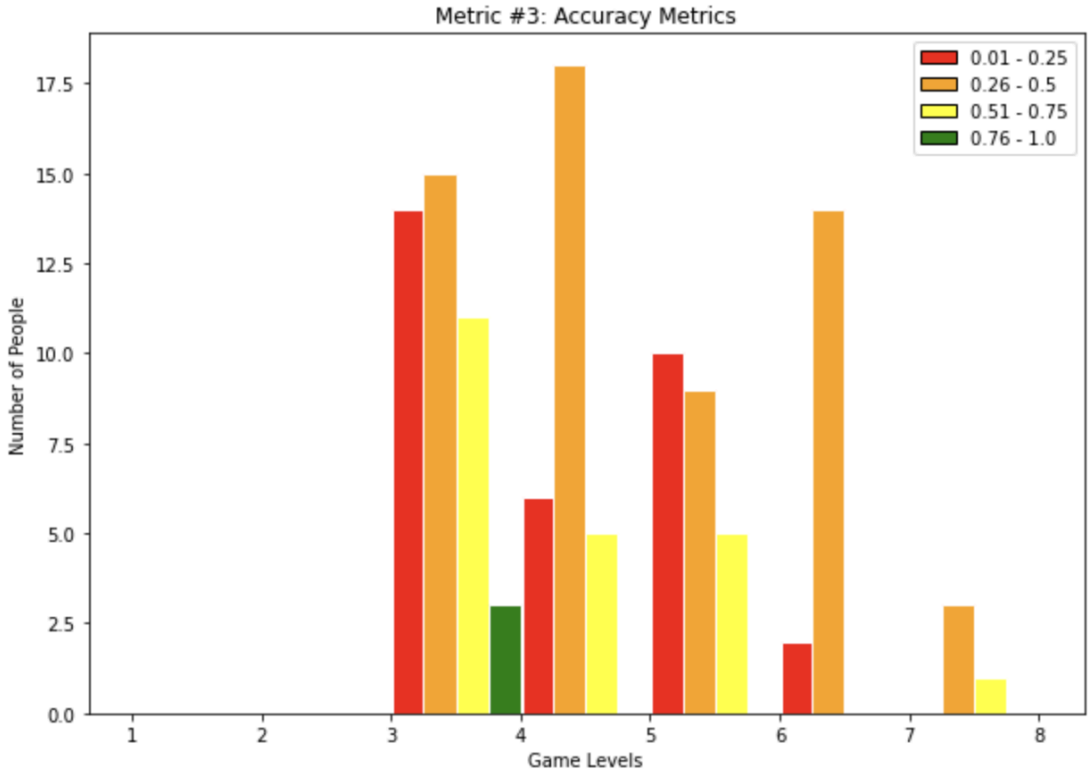


Analysis: This metric helped us understand if the players are successful in killing the enemies. This is a great insight for us to know whether a specific level is an easy one or stronger based on enemy opposition. This is going to help us strategize and place the enemies at the right levels and positions to make the game more interesting.

Metric #3: Accuracy of bullets fired by the player towards enemies

Description: This metric is used to understand how good the players are shooting bullets at the enemy troops. This is to understand the ease at which players are getting used to killing the enemies. If the accuracy level is too low, that means the enemies are moving too fast or the game level is tough to kill the enemies, and the player would eventually die from the enemy bullets, thus can't proceed to the next level.

Midterm Graphs: (Image)

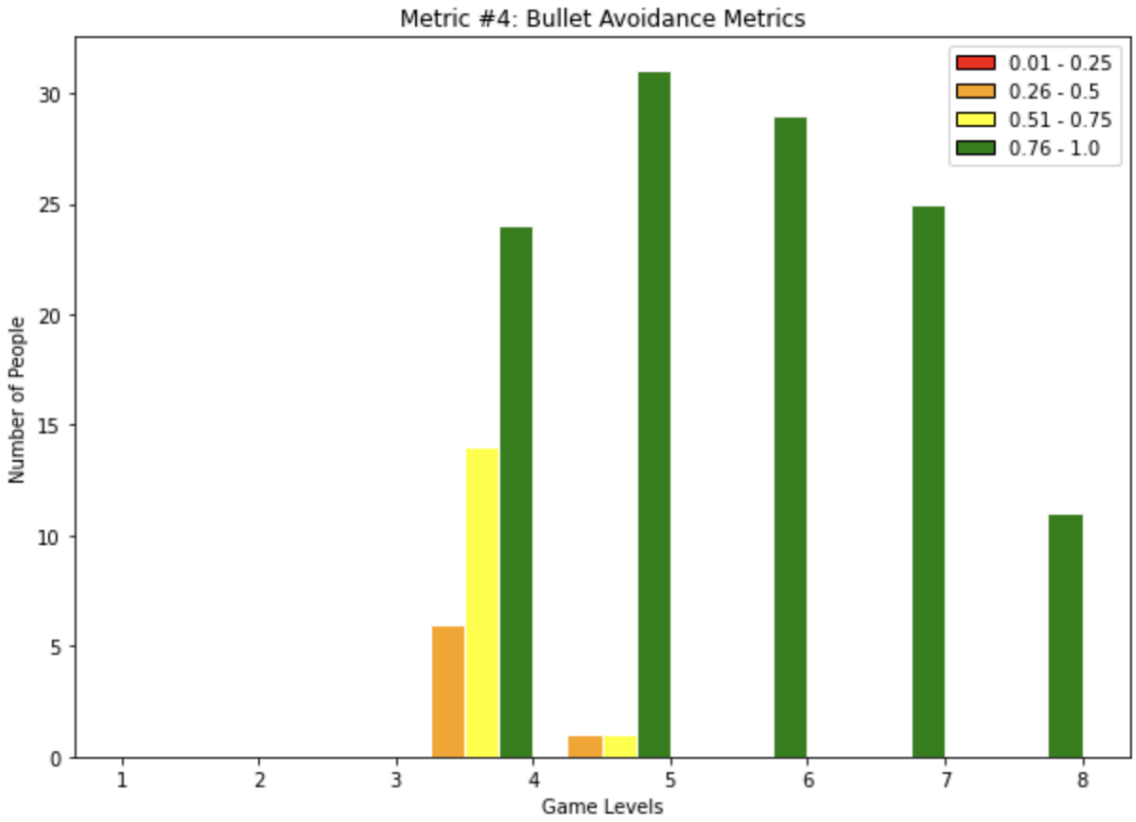


Analysis: This metric helps us understand the player's shooting abilities and the ease at which they are killing the enemies. The higher accuracy at certain levels shows that the enemies are easy to be killed since most of the bullets fired actually hit the enemies. This helps us to understand where we can introduce mobile enemies to make the game more challenging and interesting at the same time.

Metric #4: Bullet Avoidance by the player from enemy troops

Description: This metric explains how difficult it is to play against the enemy troops while achieving the goal under the influence of gravity. If the bullet avoidance is high, then we can conclude that the players are comfortable playing that level against the enemy troops by avoiding their bullets and end up finishing the level. If the bullet avoidance is extremely low, then the difficulty level at which enemy troops are attacking the player is high and might end up in killing all the players and not letting anyone finish a specific level.

Midterm Graphs: (Image)

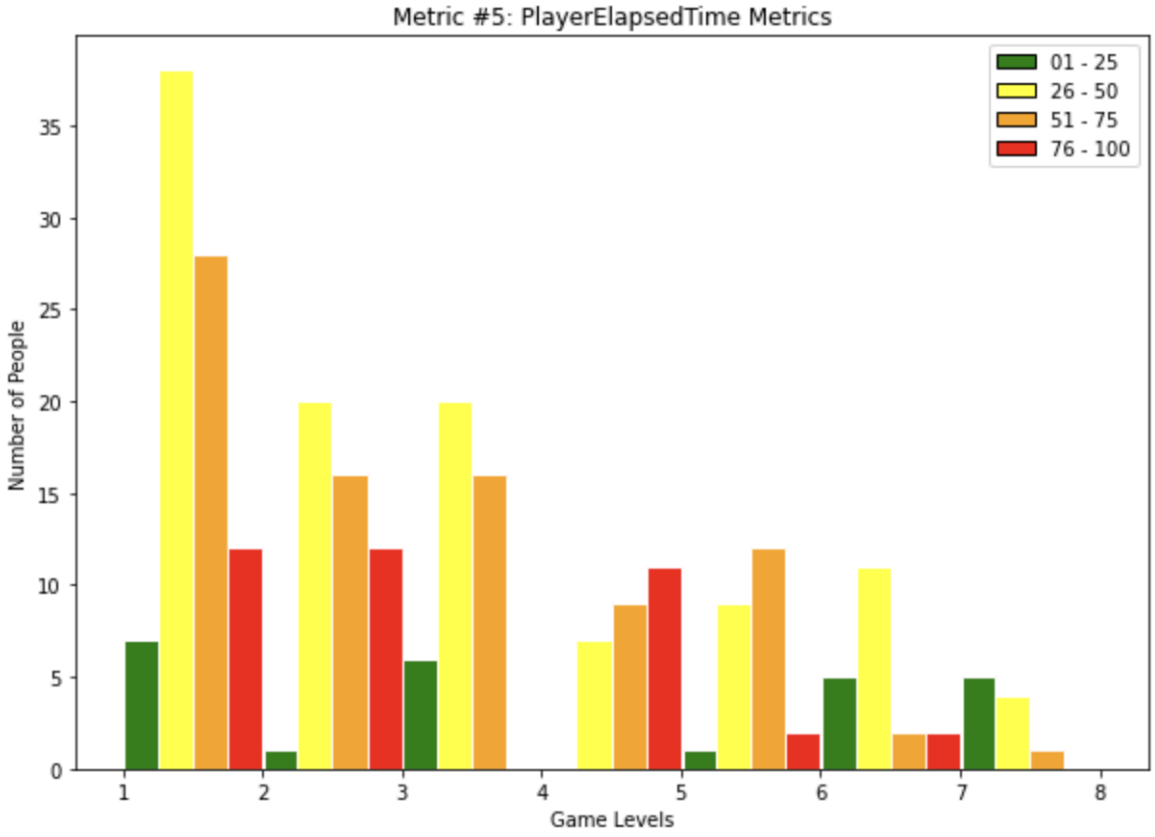


Analysis: The above metrics gave good insights into how well the players are avoiding enemy bullets. The higher the accuracy, the higher their ability in shielding themselves or making the enemy bullets not hit the players. Level 4 is when the enemy starts coming and we could see the people's learning curve. As the level increases, players get good at it.

Metric #5: Time spent by a player on each level

Description: This is the most crucial metric of our game. The amount of time every player spends on a specific level will give insights into the difficulty of the level. If a specific level has been completed by most of the players within a given time threshold, then we can consider that as a very easy level and work on placing more difficulties to complete the level. This metric is being pushed from the game on every level completion, as well as player termination.

Midterm Graphs: (Image)

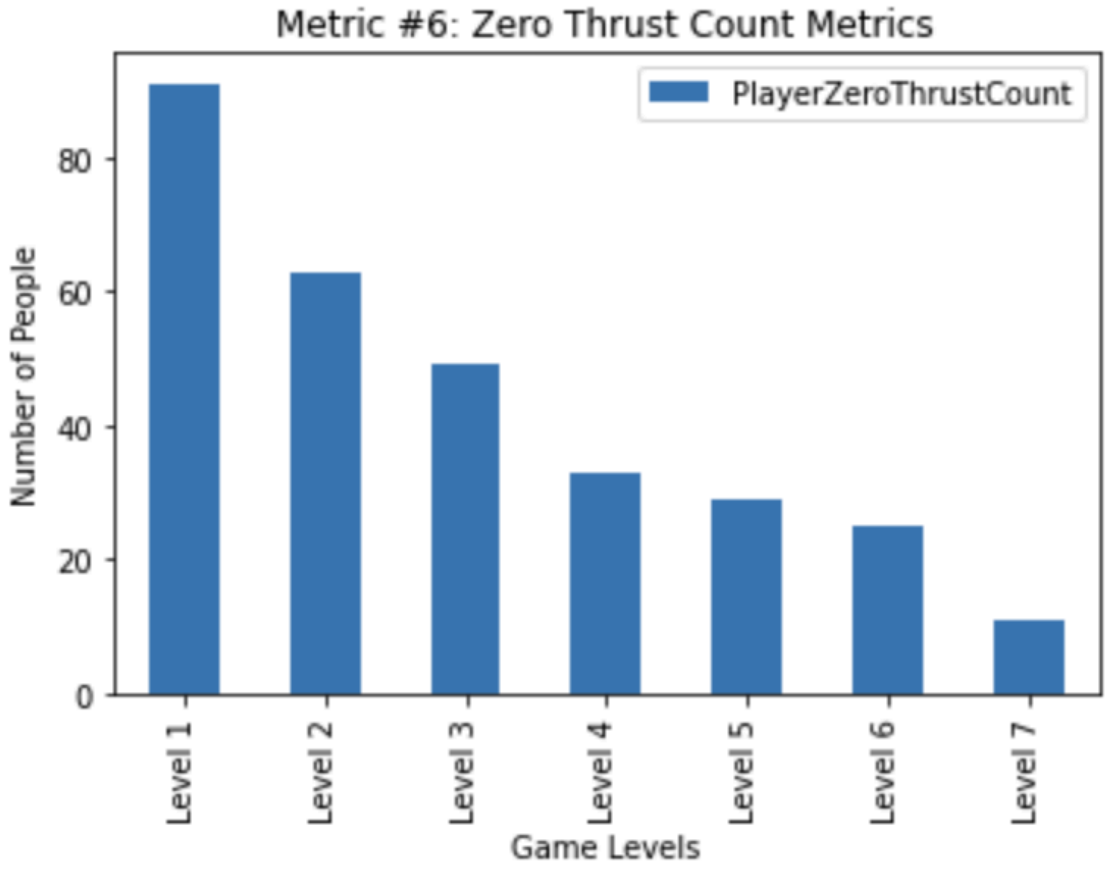


Analysis: This metric gave us insights into how difficult a level is, based on the time spent by the players in a specific level. We can see that people are taking 25 to 75 seconds in initial levels and then once they are getting hold of the game, spending less time on subsequent levels. This helps us understand the player's adaptability to our game and ease at which they are able to complete the levels.

Metric #6 : Average number of times thrust reached zero

Description: This metric is used to understand the average no. of times the thrust reached zero for a player in a specific level. Thrust is used to maneuver the player in the space in between the planets and around their gravitation fields to collect the collectibles and kill the enemies.

Midterm Graphs: (Image)

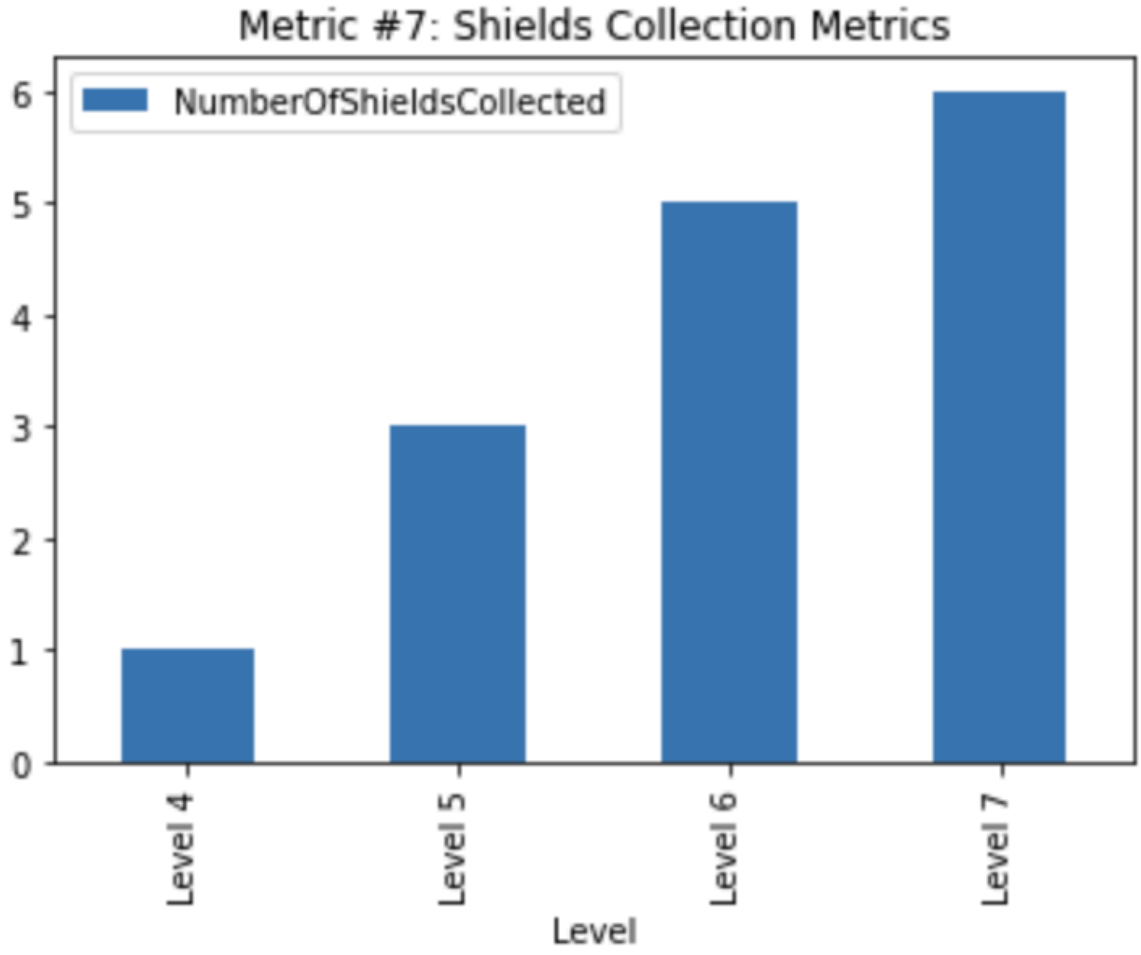


Analysis: This metric is used to undestand the usage of thrust power while playing. We want to fine tune the thrust for each level and understand whether the players are good at using the thrust as we intended. We collect the metrics when they hit zero thrust to understand fine tuning at the backend.

Metric #7 : Number of shields collected at a specific level.

Description: This metric is used to understand the no. of shields collected at a specific level. Shields are used to save against an enemy shooting. This graph along with Metric #2 will help us decide to further help players to complete a tougher level by placing one or more shields in a specific level, thus allowing the player a chance to fight against the tough enemy line and still complete the level.

Midterm Graphs: (Image)

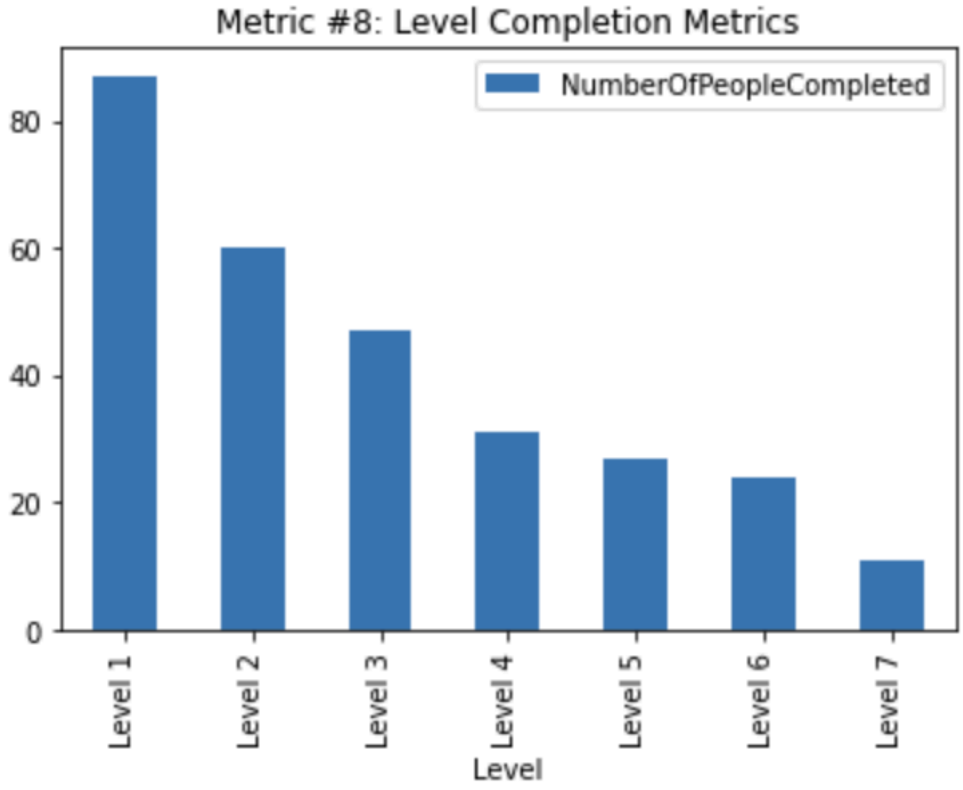


Analysis: This metric is to understand if the players are actually collecting the shields. This is to understand if this mechanism is actually being used by the players. From the above metrics, it's evident that players are using the shields as intended.

Metric #8 : Number of players that complete a specific level

Description: This is the most crucial metric of our game. This graph will let us know how many players completed a specific level. Drop in numbers in any specific level shows that a particular level is tougher and can re-design the level based on that.

Midterm Graphs: (Image)



Analysis: This is a major metric from the development point of view to understand if a specific level is able to complete or not. As the graph shows, players are completing the initial levels without any issues but as the game proceeds to subsequent levels, the completion numbers drops. This maybe due to players are not playing the complete game but just the initial levels to feel the game, but we'll put more game loop to indulge players to complete all levels going forward.

Post-Midterm improvements

Issue #1 - High thrust makes it difficult to control player

Feedback:

> Movement needs to be difficult to control as we move forward in levels not from LEVEL 1
>
> I am already fighting with thrust/gravity to keep myself within the bounds of the game

Explanation: Some players have trouble controlling the player's movements and feel that the game is too fast-paced.

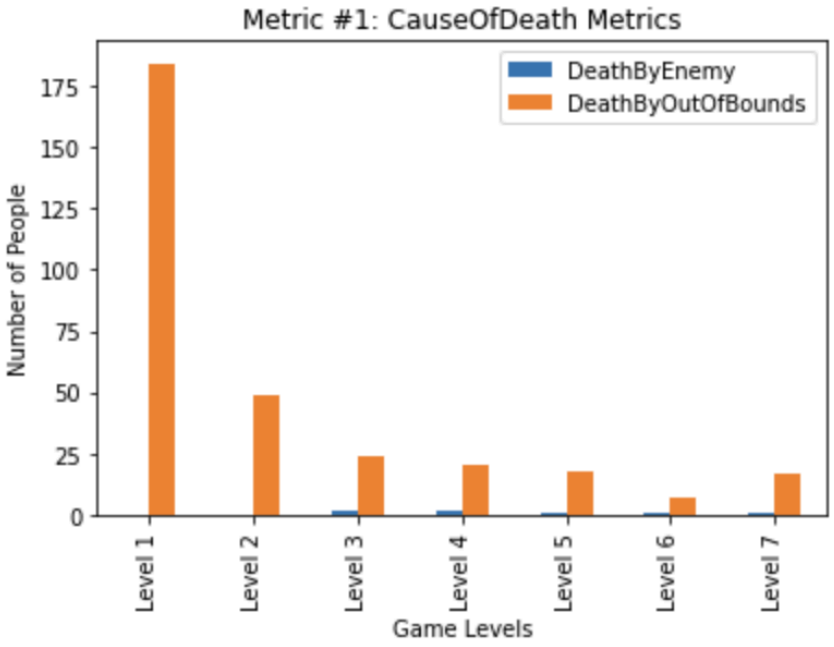
Potential Solutions:

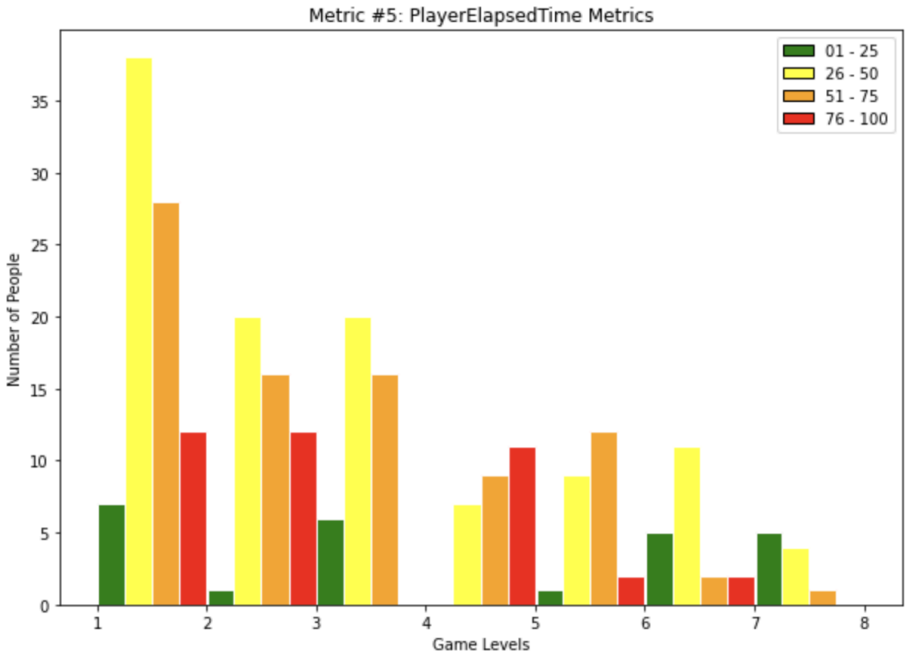
1. ***Implement a braking mechanism that allows players to slow down or stop the player's movements.***
2. Automatically stop the player gradually when 'W' is not being used. Need to handle complexity in gravity

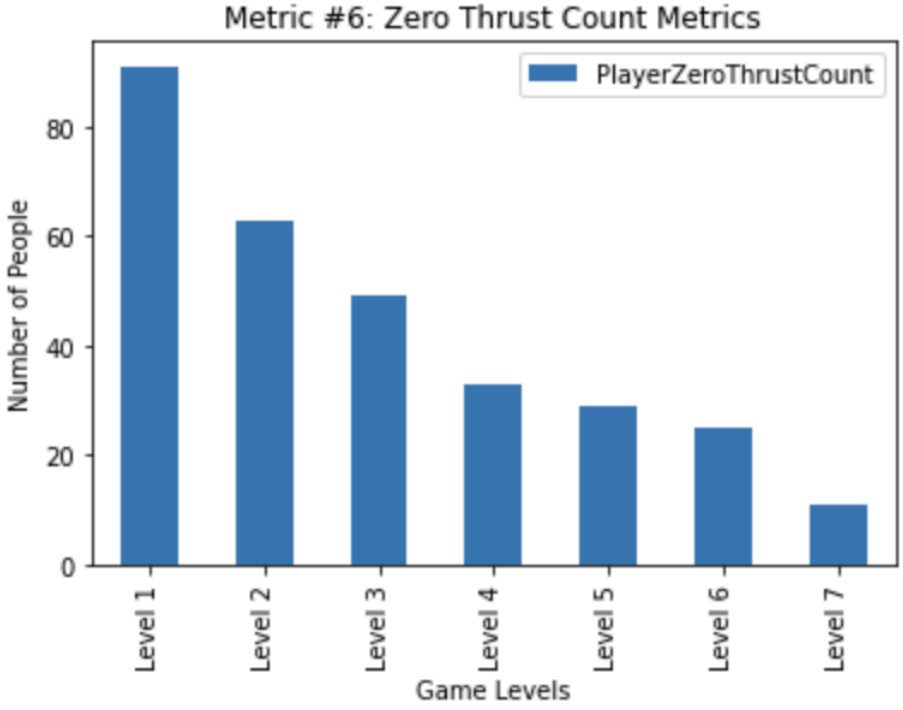
Reference: Response #34,#38,#40. Metric #1, #5 and #6

Here we can understand that the players are flying out of bounds and the number 1 cause for it is no control over thrust. Around ***13% users*** cite controls are not intuitive

Status: Started work in week 11. **Completed on 3/26/2023.**




Metric #5: PlayerElapsedTime Metrics
01 - 25
26 - 50
51 - 75
76 - 100
Number of People
Game Levels
0
5
10
15
20
25
30
35
1
2
3
4
5
6
7
8


Metric #6: Zero Thrust Count Metrics
PlayerZeroThrustCount
Number of People
Game Levels
0
20
40
60
80
Level 1
Level 2
Level 3
Level 4
Level 5
Level 6
Level 7

Issue #2 - Out of Bounds very frequent in initial levels

Feedback:

> more frustrating thing is that I was getting out of safe zone by improper thrust.

> When I was playing initial level I went away from the screen beacuse of obvious reason i was noob. Its better to put walls surrounded all side for the initial levels.

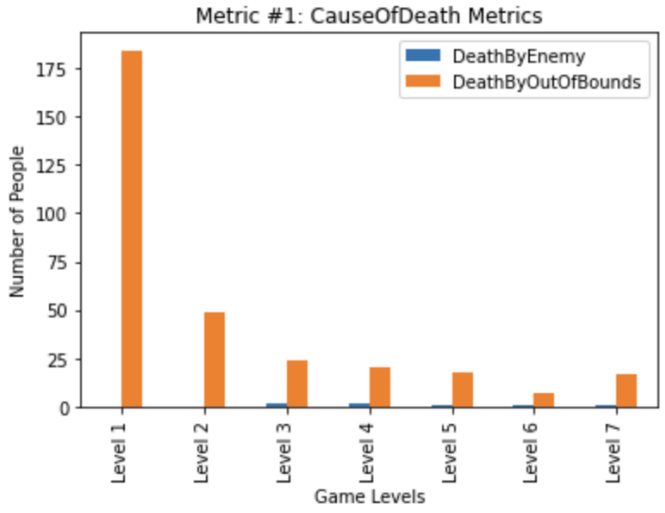
Explanation: Some players have expressed that the spaceship goes out of bounds very often and they are unable to progress in the game.

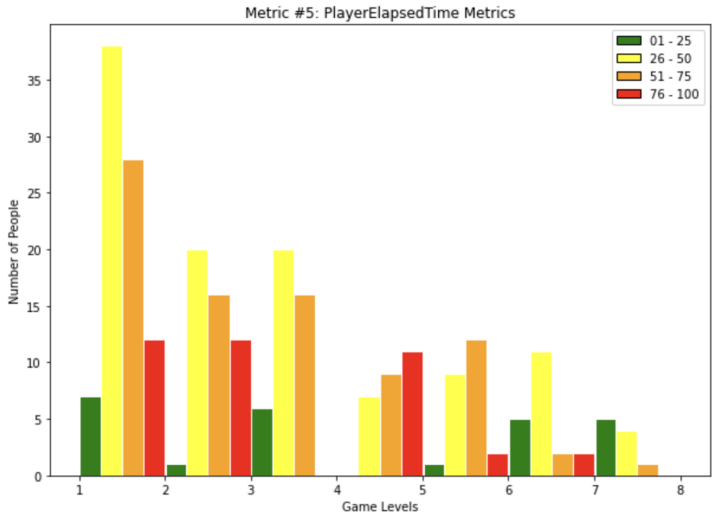
Potential Solutions:

1. ***Add a collider for out of bounds so that players stay inside the bounds***
2. Increase the size of the playing area or allow players to drift beyond the bounds for a limited time
3. For initial levels add a boundary & once the player gets used to the movement remove the boundary in later levels

Reference: Feedback #34, #43, #47, #55. Metric #1 and #5. High number of deaths due to out of bounds than enemy kills. Around ***37% users*** cite it as a difficult game

Status: Started work in week 12. **Completed by 4/9/23.**





Issue #3 - Enemies are not a central part of the game

Feedback:

If shooting/enemies is going to become important to the game, please make it more engaging.

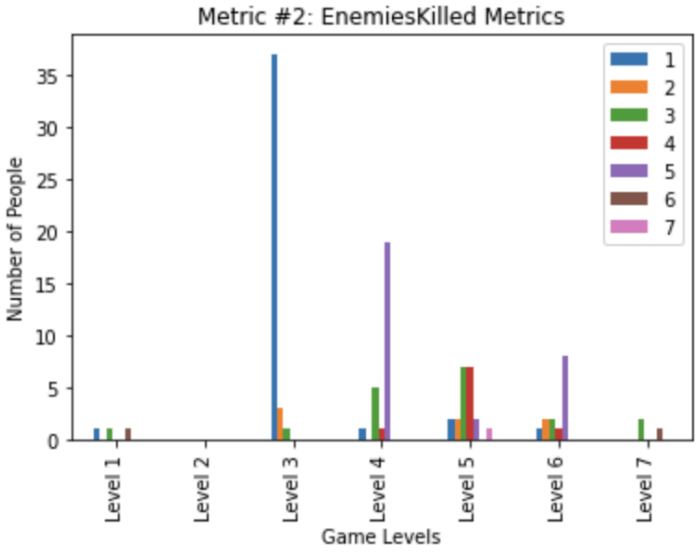
Explanation: Right now the enemies are not a central part of the game and the game can be completed without engaging the enemies.

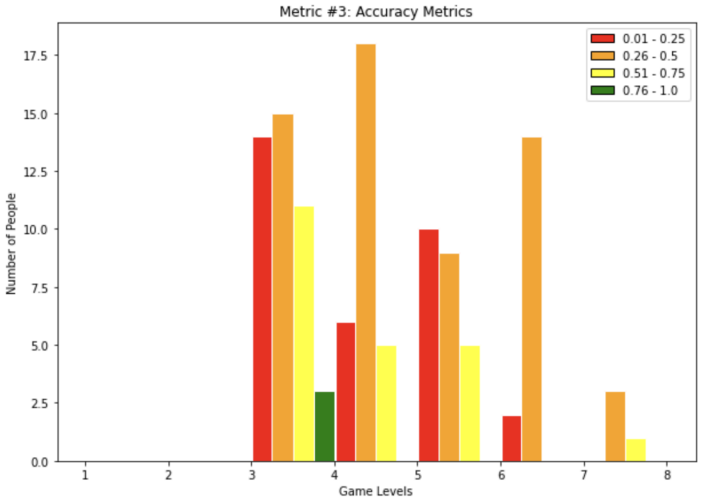
Potential Solutions:

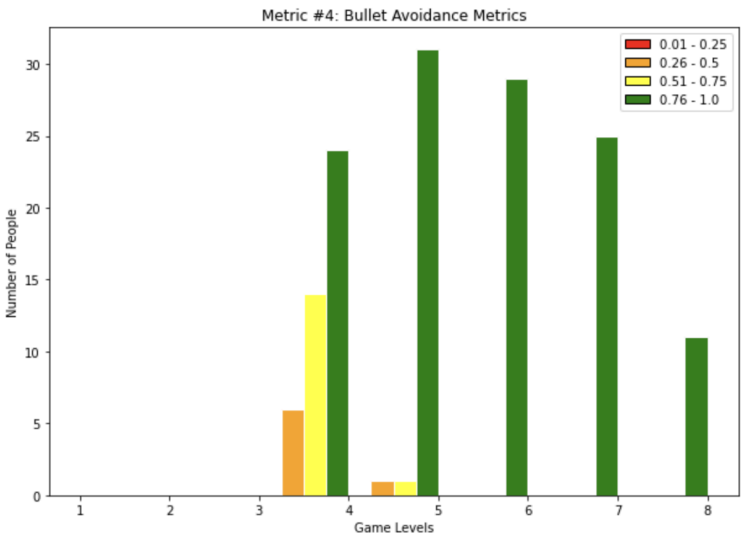
1. Change the game loop to necessitate players engaging with enemies
2. Make enemy shooting back more difficult so that players have to deal with enemies first
3. **Added a boss enemy for players who wanted more game play with the enemies.**
4. **The player can trap enemies in gravity and create strategies around it.**

Reference: Feedback #34, Metric: #2. As it can be seen from the graph most players do not bother to kill enemies, and continue on with collectibles. From Metric #3 graph we can also tell it is difficult to kill the enemies and hence the players choose not to engage with them and continue with the game. Further Metric #4 We see that most players can avoid enemies more than 76% of the time, hence they need not engage them.

Status: Started work in week 11. **Completed on 4/17/2023**


Metric #2: EnemiesKilled Metrics
Number of People
Game Levels
Level 1
Level 2
Level 3
Level 4
Level 5
Level 6
Level 7
1
2
3
4
5
6
7


Metric #3: Accuracy Metrics
Number of People
Game Levels
0.01 - 0.25
0.26 - 0.5
0.51 - 0.75
0.76 - 1.0


Metric #4: Bullet Avoidance Metrics
Number of People
Game Levels
0.01 - 0.25
0.26 - 0.5
0.51 - 0.75
0.76 - 1.0

Issue #4 - Level Design should be progressive

Feedback:

Player too difficult to control even after learning

Not able to clear the 1st level

Movement needs to be difficult to control as we move forward in levels not from LEVEL 1

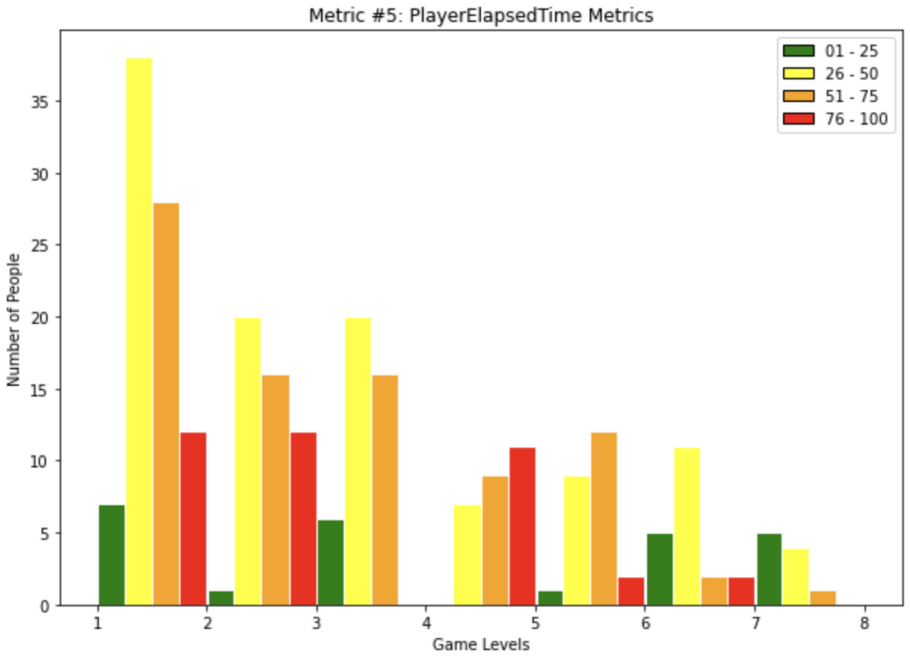
Explanation: Many users are unable to clear a few levels, especially Level 1 because the level seems too difficult for them.

Potential Solutions:

1. **Change thrust and out of bounds to accommodate easy completion of level (completed by 4/9).**
2. **Design initial levels more easily for people to get through (completed by 4/9)**
3. **Add a goal/ game-loop to the game in the form of a boss enemy (completed by 4/17)**
4. Design later levels catering to the fact that people now have learnt the controls.

Reference: Feedback: #25 #26 #33, Metric #5, Looking at the graph we can hypothesize that a lot of users spend time completing level 1 than later levels, hence catering to the audience, we need to design levels more easier in the beginning

Status: **Done**. Started in week 11, **completed by 4/17**


Metric #5: PlayerElapsedTime Metrics
01 - 25
26 - 50
51 - 75
76 - 100
Number of People
Game Levels
0
5
10
15
20
25
30
35
1
2
3
4
5
6
7
8

Issue #5: Collectibles are too hard to reach

Feedback:

not able to collect the collectibles.

Making the collecting points and other objects bigger.

Sometimes arrow didn't pop up so it wasn't clear which collectible I should approach first

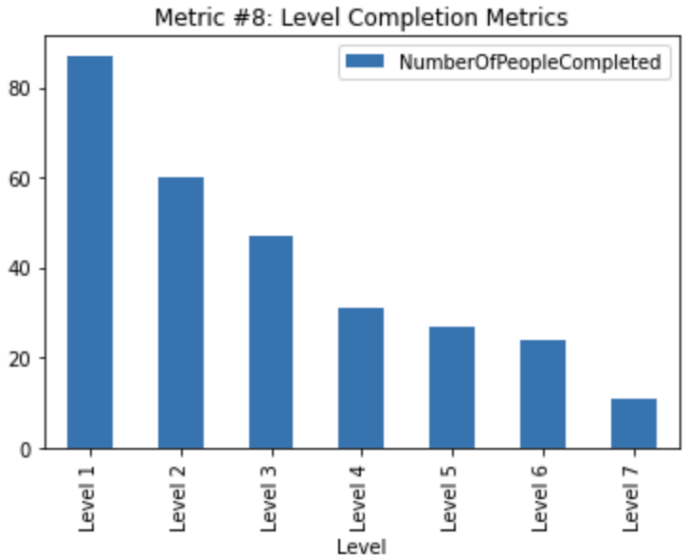
Explanation: The size of collectibles, along with their location in the more expansive levels makes them difficult to obtain.

Potential Solution:

1. **Add reverse thrust and box collider, making it easy**
2. Make collectibles gravitate towards the player when they are in range (a short distance away from the collectible)
3. **Make the level design such that collecting the collectibles are a bit easier, like making the gravity a bit weaker etc.**

Reference: Feedback #12, #18, #22. Metric #8. With this graph we can hypothesize that as the levels get difficult, less number of people are able to collect them and complete the level

Status: Started in week11, **completed by 4/9**



Issue #6 - Have control schema available at all times

Feedback:

> no back button, can't find instructions when stuck

> couldnt figure out the movements & controls when stuck; how to move & proceed further

Explanation: After the starting scene we do not display all the controls. This leads to users getting frustrated if they do not have a way to check the control schema

Potential Solutions:

1. **Have a button (esc button) that navigates to the main menu, which has a controls option that displays all the controls for the game.**
2. Have the control schema on the UI on the bottom right.

Reference: Feedback #38. Metric #5. The total time spent by players is gradually reducing, which when interpreted along with this feedback, we can hypothesize that people are unable to play the game after they are unable to find the control schema.

Status: Started in week 12, **completed by 4/17**

Issue #7 - Instructions for shields did not get through to users

Feedback:

> I liked that the game provided control information before each level. However, I had difficulty reading the information about how to use the shield in level 3 as it disappeared too quickly. It would be helpful if the game included a button somewhere that allows players to access the control information in case they missed it.

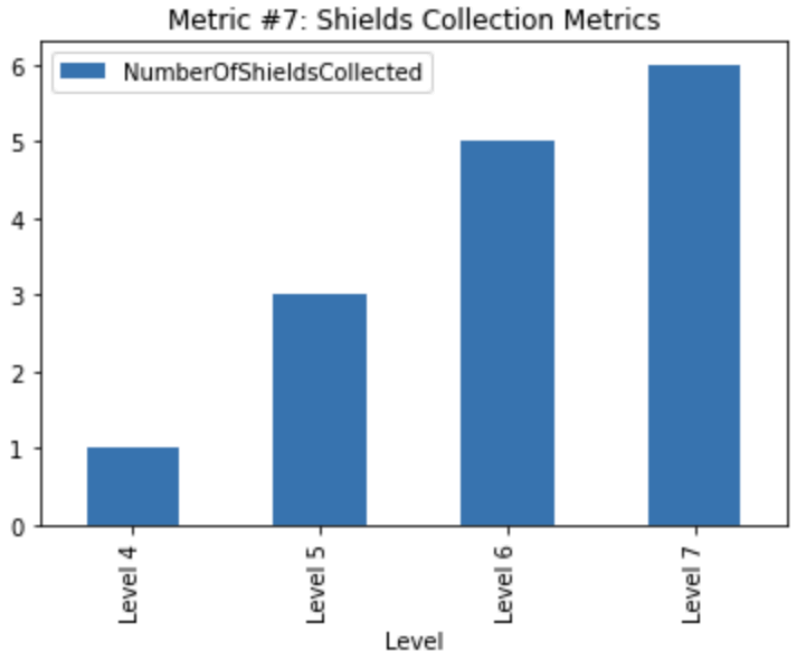
Explanation: People did not realize that there is a shield powerup in level 5.

Potential Solutions:

1. **Get across instructions in a better way. Add a dialogue during the level for a few seconds**
2. **Highlight the shield more in the level to make it noticeable and change the asset to a shield.**

Reference: Feedback #35, Metric: #7 Total number of shields collected across all the players is less than 10, meaning not a lot of the players collected shields.

Status: **Done, completed by 4/17**



Issue #8 - Bullets are too small

Feedback:

> It may be better to make the bullet bigger.

> Bullet not clear

> bullets too small and leave no trace

Explanation: Right now the bullets are too small and are very difficult to see.

Potential Solutions:

1. **Changed the art for the bullet.**
2. Choose a contrasting bullet color to the background
3. Increase size of the bullet

Reference: Feedback #34, #46, Metric: #3. As seen from the graph, the majority of the players in all levels have less than 50% accuracy.

Status: **Done.** **completed by 4/17**

Issue #9 - Spacebar not intuitive as a shoot button

Feedback:

> Shooting was fine, it was straightforward. But, I wonder why the developers have chosen spacebar as the key to shoot. I think it makes more sense and is more intuitive to use the left mouse key to fire, while simultaneously using the mouse to control aiming.

Explanation: Usually the mouse is used for shooting, but since navigating and shooting cannot be easily done on a trackpad, we have used a spacebar, which is not that intuitive.

Potential Solutions:

1. **Change controls to some other button or add more options**
    a. **Key “E” - completed by 4/10**
    b. **Left mouse button - will be completed by 4/17**
2. Place the instructions more strategically or put more emphasis on shooting during tutorials

Reference: Feedback #37, #41. Metric #2, number of enemies killed are less which points to the fact that no one thought to shoot enemies. Around ***13% users*** cite controls are not intuitive and ***23%*** are neutral about how intuitive controls are

Status: **Done.** Started on March 27th, **completed by 4/23**.

Issue #10 - No levels menu

Feedback:

> I didn't find anything frustrating about the game, I would like the ability to replay level 3 and potentially other levels as well. It would be great if the game could include a feature that allows players to select any level they want to play.

> Cannot select levels, have to start from level 1 each time

Explanation: Some players would like to know which levels they have completed and which ones they still need to complete.

Potential Solutions:

1. Add a menu that shows which levels have been cleared and which ones are still available.
2. **Add a level menu from which players can navigate to any level.**

Reference: Feedback #34. Metric #8. From the graph we can see that some users are unaware of how difficult or easy the future levels are, and may want to skip levels.
Status: Started work in week 11. **Completed by 4/9**.

Other feedback issues

Issue #11 - Double toggle presses Issue in Godview mode

Feedback:

> Possibly a bug, but anytime I tried to hit 'Q' at the start of a level, it would zoom me out, then in, and then out. This didn't happen after I started playing.

Explanation: Issue with god view where if user presses Q during the god view, mode gets activated, which leads to a double press.

Potential Solutions:

1. If someone already presses Q at the start, do not enable god view
2. Disable god view toggle from user input

Reference: Feedback: #34
Status: Skipped. (Not a major issue right now, will pick up only if time permits).

Issue #12 - Game text

Feedback:

no back button, can't find instructions when stuck

couldnt figure out the movements & controls when stuck; how to move & proceed further

Explanation: Some players find information in the text unclear and more importantly, find it hard to remember the information given after the text disappears.

Potential Solutions:

1. **Develop a way for users to pause the level and access any information pertinent to that level.**
2. Another solution is to make the text visible in the top right or left corner of the game screen at all times for the user to refer to.

Reference: Feedback #33, user points out the difficulty in understanding the text because it is unclear

Status: Fixed as part of Issue #6 fix, **completed by 4/9**

Issue #13 - Enemy health bar

Feedback:

enemy health bar of some sort. Show the enemies' health status.

Explanation: Currently it takes 5 hits to the enemy in order to kill it but since there is no indicator of the enemy's health on the screen, the players get confused about whether shooting the bullets at the enemies have any effect

Potential Solutions:

1. **Add an enemy health bar on top of the Boss Enemy to indicate their health.**
2. **Animated sprites can be used to indicate an enemy's health by changing the sprite animation when the enemy is hit. For eg, the enemy's sprite could change to a wounded animation or become progressively more damaged as its health decreases.**

Reference: Feedback #52 : indicates that the user gets confused about how to kill the enemies

Status: Done, **completed by 4/17**

Weekly Notes and Feedback

Jan 23

Prototype Work

- Gravity games are cool, and there aren't many space shooters that utilize gravity properly or in an interesting way
- We spent a few min playing wings.io
- We need to make prototypes for the next class
    - Split off into three groups
        - Group 1 (darshan and mehvish):
            - A 2d game with shooting enemies and orbiting around a sphere
        - Group 2 (shreyas, vinay, and kohinoor):
            - A 2d game where you propel yourself from orbit to orbit using a slingshot
        - Group 3 (kavya and anir):
            - A 2d game where the player collects power ups and uses them to strategically destroy enemies
- Our grader's name is chelsea
- Professor Easley and TAs mentioned to us to explore concepts, not artwork or story

Jan 30

Game Notes and Feedback

- Chelsea: Great ideas, game prototypes are cool and exceeds expectations for this week, but make sure to keep making consistent progress week to week
- For POD testing we want to have some basic level design completed, moving/shooting/powerups
- We want to design a HUD that reveals valuable information and is explanatory enough for the player to navigate through the level
- Our game loop should involve clearing the level of enemies and or reaching a goal destination
- As the player progresses the levels should become increasingly complex, and reveal to the player new abilities and techniques for tackling new challenges
- Gravity needs to be less Newtonian, and more fun. Think super mario galaxy style
- Professor: It would have been better to diverge further in our prototypes to test more outlandish ideas than the ones we chose. Because our

prototypes are only testing a single game design, we've effectively barred ourselves from exploring other ideas.

Feb 6

Game Play and Feedback

- Before POD testing, things to fix:
    - W to thrust from orbit is not intuitive, should be able to hold boost button to charge thrusters and let go (thrust should be dependent on long press)
    - Enemies fire at you before they are visible to you
    - Orbit speed is arbitrary and not dependent on entry speed
    - The orbit radius is also not intuitive
- Chelsea:
    - Kept flying off map
    - Didn't understand shooting mechanics vs flight mechanics
    - Confused about not being able to shoot during transit
    - Crashed into planets
- POD Testing:
    - Notes taken by Anir
    - We need to be able to complete a level, each prototype should have actual level design
        - Game should terminate afterwards
        - Game should have very clear win/lose conditions
    - Make sure to answer the professor's questions:
        - The goal is not to defend your game, it is to receive constructive feedback
    - Great Beginnings:
        - Easy to ask basic premise
        - But needs to track to reality a bit more, or pull from physics even more
        - There are already several mobile games that do this
        - Limited gas, using the slingshot effect to get to goal?
        - Burst apart, think of a reasonable premise and core direction for the game to go (the game should not just be go *here*)
        - No great overarching point (needs more mechanics and a better structured game loop)
        - MORE GRAVITTTTTY. "You've got something cool, lean into it"

- 

Feb 13

Game Play and Feedback

- Chelsea:
    - Good progress overall, but lacking in a few areas
        - You need clear level design, and variation and scaling in level difficulty
        - Each level should introduce mechanics to the player slowly, so that players can learn by doing
        - Overall the game should have up to 10 levels
        - We need to perfect visuals to prevent disorientation
            - Remedied by:
                - Minimap
                - Displayable Velocity Vector
                - Current Trajectory given thrust + gravity
        - Game needs a HUD and UI
            - We need to include more user friendly information about the state of the player, and the relative state of the enemies
- POD Testing:
    - Notes taken by Kavya
    - Track if people even know what to do in our game in analytics/ even know how to play our game
        - For example: track how many shots are fired vs how many enemies are killed
    - Have more of a game feel rather than assignments in your game telling player what to do
    - Find a way to display problems and solutions taken to resolve them on the GDD

Feb 27

Issues addressed from previous week

- **Problem:** You need clear level design, and variation and scaling in level difficulty
    - **Solution:** Storyboarded and created different levels in order of increasing difficulty
- **Problem:** Each level should introduce mechanics to the player slowly, so that players can learn by doing
    - **Solution:** Earlier levels have fewer mechanics, such as only one or two enemies and a couple planets with gravity. Later levels introduce more enemies, different types of mechanics such as meteors and asteroids, more planets with more gravity, etc.
- **Problem:** Overall the game should have up to 10 levels
    - **Solution:** Currently have about 6 levels, more will be added by next week.
- **Problem:** We need to perfect visuals to prevent disorientation
    - **Solution:** Added Displayable Velocity Vector and Current Trajectory given thrust + gravity
- **Problem:** Game needs a HUD and UI
    - **Solution:** Added HUD and UI
- **Problem:** Track if people even know what to do in our game in analytics/ even know how to play our game (for example: track how many shots are fired vs how many enemies are killed)
    - **Solution:** Analytics started tracking this
- **Problem:** Have more of a game feel rather than assignments in your game telling the player what to do
    - **Solution:** Changed the UI, and have enemies start shooting at player, indicating the player needs to start moving and shooting at the player
- **Problem:** Find a way to display problems and solutions taken to resolve them on the GDD
    - **Solution:** Added a section in our GDD tracking how we resolved each problem noted the previous week

Game Play and Feedback

- Chelsea:
    - Add more level design and level select menu
        - Ex. let the user know that they're going into level 1, level 2, etc, needing collectibles to go to the next level is not clear
    - Add menu level
- POD Testing:
    - Couldn't figure out W for thrust immediately, tried clicking first

        - Same with spacebar
    - Didn't understand gravity right away
    - No out of bounds, spacial disorientation
    - Too much thrust , "moving too fast to understand"
    - "I know I need to use planet's gravity to slingshot, but I'm not sure what the goal is"
        - User thought we weren't supposed to collect the collectible because we moved away from it
    - Need to figure out how to slow down the game
    - A minimap could help
    - Boundary condition not clear
    - Add a level for just users to get used to movement with planets' gravity
    - velocity to come back from out of bounds into game screen can be really fast
    - In earlier levels, gravity should be really low, so if a user thrusts, the thrust wins over the gravity
    - Log in analytics if velocity exceeds a certain threshold
        - cap velocity in that case
    - Need show a menu before going to the next level
    - Code gets confused between minimap's camera vs. camera on player
- Professor Feedback:
    - No speed is necessary if game is not based off of speed
    - Mouse movement is helpful
    - User won't know where they are on the grid, the grid doesn't help because it's pervasive
        - Out of bound box will help
        - User won't know where to go and what to do
        - Keep it super super simple, because people won't know what to do; **sympathize with who we're giving it to**
    - User needs to know what to do
    - **It's about lack of guidance, not cleverness of game → make it super simple for the user**
    - Make arrowmark green and bigger
    - Make collectibles green, so user knows it's good (red is bad)
    - Text in pause and end level menu is very time
    - Grid needs to be stuck with planets, or be a part of the background
    - **Game is too hard, make it simple**
    - Wall of text will not help, usually ignored; visual explanation is the way to go

March 6th:

- Chelsea's Feedback
    - Game is well built
    - Problem: Thrust is hard to learn but after you do it's fun
        - Solution: Add hard boundaries to the first level
    - Problem: Reduce the learning curve
        - Solution: Unlimited Thrust in the first few levels? No enemies?
    - Problem: Controls are in the first scene, I had to memorize them. Put it in an available place
        - Solution: Put controls on the canvas for at least the first few levels

March 20th:

- Team's hypothesis to fix/improve in the game
    - Implement reverse thrust (from the feedback and metric #1).
    - Bullet Avoidance accuracy is high for the players (from the metric), which suggests that the no. of enemies or their placements are not that tough to play. We can increase them to add more fun & challenge as the level progresses.
    - Player Termination metrics (#1) suggests that the players are having issues with the acceleration or thrust and hence dying from out of bounds. Introducing reverse thrust would help players to be in bounds and have more control over the game.
    - Also, the deaths from Enemies (metric #1) is very rare. We are planning to increase the no. of enemies and strategically place them to improve the challenge and fun.
    - From Metric #3, the accuracy of the players shooting the enemies is low now. Need to discuss as a team how to increase the player's shooting accuracy.
- Professor Feedback
    - Go at the feedback with some suspicion; what do players ACTUALLY mean based on the feedback they're giving
    - Don't address feedback individually response by response, use it to solve overarching problems that are unified by multiple players' feedback
    - A potential solution to mixed bag feedback: give players the option to take more risk for thrill, and allow risk averse players to approach things more cautiously and still complete levels with ease

March 27th:

- Chelsea
    - Feedback should inform some game design and mechanical changes to the build
    - We should strive to include some of the feedback in our next few design iterations based on what we hear about our pod testing presentation
- POD Testing
    - We need to cite the percentage of users might not need help in each issue we create
        - We want a fine level of detail between our issue tracker and our survey feedback
        - Some of the solutions have a long trail to get to the core issues presented in the game, so we should show how we are justifying each hypothesis
    - We need several solutions for each issue, not just one
    - We need more detailed deadlines and planning by the end of the day for each issue

April 3rd:

- POD Testing
    - Feedback from MC DJers:
        - Art was good, and fine with what you have right now, but make it guide the player
        - Add a level screen showing progress
        - Make the thurst bar more informative - when at 0, it seems as if we can still thrust, but it's just because the thrust level is increasing much more
        - Fix the collectible count bar for all levels
        - Make instructions more accessible throughout level
        - Start off level 3 with boss enemy farther away
            - We don't know if the player needs to run away or fight, and seems dangerous either way
        - Want a potentially longer range for the bullets
            - Especially in the beginning levels, so it's easier for the player to learn to shoot. Then in later levels make the range of the bullets shorter to make them harder
        - Level 4: make the planet a little smaller because it's very easy to slingshot out of the range of the planet, which is not what we want in a beginner level → players want to

slingshot around for fun, and s for brake doesn't help much unless we're out of bounds
                - Make the gravity range smaller in level 4, and have the player halt faster
            - Level 5:
                - LEAVE MORE MARGIN BETWEEN PLANET AND OUT OF BOUNDS
                - Slingshot more intuitive in this level though
            - Main feedback: add more margin between the planets and boundaries
            - Add shockwaves to show collisions with bullets/ objects
    - Feedback from Trojan Adventures:
        - S to stop is pretty baller
        - First two level implementation is good, makes it easier to learn
        - Bullet tracing the player: should be easier, the bullet shouldn't follow the player
        - *Killed super enemy*
            - Add health bar for stronger enemies
        - Very hard to control
        - Didn't know how to brake even after reading instructions
        - No way to choose levels at the moment
        - Still can't control in curves
        - Collectibles are too hard to reach

- Professor Feedback
    - (For other teams)
        - Make changes to your game that are consistent and pervasive, don't introduce something that you use for 2 sec and never use again
        - Doom is an early 3d FPS shooter game that takes place in mazes
            - Don't make your game super similar to another game, make it unique (too derivative)
    - Planet in level 3 pixelated
        - Make your art very creative
        - Art is too tiny
    - **Make gravity be used as intended → we want people to use the gravity to collect collectibles rather than another thing to maneuver**
        - Add arrows to maybe encourage player to go in the direction you want them to
    - **Use art to provide more guidance**

    - Prioritize feedback from survey results, and then professor feedback to elevate it a bit more

April 10th:

Tasks Completed

- **Issue #2 - Out of Bounds Death is very frequent in initial levels.**
    - Add a collider for out of bounds so that players stay inside the bounds.
- **Issue #4 - Level Design should be progressive**
    - Change thrust and out of bounds to accommodate easy completion of level.
    - Design initial levels more easily for people to get through..
- **Issue #5: Collectibles are too hard to reach**
    - Add reverse thrust and box collider, making it easy.
    - Make the level design such that collecting the collectibles are a bit easier, and make the gravity a bit weaker.
- **Issue #6 - Have control schema available at all times**
    - Have a button (esc button) that shows all the controls for the game.
- **Issue #9 - Spacebar not intuitive as a shoot button**
    - Change controls to some other button or add more options (key “E”).
- **Issue #10 - No levels menu**
    - Add a level menu from which players can navigate to any level.
- **Issue #12 - Game text**
    - Develop a way for users to pause the level and access any information pertinent to that level → also addressed by solution for issue 6.

Chelsea’s Feedback

- Guidance in the levels is good
- Out-of-bounds fixes are good

POD Testing

- Feedback from MC DJers:
    - Add visual indication to show braking doesn’t work in gravity

        - Tweak the thrust and brake values to make changes with key presses more instantaneous
            - Right now, its takes 2 - 3 seconds to slow down with “w” and “s”.
        - Player wobbles a bit too much with higher thrust values (not an issue though)
        - Make users need the shield more; as of right now, the shield doesn’t seem to be very important
        - Tutorials are good
        - Change the art for the boss enemy to make it go with the vibe
        - Level 3: make it just the enemy or collectibles, not both
        - Gravity, thrust bar, and enemies are good
        - Can potentially get weapon after killing the boss enemy or collectible from regular enemy
        - Put bound for the boundary, like a force field or something
- Feedback from Trojan Adventures:
    - Instructions and arrows on the screen in earlier levels are very straightforward (good)
    - Range for the bullets is a good idea
    - Add health bar for the boss enemy so the player knows whether the enemy can be defeated
    - Shield art is good
    - Speed of the boss enemy currently is good

Professor Feedback

- Art in game is contrary to your art style
    - Make bigger more valuable art
    - Change the “slingshot” name to ScribbleSpace
    - “I was promised something on the flash screen, and I don’t get the promise of being slung around by orbits when I’m actually playing the game”
- There isn’t enough intention behind the placement of orbits and where people should move
    - It’s like there was supposed to be a roller coaster but instead I got something less thrilling
- Level design should utilize more slingshotting, “you’re sitting on a rollercoaster, bring it out”

April 17th:

Tasks Completed

- **Issue #3 - Enemies are not a central part of the game**
    - Can trap enemy in gravity and create strategies around it
- **Issue #4 - Level Design should be progressive**
    - Changed level design for levels 6,7,8
- **Issue #8 - Bullets are too small**
    - Changed the art and bullet instantiation point.
- **Other Bug fixes**

Pod Testing

- MA
    - Text on instruction screen is very small
    - Remove quit as we have main menu
    - Q doesn't work in level1 & level 2
    - Level 3 doesn't teach how to shoot at enemies. Need more intuitiveness. Add help text
    - When the player presses S then art or effect is required to give the feeling that the spaceship is stopping. Like how we have fire for w
    - Shield yellow highlight should be bigger
    - Level 9 fix placement of thrust bar & health bar
    - When enemies hit you or player collides with asteroid add effects which gives the feel to player that the player is taking damage
    - Text behind "you won" screen . (Warning message) Remove that
    - Last level is fun to play
    - Bullets get hidden behind planets.
    - Player is unsure about killing enemies or collecting collectibles to end a level. Would prefer if enemies had to be killed mandatory along with collecting collectibles
    - Would like more challenging level design
    - Prefer having out of bounds play, it's too easy now
    - Only way to die is from enemies
    - Too much instructions, can't remember s to brake, forgetting buttons
    - Inconsistent control instructions
    - More deterministic movement, don't like floating
    - No incentive to shoot enemies, they hardly bother the player
    - Not sure if I'm doing damage to the enemy, not sure if I'm taking damage

    - Controls are irritating, the movement is very unforgiving
    - Health pickups would make the shooter aspect a lot more interesting
    - Last level canvas is messed up, health and thrust are in the wrong area
    - Overall gameplay is very good, if irritating at times
        - High gravity zones are too difficult depending on the player
        - Collectibles need more forgiveness
- Ss
    - The enemy placement in level3
    - The order of collectibles is confusing along with the quest pointer
    - Collision with the enemy should take some damage
    - The current order of levels is good
    - Level 3 decrease difficulty
    - Did not notice the instruction screen
    - For ppl who need more thrust coming back, OOB change was good
    - Noticed the zero thrust going below planet bug
    - Escaping is easier than shooting
    - Liked god view better. Increase collectible size in god view
    - Wants hiding behind asteroid fields
    - Wants gravity to make enemies oscillate
    - Quit doesn't work on any level
- Professor Feedback
    - It's very clever we can trap the UFOs
        - Make it even more clear
            - It feels like you're just expected to know it
        - Make them gray, or orbit around the planet
    - Collectibles need to have a different asset, space booger not appealing
    - Don't challenge the player to figure something out, communicate what to do
    - Boss Health bar needs to be introduced earlier