Iterium Documentation

Revision: 1

Table of Contents

1. Game Overview

Genre	Top-down space shooter
Theme	1980s Arcade
Perspective	2.5D
Game engine	Unity (version 2021.3.15f1)
Scripting language	C#
Build targets	WebGL – Desktop browsers
License	GNU General Public License v3.0 – See here
Gameplay	The game follows a typical arcade genre where you try and survive as long as possible to gain the highest score and be entered on the leaderboard. Score can be achieved by shooting asteroids, collecting Iterium crystals, killing AI ships and Xoid flying Saucers. At the start of the arena battle you will receive 3 ships (lives) and the battle will end when you lose all your lives. Asteroids do not do damage to your ship but will push your ship from course depending on the weight of the selected ship and size of the asteroid. Enemy fire can be destroyed with your own fire and damage can be reduced by deploying your shield. When you or the AI are destroyed, a re-spawn delay will provide a short reprieve to the opponent to collect any floating crystals and rack up further points. Iterium crystals are dropped randomly by destroying asteroids, crystals are used to upgrade the firepower, shield, and speed of your spaceship. Picking up crystals before your opponent is crucial to prevent them from also upgrading, increasing your maneuverability, offensive and defensive capabilities will give you the edge in battle. There are 3 faction ships to choose from, each having their own balance of firepower, shields and speed. Each of the ship properties can be upgraded twice, the USSR Hammer is the heaviest ship and will be less impacted by asteroids while the lightest Chinese Rapier, although the fastest ship will struggle against flying debris. Each of the ship's statistics can be viewed on the faction selection screen, including the status after upgrading.

1.1. Game Story

You are a space miner in the year 2055, traveling through the galaxy in search of rare and valuable Iterium crystals. These crystals are a crucial source of energy for space travel by an alien race called Xoids, and the demand for them is skyrocketing. Your goal is to collect as many of these crystals as you can and become the richest space miner in the galaxy.

As you travel from planet to planet, you hear rumors of a large asteroid belt rich in Iterium crystals. With excitement, you set your course for the belt and begin your search. The asteroids are treacherous, filled with danger at every turn, but the reward is worth the risk.

You equip your mining ship with the latest technology, including offensive weaponry with enough firepower to split asteroids. With each successful mining mission, your ship fills with Iterium crystals, but your journey is far from over. As you delve deeper into the asteroid belt, you encounter rival miners and Xoid UFOs who will stop at nothing to steal your valuable cargo. You must defend yourself and your ship from their attacks, using your maneuverability, firepower, and shields to fend them off.

You are determined to come out on top of the leaderboard, collecting as many crystals as you can and earning a place in the annals of space mining history. Will you become the richest space miner in the galaxy, or will you be defeated by the dangers of the asteroid belt? The fate of your journey rests in your hands.

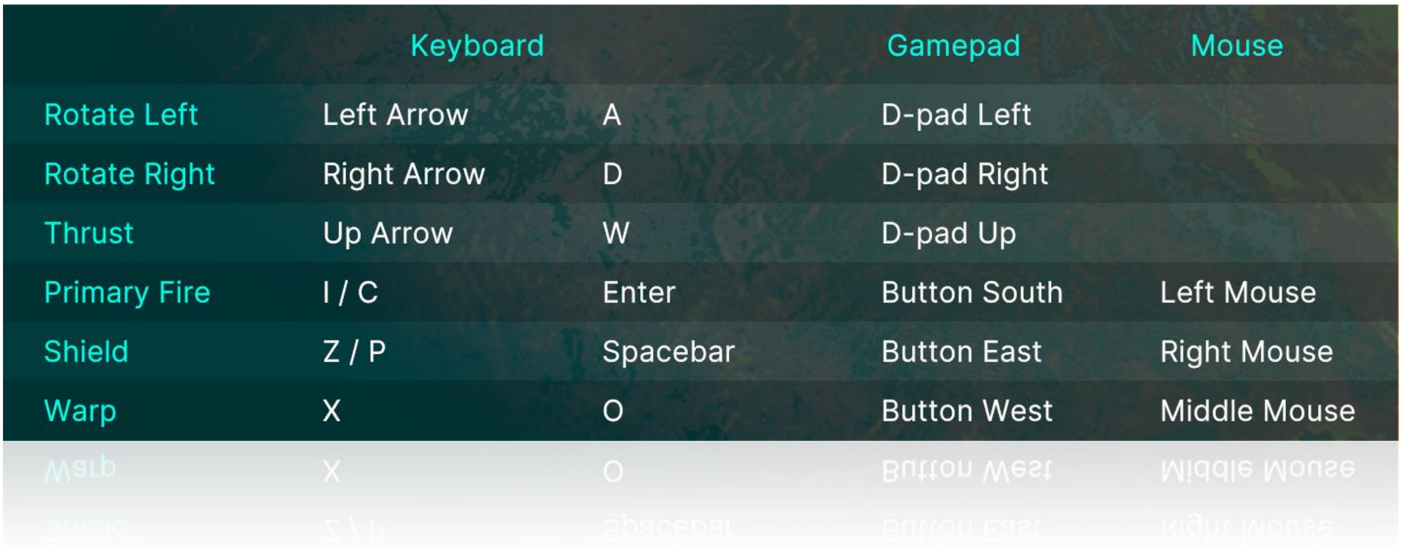
1.2. Factions and ships

The Sword belongs to the USA faction and is the most balanced of the 3 ships. The ships most impressive feature is its firepower, medium thrust capabilities with its shields being its weakest feature. Good balance of attack and maneuverability when fully upgraded. Weight is in the medium range and impacts how you are affected by the weight of an opposing asteroid.

The Hammer is the pride of the USSR faction and is built like a tank, very slow in maneuverability but makes up for it with its superb shields and medium fire power capabilities. A good ship for attack and defense during heated battles when fully upgraded. Heaviest weight of all the ships, impacts how you are affected by the weight of an opposing asteroid.

The Rapier belongs to the Chinese faction, is sleek in design with blazingly fast thrusters. Firepower is the weakest of all ships with a medium shield capability. A good ship to avoid attack and quick Iterium collection. The lightest ship which impacts how you are affected by the weight of an opposing asteroid.

1.3. Controls



	Keyboard		Gamepad	Mouse
Rotate Left	Left Arrow	A	D-pad Left	
Rotate Right	Right Arrow	D	D-pad Right	
Thrust	Up Arrow	W	D-pad Up	
Primary Fire	I / C	Enter	Button South	Left Mouse
Shield	Z / P	Spacebar	Button East	Right Mouse
Warp	X	O	Button West	Middle Mouse

1.4 Scoring & Levels

Destroying NPC

Score + 1500

XP + 100

Destroying Player

Score + 500

XP + 50

Destroying Asteroid

Score + 50

XP + 10

Collecting Iterium

Score + 250

XP + 25

Points Bonus (Game Over scene)

At the end of each battle you will receive a points bonus calculated as (Iterium collected x 100) x level.

Leveling

Every 20000 XP gained.

1.5. Upgrading

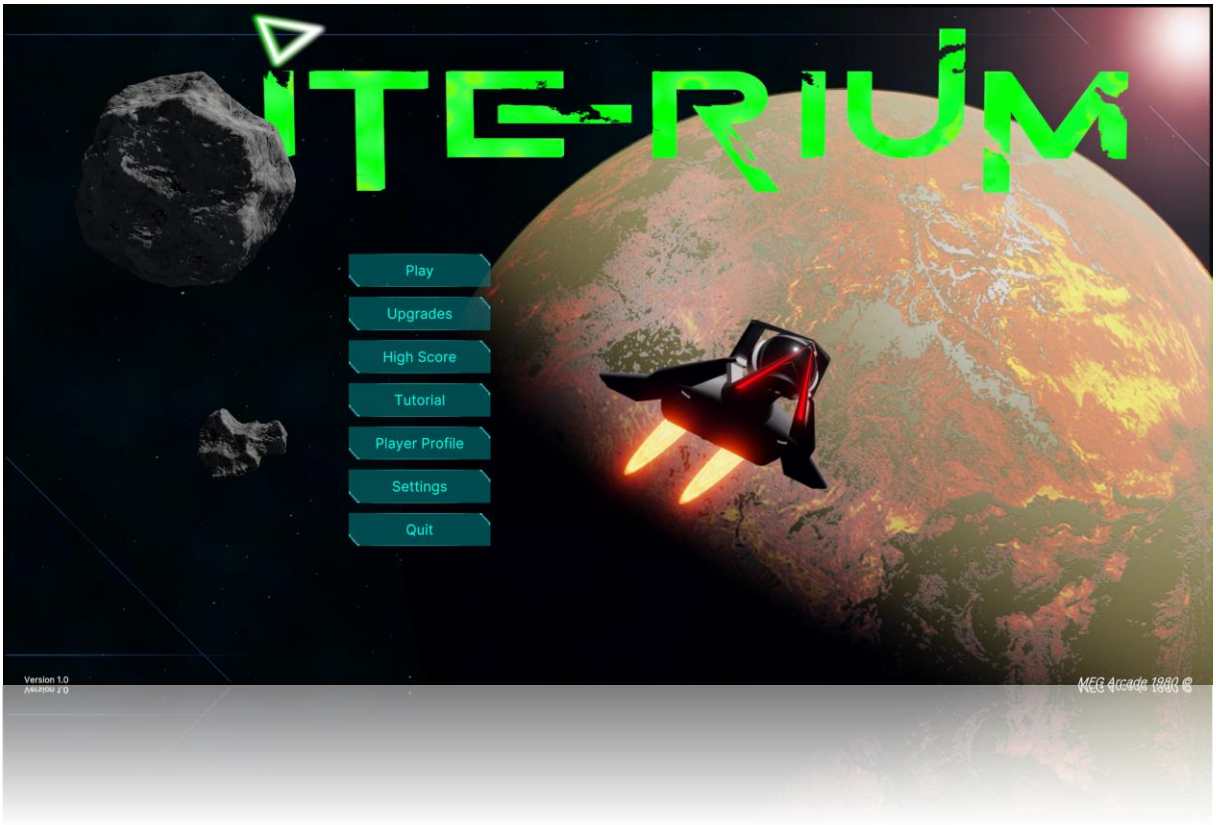
Upgrades are split into 3 different categories that can be upgraded twice each:

- Thrust
- Shields
- Firepower

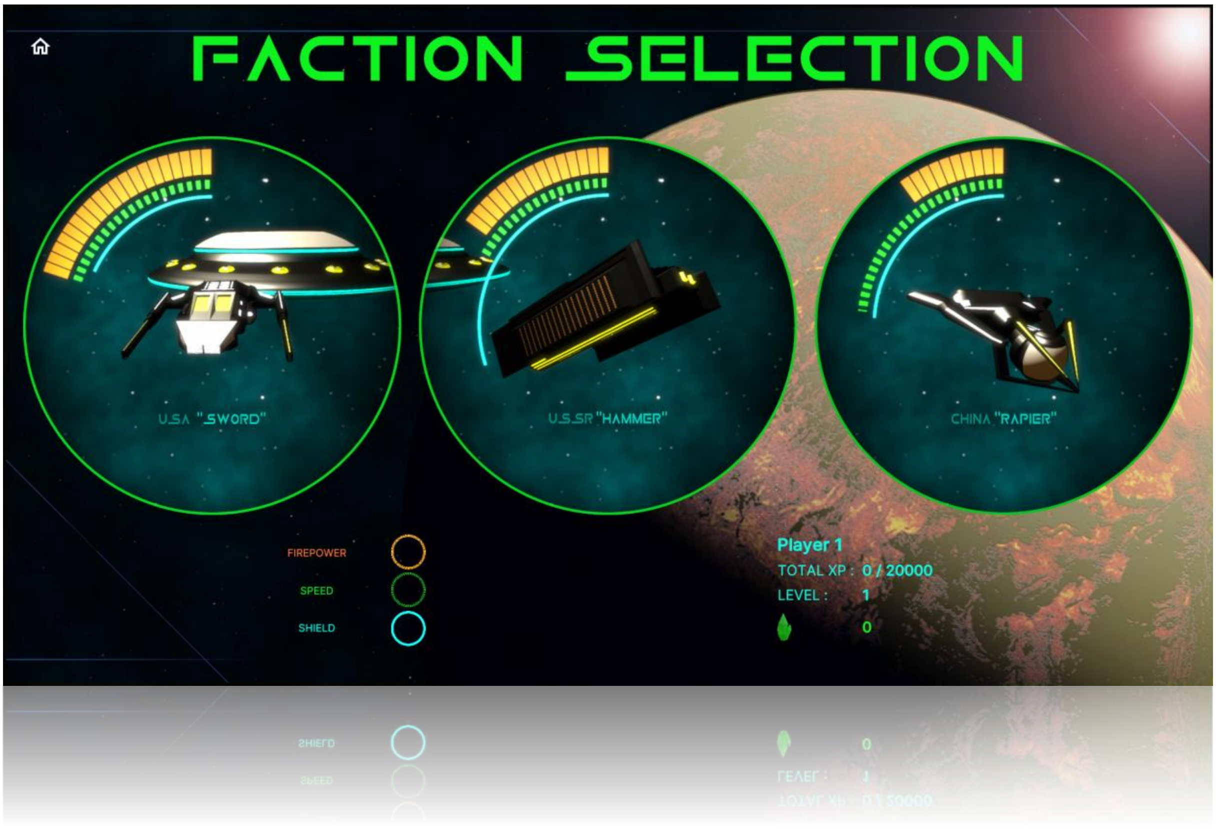
You will need the required amount of Iterium indicated by each category slider on the upgrade screen. Iterium is dropped by destroying asteroids, there is a 20:1 chance of a drop.

1.6. Screenshots

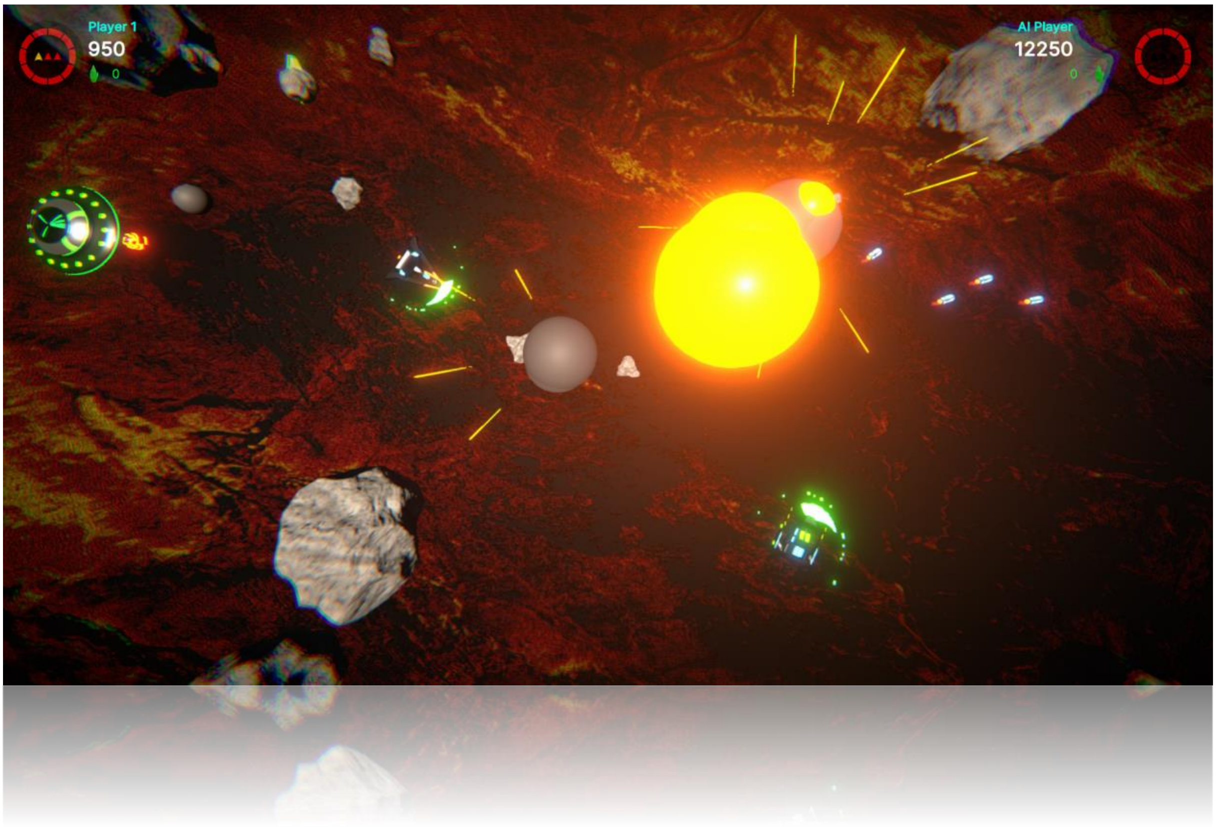
1.6.1. Main Menu



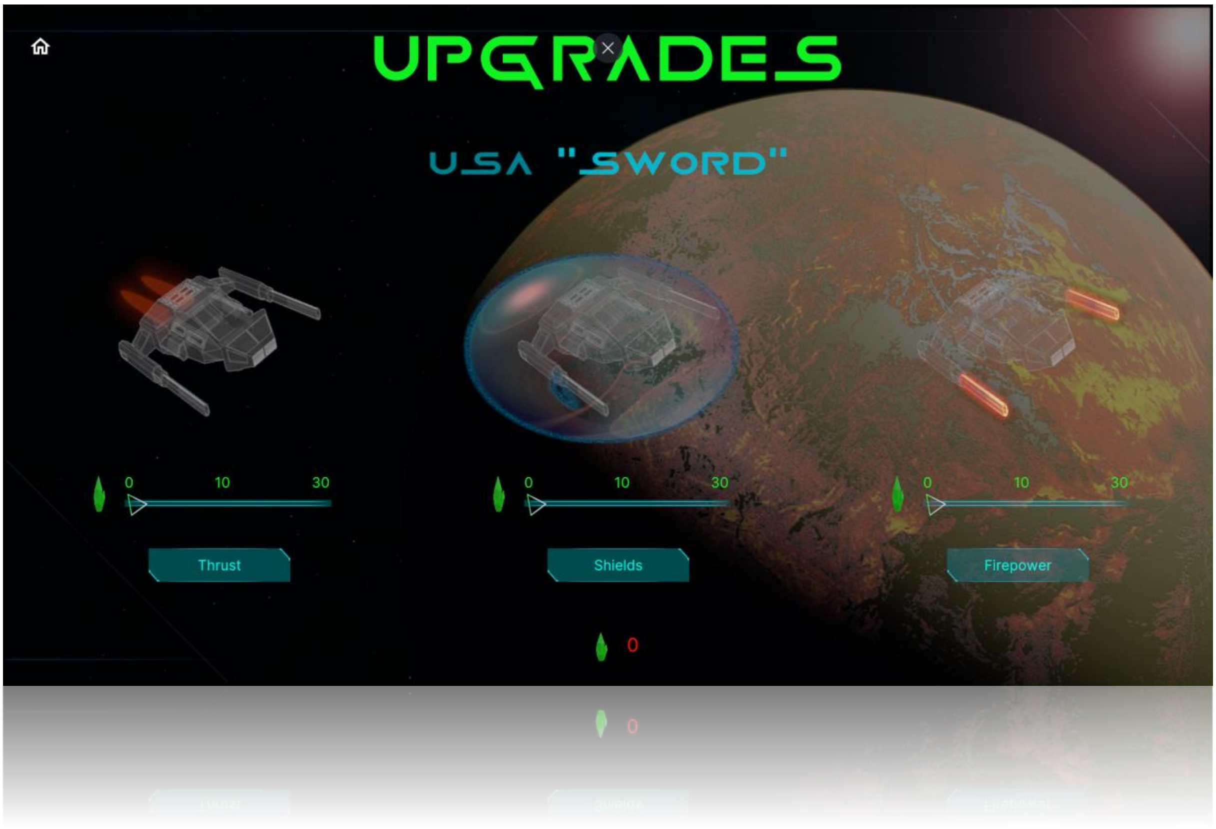
1.6.2. Faction Selection



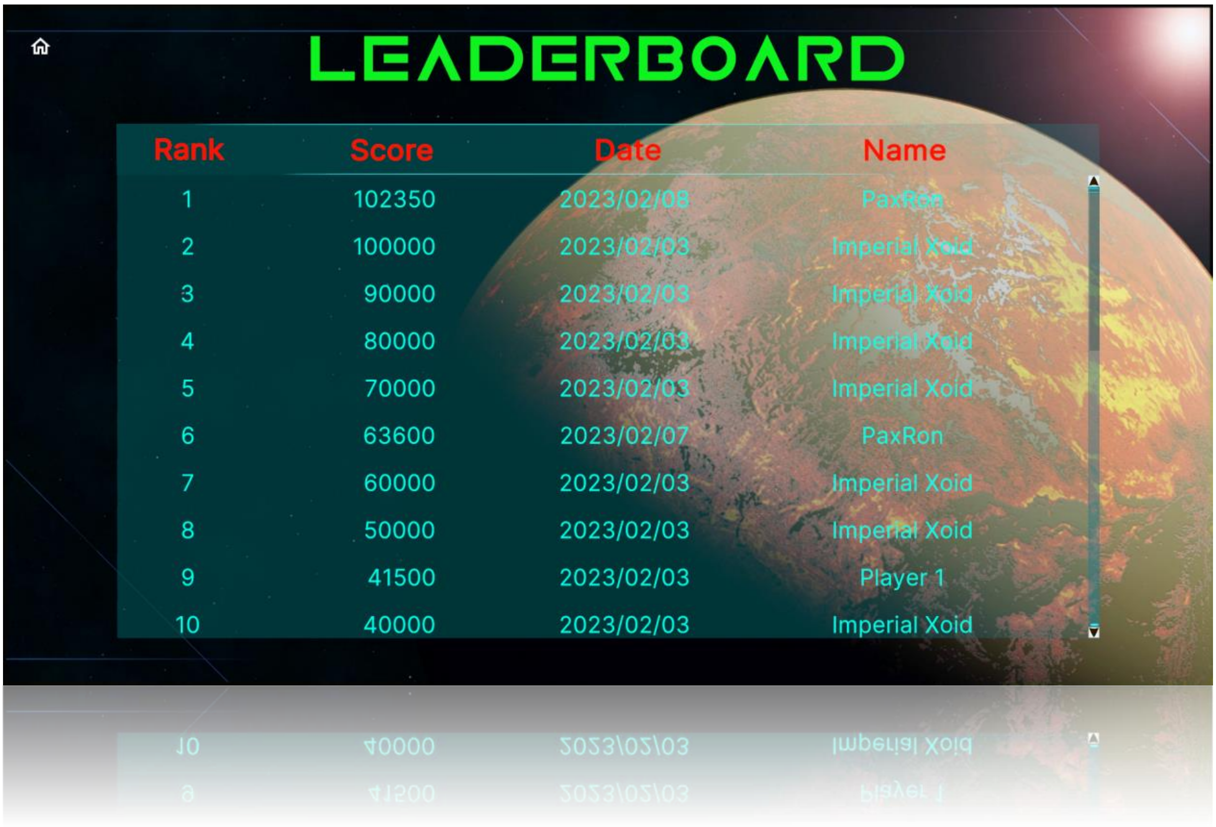
1.6.3. Game



1.6.4. Upgrades

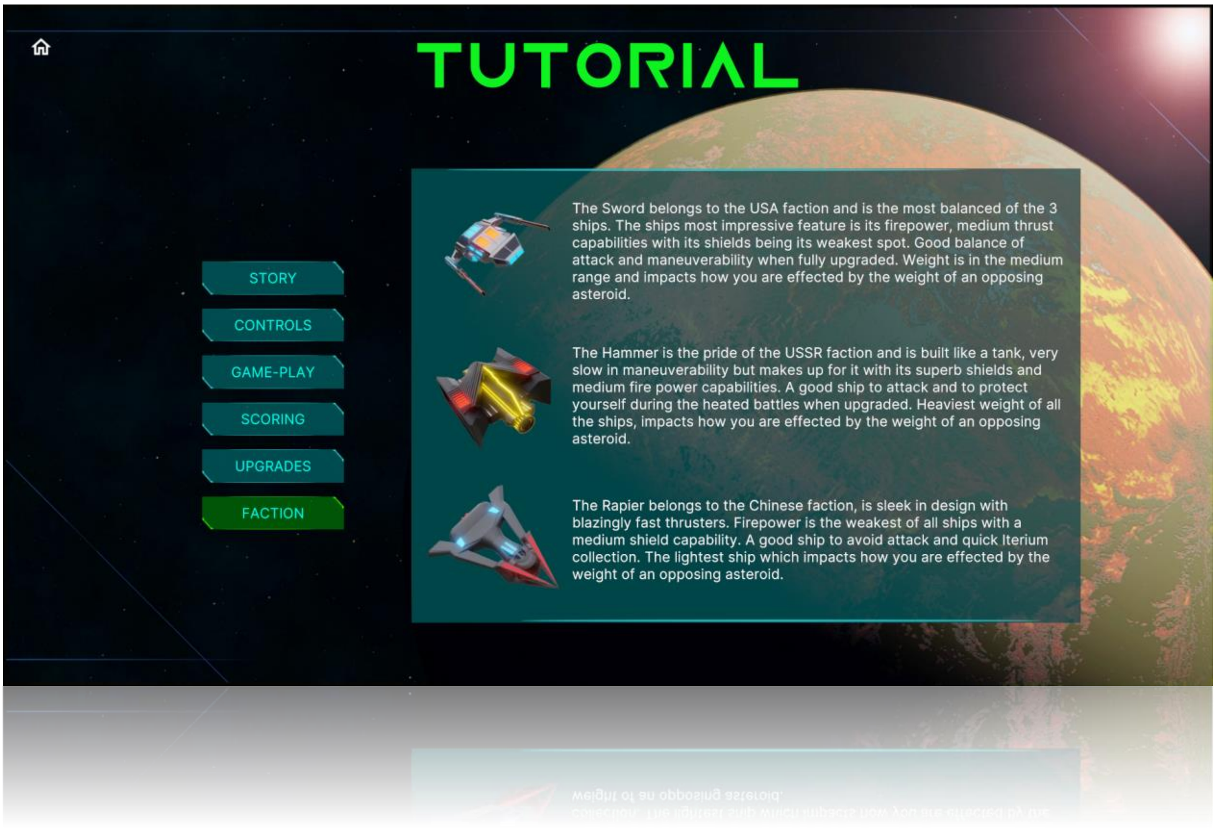


1.6.5. High Score


LEADERBOARD

Rank	Score	Date	Name
1	102350	2023/02/08	PaxRon
2	100000	2023/02/03	Imperial Xoid
3	90000	2023/02/03	Imperial Xoid
4	80000	2023/02/03	Imperial Xoid
5	70000	2023/02/03	Imperial Xoid
6	63600	2023/02/07	PaxRon
7	60000	2023/02/03	Imperial Xoid
8	50000	2023/02/03	Imperial Xoid
9	41500	2023/02/03	Player 1
10	40000	2023/02/03	Imperial Xoid

1.6.6. Tutorial

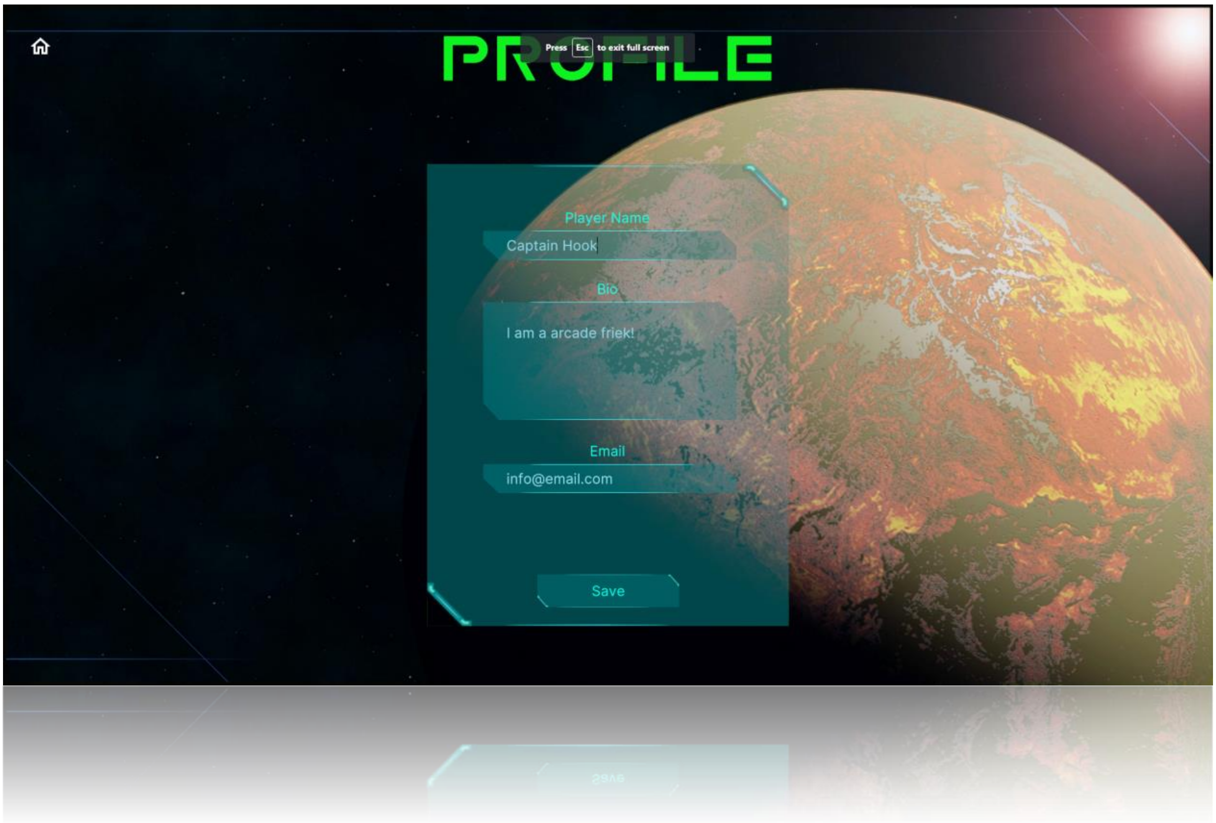


The Sword belongs to the USA faction and is the most balanced of the 3 ships. The ships most impressive feature is its firepower, medium thrust capabilities with its shields being its weakest spot. Good balance of attack and maneuverability when fully upgraded. Weight is in the medium range and impacts how you are effected by the weight of an opposing asteroid.

The Hammer is the pride of the USSR faction and is built like a tank, very slow in maneuverability but makes up for it with its superb shields and medium fire power capabilities. A good ship to attack and to protect yourself during the heated battles when upgraded. Heaviest weight of all the ships, impacts how you are effected by the weight of an opposing asteroid.

The Rapier belongs to the Chinese faction, is sleek in design with blazingly fast thrusters. Firepower is the weakest of all ships with a medium shield capability. A good ship to avoid attack and quick Iterium collection. The lightest ship which impacts how you are effected by the weight of an opposing asteroid.

1.6.7. Player Profile

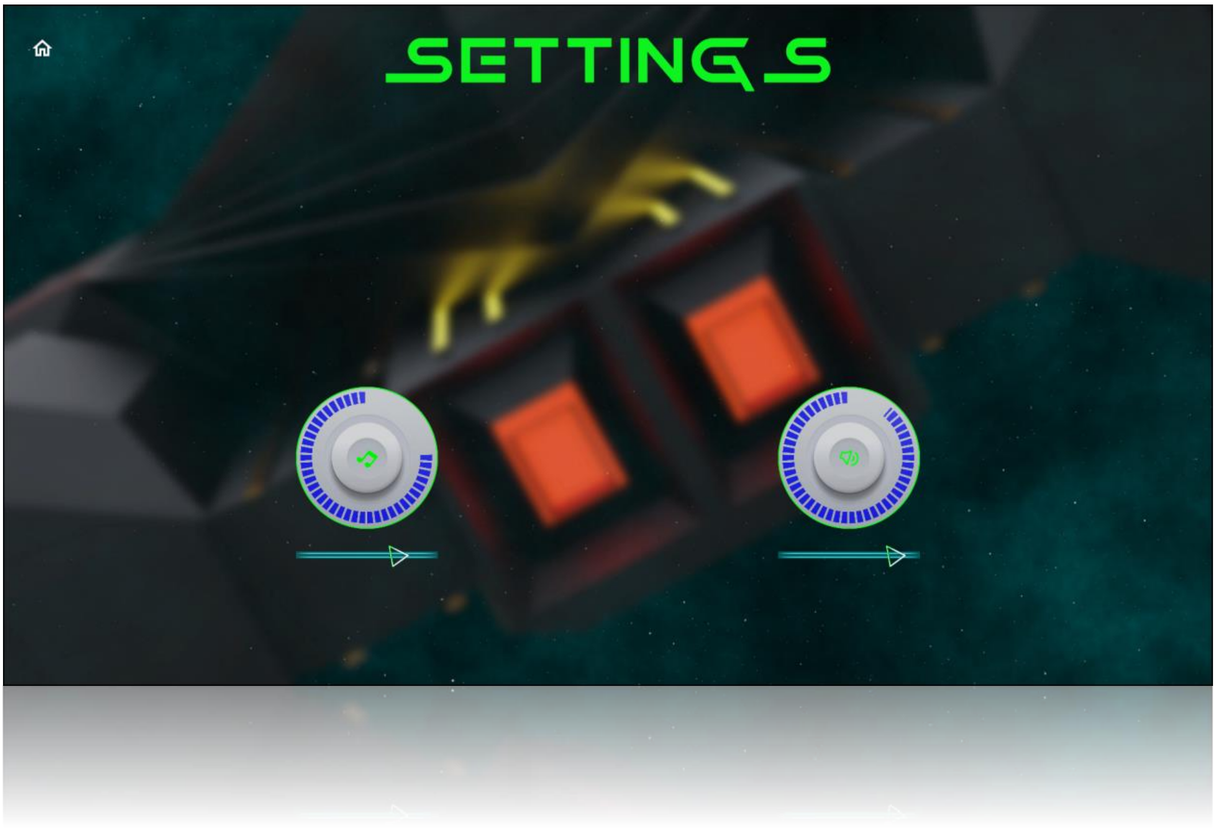


Player Name
Captain Hook

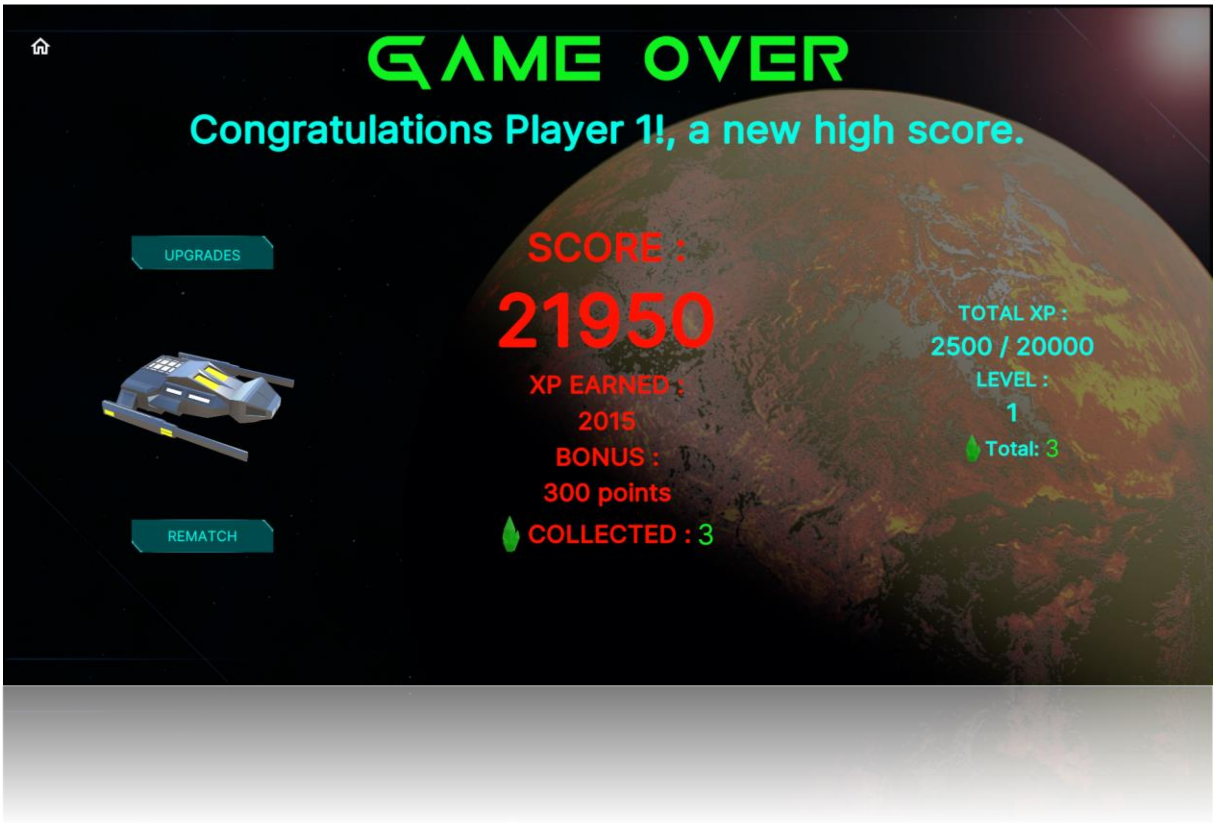
Bio
I am a arcade friek!

Email
info@email.com

1.6.8. Settings



1.6.9. Game Over



Congratulations Player 1!, a new high score.

SCORE :

21950

XP EARNED :

2015

BONUS :

300 points

COLLECTED : 3

TOTAL XP :

2500 / 20000

LEVEL :

1

Total: 3

2. Unity Project

2.1. Project requirements

- Unity version 2021.3.15f1
- Render Pipeline: URP 12.1.8
- Input System (new)
- UI Toolkit
- Timeline
- Build platform: WebGL

2.2. Folder Structure

Assets path sub folders	Description
3DModels	Ship and asteroid models
3Dmodels/Bullets	Bullet models
3Dmodels/Crystal	Iterium crystal model
3Dmodels/Environment	Planet model
3Dmodels/Materials	All model materials
3Dmodels/Textures	All model textures
Animations	Animator/timeline files
Fonts	Font files
Images	Sprites for UI
Plugins	.wslib JavaScript plugin
Prefabs	All prefabs
RenderTextures	RenderTextures used to display models and shader graph progress bars
Scenes	All Unity scenes
ScriptableObjects	Created assets that hold the data
ScriptableObjects/Bullets	3 levels of bullets per faction, Xoid bullet
ScriptableObjects/Factions	Factions – China, USA, USSR and Xoid
ScriptableObjects/Common	Common data and lists
ScriptableObjects/Common/Faction Types	Faction and NPC types
ScriptableObjects/Common/Lists	Lists of GameObjects and factions
ScriptableObjects/Common/Sound	Lists of sound effects and music
ScriptableObjects/Players	Player and AI data, holds the selected faction character SO
ScriptableObjects/Ships	Ship data, hold the ship bullet SO.
Scripts/	All C# scripts
Scripts/Characters	Player controllers, leaderboard, bullets, Iterium
Scripts/General	Utility script like rotating an object
Scripts/Interfaces	IDamage interface for character damage
Scripts/Managers	Game and sound managers, object pooling
Scripts/Save System	Save/load system
Scripts/ScriptableObjects	ScriptableObject scripts
Scripts/Spawning	Player/AI/NPC spawners and de-spawners
Scripts/UI	UI scene scripts (UI Toolkit)
Settings	URP render pipeline assets

Settings/Import	Assets import templates
Settings/Input System	Input system asset (New Input System)
Settings/Renderer	URP renderers
Settings/Sound	Sound/music mixer
ShaderGraphs	Custom shaders for ship thrust effects and radial progress bars for the UI (health/shields/firepower/speed)
Sounds	All audio files
Sounds/Effects	Sound effect files
Sounds/Music	Music files
UI	UI Toolkit files
UI/Documents	UI Toolkit Documents (UXML)
Styles	UI Toolkit styles (uss)
UI Toolkit	UI Toolkit themes
WebGL Templates	WebGL custom loader (HTML)

2.3. Classes & Assemblies

Most classes are in the Iterium namespace and form part of the Assembly-CSharp assembly, the Scripts/General folder contains a “General” assembly definition asset and the Scripts/ScriptableObjects folder contains an “SO” assembly definition asset.

Class Name	Description	Derived From	Assembly
AIController	Controls AI movement, shielding, firing etc. Added dynamically to a ship prefab when AI is spawned	MonoBehaviour	Assembly-Csharp
Asteroid	Asteroid damage/splitting, de-spawn, Iterium drop. Added to the asteroid prefabs	MonoBehaviour	Assembly-Csharp
PlayerController	Controls player movement, shielding, firing etc. Added to all ship prefabs	MonoBehaviour	Assembly-Csharp
NPCController	Controls movement and firing for Xoid UFO. Added to the Zoid prefab	MonoBehaviour	Assembly-Csharp
BulletBase	Bullet base class, asteroid destruction, bullet explosions, Iterium drop etc.	MonoBehaviour	Assembly-Csharp
Bullet	Player bullet, collision detection. Added to all bullet prefabs	BulletBase	Assembly-Csharp
BulletAI	AI bullet, collision detection. Added to a bullet prefab dynamically by the AI bullet pool	BulletBase	Assembly-Csharp
BulletNpc	Xoid bullet, collision detection. Added to the Zoid_Bullet prefab.	BulletBase	Assembly-Csharp
Iterium	Collision detection for item collection. Added to the Iterium prefab	MonoBehaviour	Assembly-Csharp
SO_Leaderboard	List<LeaderboardItem> high score leaderboard	ScriptableObject	Assembly-Csharp
LeaderboardItem	Leaderboard item fields / row template		Assembly-Csharp
AsteroidPooling	Asteroid spawn/de-spawn object pooling. Added to the Scripts empty gameObject in the Play scene	MonoBehaviour	Assembly-Csharp
BulletPooling	Bullet spawn/de-spawn object pooling. Added to the Scripts empty gameObject in the Play scene	MonoBehaviour	Assembly-Csharp
ExplosionPooling	Explosion spawn/de-spawn object pooling. Added to the Scripts empty gameObject in the Play scene	MonoBehaviour	Assembly-Csharp
Singleton	Singleton<T> base class	MonoBehaviour	Assembly-Csharp
GameManager	Player/AI objects, save/load, xp/level calculation, game settings, leaderboard, etc. Added to the Manager prefab	Singleton<T>	Assembly-Csharp
SoundManager	Playing of sounds and music. Added to the Manager prefab	Singleton<T>	Assembly-Csharp
InputManager	Device input for fire, thrust, rotate, shield etc. Added to ship prefabs	MonoBehaviour	Assembly-Csharp
PlayMusic	Play sound effect by index from the SoundManager	MonoBehaviour	Assembly-Csharp
FileSaveHandler	Load/save serialized JSON data to a file, 3 save files for Player/AI/Leaderboard, object instantiated by the GameManager		Assembly-Csharp
SaveData	Player/AI save data fields, objects instantiated by GameManager		Assembly-Csharp
AISpawner	Spawn the AI prefab after a set time, added to the AISpawner gameObject in the Play scene under Spawners gameObject	MonoBehaviour	Assembly-Csharp
AsteroidSpawner	Spawn asteroid prefabs at set intervals, added to various gameObjects under the Spawners gameObject in the Play Scene	MonoBehaviour	Assembly-Csharp

DeSpawnExplosion	Release explosion after a set time, added to the Bullet_Explosion prefab	MonoBehaviour	Assembly-Csharp
GameOverSpawner	Spawn selected ship on the game over scene, added to the Spawner gameObject in the GameOver scene	MonoBehaviour	Assembly-Csharp
NPCSpawner	Spawn xoid UFO at set intervals, added to the NPCSpawner gameObject in the Play scene under Spawners gameObject	MonoBehaviour	Assembly-Csharp
PlayerSpawner	Spawn player prefab after a set time, added to the PlayerSpawner gameObject in the Play scene under Spawners gameObject	MonoBehaviour	Assembly-Csharp
UI_Button	Generic button script to load a scene	MonoBehaviour	Assembly-Csharp
UI_FactionSelection	UI for the faction selection scene	MonoBehaviour	Assembly-Csharp
UI_Game	UI for gameplay scene	MonoBehaviour	Assembly-Csharp
UI_GameOver	UI for game over scene	MonoBehaviour	Assembly-Csharp
UI_Leaderboard	UI for the leaderboard scene	MonoBehaviour	Assembly-Csharp
UI_LoadUpgradeScene	Button script to load the upgrade scene	MonoBehaviour	Assembly-Csharp
UI_Profile	UI for the player profile scene	MonoBehaviour	Assembly-Csharp
UI_QuitGame	Button script to quit the application	MonoBehaviour	Assembly-Csharp
UI_Settings	UI for the settings scene	MonoBehaviour	Assembly-Csharp
UI_Tutorial	UI for the tutorial scene	MonoBehaviour	Assembly-Csharp
UI_Upgrade	UI for the upgrade scenes	MonoBehaviour	Assembly-Csharp
CameraZoom	Zoom camera when starting the play scene	MonoBehaviour	General
PlaySFXDelay	Play a AudioSource with a delay	MonoBehaviour	General
Rotate	Rotate any GameObject, example asteroids	MonoBehaviour	General
SO_Bullet	Bullet data, speed, firepower, prefab	ScriptableObject	SO
SO_Faction	Faction data, SO_Ship, factionId	ScriptableObject	SO
SO_FactionList	List<SO_Faction> of factions	ScriptableObject	SO
SO_GameObjects	List<GameObject> asteroids, explosions, Iterium	ScriptableObject	SO
SO_NPC	NPC data, Xoid ship	ScriptableObject	SO
SO_Player	Player/AI data, score, xp, health, lives, upgrades, etc	ScriptableObject	SO
SO_SFX	List<AudioClip> sound effects	ScriptableObject	SO
SO_Ship	Ship data, bullet, speed, shield power, prefab, etc	ScriptableObject	SO
SO_Types	Generic types or categories	ScriptableObject	SO

2.4. Save System

2.4.1. Save files and methods

The system saves all data in 3 separate JSON files, namely Player.save, AI.save and Leaderboard.save. The file names can be changed in the GameManager inspector.

GameManager Method()	**Function**
SaveGame	Save player and AI data to a JSON files
LoadGame	Load player and AI from a JSON files
SaveLeaderboard	Save leaderboard entries to a JSON file
LoadLeaderboard	Load leaderboard entries from a JSON file

2.4.2. Path

The save path is set as Application.persistentDataPath by passing the value to the FileSaveHandler constructor when the new object is created inside the GameManager Start() method.

2.4.3. Player data

The SaveData class, contains all player data fields that are saved. Two objects are created from this class inside the GameManager and passed to the FileSaveHandler for saving…

GameManager Object	**Function**
saveData	SaveData class object to store the Player data for saving to file
saveDataAi	SaveData class object to store the AI data for saving to file

2.4.4. Leaderboard data

The LeaderboardItem class contains the save data fields for each row in the leaderboard. ScriptableObject SO_Leaderboard contains a List<LeaderboardItem> that stores the leaderboard rows, this list is passed to the FileSaveHandler for saving.

Object	**Function**
Leaderboard	ScriptableObject with a List<LeaderboardItem> used for storing the leaderboard items
leaderboard.Leaderboard	GameManager Leaderboard List<> instance that is passed to the FileSaveHandler for saving to the file

2.4.5. File save handler

The FileSaveHandler class writes the data to text files in JSON format.

Method()	**Function**
Load	Load Player and AI data from file
Save	Save Player and AI to file
LoadLeadboard<T>	Load the leaderboard from file, converts array to a List<> before being returned
SaveLeaderboard<T>	Save the leaderboard to file, converts List<> to array before saving

The Unity JsonUtility does not serialize a List<> to JSON if it is the top node, the FileSaveHandler contains a JsonHelper class as a wrapper to wrap the leaderboard List<> inside a "Items" node. The List<> is cast to an array for serialization and cast back to a List<> before being returned by the FileSaveHandler to the GameManager.

JsonHelper Method()	**Function**
FromJson<T>	Return a string from the de-serialised JSON
ToJson<T>	Converts to an array to serialise to JSON
ToJson<T>	Converts to an array to serialise to JSON (formatted)

2.5. ScriptableObjects

To reduce memory, aid testing and provide a designer friendly approach, data is stored inside ScriptableObject assets, this includes runtime data. Runtime data is loaded from the game save when the game starts, and updated data is saved at certain points back to the game save.

As summery of the main ScriptableObject game structure, the Player SO holds the players data including faction, the Faction SO holds the ship, the Ship SO holds the bullet and Bullet SO holds the bullet types as the player upgrades.

2.5.1. SO_Leaderboard

The leaderboard asset contains a list that stores the high score rows. This List<> object is created inside the GameManager and passed to the FileSaveHandler for a game save.

Asset	**Function**
Leaderboard	List<LeaderboardItem> containing all the high score rows

2.5.2. SO_Player

3 Assets are created from this class that hold all the characters data – example health, level, upgrades, and faction.

Asset	**Function**
Player	All Player character data
AI	All AI character data
Xoid	All NPC character data

2.5.3. SO_FactionList

Single asset contains a list of factions.

Asset	**Function**
Factions	Holds 4 SO_Faction assets

2.5.4. SO_Faction

4 Faction assets that contain data about the faction, faction ID and the game can also retrieve the faction ship from this asset.

Asset	**Function**
CharChina	All faction data, most importantly the faction ship
CharUSA	All faction data, most importantly the faction ship
CharUSSR	All faction data, most importantly the faction ship
CharXoid	All faction data, most importantly the faction ship

2.5.5. SO_Ship

4 Assets that contains data about thrust, turn speed, shield, prefab, and the game can retrieve the ships bullet data.

Asset	**Function**
Hammer	USSR faction ship data, AI uses the same faction
Rapier	Chinese faction ship data, AI uses the same faction
Sword	USA faction ship data, AI uses the same faction
Xoid	NPC faction ship data

2.5.6. SO_Bullet

Contains bullet data like speed, firepower, and list of 3 bullet level prefabs.

Asset	**Function**
Photon Shell Chn	Chinese faction bullet data and types
Plasma Ball Ussr	USSR faction bullet data and types
Rocket Us	USA faction bullet data and types
Xoid	NPC faction bullet data and types

2.5.7. SO_SFX

Contains a list of sound effects or music to play, has a method that returns a random clip.

Asset	**Function**
AsteroidExplosions	Asteroid explosion sounds
Effects	Various sound effects, played by index
Music	Game music for menu, game play, high score etc
ShipExplosions	Ship explosion sound effects

2.6. Managers

2.6.1. Singleton

Base class for all singleton managers, namely GameManager and SoundManager classes. These 2 manager scripts are attached to the Managers prefab and gets instantiated from the main menu scene and lives for the lifetime of the game.

2.6.2. GameManager

The main class from where most game functions are managed – save/load game, Player/AI/NPC objects, leaderboard management, calculations for bonuses, XP, levels, upgrades etc.

Exposed Property	**Function**
SaveFile	Name of the player game save file.
SaveFileAi	Name of the AI game save file.
SaveFileLeaderboard	Name of the leaderboard game save file.
Leaderboardsize	The number of rows the leaderboard can have.
Player	Player Scriptable Object for the game to access all player and faction data.
factions	List of all factions to assign a faction to a player when he selects a ship.
deathRespawnRate	Delay in seconds before a player or AI is re-spawned after death.
xpLevelSteps	XP to gain for the player to gain a new level.
maxLevel	The maximum number of achievable levels.
freeShip	Score needed to gain a free live, this only happens once per battle.
aiPlayer	The AI ScriptableObject for the game to access all AI data and faction data.
aiPermadeath	Does the AI have lives or continuously re-spawns which is the default.
npcPlayer	Xoid ScriptableObject for the game to access all NPC data.
iterium	Iterium ScriptableObject contains the Iterium prefab
iteriumChance	Chance for a splitting asteroid to drop a Iterium crystal. Default = 20:1
speedLevel1	Number of Iterium need to upgrade speed to level 2.
speedLevel2	Number of Iterium need to upgrade speed to level 3.
shieldLevel1	Number of Iterium need to upgrade shield to level 2.
shieldLevel2	Number of Iterium need to upgrade shield to level 3.
firepowerLevel1	Number of Iterium need to upgrade firepower to level 2.
firepowerLevel2	Number of Iterium need to upgrade firepower to level 3.
Leaderboard	Leaderboard ScriptableObject containing a list of leaderboard rows.

Private Property	**Function**
fileSaveHandler	Instance of the FileSaveHandlder class to serialize data objects to a JSON file. Data objects passed are saveData, saveDataAI and leaderboard.Leaderboard
currentAIFaction	Current AI faction index, used to check if a AI respawn if a different faction so the AI bullet pool can be cleared. See SelectAiPlayer() method

Hidden Public Property	**Function**
saveData	Instance of the FileSaveHandlder class to serialize data objects to a JSON file. Data objects passed are saveData, saveDataAI and leaderboard.Leaderboard
saveAIData	Current AI faction index, used to check if a AI respawn if a different faction so the AI bullet pool can be cleared. See SelectAiPlayer() method
playerSpawner	Position of the spawner for respawns, position passed to this property by the spawner prefab on the first spawn
aiSpawner	Position of the spawner for respawns, position passed to this property by the spawner prefab on the first spawn
upgradeChnScene	China upgrade scene name

upgradeUsScene	USA upgrade scene name
UpgradeUssrScene	USSR upgrade scene name
isPlaying	Bool if the player is playing a game, if true don't spawn player/AI from the co-routines when player exits the play before they spawn.
targetPlayer	Holds the spawned player, used by AI/NPC controllers to use as a target
targetAi	Holds the spawned AI, used by Player/NPC controllers to use as a target
targetNpc	Holds the spawned NPC, used by Player/AI controllers to use as a target

Method()	**Function**
OnEnable	Subscribe to events • PlayerDamage – invoked when player takes damage • AiDamage – invoked when AI takes damage • NpcDamage – invoked when NPC takes damage • AsteroidDamage – invoked when asteroid takes damage • DropIterium – invoked when asteroid drops a crystal • CollectedIterium – invoked when Player or AI collects a crystal
OnDisable	Un-subscribe from all events
CollectIterium	Increase Player or AI score, XP and Iterium collected
PlayerDamage	Increase AI score and XP
AiDamage	Increase Player score and XP
DropIterium	Instantiate an Iterium crystal
NPC Damage	Increase Player or AI score, XP
AsteroidDamage	Increase Player or AI score, XP
Start	• Instantiate new FileSaveHandler object • Load a game save • Initialise event listeners for Player/AI bullet/shield/speed upgrades
SelectAiPlayer	Randomly select a faction for the AI at each spawn
ResetGame	Reset all data to game defaults
ResetArena	Reset player/AI score/health/lives/Iterium /XP for a new arena battle
spawnPlayer	Call co-routine to spawn player prefab
spawnAi	• Call SelectAIPlayer() to select a random faction • Call co-routine to spawn AI prefab
SpawnPlayerOverTime	IEnumerator to spawn player prefab with a set delay
SpawnAiOverTime	IEnumerator to spawn AI prefab with a set delay
RandomScreenPosition	Gets a random spawn position near a spawn point
CameraShake	Call co-routine to shake camera when player dies
Shake	IEnumerator to shake camera with time and magnitude
SaveGame	• Update SaveData object with values from the Player ScriptableObject asset • Update aiSaveData object with values from the AI ScriptableObject asset • Pass both objects to the fileSaveHandler.Save() method to serialise and save the data to a JSON file • Call SaveLeaderboard() to save the leaderboard
LoadGame	• Populate the SaveData object with the data loaded from the Player save file • Update the Player ScriptableObject asset with the SaveData data • Populate the aiSaveData object with the data loaded from the AI save file • Update the AI ScriptableObject asset with the aiSaveData data • Call LoadLeaderboard() to load the leaderboard • Assign the default faction to the player if this is a new game • Call UpgradeLevelSync to sync the loaded data to the upgrade levels

	for speed/shield/bullet
SaveLeaderboard	Pass the leaderboard.Leaderboard (ScriptableObject) property of the GameManager to the fileSaveHandler for saving to a JSON file
LoadLeaderboard	• Load the leaderboard data from file into the leaderboard.Leaderboard property (ScriptableObject) of the GameManager • Call InitLeaderboard()
InitLeaderboard	If there are no current leaderboard rows (new game) then create new rows by the leaderboardSize property with the Xoid faction, multiplying each row score by 10 000
AddLeaderboardItem	Called from the GameOver scene to add the player or AI score to the leaderboard when a high score is achieved
SortLeaderboard	Sort the leaderboard by score by descending, for a high score the last row is always checked to see if the player scorer is higher
GameOver	Load the GameOver scene
SceneUpgrade	Load the appropriate upgrade screen depending on the player currently selected faction.
MainMenu	Load the MainMenu scene
OnApplicationQuit	• Call SaveGame() • If WebGL build load a specific webpage • Else Application.Quit() is called
CalculateXP	Calculate if a player or AI increases in level after gaining XP in the arena. Level steps are calculated on the xpLevelSteps property
CalculateIterium	Add Iterium collected in the arena to the player/AI total Iterium value
CalculatePlayerBonus	Calculate how much points bonus a player earns after an arena. player.IteriumCollected * 100 * player.Level
CalculateAiBonus	Calculate how much points bonus a AI earns after an arena. aiPlayer.IteriumCollected * 100 * aiPlayer.Level
BulletLvlChanged	Keeps the Player object bullet upgrade values in sync with the current upgrade level
ShieldLvlChanged	Keeps the Player object shield upgrade values in sync with the current upgrade level
SpeedLvlChanged	Keeps the AI object speed upgrade values in sync with the current upgrade level
BulletLvlChangedAi	Keeps the Player object bullet upgrade values in sync with the current upgrade level
ShieldLvlChangedAi	Keeps the AI object shield upgrade values in sync with the current upgrade level
SpeedLvlChangedAi	Keeps the AI object speed upgrade values in sync with the current upgrade level
UpgradeLevelSync	Sync the loaded game save upgrade data to the game upgrade properties for the player and AI

2.6.3. SoundManager

Play sound effects and music, runtime creation of AudioSources, volume management.

Exposed Property	**Function**
asteroidExplosion	ScriptableObject list of sound clips for asteroid explosions
shipExplosion	ScriptableObject list of sound clips for ship explosions
effects	ScriptableObject list of sound clips for sound effects
music	ScriptableObject list of sound clips for music
audioSourceNumber	Number of AudioSources to create dynamically for use by sound effects. Music has its own single dedicated AudioSource and is not included in this.
mixerMaster	Audio mixer group for the master
mixerMusic	Audio mixer group for music
mixerSfx	Audio mixer group for sound effects

Private Property	**Function**
audiosourceSfx[]	AudioSource array used in round robin for sound effects
audiosourceMusic	Single AudioSource used for music

Method()	**Function**
Start	Set the effects and music mixer volumes that are loaded from game save into the Player object when the game starts
Awake	• Init AudioSource array to be used as round robin for sound effects • Assign the sound effects mixer group to the above AudioSources • Init an AudioSource for music • Assign the music mixer group to the above AudioSource
PlayAsteroidExplosion	Play a random sound from the AsteroidExplosions ScriptableObject asset
PlayShipExplosion	Play a random sound from the ShipExplosions ScriptableObject List<AudioClip>
PlayEffect	Play a sound by index from the Effects ScriptableObject List<AudioClip>
PlayMusic	Play a music by index from the Music ScriptableObject List<AudioClip>, method has properties to loop, stop and delay the clip
GetAudioSourceSfx	Returns an available AudioSource from the audiosourceSfx[] array

2.6.4. Bullet Pooling

Bullets fired from the player, AI and NPC are object pooled to increase performance. The GameManager provides the bullet prefabs to spawn from the player, aiPlayer and npcPlayer objects, depending on the faction that is selected and the level of the bullet.

Exposed Property	**Function**
capacity	Default pool capacity of bullets for the player
maxCapacity	Maximum pool capacity of bullets for the player
capacityAi	Default pool capacity of bullets for the AI
maxCapacityAi	Maximum pool capacity of bullets for the AI
capacityNpc	Default pool capacity of bullets for the NPC
maxCapacityNpc	Maximum pool capacity of bullets for the NPC
bulletPoolPlayer	Player bullet pool (static)
bulletPoolAi	AI bullet pool (static)
bulletPoolNpc	NPC bullet pool (static)

Method()	**Function**
Awake	• Init a new bulletPoolPlayer pool object • Init a new bulletPoolAi pool object • Init a new bulletPoolNpc pool object
PoolNew_Player	Return a new instantiated Player bullet taking the Player bullet level
PoolNew_Ai	Return a new instantiated AI bullet taking the AI bullet level
PoolNew_NPC	Return a new instantiated NPC bullet
PoolGet	Get a bullet from the pool
PoolReturn	Return a bullet back to the pool
PoolDestroy	Destroy a bullet in the pool

2.6.5. Asteroid Pooling

Asteroids are spawned from a pool to increase performance. Properties exposed in the inspector are as follows…

Exposed Property	**Function**
capacity	Default pool capacity of spawned asteroids
maxCapacity	Maximum pool capacity of spawned asteroids
asteroids	Asteroids prefab to be spawned from the Asteroids ScriptableObject asset
asteroidPool	Asteroid pool (static)

Method()	**Function**
Awake	Init a new asteroidPool pool object
PoolNew	Return a new instantiated asteroid prefab
PoolGet	Get an asteroid from the pool
PoolReturn	Return an asteroid back to the pool
PoolDestroy	Destroy an asteroid in the pool

2.6.6. Explosion Pooling

Explosions are spawned from a pool to increase performance. Properties exposed in the inspector are as follows…

Exposed Property	**Function**
capacity	Default pool capacity of explosions to spawn
maxCapacity	Maximum pool capacity of explosions to spawn
explosion	Explosion prefab to spawn
explosionPool	Explosions pool (static)

Method()	**Function**
Awake	Init a new explosionPool pool object
PoolNew	Return a new instantiated explosion prefab
PoolGet	Get an explosion from the pool
PoolReturn	Return an explosion back to the pool
PoolDestroy	Destroy an explosion in the pool

2.6.7. Input Manager

InputManager class is the main class to use the InputSystem for all actions except the Pause action, this is actioned from the UI_Game class.

Exposed Property	**Function**
rotateInput	Vector2 to rotate the ship left or right
thrustInput	Vector2 to thrust the ship forward
isFire	Bool to know if the ship is firing
isShield	Bool to know if the ship is shielding
isWarping	Bool to know if the ship is warping

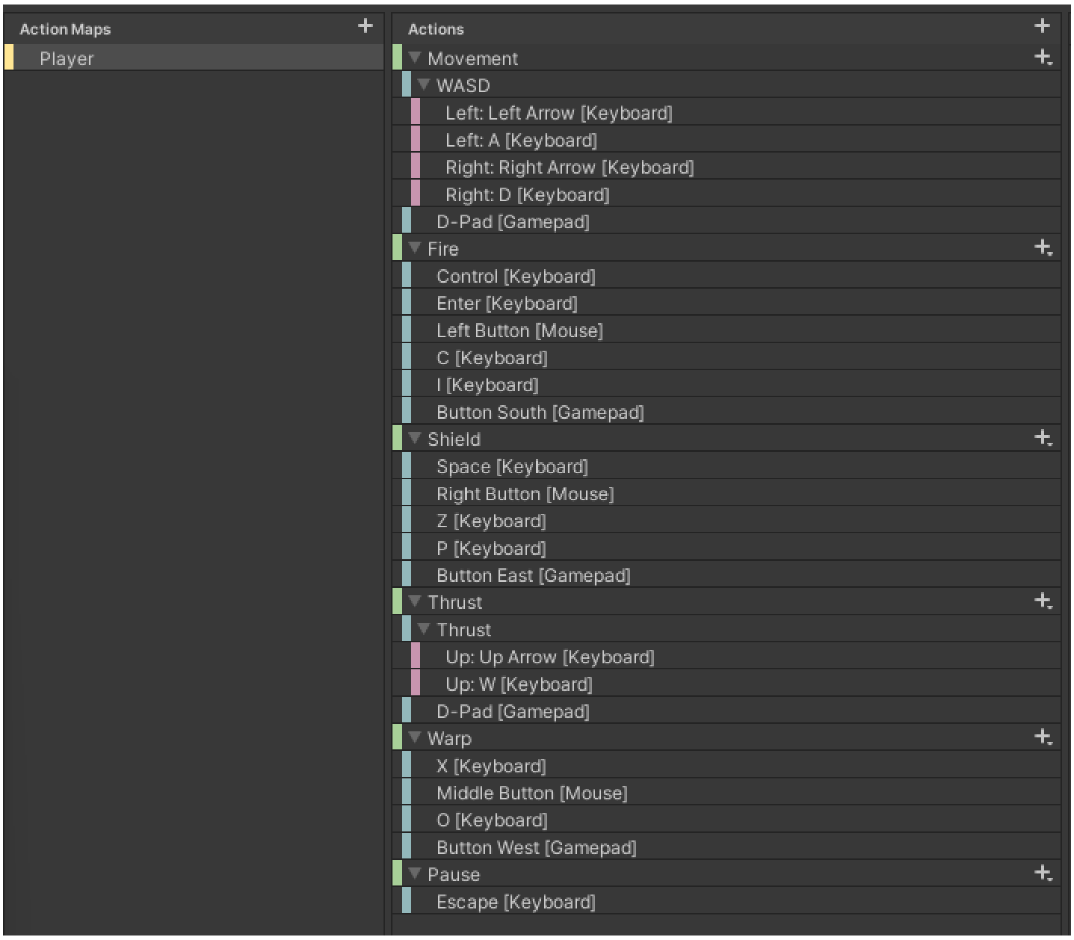
Private Property	**Function**
input	Object of the InputSystem

Method()	**Function**
OnEnable	• Create new InputSystem object • Enable the Player action map • Create input action events
OnDisable	Disable all input action events
Rotate	Update rotateInput property, Player controller reads this from its Update() method
Thrust	Update thrustInput property, Player controller reads this from its FixedUpdate() method
Fire	Update isFire property, Player controller reads this from its Update() method
Shield	Update isShield property, Player controller reads this from its Update() method
Warp	Update isWarping property, Player controller reads this from its Update() method

2.6.7.1. Input Actions – Input System

Single action map called “Player” with the following actions.

- Movement and firing
- Shield
- Thrust
- Warp
- Pause



Action Maps

- Player

Actions

- Movement
  - WASD
    - Left: Left Arrow [Keyboard]
    - Left: A [Keyboard]
    - Right: Right Arrow [Keyboard]
    - Right: D [Keyboard]
  - D-Pad [Gamepad]
- Fire
  - Control [Keyboard]
  - Enter [Keyboard]
  - Left Button [Mouse]
  - C [Keyboard]
  - I [Keyboard]
  - Button South [Gamepad]
- Shield
  - Space [Keyboard]
  - Right Button [Mouse]
  - Z [Keyboard]
  - P [Keyboard]
  - Button East [Gamepad]
- Thrust
  - Thrust
    - Up: Up Arrow [Keyboard]
    - Up: W [Keyboard]
  - D-Pad [Gamepad]
- Warp
  - X [Keyboard]
  - Middle Button [Mouse]
  - O [Keyboard]
  - Button West [Gamepad]
- Pause
  - Escape [Keyboard]

2.8. Gameplay Scripts

Scripts that control character (player/AI/NPC) movement, firing, bullets, and Iterium collection etc.

2.8.1. PlayerContoller

Control the player ship movement, thrusting, firing, shielding, warping and damage.

Private Property	**Function**
input	InputManager component for input
firePosition	Empty GameObject on the ship from where bullets are spawned
shield	Plane with a Shader Graph material, gameObject is disabled at start, enabled when shielding
shieldCooldown	Used as a timer to limit when the player can deploy the shield
IsShielding	Bool to indicate if the ship is shielding to limit continuous shielding
thrusters	Plane with a Shader Graph material, gameObject is disabled at start, enabled when thrusting
isThrusting	Bool to indicate if the ship is thrusting to limit continuous thrusting
rigidBody	Used to add velocity to the ship

Medthod()	**Function**
Start	Get components for… • input – InputManager for user input • shield – transform child under the ship at index 0, plane with a Shader Graph material to give a shield effect • thrusters – transform child under the ship at index 1, plane with Shader Graph material to display a thruster effect • firePosition – transform child under the ship at index 2, position from where the bullets will fire from • rigidBody – velocity is added from thrusting
Update	Calls methods to operate the player ship • Rotate() • Fire() • Shield() • Warp()
FixedUpdate	Calls methods to move the ship • Thrust()
Rotate	Rotation of the ship left or right, plus added turn speed with any upgrade bonus
Fire	Fire a bullet from the bullet pool, the velocity of the bullet is the factions ship speed plus any upgrade bonus
Thrust	• Thrusts the ship forward • Enables/disables the thrust effects – plane as child on the ship • Adds force to the rigidBody using the faction ships thrust plus any upgrade bonus
Shield	• Enables/disables the shield effects – plane as child on the ship • Play a sound from an AudioSource attached to the shield plane • Starts a co-routine ShieldTime() as a count-down timer • Start the shield cooldown so the player cannot deploy the shield again until the cooldown is finished
Warp	Warp the player to a random position near the player spawner
ShieldTime	IEnumerator that waits until the shield time is up and disables the shield
Damage	• Calculates damage using firepower and shield status

	• Destroys the ship if health is depleted • Invoke PlayerDamage event, pass the attacker in a string • GameManager subscribes and increases score and XP to attacker • Calls co-routine DestroyShip() if the health reaches zero
DestroyShip	IEnumerator to destroy the ship • Instantiate 3 explosion effects from the explosions pool • Play the ship explosion sound effect from the SoundManager • Call the SpawnPlayer() method of the GameManager passing in the required delay time • Destroy the ship gameObject
OnBecameInvisible	When the ship leaves the screen it returns (wraps) the ship to the opposite side of the screen

Event	**Function**
PlayerDamage	• Event for when the Player takes damage

Interface	**Function**
IDamage	Contracts the Damage() method passing in bullet firepower and attacker

2.8.2. AIContoller

Control the AI ship movement, thrusting, firing, shielding and damage.

Exposed Property	**Function**
fireDelay	Delay before the AI starts firing when spawning
fireInterval	Interval that the AI fires a bullet
warpInterval	How often does the AI decide warp or not
descisionCycle	How many shots are fired at a target before making another decision on who to target (Player or NPC)

Private Property	**Function**
firePosition	Empty GameObject on the ship from where bullets are spawned
shield	Plane with a Shader Graph material, gameObject is disabled at start, enabled when shielding
IsShielding	Bool to indicate if the ship is shielding to limit continuous shielding
thrusters	Plane with a Shader Graph material, gameObject is disabled at start, enabled when thrusting
isThrusting	Bool to indicate if the ship is thrusting to limit continuous thrusting
rigidBody	Used to add velocity to the ship
shots	How many shots have been fired to make a target decision if equal to the descisionCycle
attackNPC	Must the AI attack the NPC, 50/50 decision or the player could be dead

Medthod()	**Function**
Start	Get components for… - shield – transform child under the ship at index 0, plane with a Shader Graph material to give a shield effect - thrusters – transform child under the ship at index 1, plane with Shader Graph material to display a thruster effect - firePosition – transform child under the ship at index 2, position from where the bullets will fire from - rigidBody – velocity is added from thrusting - InvokeRepeating for firing - InvokeRepeating for shielding, using the shield cooldown value as the interval
Update	Calls methods to operate the player ship - Rotate()
FixedUpdate	Calls methods to move the ship - Thrust()
Rotate	Rotation of ship to either target the Player or NPC
Fire	- Shoots a set number of bullets in a decision round, then decides again on its target, taking into account who is on the screen regards Player and NPC - Fire a bullet from the bullet pool, the velocity of the bullet is the factions ship speed plus any upgrade bonus
Thrust	- Thrusts the ship forward - Enables/disables the thrust effects – plane as child on the ship - Adds force to the rigidBody using the faction ships thrust plus any upgrade bonus
Shield	- Enables/disables the shield effects – plane as child on the ship
ShieldTime	IEnumerator that waits until the shield time is up and disables the shield
Warp	50/50 descition to warp or not
Damage	- Calculates damage using firepower and shield status - Destroys the ship if health is depleted - Invoke AiDamage event, pass the attacker in a string - GameManager subscribes and increases score and XP to attacker
DestroyShip	IEnumerator to destroy the ship - Instantiate 3 explosion effects from the explosions pool - Play the ship explosion sound effect from the SoundManager - Call the SpawnAi() method of the GameManager passing in the required delay time - Destroy the ship gameObject
OnBecameInvisible	When the ship leaves the screen it returns (wraps) the ship to the opposite side of the screen

Event	**Function**
AiDamage	- Event for when the AI takes damage

Interface	**Function**
IDamage	Contracts the Damage() method passing in bullet firepower and attacker

2.8.3. NPCContoller

Control the NPC ship (Xoid UFO) movement and firing.

Exposed Property	**Function**
fireDelay	Delay before the NPC starts firing when spawning
fireInterval	Interval that the NPC fires a bullet

Private Property	**Function**
target	Random number to select the target of player or AI

Medthod()	**Function**
Start	InvokeRepeating for firing
Fire	Decides randomly on its target, considering who is on the screen regards Player and NPC
Damage	• Passes the attacker in a string • Invoked when the AI takes damage • GameManager subscribes and increases score and XP to attacker • Destroy ship
OnBecameInvisible	When the ship leaves the screen it gets destroyed
DestroyShip	Destroy the NPC gameObject

Event	**Function**
NpcDamage	• Event for when the NPC takes damage

Interface	**Function**
IDamage	Contracts the Damage() method passing in bullet firepower and attacker

2.8.4. BulletBase

Interface	**Function**
IDamage	Contracts the Damage() method passing in bullet firepower and attacker

Method()	**Function**
OnBecameInvisible	• Call ReleaseBullet()
Damage	Release bullet when it collides with an object
BulletExplosion	• Call bullet explosion sound from SoundManager • Instantiate an explosion visual effect from pool • Call ReleaseBullet()
ReleaseBullet	• Releases bullet back to pool, empty in the base class and is overwritten individually on each player/AI/NPC bullet

2.8.5. Bullet

Players bullet collision.

Exposed Property	**Function**
firePower	Firepower of the bullet, value updated when the PlayerController instantiates

Method()	**Function**
OnTriggerEnter	• If IDamage interface contracted by PlayerController, then call the Damage() method • Do a bullet explosion
ReleaseBullet	• Release bullet to pool

2.8.6. BulletAI

AI bullet collision.

Exposed Property	**Function**
firePower	Firepower of the bullet, value updated when the AiController instantiates

Method()	**Function**
OnTriggerEnter	• If IDamage interface contracted by AiController, then call the Damage() method • Do a bullet explosion
ReleaseBullet	• Release bullet to pool

2.8.7. BulletNpc

NPC bullet.

Method()	**Function**
OnTriggerEnter	• If IDamage interface contracted by AiController, then call the Damage() method • Do a bullet explosion
ReleaseBullet	• Release bullet to pool

2.8.8. Iterium

Attached to a drop Iterium prefab, collection of the crystal by player or AI.

Method()	**Function**
OnTriggerEnter	• invoke CollectIterium event passing who collected the crystal in a string

Event	**Function**
CollectIterium	• Passes attacker in a string • Invoked when the crystal is collected by Player or AI • GameManager subscribes to increase Iterium collected, score and XP

2.7.9. Asteroid

Exposed Property	**Function**
IteriumChance	Chance of an Asteroid dropping a Iterium crystal when split

Method()	**Function**
Damage	• Splits and scales the Asteroid when hit by a bullet • Can be split 3 times into a random 2-4 pieces. • Asteroid weight is also adjusted according to the split size. • Chance to spawn an Iterium crystal • Release asteroid to pool

Event	**Function**
AsteroidDamage	• Passes the attacker in a string • Invoked when asteroid takes damage • GameManager subscribes to increase score and XP to the attacker
DropIterium	• Passes a Vector3 position of the asteroid • Invoked when an Iterium crystal is dropped • GamaManager subscribes to spawn the crystal at the position

2.8. User Interface

The only UI used is the project is the Unity Toolkit and the look and feel of the game can be changed through the Unity style sheets (uss).

2.8.1. Unity Style Sheets

MainStyle USS Classes	**Function**
.button	Buttons throughout UI
.text	Used across multiple UI documents to affect text's
.footer	Footer text used in Main Screen
.menu-button	Home screen menu sprite
.image-button	Images on Faction Selection Screen
.header	Headers on all UI documents
.text-score	Leaderboard Item text
.score-panels	Leaderboard score panel adjustment
.text-scoreHeading	Leaderboard title headings
.progress-slider	Sliders in all documents
.progress-slider-text	Upgrade document slider text
.button:hover	Transitions for buttons on hover
.button:active	Transitions for buttons when active
.button:focus	Transitions for buttons when it has focus
.unity-base-text-field__input	Base input text fields
.unity-base-field__input	Base input text fields
.unity-text-field__input	Base input text fields
.gridcontroller	Tutorial document Control Panel alignments and spacing
.ship-icon	Tutorial document Scoring Panel ship sprites
.controller-text	Tutorial document text
.unity-base-slider__dragger	Sprite for slider handle
.unity-base-slider__tracker	Sprite for slider background
.unity-scroller__slider	Leaderboard scroller sprite
.unity-scroller__low-button	Image on vertical scrollbar for the Leaderboard scroll view
.unity-scroller__high-button	Image on vertical scrollbar for the Leaderboard scroll view
#scrollViewHS .unity-scroller--vertical .unity-base-slider__dragger	The scrollbar dragger image and background colour for the Leaderboard scroll view

2.8.2. UI Toolkit Documents

UI Toolkit Documents	**Function**
FactionSelection	Used on the FactionSelection scene, contains 4 Render Textures in the UI for the Shader Graph progress bars of the 3 ships and the legend
GameOver	Used on the GameOver scene, contains 1 Render Texture to display the player ship
Leaderboard	Used on the Leaderboard scene, contains a scroll view named scrollViewHS
LeaderboardItem	Used as a row template inside the Leaderboard scrollViewHS
MainMenu	Used for the menu on the MainMenu scene
Play	Used to display the player/AI health, lives and Iterium collected, pause menu
Settings	Used on the Settings scene, contains 2 Render Textures to display the volume/effects radial shader graph progress bars.
Tutorial	Used on the Tutorial scene, various tutorial panels that are shown/hidden via script depending which tutorial button is clicked
UpgradeChina	Used on the UpgradeChina scene, has 3 sliders (thrust/shields/firepower) with a corresponding upgrade button, the sliders are disabled and are moved by a script when an upgrade is actioned.
UpgradeUSA	Used on the UpgradeUSA scene, has 3 sliders (thrust/shields/firepower) with a corresponding upgrade button, the sliders are disabled and are moved by a script when an upgrade is actioned.
UpgradeUSSR	Used on the UpgradeUSSR scene, has 3 sliders (thrust/shields/firepower) with a corresponding upgrade button, the sliders are disabled and are moved by a script when an upgrade is actioned.
UserProfile	Used on the UserProfile screen, various input boxes, the information is saved when the save button is clicked

2.8.3. UI Scripts

Refer to the Classes & Assemblies section for additional reference.

2.8.3.1. UI_Button

General button UI script to load a scene.

Exposed Property	**Description**
sceneName	Holds the scene to load
buttonName	Button element – load the scene when clicked

2.8.3.2. UI_FactionSelecton

FactionSelection scene UI script to allow the player to see the firepower/shield/speed stats of each ship

Exposed Property	**Description**
sceneName	Holds the Play scene to load
chinaFaction	Button element – Selects the China faction and updates the Player.Faction property of the GameManager
usaFaction	Button element – Selects the USA faction and updates the Player.Faction property of the GameManager
ussrFaction	Button element – Selects the USSR faction and updates the Player.Faction property of the GameManager
playerStatsText	Label element – Profile name of the player
playerXPTotal	Label element – Players total XP value
playerLevel	Label element – Players current level
playerIteriumTotal	Label element – Players total Iterium value
shipChn	Chinese ship ScriptableObject asset to get the ships base firepower/shield/speed
shipUs	USA ship ScriptableObject asset to get the ships base firepower/shield/speed
shipUssr	USSR ship ScriptableObject asset to get the ships base firepower/shield/speed
progressChnFirepower	Plane with Shader Graph material, China firepower radial progress bar, set float on the progress bar material (“`_RemovedSeg`”)
progressChnSpeed	Plane with Shader Graph material, China speed radial progress bar, set float on the progress bar material (“`_RemovedSeg`”)
progressChnShield	Plane with Shader Graph material, China shield radial progress bar, set float on the progress bar material (“`_RemovedSeg`”)
progressUsFirepower	Plane with Shader Graph material, US firepower radial progress bar, set float on the progress bar material (“`_RemovedSeg`”)
progressUsSpeed	Plane with Shader Graph material, US speed radial progress bar, set float on the progress bar material (“`_RemovedSeg`”)
progressUsShield	Plane with Shader Graph material, US shield radial progress bar, set float on the progress bar material (“`_RemovedSeg`”)
progressUssrFirepower	Plane with Shader Graph material, USSR firepower radial progress bar, set float on the progress bar material (“`_RemovedSeg`”)
progressUssrSpeed	Plane with Shader Graph material, USSR speed radial progress bar, set float on the progress bar material (“`_RemovedSeg`”)
progressUssrShield	Plane with Shader Graph material, USSR shield radial progress bar, set float on the progress bar material (“`_RemovedSeg`”)

and to select a shift for the arena.

2.8.3.3. UI_Game

Exposed Property	**Description**
sceneName	Holds the MainMenu scene to load
player1	TextElement to display player profile name
scorePlayer1	TextElement to display player score
iteriumPlayer1	TextElement to display the player stage collected Iterium
player2	TextElement to display AI profile name
scorePlayer2	TextElement to display AI score
iteriumPlayer2	TextElement to display the AI stage collected Iterium
ship1Player1	VisualElement to display the player 1st ship life image
ship2Player1	VisualElement to display the player 2nd ship life image
ship3Player1	VisualElement to display the player 3rd ship life image
ship4Player1	VisualElement to display the player 4th (extra life) ship life image
ship1Player2	VisualElement to display the AI 1st ship life image
ship2Player2	VisualElement to display the AI 2nd ship life image
ship2Player2	VisualElement to display the AI 3rd ship life image
progressPlayerHealth	Plane with Shader Graph material, player radial health progress bar, set float on the progress bar material (“`_RemovedSeg`”)
progressAiHealth	Plane with Shader Graph material, AI radial health progress bar, set float on the progress bar material (“`_RemovedSeg`”)
pause	VisualElement – pause panel with 2 buttons
exitGame	Button element – pause button to exit the stage and return to MainMenu scene
continueGame	Button element – pause button to close pause panel and continue stage

Game scene UI script to display the health/Iterium collected, and lives of the player/AI.

2.8.3.4. UI_GameOver

Game Over scene UI script, displays the users arena score, collected XP, Iteruim, high score music, custom message based on how good the score is.

Exposed Property	**Description**
gameScene	Holds the Play scene to load
playerScore	Label to display player stage score
playerIterium	Label to display player stage collected Iterium
playerIteriumTotal	Label to display player total Iterium
message	Label to display custom score related congratulations message
rematch	Button element to re-start the Play scene
playerXPTotal	Label to display player total XP
playerXPEraned	Label to display player stage XP
playerLevel	Label to display player level, level progression is calculated in this scene
playerBonus	Label to display player stage score bonus, (Iterium collected x 100) x level

2.8.3.5. UI_Leaderboard

Exposed Property	**Description**
scoreRow	VisualTreeAsset – leaderboard row document template
rank	Label to display leaderboard item ranking
score	Label to display leaderboard item score
date	Label to display leaderboard item date
playerName	Label to display player name

Leaderboard scene UI script to display the top player/AI high scores by score ranking

2.8.3.6. UI_LoadUpgradeScene

Exposed Property	**Description**
upgradeButton	Button element to ask the GameManger to load the upgrade scene of the player selected faction

Loads the appropriate faction upgrade scene depending on the players currently selected faction.

2.8.3.7. UI_Profile

Exposed Property	**Description**
sceneName	Holds the MainMenu scene to load
playerName	TextField input for the player name
bio	TextField input for the player bio description
email	TextField input for the player email address
save	Button element to save input field data and load the MainMenu scene

PlayerProfile scene UI script, display and save input for player name, bio and email address.

2.8.3.8. UI_QuitGame

Attached to the quit button to call the Game Manager OnApplicationQuit() method, which saves the games and redirects to a chosen webpage.

Exposed Property	**Description**
buttonName	Button element to exit the game

2.8.3.9. UI_Settings

Settings scene UI script to display the current music and sound effects volumes, adjusts the volumes when the appropriate sliders are changed.

Exposed Property	**Description**
musicSlider	Slider element to change the music volume
soundslider	Slider element to change the sound effects volume
musicIcon	VisualElement to rotate the music icon image
soundIcon	VisualElement to rotate the sound effects icon image
progressMusic	Plane with Shader Graph material, music radial progress bar, set the float on the progress bar material (“`_RemovedSeg`”)
progressSound	Plane with Shader Graph material, sound effects radial progress bar, set the float on the progress bar material (“`_RemovedSeg`”)
audioMixer	Sound Audio Mixer asset to set the volume of the music float (“`Music`”) and sound effects float (“`Sound`”)

2.8.3.10. UI_Tutorial

Tutorial scene UI scripts to hide/show tutorial VisualElements depending on what tutorial buttons are selected.

Exposed Property	**Description**
buttonStory	Button element to show the story tutorial panel
controls	Button element to show the controls tutorial panel
gameplay	Button element to show the gameplay tutorial panel
scoring	Button element to show the scoring tutorial panel
upgrades	Button element to show the upgrades tutorial panel
faction	Button element to show the factions tutorial panel
storyPanel	VisualElement of the story panel
controlsPanel	VisualElement of the controls panel
gameplayPanel	VisualElement of the gameplay panel
scoringPanel	VisualElement of the scoring panel
upgradesPanel	VisualElement of the upgrades panel
factionPanel	VisualElement factions panel

2.8.3.11. UI_Upgrade

Faction upgrade scene, upgrades the ships firepower, speed and shield if the player has the required Iterium.

Exposed Propertt	**Description**
thrustSlider	Slider element to indicate the current thrust upgrade
shieldSlider	Slider element to indicate the current shield upgrade
firePowerSlider	Slider element to indicate the current firepower upgrade
thrustUpgrade	Button element to upgrade the thrust if the player has the indicated Iterium
shieldUpgrade	Button element to upgrade the shield if the player has the indicated Iterium
firepowerUpgrade	Button element to upgrade the firepower if the player has the indicated Iterium
iteriumAmount	Label element to indicate the player total Iterium
thrustLevel1	Label element to show the required Iterium needed for the thrust level 1, set in the GameManager class
thrustLevel2	Label element to show the required Iterium needed for the thrust level 2, set in the GameManager class
shieldLevel1	Label element to show the required Iterium needed for the shield level 1, set in the GameManager class
shieldLevel2	Label element to show the required Iterium needed for the shield level 2, set in the GameManager class
firepowerLevel1	Label element to show the required Iterium needed for the firepower level 1, set in the GameManager class
firepowerLevel2	Label element to show the required Iterium needed for the firepower level 2, set in the GameManager class

2.9. Spawning

The spawning folder holds all the spawning scripts, these are attached to an empty prefab and placed where you want the player/AI/NPC to spawn. The asteroids have 4 empty gameObject spawners in the scene and a target object where they point to when spawned.

Script Name	**Description**
PlayerSpawner	Player spawner that has a spawnTime property to set the delay in seconds of the spawn. The game GameManager has a SpawnPlayer(spawnTime) method that is called and does the spawning. The position of the spawner is also passed to the GameManager playerSpawner property so it knows where to do a re-spawn later
AISpawner	AI spawner that has a spawnTime property to set the delay in seconds of the spawn. The game GameManager has a SpawnAI(spawnTime) method that is called and does the spawning. The position of the spawner is also passed to the GameManager aiSpawner property so it knows where to do a re-spawn later
NPCSpawner	Spawns the Xoid NPC UFO periodically, setting the target and random speed • spawnInterval property is the time before each spawn • minSpeed is the min range of the random speed selection • maxSpeed is the max range of the random speed selection The player is the first target, if the player is dead, it will target the AI, if they are both dead it will fly towards the center of the screen
AsteroidSpawner	This script is attached to 4 empty gameObjects in the scene, but more can be added, and move towards a target gameObject (AsteroidTarget). A random speed is selected and the AsteroidPooling class selects a random prefab to spawn. • spawnInterval property is the time before each spawn • minSpeed is the min range of the random speed selection • maxSpeed is the max range of the random speed selection • spawnOnceOnly allows only 1 asteroid to spawn • target is the AsteroidTarget game object that the asteroid moves towards
DeSpawnAsteroid	Releases the asteroid back to the pool when the asteroid leaves the camera view
DeSpawnExplosion	Releases the explosion effect back to the pool after a set time delay
GameOverSpawner	Spawn the players currently selected ship on the Game Over scene.

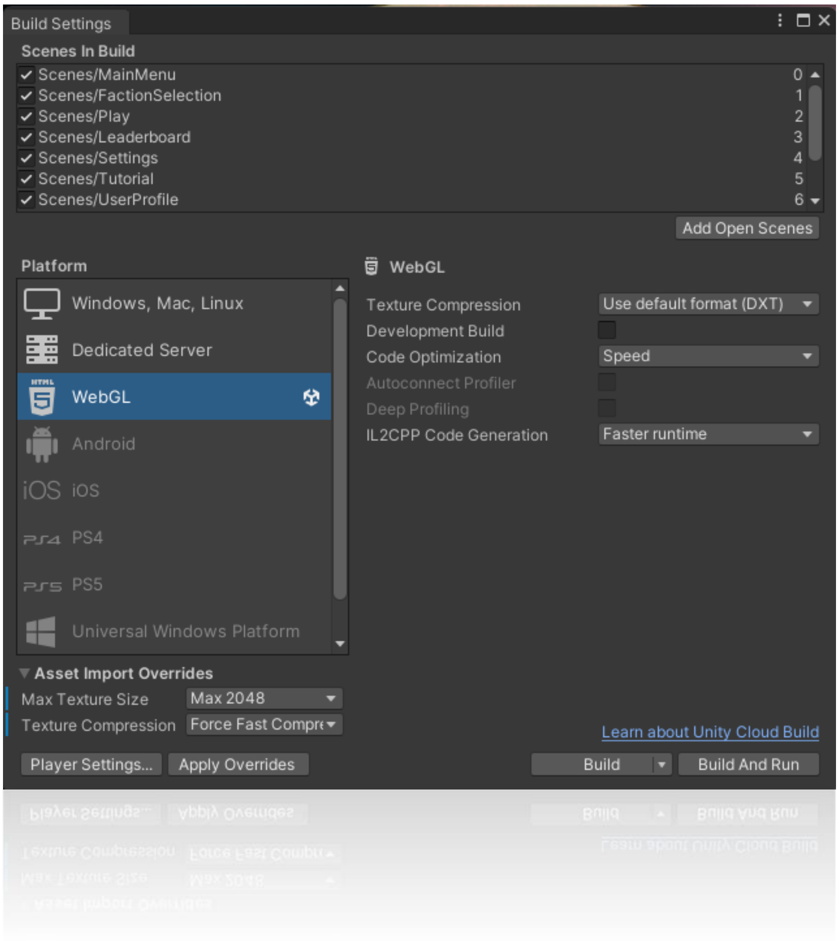
2.10. Plugins

WebGL.jslib plugin file in the Plugins folder for various JavaScript methods.

Functions	**Description**
syncSave	Force update of IndexedDB immediately when actioning a save game, otherwise the browser session will not save the data immediately
alert	Alert popup to display a message to the user or debugging, used in the FileSaveHandler for error alerts

2.11. Builds

Build settings platform must be set to WebGL



Build Settings

Scenes In Build

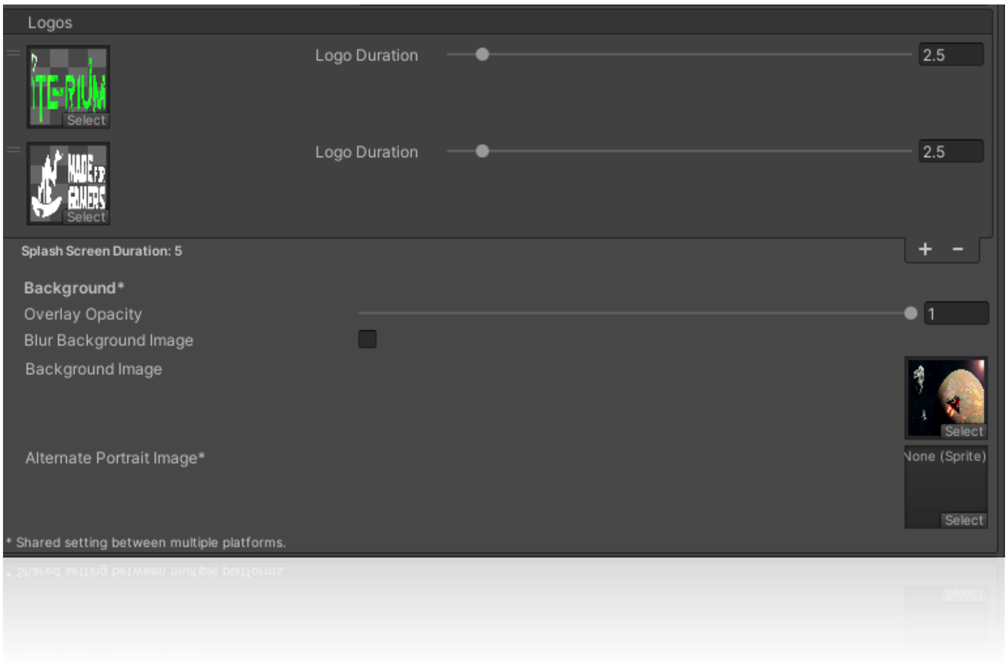
- ✓ Scenes/MainMenu 0
- ✓ Scenes/FactionSelection 1
- ✓ Scenes/Play 2
- ✓ Scenes/Leaderboard 3
- ✓ Scenes/Settings 4
- ✓ Scenes/Tutorial 5
- ✓ Scenes/UserProfile 6

WebGL

Texture Compression: Use default format (DXT)
Development Build: □
Code Optimization: Speed
Autoconnect Profiler: □
Deep Profiling: □
IL2CPP Code Generation: Faster runtime

Asset Import Overrides
Max Texture Size: Max 2048
Texture Compression: Force Fast Compre

Change the background, game and studio logos for the splash page under Player Settings.



Logos

Logo Duration: 2.5
Logo Duration: 2.5

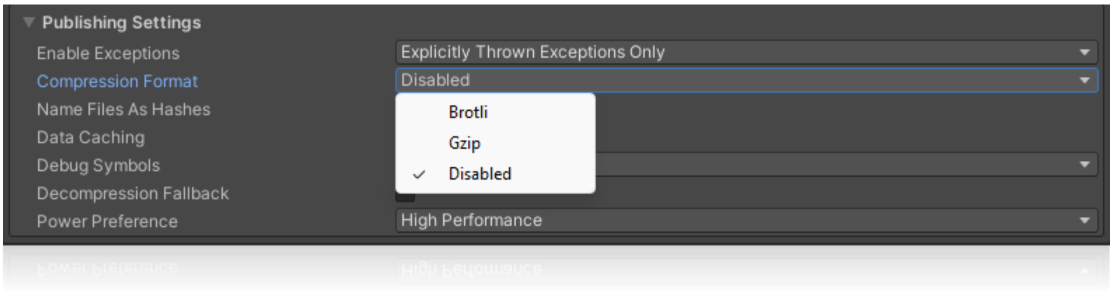
Splash Screen Duration: 5

Background*
Overlay Opacity: 1
Blur Background Image: □
Background Image
Alternate Portrait Image*: None (Sprite)

* Shared setting between multiple platforms.

Also, under Player Settings / Publishing Settings, change the compression accommodated by your web server, depending on the server OS and web server you may need to include an appropriate config file for the compression and handling of the wasm files and mime types such as .mem and .data.

See Unity documentation here on web server config files - Link



Publishing Settings

Enable Exceptions: Explicitly Thrown Exceptions Only

Compression Format: Disabled

Name Files As Hashes

Data Caching

Debug Symbols

Decompression Fallback

Power Preference: High Performance

You can change the HTML loading template under the folder WebGL Templates / MFG

Finally build the project and copy the build folder contents to your web server, open the URL to the website in your browser, sometimes you would have to add the Index.html (case sensitive) at the end of the URL, for example the GitHub Pages hosting environment.