**JASP**

*Just Another Space Pilot*

*Sound Design Document*

**

*Zachery Dyer, Blake Andrews & Bayley Fisher*

*Changelog*

*30th June v1.0 initial document creation (GDD)*

*19th July v.1.1 SFX list*

*27th July v2 feedback + documentation alterations*

*Table of Contents*

*GDD*

*Overview*

*Gameplay*

*Core Mechanics*

*Level Overview*

*SDD*

*SFX List*

*Description of SFX*

*Sound Files*

*Risk Assessment*

*Feedback*

**GDD**

**Overview**

| Genre | Arcade / Shooter / Simulator |
| --- | --- |
| Platform | PC, Mac, PS3 |
| Point of View | Third Person |
| Theme / Mood | Sci-Fi, Epic, Futuristic, Heroic |
| Target Audience | Fans of intense and fast paced arcade shooters that progressively get harder relatively quickly |

**Gameplay**

‘*JASP* ‘ has the player put into the majestic void of space with only their StarShip and blasters in a wave-like space shooter, trying to fight off enemy attack ships that are trying to hinder and destroy the player.

**Core Mechanics**

At its heart, JASP is heavily inspired by the likes of Space Invaders and Galaga, but put into a 3D environment with complimentary colours of blue and orange to highlight the distinction between the world, the player and its enemies.

The game consists of an infinitely growing wave system that makes the game progressively get harder as the player tries to survive as long as possible against the enemy ships and asteroids.

**Player Ship**

The main ship is given full control to the player via omnidirectional, real world inspired ship controls. While not entirely accurate, it will give the player a feeling like they’re in the black abyss of space.

The player ship features a health bar, blaster heat, speed drive and a space rader in the UI in which the player must manage.

**Health Bar** is the player's overall health, which will be used to visually display the damage the enemy has done to the player.

**Blaster Heat** is a cooldown system that stops the player from spamming the blaster - doing so will lockout the player from shooting for an extended period of time.

**Speed Boost** is the extra speed the player can access to propel themselves faster and further in quick succession, but will need time to cool down to reuse, to deter the player from constantly using it.

**Radar** displays all the current enemies in the general vicinity of the player - however, asteroids do not appear here so the player must remain vigilant.

**Asteroids** try to hinder the player by randomly flying in straight lines towards the player's general direction at different speeds. Colliding with the player will result in Game Over.

**Enemy Ships** have the distinction between three main attributes: Health, Speed & Damage, which progressively goes up the longer the player survives.

Come in the form of swarms (BOIDS), and Motherships. BOIDS will clump together and target the player, while Motherships spawn the BOIDS, and spawn them until the Mothership is destroyed.

While enemies spawn, their rotation is randomised to give a better feel for open space gameplay. This helps solidify to the player that an “up” or “down” is not important in this setting.

**Level Overview**

*JASP* pits the player against progressively harder and more enemies as the player survives more and more rounds of *JASP*. When the player starts, they find themselves in an open environment that is filled with the void of space, stars, and asteroids… ASTEROIDS? The player will be forced to manoeuvre around these obstacles, and soon have to fend off against enemy starships - blowing them up!

**SDD**

**SFX**

| **Sound** | **Filename** | **Loop** | **Variations** | **Notes** | **Status** | **Priority** |
| --- | --- | --- | --- | --- | --- | --- |
| Player Ship Fly | player\_fly | Yes | 1 | Flying | Completed | High |
| Player Taking Damage | player\_hurt | No | 1 | Rattling metal when the player is shot | Not Started | High |
| Player Exploding | player\_death | No | 1 | Explosion effect when the player dies in any way | Not Started | Moderate |
| Asteroid Exploding | ast\_death | No | 1 | Rubble explosion | Completed | Moderate |
| Enemy Exploding | enemy\_death | No | 1 | Explosion of the enemy | Completed | Moderate |
| Player Shooting | player \_shoot\_1 | Yes | 2 | Two different shoot sounds - turret & cannon | Completed | Moderate |
| Bomb Deployed | bomb\_deploy | No | 1 |  | Not Started | Low |
| Bomb Detonate | bomb\_explode | No | 1 |  | Not Started | Low |

**Description of SFX**

* The **Ship Flying** and **Accelerating** will be used throughout the entire game as they are the main mechanics the player will be using to navigate the environment. Ship flying is the base sound effect used, accelerating is used to amplify it, and decelerating is used to decrease the sound so the player is informed they are slowing down.
* **Player Boost** is used when the player dashes forward to propel themself further faster when they need to manoeuvre around faster for that short duration - for whatever the reason may be. This would be the ship accelerating but to a much more drastic effect. It is used in conjunction with the camera pulling back to really enforce how fast they are moving. It’s also a great way for the player to reach top speed faster.
* **Player Damage** is used when the player is shot at by an enemy spacecraft. This metallic like sound is used to inform the player they have taken damage, as it’s meant to imply that the spacecraft is collapsing in on itself.
* **Player, Asteroid** and **Enemy Exploding** is used when they explode! When the player explodes into a fiery ball of metal and scrap - this becomes the end screen for the player. The Asteroid exploding will sound like rocks crumbling, and it occurs when the player successfully blows up an incoming asteroid. Enemy exploding will sound relatively similar sound to the player exploding, but will be poppier and crunchier
* **Player Shooting** is used when the player is shooting with its two front facing turrets, an active mechanic the player will always be doing. The main cannon is fired in front for a large amount of damage, but with a significant cool down.
* **Bomb Deployed & Explode** occurs when the player drops a space bomb behind themselves - also with a significant cool down. The explosion occurs after the bomb has been floating in space for a duration, or when it collides with a rock or enemy spacecraft.

**Sound Files**

*32 bit, 44.1 kHz, 5.1 Channel .WAV*

The file format of .WAV is the decision for our audio because we don’t necessarily want our audio to be loaded into memory - when we can play it with very little latency. This is especially so when we want our lasers and ships to be precise with player input. WAV is also important for us because it is supported on our engine of Unity, and it is supported for PS3 which is our first major console release.

Since we don’t use any long audio files or tracks in our game, it makes sense to use WAV over MP3 since it allows us to retain that audio detail when we want it to be high in sample rate for crisp audio. MP3 files also don’t loop seamlessly which would be an issue when a lot of our audio design is centred around looping files.

**Risk Assessment**

| **Hazard** | **Risk** | **Solution** |
| --- | --- | --- |
| Using Headphones for a Long Time. | Potential ear infection and risk of tinnitus - loud sound can cause a headache. | Periodic breaks with headphones off - including taking hourly breaks away from the screen. |
| Issues with Adobe Audition. | Application issues with crashing and unresponsiveness caused by using Windows. | Switching to using Mac, constantly creating back-ups, or outright switching to Audacity. |
| Issues with Serum. | Application issues with crashing and unresponsiveness caused by using Windows - free licence ended. | Switching to using Mac, constantly creating back-ups, or outright switching to Vital. |

**Feedback**

**21st July**

***Joel*** *- A radar to show enemy location.*

Out of scope for us at the time being, there not being enough time to implement regardless of how useful it would have been for player feedback - and it is useful.

***Tom*** *- Enemy pinging via a voice com that indicates to the player where the enemies are coming from.*

Out of scope for us at the time being, there not being enough time to implement regardless of how useful it would have been for player feedback - and it is useful.

***Joel*** *- Highlight crosshair when looking at an enemy.*

Out of scope for us at the time being, there not being enough time to implement regardless of how useful it would have been for player feedback - and it is useful.

***Joel*** *- Implementation of 2-channel audio to indicate where the enemies are coming from.*

*This is used in game to indicate to the player where the ships are coming from as it is the minimum we could implement in the time we have.*