A picture containing text

Description automatically generated

LOST DROID

# 1. Characters

## 1.1 Player

There is one character in this game: The player-controlled character. You are the space droid.

**A picture containing water

Description automatically generated**

# 2. Story

In 2030, NASA has successfully discovered a wormhole that connects the Milky Way Galaxy to a galaxy millions of lightyears away, named *GardenOfEden-012*. Probes have discovered over a dozen planets like Earth. NASA believes that these planets can sustain life. However, human astronauts cannot safely make the trip into the wormhole and the probes cannot send information back once they’ve entered the atmosphere of these planets. In response NASA began Project Robot L. Robot L would be the first android created with a human consciousness uploaded into its programming. If successful, he would be the most advanced artificial intelligence being in existence. In 2035, Project Robot L was successfully completed. Robot L was placed on a small shuttle and launched into the wormhole with a mission to land on the nearest planet to the wormhole exit. The mission went south as the shuttle entered an asteroid field upon exiting the wormhole. The shuttle was destroyed but the droid survived. Now the droid is flying through space while dodging pieces of the wreckage and the asteroids.

## 2.1. Theme

The game focuses around a space theme. The player is supposed to feel overwhelmed by how massive the area is around them while also feeling like they are in constant danger. Obstacles start flying towards the player as soon as the game begins.

# 3. Gameplay

## 3.1. Goals

The player’s main goal is to avoid space debris and asteroids while racking up as many points as they possibly can. The point system is based around the in-game time so the longer you stay alive the higher your score will be.

## 3.2. User Skills

The only skill the user needs to have is the reaction time to move the droid around the screen to dodge the obstacles.

## 3.3. Game Mechanics

* ‘W’ or Up Arrow Key: Moves player in Upwards Direction
* ‘A’ or Left Arrow Key: Moves player in Left Direction
* ‘S’ or Down Arrow Key: Moves player in Downwards Direction
* ‘D’ or Right Arrow Key: Moves player in Right Direction
* If a player collides with an obstacle, then the game ends.
* If a player collides with an asteroid, then the game ends.
* If a player moves out of bounds (outside of the camera view) on either side of the screen, then the game ends.

## 3.4. Progression and challenge

The obstacles and asteroids spawn in the same location at set intervals but the actual obstacle or asteroid that spawns in randomized. This allows each game session to be completely random. For example, sometimes you may get three obstacles in a row that spawn on the left side and so the player just has to huge the right side. Sometimes the obstacles spawn in every part of the screen so the player has to be more proactive in dodging. Sometimes a large asteroid will spawn and collide with an obstacle before it even comes into the player’s view so an obstacle may come spiraling from an unexpected direction. The chances of this happening increase as the game goes on.

## 3.5. Losing

The player loses whenever they collide with:

* Obstacle
* Asteroid
* Off Screen Boundaries (Left, Right, Top, Bottom).

# 4. Art style

A star filled sky

Description automatically generated

**Graphical user interface, application

Description automatically generatedA picture containing object, star, sky

Description automatically generated**

# 5. Assets Used

Below is a list of free assets we used from the Unity Asset Store. We only used the material files from these sources for textures. The entire asteroids were prefabs from a package.

**Player:**

* Jammo Character | Mix and Jam : <https://assetstore.unity.com/packages/3d/characters/jammo-character-mix-and-jam-158456>
* Space Robot Kyle: <https://assetstore.unity.com/packages/3d/characters/robots/space-robot-kyle-4696>

**Obstacles:**

* Sci-Fi Barrel: <https://assetstore.unity.com/packages/3d/environments/sci-fi/sci-fi-barrel-175837>

**Asteroids**:

* Asteroids Pack: <https://assetstore.unity.com/packages/3d/environments/asteroids-pack-84988>

**SkyBox:**

* SpaceSkies Free: <https://assetstore.unity.com/packages/2d/textures-materials/sky/spaceskies-free-80503>

**Sounds:**

* Free Casual Game SFX Pack: <https://assetstore.unity.com/packages/audio/sound-fx/free-casual-game-sfx-pack-54116>

**Smoke**:

* White Smoke Particle System: <https://assetstore.unity.com/packages/vfx/particles/white-smoke-particle-system-20404>

**Misc.:**

* Game Design Document Template:
* **http://trickgs.com/blog/wp-content/uploads/2016/01/Tricks-GDD-Template-1.docx**

# 6. Technical Description

This game was designed to run on a Windows operating system. It was created using Unity 2019.4.12f1 and C# scripts.

# 7. Group Contributions

## Gezim:

* Obstacles Spawn
* Obstacles Movement
* Obstacle Hit Detection
* Border Hit Detection
* Instructed team on using version control and was responsible for merging and correcting conflicting file issues.
* Options Integration

## Dennis.

* Player Movement
* Asteroid Movement/Spawn/Collision (Edits to Gezim’s Original Obstacle Mechanic)
* Particles
* Asset Collection (Free from Unity Asset Store)
* Game Design Document Writeup

## John:

* Menu Creation
* Point Tracking System
* Game Design Document Writeup
* Skybox Rotation