Space Game (Prototype)

Game Design Document

April 2019

Contents

Game Overview

Game Developers:

Anthony Reese

Armando Perez

Dwayne Wilkes

Irina Brigido Chan

Game Title: Space Game (Prototype)

Game Platform: Mac OS10.11+ and Windows 7 SP1+

Game Style: Third-Person Action Exploration

Rijeka Repository: CS596S19_3D_Game_Team04

URL: https://tinyurl.com/y482yhe3

High Concept

Space Game is a third-person Action Exploration game where the player is a space traveller who crashed into an unknown world. The player needs to rebuild his spaceship by exploring the planet's environment to find the missing pieces, avoiding and defeating the different types of enemies encountered along the way.

Game Objective

The main objective of the game is to recover all lost ship pieces and successfully deliver them back to the main ship. There will be different types of enemies spread around the environment, as well as pickup items that will empower the player or replenish health.

Gameplay

Game Camera: The main camera is positioned behind the main character and view angle can be controled by moving the mouse .

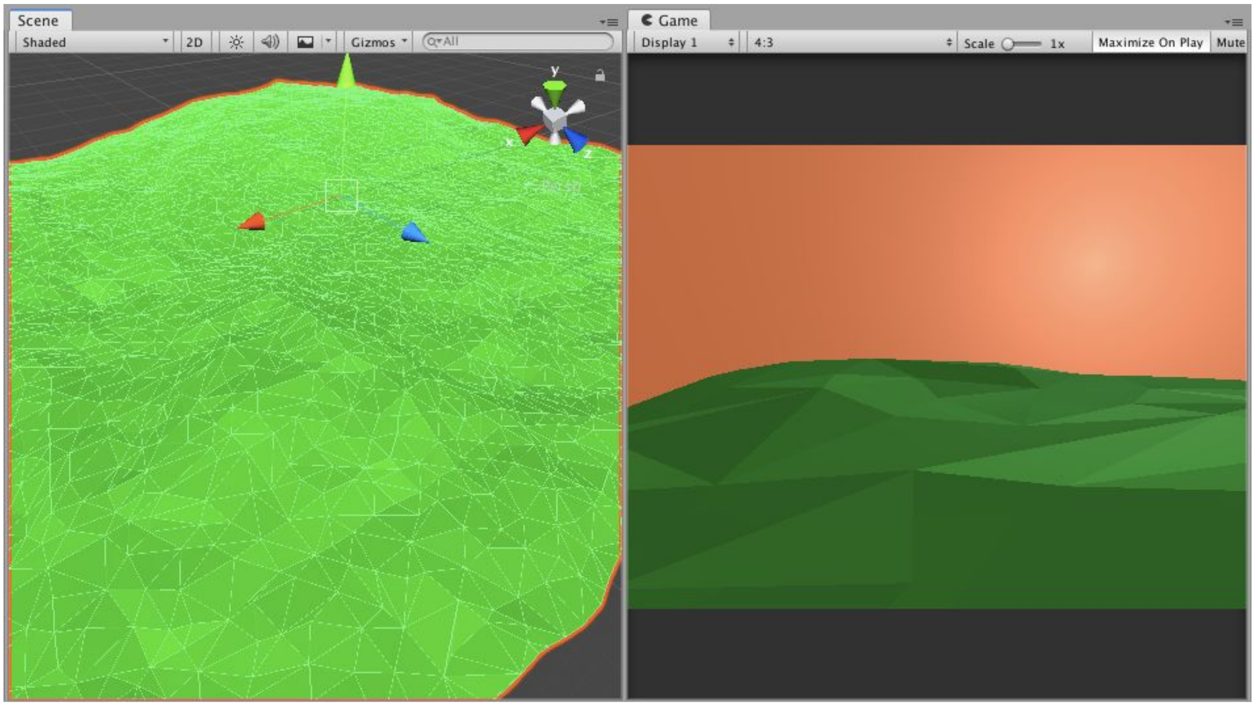
Game Controls:

- (WASD) and Space bar - Standard character movement and jump.
- (E) - Item pickup

Level Design

The art style of the game will be “low poly” in order to deliver a distinct environment feel, provide easier game asset creation and modification, and less computational demand for both development and gameplay.

The main level is the planet’s surface which consists of a large mesh with slight terrain variations. The player is set at the center of the terrain, along with the spaceship.



There will be environment enriching assets populating the planet, like rocks and vegetation. For easier development, the environment won't be fully populated at earlier prototype stages.

Ship Mechanics

The ship pieces are instantiated at the beginning of the game accross a set of available waypoints. The pieces are distributed randomly and no more than 1 piece can be instantiated at a waypoint.

As the player successfully delivers a missing piece, the ship's model will be updated to reflect its level of completion.

AI Mechanics and Behavior

Initially, there will be three types of Enemies AI:

1. Ranged static: Enemy that has low or no mobility and shoots projectiles at the player's direction;
2. Ranged dynamic: When detecting the player within a radius, the enemy will start following and shooting projectiles at the player's direction;
3. Melee dynamic: When detecting the player within a radius, the enemy will start following the player and will perform a melee attack when reaching the player's current position.

The team will implement additional AI mechanics as the above features are developed.

Game Asset and Script Bibliography

The following resources were used as reference for asset creation and game development during the prototype phase under Creative Commons[1] (CC) licenses.

Game Assets

The basic spaceship and its pieces were modeled on Blender and on Unity Editor. The following tutorials were used as reference for modeling and UV wrapping:

https://www.blendernation.com/2015/10/14/tutorial-how-to-model-a-low-poly-spaceship/

https://www.youtube.com/watch?v=V6OXSR5Ynyc

Script

The Unity Scripting Reference[2] was used extensively, along with the following tutorials:

- Instantiating Projectiles:

https://learn.unity.com/tutorial/instantiate?projectId=5c8920b4edbc2a113b6bc26a#

- Item drop and pickup mechanics:

https://www.youtube.com/watch?v=IEV64CLZra8

- AI Behavior:

https://www.youtube.com/watch?v=gXpi1czz5NA&t=491s
https://www.youtube.com/watch?v=xppompv1DBg&

[1] https://creativecommons.org/ Please refer to individual asset for specific version.

[2] https://docs.unity3d.com/ScriptReference/ version 2018.2