Shadow of the Dunes: Technical Risk & Assessment

Delivery Platforms

Windows, Mac, Linux and Modern Consoles (Difficulty: 2/5)

Shadow of the Dunes will be released on all three major operating systems (Windows, Mac, Linux) as well as modern consoles due to Unity's ease of creating builds for these platforms. Its third person camera and control scheme lends itself well to both controller and keyboard players alike.

Development Environment

Unity

Shadow of the Dunes will be developed using the Unity engine (Version 2021.1.19f1). This is because of our two programmers being comfortable with working in this engine, as well as being able to easily create builds for other platforms without massive overhauls in code for the sake of compatibility.

Git

Our choice for version control will be Git, and the main reason for this choice is that our repository service, Redmine, supports Git. While the programmers have experience in other software such as Subversion, all members of the team have experience using Git.

Gameplay Systems

Boat Movement (Difficulty: 2/5)

Half of Shadow of the Dunes takes place on the player's boat, and while this game is not a sailing simulator, we want the boat to float in the sand as if it was floating in a real ocean. This system is a matter of learning the necessary physics and tweaking it to feel responsive to the player.

"Boots-on-the-ground" Movement (Difficulty: 2/5)

A significant portion of the game will also be spent outside of the boat on foot, exploring landmarks the player comes across while exploring the sand ocean. The player will still be able to perform some actions, such as crouching and interacting with puzzle elements such as blocks and switches, or picking up/throwing objects.

Boat Upgrades (Difficulty: 3/5)

The primary method of player progression will be in the form of upgrades to the boat. This will allow the player to solve more complex puzzles in dungeons. Exactly what kind of tools the player will be using is unknown at this time, but it is likely that each will drastically change how puzzles are approached.

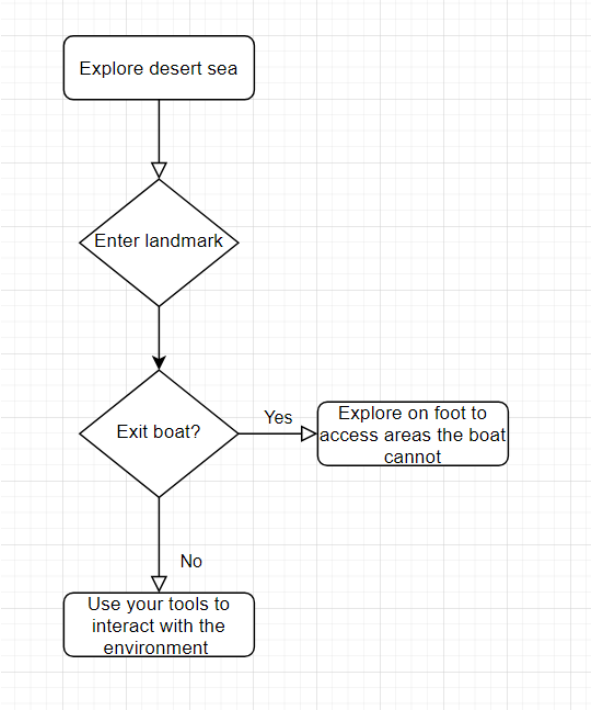
Cutscene and Dialogue Systems (Difficulty: 1/5)

The game will have many cutscenes, featuring dynamic camera angles and dialogue. Many of the tools needed for this are either already available in Unity or are simple to create - making this system simple from a technical standpoint.

Swarm Encounters (Difficulty: 4/5)

At various points in the game, the player will encounter an enemy known as a Swarm. The player is largely defenseless against these creatures, so stealth will be the key. This requires us to create AI for the Swarm to attack the player if it enters line of sight and to move towards noises. The player will be able to throw objects to distract them and make their escape. There will be many other swarm encounters that could call for new behaviors, such as an encounter in the desert sea, so this will be a difficult system to implement.

Logical Design Flow



Pipelines

Art Pipeline

Our artists will have a special folder in the repository that they will use to upload art assets for the prototype through Git.

For art pieces that serve as destructible objects, the programmers will likely have to spend time implementing them manually

Sizing

n/a

Naming Conventions

n/a

Design Pipeline

Due to both of our designers having familiarity with Unity, design changes can be easily made by the designer at any time by editing values through the inspector. Nearly any variables that change up gameplay are available to edit without the need of editing the code itself. The designers will meet with the programmers often to ensure they understand their direction fully, and the programmers will maintain a google sheet including all the game's variables, what they do and how to change them for the designers to have an easier time changing things.