Shrinky World Design Document

3GB3 Assignment 5

Guang Yu Zhang

April 9, 2018

Introduction

The goal of Shrinky World was to create a mobile game targeted at casual audiences. There are several design principles to consider while creating a mobile game. This document will outline some of these principles while evaluating the implementation and design choices. “Less is more” is the underlying design principle for this project. Many mobile games such as “Flappy Bird” employ simple mechanics to achieve significant results in gameplay, Shrinky World attempts to apply the same principles from these games. Note: This game was heavily inspired by a project from Ludum Dare. As this assignment is focused on the polish aspect of the game, several improvements to the game were made.

Gameplay

Shrinky World is a game where players drive around a spherical planet that is constantly decreasing in size. Meteors spawn and crash on the planet which create craters that players must avoid. Green diamonds also spawn on the planet for players to collect. Upon collection, the planet will increase in size. The goal is to survive as long as possible and achieve the best score.

Menu

The menu consists of several elements including the Play, Quit, Next and Previous buttons as well as a preview of the currently selected planet. The focus of the menu is to provide players with a gateway to access the game. Several options such as customization and high score are also pre-gameplay features that are included in the menu. Originally the menu only had the Play and Quit buttons but options to change planets and display score were added during the polishing stage of the game. The text colours for the UI items were also changed to black for clarity as per player request. A theme overhaul was necessary after changing the planet model.

Design Choice: Players are able to view their high score after the game but should also be able to view it when the game is launched. The score display is a core component of this game and is provided with clarity. Customization features are also provided prior to the start of the game for obvious reasons.

Score

The original scoring system was based on the size of the planet; the score displayed the size the planet shrunk to when the player died/lost. This system was changed because the growing mechanic was added (View Power Ups section). Planet size was no longer a valid metric for determining skill as players had methods of increasing the planet size so the scoring was changed to be based on time. Following the “less is more” approach, no further metrics are provided. The score system enables players to attempt to beat their personal high scores as well as challenge others’ scores. “Flappy Bird” became widespread through people's’ competitive nature and this game seeks to recreate that. After several playtests

with friends, it appears to have been a success as many people played for more than 10 minutes in order to beat each other's high scores.

Design Choice: Time provides a consistent metric for measuring progression however, through player testing, many people reported that distance is more meaningful in terms of player satisfaction so the time is displayed as distance traveled in meters.

Customization Features

Although the focus of this game is to be simplistic, customization features can add flavour and variety without changing the underlying mechanics of the game. Throughout the polishing phase of the game, a system to select the planet is provided. Avatar selection was not implemented due to time constraints but is another aspect that can add player individuality without changing the underlying principles of the game.

Design Choice: For part 3, different planets were added which had different properties such as driving speed and different shrinking speeds and meteor spawn speeds but were ultimately removed as it interfered with the underlying design principle of “less is more”. Instead, different models for planets were added without changing its’ properties and challenges.

Player and Controls

The avatar moves at a constant speed around the planet without speeding up or slowing down. Players have restricted control of the direction the car is going; the directional change is static. This is essential in creating a “less is more” type of game. For reference, “Flappy Bird”’s only control is tapping to fly.

Design Choice: The controls were intended to be restrictive to add a layer of depth and a learning curve. For mobile games, simple controls are intuitive and required. Accelerometer controls were experimented with but were discarded as they were unreliable. Instead, the screen is divided into two halves that represent directional input (Simulated by “a” and “d” keys).

Obstacles

Meteors spawn 20 units away from the planet and get pulled towards to planet through Unity’s Flux Gravity script. As they land on the planet, they create creators that players need to avoid. Once struck, the game is over.

Several other obstacles were tested such as puddles of sludge that slow the avatar and cracks that the player must jump over. These were ultimately removed as they added unnecessary clutter and did not add enough depth.

Balls with physics enabled were tested to spawn instead of meteors. On impact with the planet, they bounced and rolled around the planet. This could be included as another game

mode as it changed the dynamic of the game while maintaining the core mechanics of avoiding obstacles.

Design Choice: Some obstacles were tested and removed because they created unnecessary clutter. With one obstacle, the objective is clear; avoid meteors and craters.

Power Ups

Originally the game did not have any way to recover so the average score was similar for all players. The lack of a “grow” mechanic restricted players from achieving high scores past a certain point as the planet would become too small to maneuver around craters. The green diamond power up was introduced to provide more depth. These powerups spawn every 5-10 seconds randomly around the world. Upon collecting the power up, the planet would grow by a set factor. Several restrictions were added to ensure that the planet would not grow past it’s original size. If the planet is smaller than 50% of its original size, collecting a power up will set the size back to 50%. If this were not implemented, an issue with skill cap would be introduced as players would eventually lose regardless of how skilled they are.

Design Choice: Only the growth mechanic was added because it increases the skill range a player can attain without over complicating the game.

Note: Several testers have noted that they did not initially notice the effects of collecting the power up however, after several playthroughs and increases in player skill, they did notice the planet growing.

Conclusion

The focus of this project was to add depth and polish to a game while maintaining a simple “less is more” approach targeted towards casual mobile gamers. This was achieved by providing customization without changing any mechanics and maintaining a simple control scheme. Depth was added through the inclusion of the “growth” mechanic to allow for potentially infinite gameplay.

Sources

https://ldjam.com/events/ludum-dare/38/shrinking-planet - Original Concept
https://assetstore.unity.com/packages/3d/environments/sci-fi/low-poly-planets-pack-96311 - Low-poly planets
https://assetstore.unity.com/packages/3d/props/3d-balls-collection-103428 - Balls collection
https://assetstore.unity.com/packages/audio/sound-fx/sound-pack-free-pack-78186 - Free sounds