Lappeenranta University of Technology
CT60A5400 – Fundamentals of Game Development
Autumn 2019

GAME DESIGN DOCUMENT

Sweet Duck

Author:
Laplaza Osta, Irene - Student Number: 0556340

Course Professor:
Jussi Kasurinen

Course Assistant:
Syed Naqvi

Game Design Document

- **Define your own game idea with minimum of 1-3 sentences per point.**
- **Consider the developed game from the viewpoint that you are actually making something which will be published at the end.**
- **Focus on the aspects 1-4, others you can define in more general terms since this is only the intro course into game development.**
- **The main aspects you are defining here are the following:**
    - ○ **What would you like your game to look? Search or draw some concept art for your game, levels and characters.**
    - ○ **What is the player doing in the game, what are the abilities of the characters?**
    - ○ **What technological demands or requirements do you have? What technical aspects have to be solved for your idea to work?**
    - ○ **What will your interface look like? What control scheme are you using and how does it work?**

Topics with (*) usually extended to separate detailed documents, in this project has been simply summarized

1. Executive Summary, Quick overview

Sweet Duck is a side-scrolling game featuring 2D style graphics inspired on the 2013 smartphone game. The main objective is to obtain the maximum score by directing a flying duck, who moves continuously to the right between sets of lollipops. If the player touches the lollipops, he/she loses. As the duck moves forward, enemies will start to appear (butterflies in this case) that the player will have to avoid if he does not want the game to end.

2. Target Audience

I honestly think that the assumption that a target audience with certain age, gender and country plays a specific game is not as accurate anymore. Therefore, my target audience will be **anybody** (women, men, even pets, etc.) who are looking for a fun and entertaining time. It will be really useful for example when people are having short breaks during work/study and need to relax and recover energy, since Sweet Duck does not have a story and then people can play it whenever they want without wasting too much time.

3. Main Characters

The main character is the duck. It is in charge of performing actions such as flying, dodging obstacles and enemies as well as making the player get the highest possible score.

4. Main Features

4.1 Main mechanics

Game mechanics revolve around how well the player controls the duck. The main character briefly flaps upward each time that the player taps the screen. If the screen is not tapped, the duck falls because of gravity.

There is no variation or evolution in the map throughout the game, as the lollipops always have the same gap between them. However, the enemies will randomly appear on different places hindering the player to continue scoring several seconds after start playing.

There is no end to the running track, having only the flap and ding sounds and the rising score as rewards.

There is no end or victory condition, the game will continue until the main character dies. If the main character does not avoid an obstacle or enemy, it will die instantly.

4.2 Movement

The game is controlled with the mouse, tapping the screen. To obtain points in the game, the player must control their tap force and handle the drift of the duck.

As mentioned above, if the screen is not tapped, the duck falls because of gravity.

4.3 Physics

The duck will die if it collides with a lollipop or with a butterfly. So the physics in this game are mainly Rigidbodies to react real-time collisions and Colliders adapted to the different GameObject's shapes to kill the duck if it touches them.

The behavior of the GameObjects is controlled by the script components. The editor used for scripting has been Visual Studio 2017 and the programming language has been C#.

4.4 Multiplayer mode

There is currently no multiplayer mode in this game, but it will be a good option to develop it in case I extend the game for implementing it as a commercial project

5. Genre, Setting, Concept Art book*

Sweet Duck is an arcade type game. It has been built to be enjoyed alone or with some friends (each of them playing from their own devices, as there is not currently multiplayer mode). Below there are some pictures of how the game looks like:

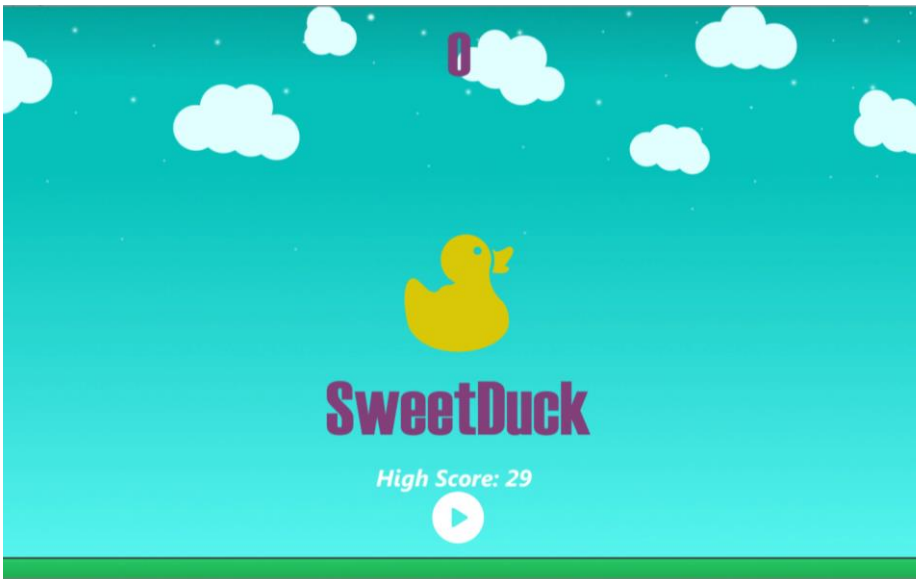

Figure 1. Start Page

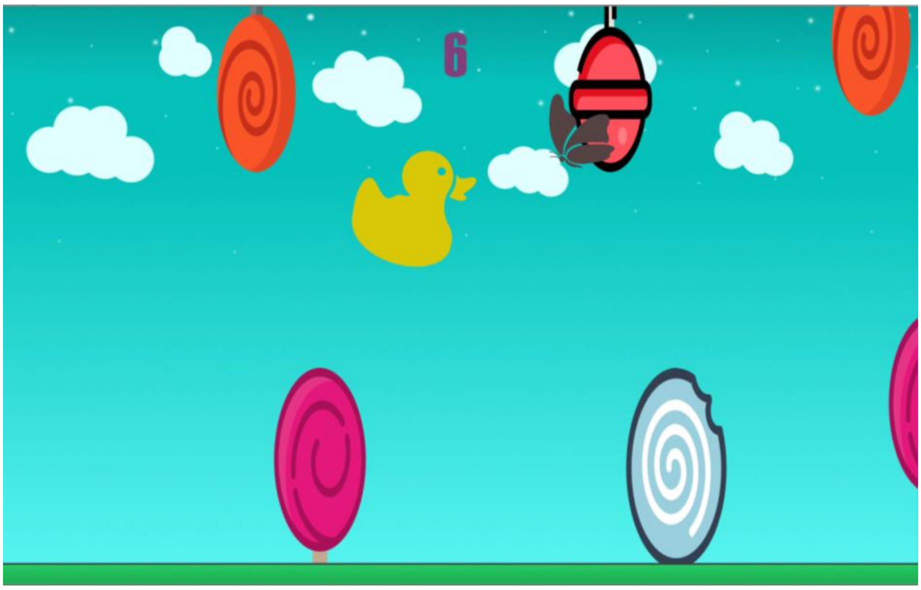

Figure 2. Main map

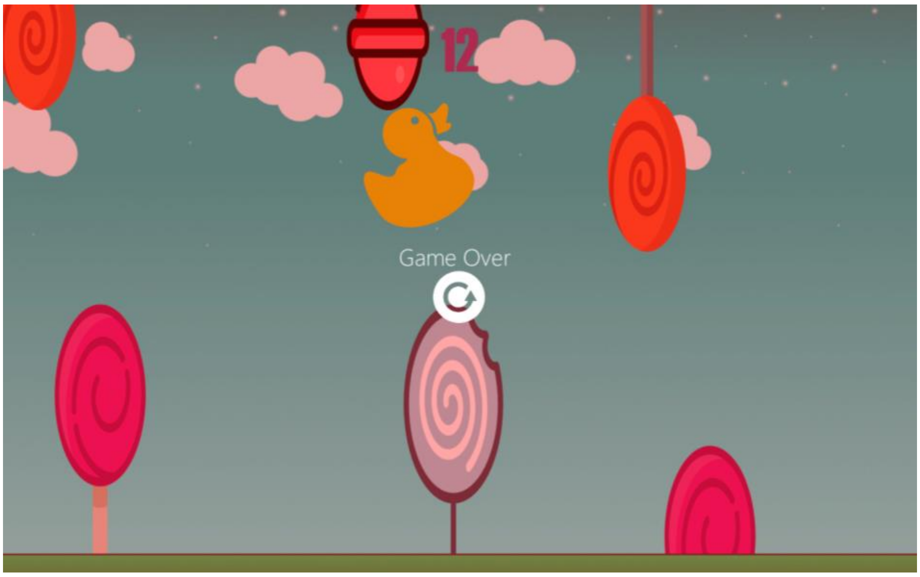

Figure 3. Game Over Page

6. Enemies, NPCs, Other objects

The butterflies are the duck’s archenemies. They can appear everywhere in the map (up, down, in the middle) going to the duck’s opposite direction (i.e. to the left). They are not dangerous, but it is better not to touch them, otherwise they will kill our main character.

The lollipops are the obstacles that the player has to avoid. They look really sweetie but our main character will die if it touches them. They always have the same gap between them but they appear in randomly-different positions.

*7. Story board, script**

Unfortunately, there is no story in this game.

*8. Technical definitions, Tech guide**

8.1 Platforms, versions

It is built as a Standalone application. Its main target platform is Windows, but it can also be run for Mac and Linux.

8.2 Control Scheme

The game is controlled only with the tap force of the mouse. Depending on how many times the player hits the mouse left button, the duck will fly higher or lower.

8.3 Limitations

Since it was the first time I do programming, my knowledge did not allow me to do incredible things. The playable levels do not differ much from each other and the duck does not change its speed as the game progresses. I would like to improve it in the future.

*9. Business definitions**

Business-wise, the original Flappy Bird game generated millions of downloads, so Sweet Duck could probably follow the same path. However, to implement it as a commercial project, several improvements must be done.

9.1 In-app purchases

There are no in-app purchases.

9.2 DLC packs

Possible DLC packs to change the main character and enemy's appearance as well as the main map.

10. Outsourced/Bought Assets

I have not used any Asset from the Unity Assets Store. I have created almost everything by myself:

- Background, clouds, stars and ground with GIMP editor.
- Sounds with Bfxr mixer (Countdown, Die, Score and Tap)
- Menu Theme Music with an online mixer

The rest of Sprites (Duck, Lollipops, Butterfly, Play Button and Replay Button) were downloaded from Flaticon (all of them in white to be able to change their color).