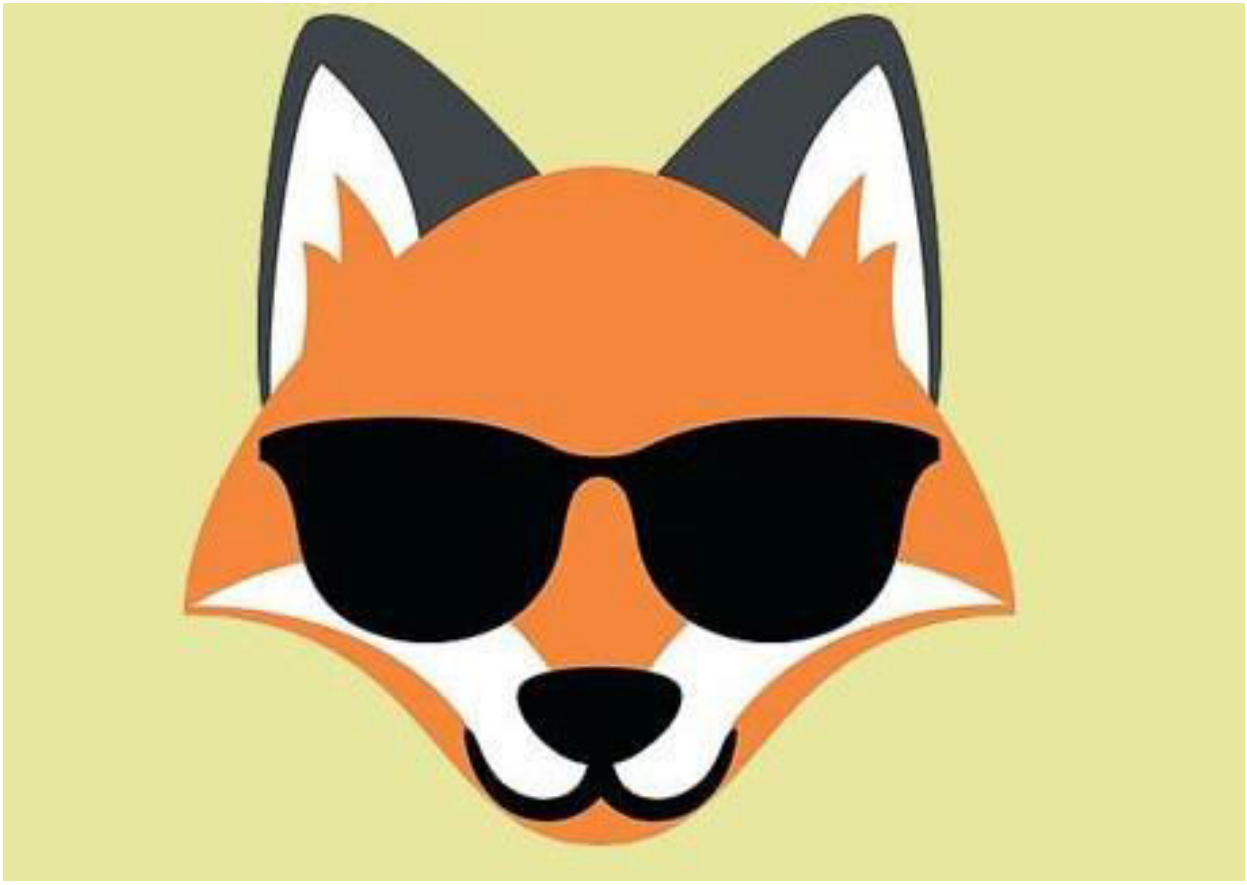
GAME DESIGN DOCUMENT

"Cool Fox" Unity Game



Filippidis Odysseas

12.09.2019

Games and Media Technologies

The University Of Sheffield.

CITY College An International Faculty Of The University.

OVERVIEW

Cool Fox is a side-scrolling 2d action game inspired from the well-known Super Mario game. The main character is a fox that wanders in the forest, trying to assemble as much food as he can in order to feed his family. This game is a one-player sprite-based game which is intended for desktop computers.

CHARACTERS

As mentioned before, the main character is a male fox, which stands on two rear legs and shown in figure 1. He is able to run and jump on the two legs and thus collect the food needed for his family. The food in this game is cherries that are found on trees and the fox need to jump in order to collect them. The fox does not have any name but he considers himself as “cool”, hence the name of the game.

Figure 1: Fox Avatar.

Another character found in this game is a poisonous frog, shown in figure 2. Actually there are a number of these frogs in the game and they are regarded as villain characters of the game, as they are able to hurt the fox. In detail, the fox must not make a physical contact with any of these frogs and needs to simply avoid them by jumping over them. Otherwise, in a single touch with one of these poisonous frogs, the fox will die and will lose one of his lives.

Figure 2: Frog enemy

STORY

The fox has a family, a wife and three kids. After giving birth, his wife got sick and can not hunt with him anymore and she is hardly able to look after the kids. Consequently, the main character needed to look after his family on his own and thus bring more food in order to feed all the family members. This way, he decided to launch a short journey to the forest in order to collect and bring back cherries. The particular choice of cherries is related to the illness of his wife, as cherries are antioxidant. Finally, the fox starts his journey with a vivid mood and “cool” style.

THEME

This game is mostly about a fun story. The story is mostly related to hardships indeed, but the main character is optimistic and a very strong personality. For this reason he considers his problem as a challenge and will try to have fun in the process of solving it. This way, the game is supposed to have a sense of humor and be funny.

GAMEPLAY

Before proceeding with the analysis of the gameplay and its subsections, it is important to point out that the definitions of Gameplay and Game mechanics are not universally agreed in the literature of Game studies and ludology. There are various definitions of those two concepts in the literature. Therefore, it is

crucial to define these definitions for the current case beforehand so that the readers of this document understand the same thing and be on the same page.

Game mechanics is a set of rules which define the way that the player interacts with different objects or elements of the game, which in turn influences and changes the state of the game. In essence, game mechanics describe what the player and other entities in the game are able to do and how they do it. These mechanics are understood by the player while he plays the game and explores its simulated world through the use of a feedback mechanism. This way, the player is guided to behave in a particular way, since these rules constrain the space of possible plans to attain the goals of the game.

Gameplay is the result from interacting with the game mechanics in order to achieve a goal. Hence, gameplay is tightly connected with the notion of game mechanics. This way, it would not be wrong to say that gameplay defines what a player can and can not do in the game and how the game responses to these actions. Additionally, gameplay defines separate actions in the virtual world regardless of the player's actions, showing the liveliness of the world. However, despite the fact that gameplay and game mechanics are strongly related, it is important to note that gameplay is not merely a set of game mechanics. Gameplay uses game mechanics as a tool in order to provide the player an enjoyable experience and ultimately entertain him. In other words, the gameplay provides a player the illusion of a magical world where he has abilities, goals and so forth. The game mechanics, on the other hand, are the underpinning mechanics of movement, shooting, fighting and others, that enable the game to provide this illusion.

GAME MECHANICS

The core game mechanic of the game is the mechanic of the jump act, since it is supposed to be the most used and vital act of the player to progress the game. To jump the player needs to push the "Space" button and the avatar at that point of time needs to be located either on the ground, plank or any other rigid body. This way the player is restrained to push subsequently the "Space" button and jump even at the air higher and higher.

The above game mechanic is the only core game mechanic in the game. The rest are considered to be secondary game mechanics, as they are not used so extensively by the player. One of them is the collecting game mechanic, which dictates the exact way the cherries are collected. According to this mechanic, the avatar need to touch the cherries on the tree in order to collect them. In other words, he needs to collide with the cherries object in order to trigger the collecting game mechanic. Finally, whenever this mechanic is triggered, the score of the game increments by one.

Another game mechanic is the "hurt" mechanic, which represents the death of the avatar and loss of one life. This mechanic is triggered whenever the avatar comes in contact with either the bomb object or the poisonous frog object. The contact needs to be the same as with the collecting mechanic in order for the "hurt" mechanic to be triggered.

Obviously, whenever the "hurt" mechanic is triggered, another game mechanic needs to be triggered for

the object that the avatar came in contact with. For instance, when the avatar comes in contact with a bomb object, explosion game mechanic is triggered in order to play the explosion animation, the explosion sound and so forth. The same goes for the case of touching a poisonous frog, where "frog attack" game mechanic is triggered.

Also, there is the "fall" game mechanic which covers the case where the player falls into a gap. This game mechanic is triggered when the player falls into a gap and the avatar touches an invisible object below the camera as he falls down. When the mechanic is triggered, the player loses a life and the avatar respawns at the last checkpoint.

Respawning feature is yet another game mechanic employed in the game. This mechanic is triggered right after the "hurt" mechanic or the "fall" mechanic. Whenever it is triggered the avatar disappears and reappears again in the location of the last checkpoint that he reached or the beginning of the game.

Another secondary game mechanic in the virtual world is "run" game mechanic, which is triggered whenever the player pushes either the "left" or the "right" button. Whenever this mechanic is triggered, the running animation is played and the avatar moves to the corresponding direction. In order to make this mechanic to work properly it was required to make the avatar flip to the correct direction whenever the player wanted to run either right or left.

One more secondary game mechanic, which is not very obvious, is the frog's "flip" game mechanic. The purpose of this mechanic is to make the frog to always look at the avatar's direction. This way, it seems like the frog always stares at the avatar in order to hurt him whenever he comes close enough.

All these game mechanics are responsible to deliver the illusion to the player that the world of the game is real and vivid. That is, they are combined in order to create the gameplay for the player to enjoy.

GAME PHYSICS

The game physics that are incorporated in this game are the ones provided by the Unity 2d game engine. More particularly, the Unity's Rigid Body Dynamics are used, which use Newtonian principles of movement and mass to simulate rigid objects such as boxes, spheres, walls and so forth. Also, Unity's 2d colliders are used to detect the collisions of these rigid objects. After detecting a collision, a collision resolution is followed and applied. This collision resolution uses an animation, which should be appropriate for each case. For example, when a collision is detected between the player and the bomb, an explosion animation is played.

GOALS

The long term goal of this game is to gather enough food through all the levels and bring it back home his family. The short term goal is to pass the current level collecting as much cherries as possible and progress to the next level.

USER SKILLS

The avatar does not have any special skills in this game. The game is very simple, so the avatar needs only to be able to avoid gaps, obstacles and frogs by simply jumping over them. Thus jumping is the only skill that is provided to the player to achieve all the goals.

LOSING

This section describes the losing conditions of the game. In other words, the cases where the player loses and also the cases where his avatar loses a life are described here. First of all, it is important to note that the player has three lives in total at his disposal. After spending all three lives, the player loses the game and starts over again from the previous level, with the score that he started that level. In case where the player lost in the first level, he is redirected back to the start screen. At each time that the player loses the game, a "losing" sound is played and a message pops up at the screen, notifying the player that he lost the game.

So, there are three conditions under which the avatar loses a life. One is when he falls into a gap, the other is when he is hurt by the bomb and the last one is when he is hurt by the poisonous frog. After losing a life in either of the cases, the avatar has one less life and is respawned again (in case there are still lives left).

MUSIC AND SOUND EFFECTS

In this game various sound effects were used in order to deliver pleasant aesthetics and a more vivid atmosphere in the game. These sound effects are all royalty free and found mainly in two websites, the www.freesound.org and www.freesfx.co.uk websites. More particularly, the sound effects used in the game world are the following:

- jumping sound effect of the fox avatar
- explosion sound effect of the bomb
- looping sound effect of the poisonous frog
- frog attack sound effect
- collecting cherries sound effect
- sound effect for losing a life
- sound effect for losing the game
- sound effect for winning a level
- sound effect for reaching a checkpoint

Additionally, a soundtrack is incorporated in the aesthetics of the game in order to improve the player's experience even more. This soundtrack is also royalty free and was found in the www.purple-planet.com website.

TECHNICAL DESCRIPTION

As mentioned in the overview of the document the game is intended for stand-alone PCs. There are no specific system requirements for the game to be able to operate at its full capacity, as it is a simple indie game intended for low-end PCs.

The game is intended to operate only on the Windows platform, which uses either 64 bit or 32 bit architecture. Also, for the development of the game Unity engine was used. More specifically Unity's 2d engine version was used to develop and build the game with the preferred configurations.

TARGET AUDIENCE

The game targets mainly casual players of any gender. The age of the target audience ranges from 10 to 50 years old. Mostly though it is designed for secondary or high school children, since it has a very simple long-term goal, cheerful atmosphere and a family related story.

PROJECT SCOPE

There are three levels in this game, each of which has the same objectives: to collect as many cherries as possible, avoid obstacles and reach the final destination. All three levels use the same virtual world, which means there is only one virtual world in the game.

USER INTERFACE

After the splash screen, the game starts with a start screen shown in figure below.

Figure 3: Start screen

At this point the player can either start the game by clicking on the fox picture or terminate the game by clicking the “QUIT GAME” button.

Apart from that, there is also the pause menu screen shown in figure 4, which pops up whenever the player pushes the “Escape” key on his keyboard. At this screen the player can either resume the game by pushing the “RESUME” button on the screen or the “Escape” key again. He can also go back to start screen by pushing the “MENU” button or terminate the game by pushing the “QUIT” button on the screen.

Figure 4: Pause menu screen

Furthermore, as can be seen in figure 4, while playing the game three different UI elements are provided to the player. In the upper-left corner two UI elements can be noticed, the “Number of lives: “ element and “Score: “ element. Also, in the upper-right corner “Level: “ UI element can be seen. The meaning of these UI elements are self-explanatory. These three elements provide vital information to the player at real-time while he plays. This way the player knows the state of the game at any point of time and is able to make better decisions.

At this point it would be useful to show a basic flowchart in order to illustrate in detail how the player can navigate through these screens. The detailed flowchart in figure below is used for that purpose.

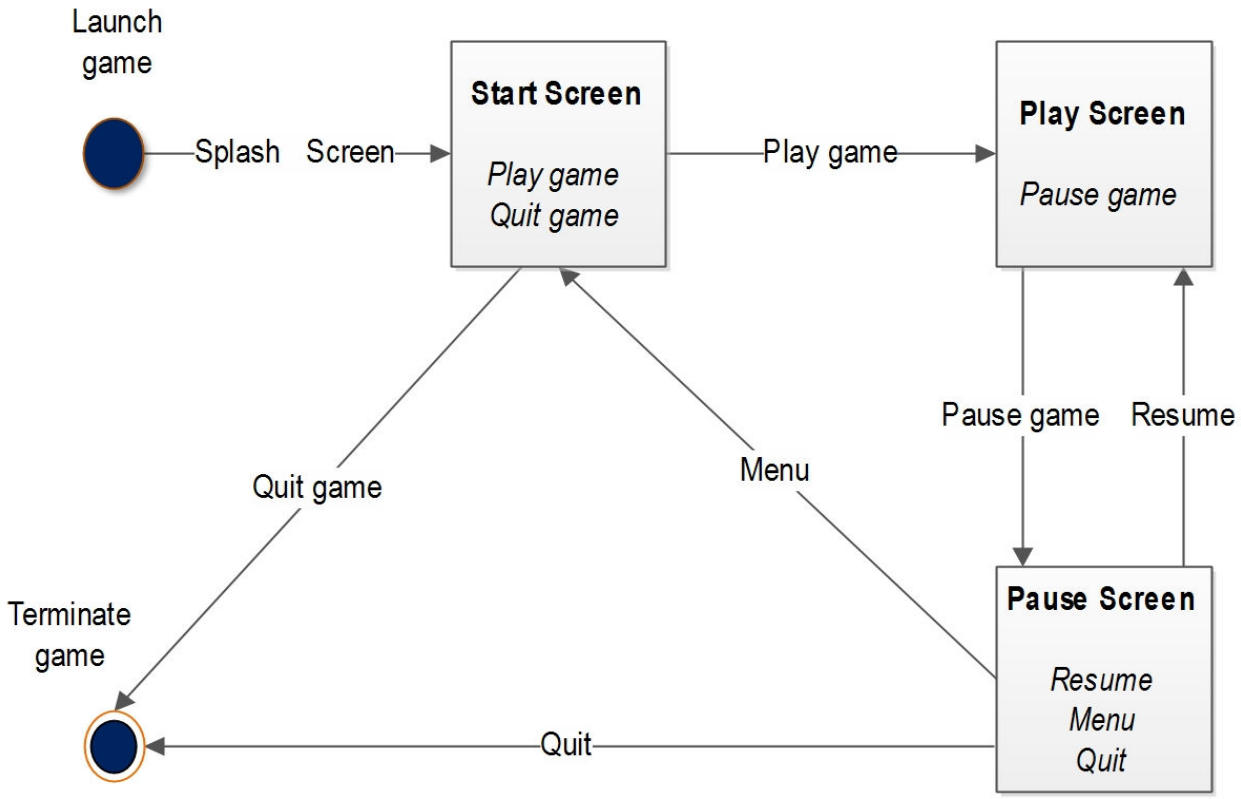


Figure 5: A flowchart showing the navigation through the screens of the

GAME ART

The sprites that are used in the game are all free and can be found in the Unity's asset store. More precisely, all the sprites of the game come from two free assets: Sunny Land and 2D Sprites pack. The sprites of the fox avatar, the frog, the cherries and the tree are taken from the Sunny Land asset. All the rest sprites are taken from the 2D Sprites pack.

However, some sprites in the game were created manually, since the two assets did not have the needed sprites. These sprites are the two crossed finish race flags, shown in figure 6, and the checkpoint flags (red and green), shown in figure 7. Also, the picture of the start screen, shown in figure 3, is also manually created, as it needed to have the appropriate fox drawing and title.

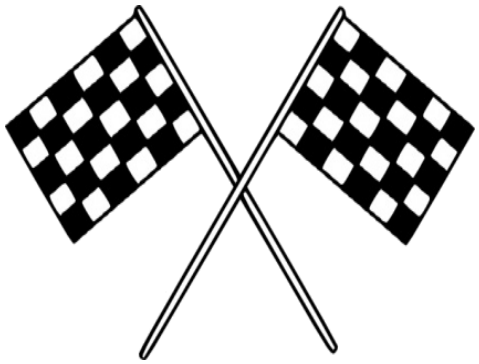


Figure 6: The two crossed

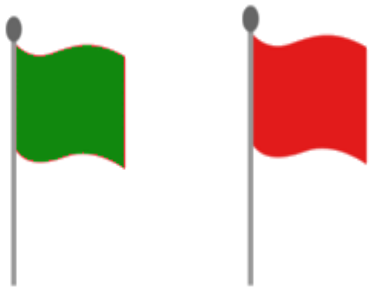


Figure 7: The red checkpoint