

PONG

Analysis

by James Dimech

BA INTERACTIVE MEDIA YEAR 2

WHAT IS PONG?

Pong (1972) is one of the many early forms of video games that emerged back in the early 70s. While not being the first game ever created, many assume it's the cause for putting the video game industry to where it is today. Most of the video games that were out during the time were viewed as being too complex and thus helped pong become more popular to play due to its simplicity. Pong was simple in design as well as functionality, three objects and one goal, don't miss the ball. (Editors of Encyclopaedia Britannica, 2018)

THE ORIGINAL GAME

Ralph Baer and Bill Harrison tested out a demo of Pong using a 'module' that consisted of two controllers connected to a main system that was plugged into a TV screen. The controller consisted of a few buttons, a knob that moved the paddle left and right, another knob that moved the paddle up and down, another knob that allowed them to control the curves of the ball and finally a button that spawn the ball onto the screen.

However, the original pong was not programmed using code, it was engineered using a circuit board which consisted mostly of transistors.

The game's physics were developed realistically. As soon as the ball struck the paddle at a specific point, the ball's trajectory would be reflected in the other direction. Holden wrote, "This was done by dividing the bat into segments...", and further on explains that as the ball travels back and forth on its horizontal path, it will increase speed over time. Once either player misses the ball, the speed is reset. (Holden, 2013)

SOFTWARE DEVELOPMENT MODEL

I will be creating a clone of the classic pong game using Unity and code instead of circuits and transistors. The game will function just like the original with two paddles, one being the player and the other being the AI, and a ball that both sides need to successfully send into their respective goals.

I will be using the Waterfall development model which consists of six steps.

1. REQUIAEMENTS

The first phase of development is listing the 'potential requirements' of the software which will help the developers as a reference. "The result is typically a requirements document that defines what the application should do, but now how it should do it." (Powell-Morse, 2016)

For my Pong clone, there are three requirements. First the game must have a ball that can mimic the physics of the original. Being able to change trajectory depending on what part of the paddle the ball hits. Secondly both paddles must be working, one for the player which will be manipulated with the arrow keys and the other one as the computer paddle which will act as the players opponent. Finally, a working score system that will keep the score of the game.

2. ANALYSIS

The next step where the software or system being developed is analysed thoroughly to see what exactly has to be made first and then afterwards and so on. (Powell-Morse, 2016)

First the paddles need to be set up, initially allowing the player to use the arrow keys successfully. Following that, the paddles need to reflect the ball upon contact. Finally, once that is done, the final step is to add a simple AI into the opposing paddle.

Second is the ball which needs to be able to bounce off the paddles upon contact and not pass through. The ball must also change trajectory, depending on where it touches on the paddle. Next the ball needs to be set up so it won't pass through the top and bottom walls and instead bounce off. Finally, the ball has to reset when it reaches the left and right walls past either paddle.

Finally, the scoring system. The system will add one point to the deserving sides' score once the ball reaches either end. Once the point is given, the ball will go back to the middle of the screen and fire in a random direction.

3. DESIGN

The part of the model where we take what was gathered in analysis and begin preparing what is needed to implement those requirements. Such as assets, scripts, what programming language can be used. (Powell-Morse, 2016)

Three assets will be used with 1 of them having no particular functionality. Pong is essentially overhead tennis, therefore a field will be implemented. The first asset is the background which will only be a reference for players to know exactly where the ball is. The other two assets are the paddles which is a white rectangle that can be duplicated for the other paddle and the ball which is a white square. Csharp will be used as the coding language in UNITY.

4. CODING

This section is where the code is finally starting to be implemented. "...implementing all models, business logic, and service integrations that were specified in the prior stage." (Powell-Morse, 2016)

Since my game only consists of three main assets, I essentially need them to be placed in before I can start coding. After the assets are setup I will be coding and adjusting as needed.

5. TESTING

The stage where the software is tested, and errors are found and fixed. "It is not uncommon for this phase to cause a 'necessary repeat' of the previous coding phase, in order for revealed bugs to be properly squashed." (Powell-Morse, 2016)

6. OPERATIONS

This is the final stage where the software or game is compiled and exported. In terms of the game, the final step is to create a "build" which will be the actual finished product itself. In most development projects, maintenance should be performed, and updates should be made to keep the software up and running. (Powell-Morse, 2016)

REFERENCES

Editors of Encyclopaedia Britannica, 2018. Pong. [Online]
Available at: <https://www.britannica.com/topic/Pong>
[Accessed 17 04 2018].

Holden, D. H., 2013. Building a Digital Video Game with 74 Series TTL IC's. Atari Pong E circuit Analysis & Lawn Tennis.

Powell-Morse, A., 2016. Airbrake.io. [Online]
Available at: <https://airbrake.io/blog/sdlc/waterfall-model>
[Accessed 01 05 2018].