# Final Report CS 443 Project

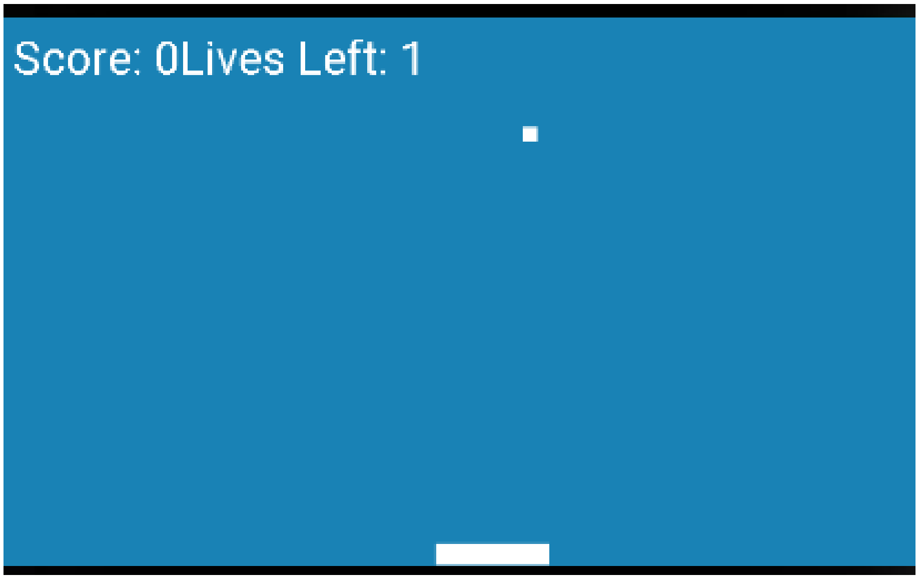
# Paddle Ball

Akash Subedi

05/26/2019

## Project Statement

The purpose of this project is to build a simple game for entertainment. The name of the application is “Paddle Ball”, an old school game we used to play on GameBoy and other similar consoles. The game is very simple, and it has only ball and paddle as the parts, but it can get very challenging and frustrating to get good score.



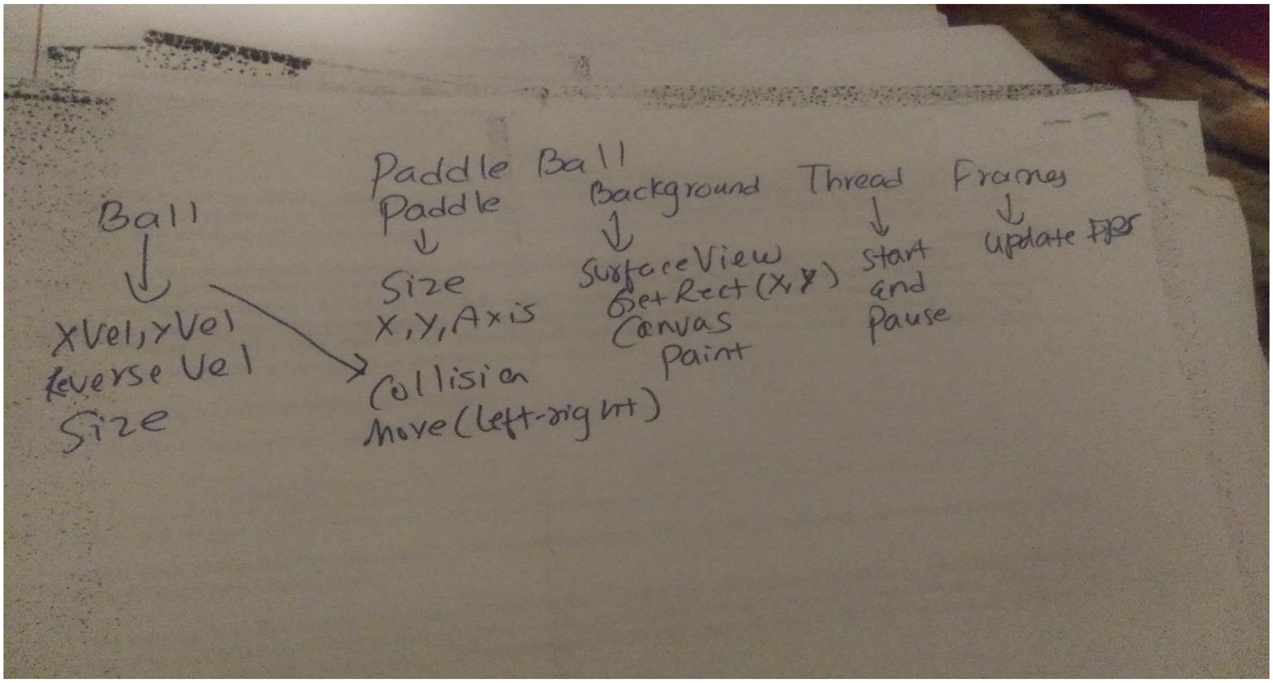
Score: 0Lives Left: 1

This picture demonstrates the game. The goal is to block the ball with paddle as many times as you can. After each block, the velocity of the ball increases. User can tap either right or left side of the screen judging the position ball returns to. The balls can hit any side of the walls.  The user is given 3 lives at the start and after he loses all the life the game restarts. Usually, after you get score of 5-6 the game can get unbelievably tough as you must have superhuman reflexes to keep blocking the ball with the paddle. The physics of velocity, canvas and thread used in Bouncing Ball sample provided to us made it little easy for me to build this game.

This app needs to special requirements and instructions to compile. You can build and run the apk from Android Studio and it runs on devices having 6.0 Marshmallow or higher.

# Application Design

This is general design of the application.



# Implementation and Evaluation

Ball Class – This contains implementation ball's size , its velocity , collisions, reverse velocity and its position in surface.

Paddle Class – This contains implementations of Paddle's length/height , its movement and frames.

MainActivity – It is where the MainActivity class communicates with the events and pass information to BounceThread. I created BounceThread here as an object as it is very important to make application successful. It has getter methods to pass references in BounceThread .

BounceThread – This is the final and most important method that combines all the action together. The thread will start and stop from the methods implemented. SurfaceHolder holds the canvas into place and makes our background. When game is playing , Boolean gamePlaying is always true .Paint and Canvas Object are used to draw .

Most important parts of frames are used here as they determine the correct amount of movement to ball and the paddle and pass them to update() methods.

sizeX and sizeY are used to hold the resolution together and they get passed into MainActivity Constructor. Text Score and Lives Left are used to track the user scores.

The update method will keep updating the positions and frames. Ball and Paddle collision is checked by intersection method. When the intersections return true, it means that they are indeed collided and ball bounces in reverse.

I also implemented clearX and clearY methods because when I tested the application , there were many instances where ball hit the paddle and never appeared again. These methods skips frame pixels animation so that this problem is avoided.

Finally, the canvas was locked into surface to draw and check if it is correct. If that is so , paint and background color are drawn and unlock Canvas is called at the end of drawing.

For some reason I couldn’t figure out, this application would not run in portrait mode often. It might have because of the miscalculations in background and co-ordinates. Therefore, I added forced landscape mode to AndroidManifest.xml file so that this game can only run in landscape mode.

## References

As I stated earlier, BouncingBalls animation was a major help for this project. Apart from that I took ideas from khanAcademy and the BreakOut game which is much more complex than this. I also watched some videos on Youtube that included general ball bouncing mechanics. For my problem that occurred when drawing paddle and aligning it properly as well as problem of ball disappearing into bat , some posts of stackOverFlow were of help.

https://www.khanacademy.org/computer-programming/paddle-ball/830543654

http://zetcode.com/tutorials/javagamestutorial/breakout/

## Experiences and Thoughts

Although I had plans to build very beautiful and complex game as stated in my project proposal, due to the lack of time and illness I had to create this basic game, but it was in no way easier than I thought. I had to struggle a lot and it took me 10-12 hours to build this. Meanwhile, I will keep improvising the game and add extra objects to the upper background that ball can hit and achieve extra scores. This project taught me how to make basic games and apps in android and I will always apply this knowledge to whatever programming steps I climb in future. Overall, It was great taking this class and I learned a lot about app development in android. I thank Professor Bo for this great effort on making us learn.