



BALL FIGHT

Design Specification

A

Daniel Roberto and Greg Loudon

Game

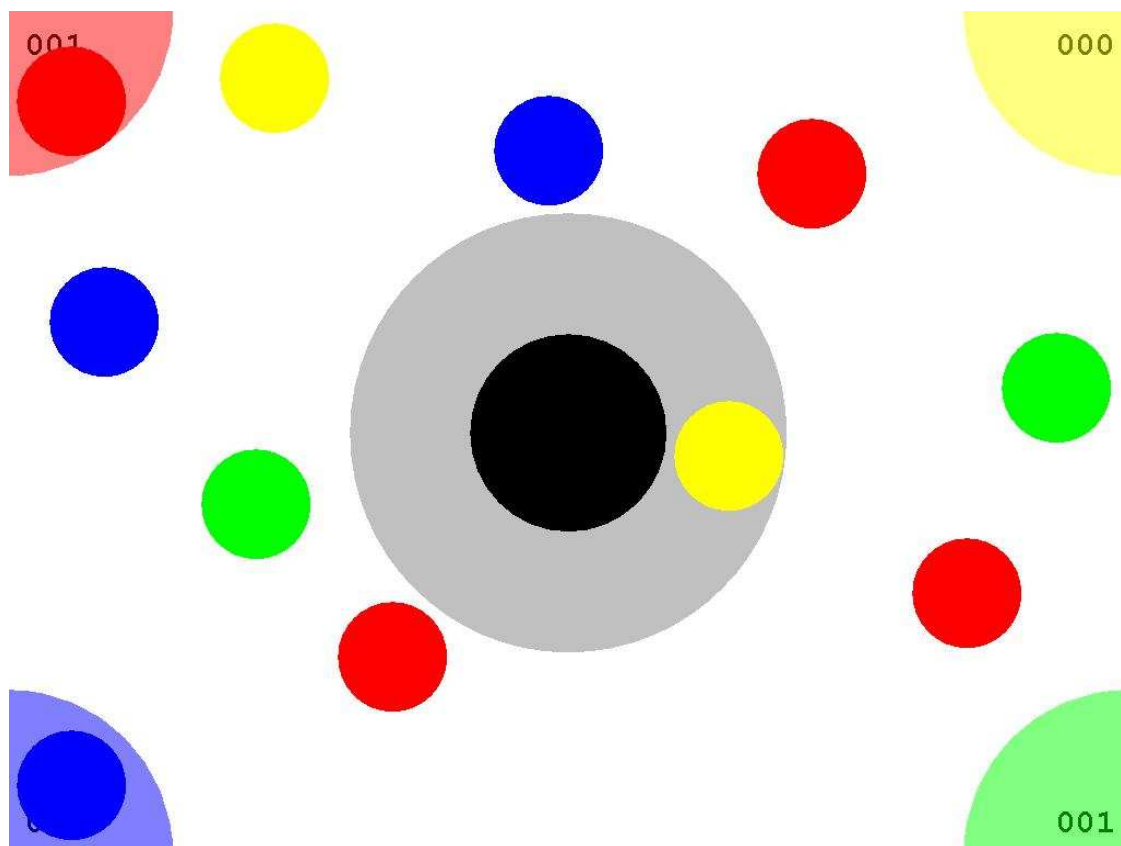
Table of Contents

Overview	3
Project Background	4
Console Application.....	4
X and Y Co-ordinate Application.....	6
Drawing Application	7
Ball Fight.....	8
How to Play Ball Fight	8
Creating a Ball using a Corner Base	9
Scoring Points in Ball Fight	9
Merging Balls in Ball Fight	10
Stealing Balls in Ball Fight	10
Detailed Game Description	11
Elliptical Geometric Primitives	12
Corner bases	12
Centre Pocket (Black hole Evolution)	12
Pocket Boundary	13
Ball to Ball / Boundary Collision Detection	13
Merging Game Play	14
Stealing Game Play.....	14
Reflection	15

Overview

Ball Fight is a game made for the research and development of touch screen technology and applications for games at the University of Technology Sydney (UTS) Games Studio. The objective of Ball Fight was to enable multiple players using one touch table to compete against each other to achieve the highest score through putting their own colour distinct balls into a centre pocket using touch. The key focus during development was to create an accessible, simple and most importantly fun game that would appeal to all ages due to its simplicity and intuitiveness.

The development of Ball Fight was commenced by Daniel Roperto on the 31st of July 2008 and he was assisted as of the 26th of August by Greg Loudon as a project at the UTS Games Studio, part of the multi-disciplinary Cognition and Creativity Studios (CCS). Ball Fight was developed using a Mitsubishi Electric Research Laboratory (MERL) Diamond Touch Table with the Java programming language 2.1 Software Development Kit (SDK). The game concept of the core centre pocket game play was designed by Associate Professor Yusuf Pisan, director of the UTS Games Studio and co-director of CCS. However as detailed in the document during the development and feature driven evolution, Ball Fight was enhanced for the purposes of accessibility, simplicity and last but not least fun.



UTS Games Studio "Ball Fight" Screen Capture

Project Background

As mentioned Associate Professor Yusuf Pisan, director of the UTS Games Studio and co-director of CCS designed the basic game concept of Ball Fight where competition was used as the key game play mechanism. The game concept conceived was that players using touch would drag their balls into a centre pocket to achieve a point. Ball Fight was developed using a Mitsubishi Electric Research Laboratory (MERL) Diamond Touch Table which acts as the input when the developed softwares image is projected on the touch table. The software was developed with the Java programming language based 2.1 Software Development Kit (SDK) of the Diamond Touch table. The development of the project went through three key iterations in which the hardware and SDK was explored and understood through the development of three smaller projects that would form as the basis of Ball Fight. The three stages of the development of Ball Fight were:

1. **Console Application**
2. **X and Y Co-ordinate Application**
3. **Drawing Application**
4. **Ball Fight**

Console Application

The first stage in the development of Ball Fight was the console application stage, where the hardware and SDK was explored and understood to provide a simple input and output system. The objective of the stage was to detect the amount of touches by each user on the Diamond Table through outputting the amount X and Y co-ordinate readings from each of the four users.

The project produces three simple output statements that allow the different users and differing amount of touches on the Diamond Touch to be processed and understood. The output line `EVT: touchDown USER: USER NUMBER Xs: NUMBER OF X CO-ORDINATES Ys: NUMBER OF Y CO-ORDINATES` shows that a user has touched the input device, `EVT: touchDuring USER: USER NUMBER Xs: NUMBER OF X CO-ORDINATES Ys: NUMBER OF Y CO-ORDINATES` shows that there has been sustained touch on the input device and `EVT: touchReleased USER: USER NUMBER Xs: NUMBER OF X CO-ORDINATES Ys: NUMBER OF Y CO-ORDINATES` is used to show that the user has removed input onto the touch table.

The following lines are a direct copy of output from using the Diamond Touch and the Console Application stage of Ball Fight. Further documentation has been added to allow the output to be easily understood and the actions performed to be more effectively illustrated.

Devices:

DT81M4-07-50-003

DiamondTouch: using device DT81M4-07-50-003

// The output statements below illustrate touch from a users (USER: 0) single finger or point of input that is being placed on the touch table and then removed after a period of sustained touch

EVT: touchDown USER: 0 Xs: 1 Ys: 1

EVT: touchDuring USER: 0 Xs: 1 Ys: 1

EVT: touchDuring USER: 0 Xs: 1 Ys: 1

EVT: touchDuring USER: 0 Xs: 1 Ys: 1

EVT: touchDuring USER: 0 Xs: 1 Ys: 1

EVT: touchReleased USER: 0 Xs: 0 Ys: 0

// The output statements below illustrate touch from a user (USER: 0) using multiple fingers or points of input that are being placed on the touch table and then removed one by one after a period of sustained touch

EVT: touchDown USER: 0 Xs: 2 Ys: 2

EVT: touchDuring USER: 0 Xs: 2 Ys: 2

EVT: touchDuring USER: 0 Xs: 2 Ys: 2

EVT: touchDuring USER: 0 Xs: 2 Ys: 2

EVT: touchDuring USER: 0 Xs: 1 Ys: 1

EVT: touchDuring USER: 0 Xs: 1 Ys: 1

EVT: touchReleased USER: 0 Xs: 0 Ys: 0

// The output statements below illustrate touch from a user (A different user in this case, USER: 1) using a single finger or point of input that is being placed on the touch table and then removed after a period of sustained touch

EVT: touchDown USER: 1 Xs: 1 Ys: 1

EVT: touchDuring USER: 1 Xs: 1 Ys: 1

EVT: touchDuring USER: 1 Xs: 1 Ys: 1

EVT: touchDuring USER: 1 Xs: 1 Ys: 1

EVT: touchReleased USER: 1 Xs: 1 Ys: 0

The ability to recognise the differing users and input are the foundation to Ball Fight, and were necessary for the following X and Y co-ordinate application to be developed where the input recognition from this stage of development could be located using Cartesian co-ordinates. Thus during this stage of the Ball Fight development, the dragging motion or location of input was not featured. The stage was largely to allow the understanding of the hardware constraints and demonstrate the identification of multiple users, and as a means to read in input using the examples given in the Diamond Touch SDK 2.1 examples.

X and Y Co-ordinate Application

The second stage of the development of Ball Fight was the X and Y Co-ordinate application that built on the Console application in which the input was able to located specifically using a display window and Cartesian co-ordinates. Whereas the Console application merely outputted the amount of X and Y co-ordinates this next stage of development allowed the input to be located and displayed through drawing a dot and X-Y axis that followed the users touch. Furthermore the users were visually differentiated using four different colours Red, Green, Blue and Yellow a contrast to the numbering system used in the previous stage.

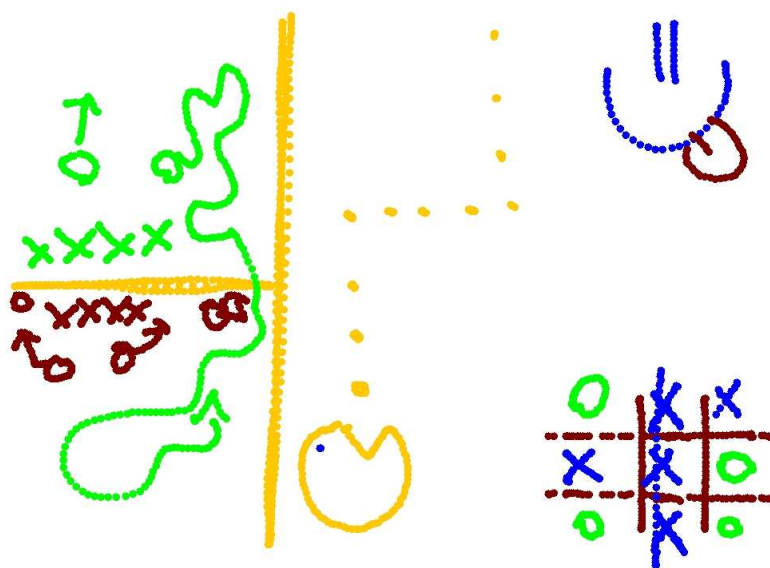
The X and Y Co-ordinate application built on the previous stage and implemented a display window, user colour differentiation and the location of the points of input. The listed features were used in the final version of Ball Fight for the creation of balls, dragging and as mentioned user colour differentiation and will be detailed in the section titled Detailed Game Description.



UTS Games Studio "X and Y Co-ordinate" Screen Capture

Drawing Application

The Drawing Application stage was a precursor to the dragging motion required for the movement of the player's ball in Ball Fight. The application was initially designed to allow sustained movement to be located, rather than just single point location as present in the X and Y application. The design was to draw a point for each time a touch was located on the display window. During the design stages the ramifications of this decision were unrecognised and the discovery of the drawing application was unexpected. However it was a good learning experience for the development team and as mentioned was key precursor to the dragging motion required for the movement of the player's ball in the final version of Ball Fight.



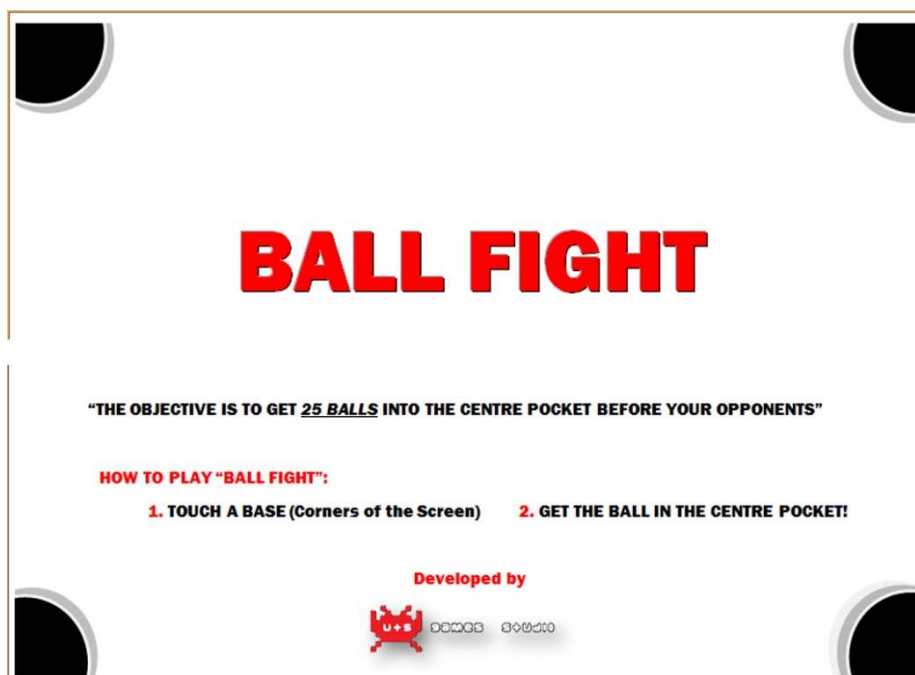
UTS Games Studio "Drawing Application" Screen Capture

Ball Fight

The final version of development was Ball Fight, which used each of the previous stages and combined them to create the game. The objective to enable multiple players using a touch table to compete against each other to achieve the highest score using touch was successfully achieved upon the foundation of the three other projects. The game however was not implemented straightaway, and a series of feature driven components were implemented during Ball Fight's development. The extra features of the ellipse geometric primitives, corner bases, pocket boundary, ball to ball / boundary collision detection, merging and stealing game play were also implemented in a similar iterative and evolutionary development process. The listed extra features will be explored fully in this document in the sections titled How to Play Ball Fight and the Detailed Game Description sections.

How to Play Ball Fight

Ball Fight has been developed from the ground up to be a simple, intuitive, accessible and most importantly fun game. As such Ball Fight is extremely easy to play and have fun with. The title screen at the start of the game details the overall objective of the game and immediately drops the player into the game, where the competition between players begins and the race to get 25 points into the centre pocket through scoring, merging and stealing begins.



UTS Games Studio "Ball Fight" Title Screen

Creating a Ball using a Corner Base

As displayed on the title screen, the player touches a base (a circle quadrant) on each of the edges of the rectangular touch table. Upon touching a corner base, an elliptical geometric primitive would appear that can be moved by simply maintaining sustained touch on the table. Whilst maintaining this sustained touch the player's ball can be moved by dragging their finger across the touch table.

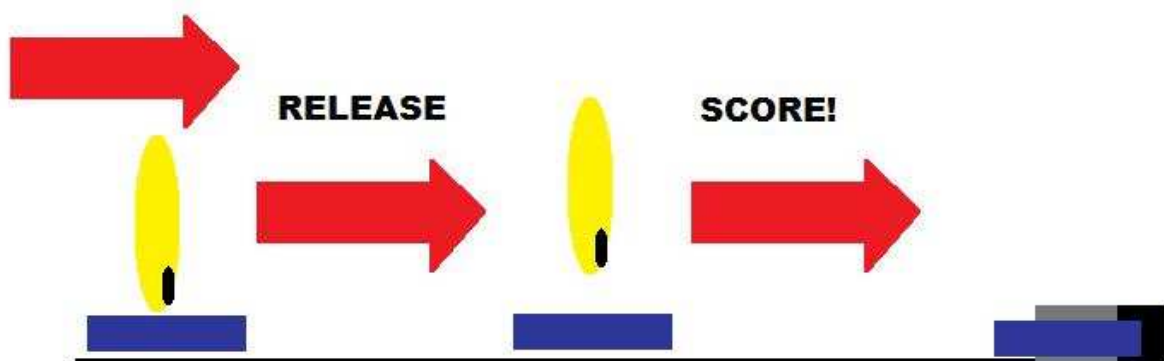
TOUCH A BASE AND DRAG A BALL OUT



Scoring Points in Ball Fight

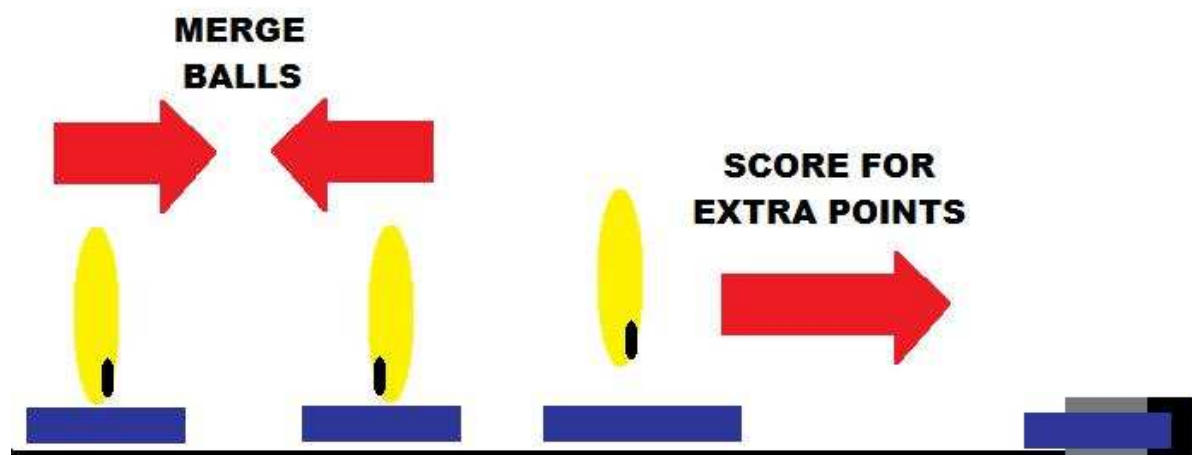
Scoring in Ball Fight is fairly easy. In order to score the player has to drag their created ball and get it inside the centre pocket to achieve a single point through releasing their ball and using momentum like in bowling. The centre pocket is very similar to a pocket in pool in which the player must aim and get the ball within the centre pocket by using enough power and accuracy. However the game is made more difficult through the "Pocket Boundary". The pocket boundary encloses the centre pocket making a region where the player loses control over their balls. Thus the centre pocket eliminates the players' ability to just drag a ball into the centre pocket in the game unobstructed.

DRAG



Merging Balls in Ball Fight

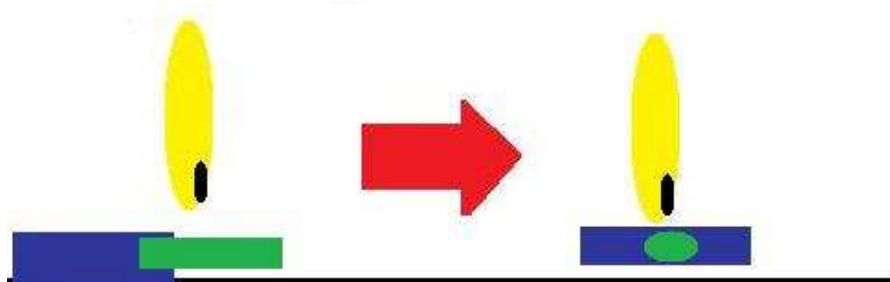
As well as having competition as a key game play aspect, the feature of merging balls was added to the features of Ball Fight. Merging is done through bringing two balls together through using two fingers. Once the balls are merged the balls will combine to create a bigger ball equal to a score value of the merged balls. Upon merging the player must still score to achieve the bonus points granted through merging. The ability to merge balls for bonus points adds to the competitive game play of Ball Fight and will be integral to a player if they wish to win.



Stealing Balls in Ball Fight

Stealing Balls is another game play feature of the game that will be integral to a player if they wish to win. Stealing also follows the simple, intuitiveness and simple framework of Ball Fight. The feature of stealing unlike other features of the game is purely decided on chance, in which it relies on a circumstance that is not common in ball fight. The option to steal is based solely on an opponent's ball landing on the player's base, which if touched allows the ball to be stolen for double points once pocketed. Thus stealing as a feature of Ball Fight differs substantially in that it is reliant on chance and some games can go on without any steals, and as such it is a bonus that should not be missed if the opportunity arises.

**IF AN OPPONENTS BALL LANDS ON
YOUR BASE QUICKLY TOUCH IT TO
STEAL FOR DOUBLE POINTS!**



Detailed Game Description

The final version of Ball Fight as a game made for the research and development of touch screen technology and applications for games at the University of Technology Sydney (UTS) Games Studio features numerous features. The features were all implemented in a feature driven evolutionary cycle, in which as each feature or idea was implemented another was initiated. The original game concept of the core centre pocket game play was designed by Associate Professor Yusuf Pisan was refined and enhanced to ensure the game was accessible, simple and most importantly fun that would appeal to all ages due to its simplicity and intuitiveness.

As mentioned development of Ball Fight was commenced by Daniel Roperto on the 31st of July 2008 and he was assisted as of the 26th of August by Greg Loudon as a project at the UTS Games Studio. Ball Fight was demonstrated at the UTS information day on the 30th of August 2008 following a series of prototypes during the feature driven cycle of the projects development. The game concept of the core centre pocket game play was designed by Associate Professor Yusuf Pisan, director of the UTS Games Studio and co-director of CCS. However during the development and feature driven evolution, Ball Fight was enhanced for the purposes of realism, simplicity and last but not least fun.

The game however was not implemented straightaway, and a series of feature driven components implemented during Ball Fight's development. The key features of Ball Fight, as an accumulation of all of the other previous projects and versions deployed for the UTS Information Day demonstration were:

- **Elliptical Geometric Primitives**
- **Corner bases**
- **Centre Pocket (Black hole Evolution)**
- **Pocket Boundary**
- **Ball to Ball / Boundary Collision Detection**
- **Merging Game Play**
- **Stealing Game Play**

Elliptical Geometric Primitives

The implementation of the elliptical geometric primitives as the balls of Ball Fight was easily implemented, due to the inbuilt ellipse primitive function in the programming language Java. The previous X and Y co-ordinate project Cartesian co-ordinate location source code was enhanced to draw solely an ellipse rather than a dot and the corresponding X and Y axis's. The way in which the ellipse geometric primitives of Ball Fight were drawn was through the use of an X and Y co-ordinate, and the hard coded values of the width and height of the circle.



Ellipse Geometric Java Built in Function Attribute Diagram

Corner bases

The corner bases of Ball Fight were a key component of the initial game design by Yusuf Pisan. Similarly to the ellipse geometric primitive, the corner bases used the built in Java function and was easily implemented. Four ellipse geometric primitives were drawn on the display window in each corner and were given a special property in code that allows balls to be created within these unique elliptical corner bases.

As mentioned the title screen would display the directions that a player would touch a base (a circle quadrant) on each of the edges of the rectangular touch table. Upon touching the corner base, an elliptical geometric primitive would be created and appear that could be moved by simply maintaining sustained touch on the table.

Centre Pocket (Black hole Evolution)

The centre pocket game play of Ball Fight was also defined in the initial game design by Yusuf Pisan; however it was described as a "Black hole", which would pull the balls in to the centre of the touch table. The gravitational aspects of this black hole game play was implemented, however the ease at which players drag a ball to the centre of the touch table, along with the corresponding pulling of the black hole the game became far too easy to score. The easiness of the game compromised

competitiveness of the players and the fun of the players had. As such it was decided the concept of the black hole would be removed and a pocket similar to pool table pocket would replace it. In order to score the player would have to drag their created ball and get it inside the centre pocket to achieve a single point through releasing their ball and using momentum like in bowling.

Upon completing the centre pocket feature Ball Fight, the original initial design of the game was completed. However upon playing the game numerous fixes and errors both in the software and the game itself were realised, addressed and enhanced to make Ball Fight even more simple, accessible, intuitive and of course fun.

Pocket Boundary

The implementation of the centre pocket and a range of testing and quality assurance caused the feature driven development of Ball Fight to continue. The centre pocket of the game was designed to be very similar to a pocket in pool or a hole in golf, in which the player must aim and get the ball within the centre pocket by using enough power and accuracy. However without the implementation of the “Pocket Boundary” feature, the player could initially easily drag the ball into the centre pocket. The ease, at which this is done, eliminates the completion, and makes takes the accomplishment scoring a vital aspect of all games.

Upon this the game was made more difficult through the Pocket boundary. The pocket boundary encloses the centre pocket making a region where the player loses control over their balls. As well as the player losing control of their ball in the centre pocket, the ball also rapidly loses momentum and the reward of scoring is returned to Ball Fight. Thus the centre pocket eliminates the players’ ability to just drag a ball into the centre pocket in the game unobstructed and brings accomplishment and reward into Ball Fight.

Ball to Ball / Boundary Collision Detection

The testing of Ball Fight for the UTS Information day also brought to light, another flaw of the game, that if fixed could expand the games accessibility, simplicity and fun. The error noticed was that if a ball collided with an opponent’s ball, both balls would stop and a visible square barrier between them could be seen. The error was immersion breaking, in which in testing ruined the experience of the game due to the lack thereof realism through the balls inability to bounce off each other.

Furthermore, the game also did not feature any boundaries, so if a ball went off screen, it would immediately be deleted from the game.

The first fix was to create a boundary for the game equal to the size of the Diamond Touch table. Upon creating the boundary the same immersion breaking error of the ball stopping and a visible square barrier between the wall and ball could be seen. The next step was to fix the error and also bring in the feature of balls bouncing into the chaos that is Ball Fight. The solution was to bring momentum into the game.

Momentum was calculated through using the equation of $Speed = Distance / Time$, in which the speed of the ball was tracked through recording the starting and ending position and time taken to get between them. The speed of the collision was then inversed and decreased so the balls would visually appear to bounce on screen. Following this, the barriers between circles was decreased significantly to the point of minor overlapping of objects to fix the visible barrier to both balls to ball / boundary. The fixes made Ball Flight a lot more fun and really added to the competitiveness through the chaos of balls bouncing everywhere. The realistic addition of bouncing and collision detection also enhanced the intuitiveness and simplicity of the game bringing increased accessibility through the user's anticipation and acceptance of bouncing as a component of game play.

Merging Game Play

The ability to merge balls was a further example of the feature driven development of Ball Fight and an overall enhancing game play mechanism. As mentioned as well as having competition as a key game play aspect, the feature of merging balls was added to the features of Ball Fight. Merging is done through bringing two balls together through using two fingers. Once the balls are merged the balls will combine to create a bigger ball equal to a score value of the merged balls. The visible feedback of having the balls combine to create a bigger ball also corresponds to the centre pocket and pocket boundary. Thus the game becomes harder if the player does not merge, through the increased size of the pocket boundaries zone of no control. The expanded pocket boundary also influences the players to use multiple balls to hit their own balls into the centre pocket.

Upon merging the player must still score to achieve the bonus points granted through merging. The ability to merge balls for bonus points adds to the competitive nature of the game play and opens up a totally new way of playing through the influential means of game play escalation.

Stealing Game Play

The stealing game play is the chance component of Ball Fight, and can turn the tide of a game upon a successful steal and score for double points. Stealing follows the

simplicity, intuitiveness of Ball Fight, but it relies on a circumstance that is not common in ball fight. The option to steal is based solely on an opponent's ball landing on the player's base, which if touched allows the ball to be stolen for double points once pocketed. Thus stealing as a feature of Ball Fight differs substantially in that it is reliant on chance and some games can go on without any steals.

Reflection

Ball Fight as a game made for the research and development of touch screen technology and applications for games at the University of Technology Sydney (UTS) Games Studio was extremely successful. The objective of Ball Fight being able to enable multiple players using one touch table to compete against each other to achieve the highest score through putting their own colour distinct balls into a centre pocket using touch was completed and enhanced significantly. Also the key focus during development "*to create an accessible, simple and most importantly fun game that would appeal to all ages due to its simplicity and intuitiveness*" was met and illustrated at the UTS Information day upon an overwhelming positive response.

The team is very happy with Ball Fight, in which the game produced met all of the standards needed and time also allowed testing, quality assurance and enhancement to the game before the deadline of the UTS Information day. The team by the end had successfully understood the constraints, limitations and possibilities of the MERL Diamond Touch Table and also discovered the importance of quality insurance to the development of a game. Nonetheless the development of Ball Fight was not perfect and a few enhancements were left unimplemented due to the time constraints of the limited deadline for full development, testing and implementation.

A key enhancement that was left unimplemented was full gesture recognition. The development team aimed to have the game recognise different gestures rather just solely touch as seen in the final version of Ball Fight. However the team aims to in the future to research and develop gesture recognition into a different touch table project that will further enhance touch screen technology and applications for games overall.

The development of Ball Fight was commenced by Daniel Roperto on the 31st of July 2008 and he was assisted as of the 26th of August 2008 by Greg Loudon. Ball Fight was a project at the UTS Games Studio, part of the multi-disciplinary Creativity and Cognition Studios (CCS). The game concept of the core centre pocket game play was designed by Associate Professor Yusuf Pisan, director of the UTS Games Studio and co-director of CCS.