Trow, Cameron

ASCII GAME

DESIGN, DEVELOPMENT AND TEST APPLICATION TO SPECIFICATION

ASSessment1/1 Design, Development AND TEST Application TO Specification

Table of Contents

[Design 2](#_Toc500852823)

[Noun/Noun Phrases 2](#_Toc500852824)

[UML Diagram 3](#_Toc500852825)

[Testing 3](#_Toc500852826)

[Menu 3](#_Toc500852827)

[Game 3](#_Toc500852828)

[End of Game 3](#_Toc500852829)

[Sound 3](#_Toc500852830)

[Color 4](#_Toc500852831)

[Paddle change size 4](#_Toc500852832)

[demo 4](#_Toc500852833)

[health 4](#_Toc500852834)

[score 4](#_Toc500852835)

[controls 4](#_Toc500852836)

[pause Menu 4](#_Toc500852837)

[Exit game esc shortcut 4](#_Toc500852838)

[Multiple control schemes 4](#_Toc500852839)

[Additional Features 5](#_Toc500852840)

[Sound 5](#_Toc500852841)

[Change size 5](#_Toc500852842)

[Color 6](#_Toc500852843)

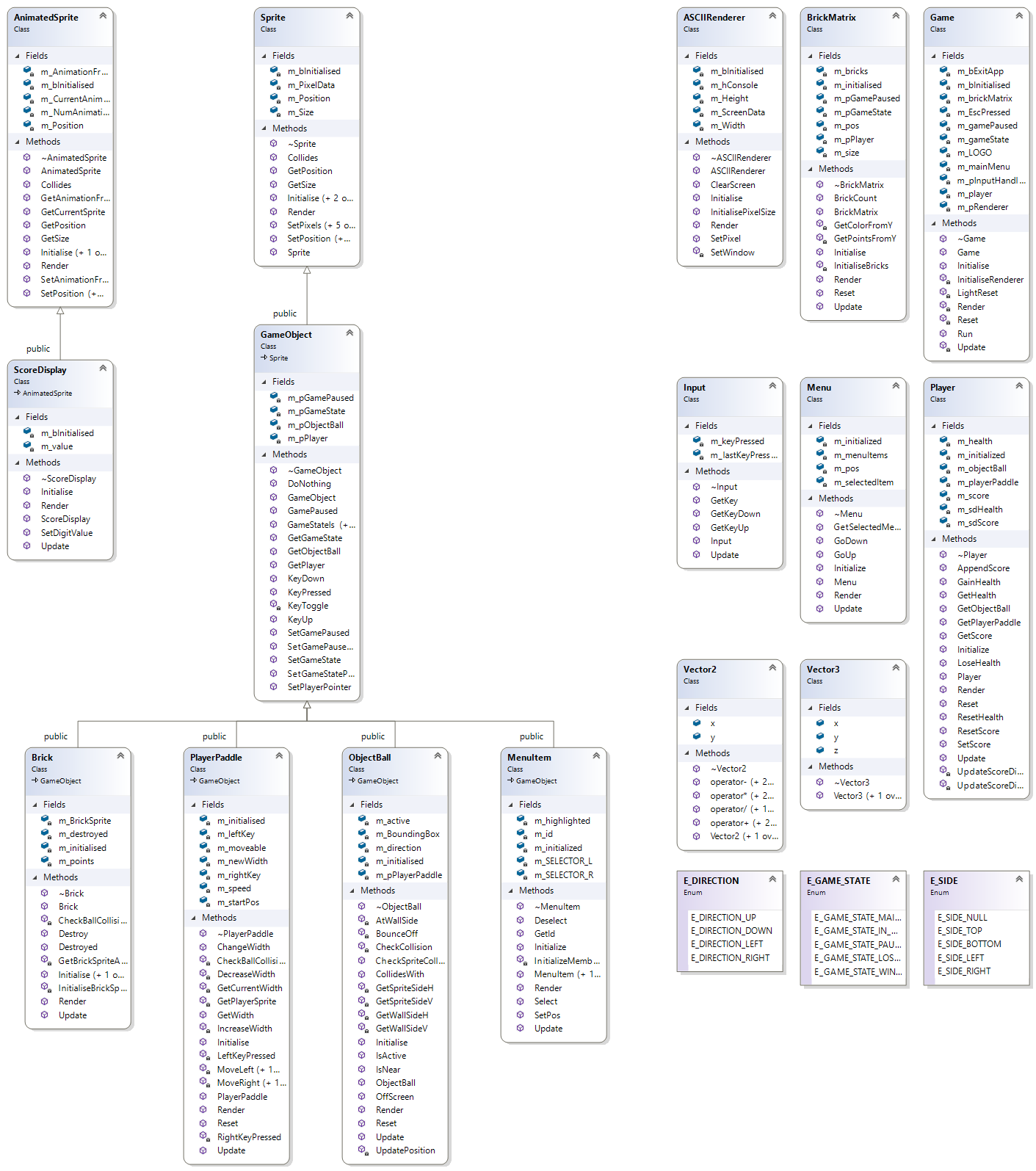
[Demo mode 8](#_Toc500852844)

[Pause menu 8](#_Toc500852845)

# Design

|  |  |  |
| --- | --- | --- |
| Noun/Noun Phrases | Strong enough | why |
| Breakout | n/a | System Scope |
| Ball | Yes | Complex Object |
| Score | No | Just a number |
| Brick | Yes | Complex Object + Many Required |
| paddle | Yes | Complex object |
| Wall | No | Just defined by 0,and screen size |
|  |  |  |
|  |  |  |
|  |  |  |

## UML Diagram



# Testing

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test numb | Location / object | Input | Purpose | Expected output | Comment / verified | Evidence location | |
| 1 | Menu | W,s | Change selected item | ‘[…]’ move | yes |  |
| 2 | End of Game |  | Changes state to E\_GAME\_STATE\_LOSE\_GAME | BOO appears on the screen | yes |  |
| 3 | Sound |  | Play sound | Sound is played when ball hits wall | yes | n/a |
| 4 | Color |  | Bricks are evenly distributed a color to be | Colorful bricks | yes | Most screenshots |
| 5 | Paddle change size |  | Paddle shrinks on health loss | Paddle changes size on death | yes |  |
| 6 | Demo |  | Changes game to demo mode | Wait a long time and demo shows | yes |  |
| 7 | Health |  | Heath goes down on ball fall out of game | Health goes gown | yes |  |
| 8 | Score |  | Detects for ball hitting a brick | Score goes up by amount specified by color of brick | yes |  |
| 9 | Controls | A, D | Moves the paddle | The paddle moves | yes |  |
| 10 | Pause Menu | ESC (in Game) | Changes state to pause game | The word pause appears on the screen | yes |  |
| 11 | Exit game esc shortcut | Double press escape on main menu | Closes game | Closes game | yes |  |
| 12 | Multiple control schemes | A, D or left, right  main menu only: enter or space | Detects for either key | Depends on the key | yes |  |
| 13 |  |  |  |  |  |  |
| 14 |  |  |  |  |  |  |
| 15 |  |  |  |  |  |  |
| 16 |  |  |  |  |  |  |
| 17 |  |  |  |  |  |  |
| 18 |  |  |  |  |  |  |

# Additional Features

## Sound

Makes sound when ball hits wall

Easter egg: if the ball bounces off these sides in odder you will hear the 4 main notes of lavender town: bottom, right, top, left

In Object Ball::Bounce Off

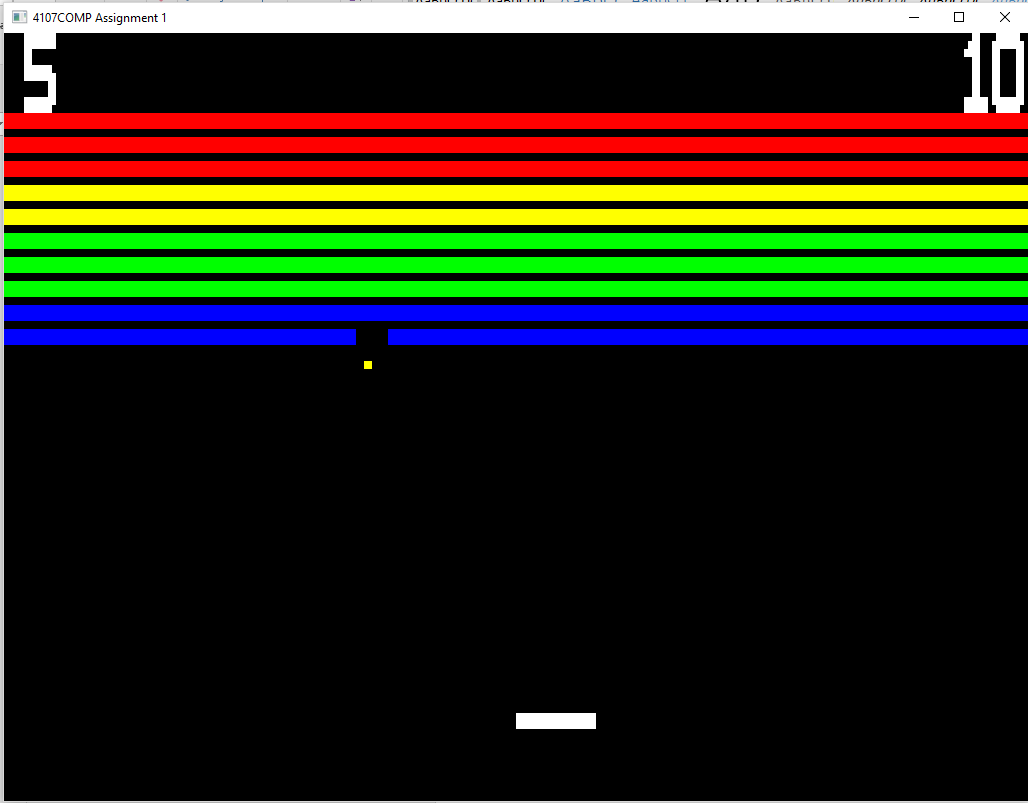
Lines 202,208,215,222

## Change size

The paddle can change size though out the game depending on the health

Location: PlayerPaddle.cpp (169-187), (156)

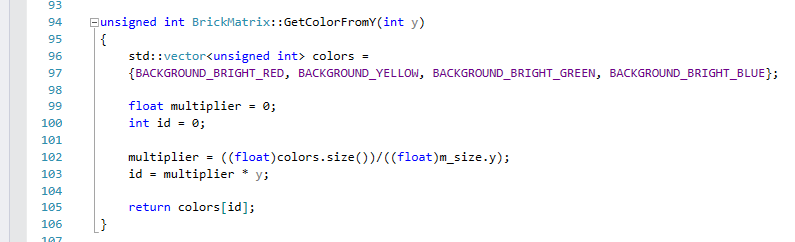
## Color



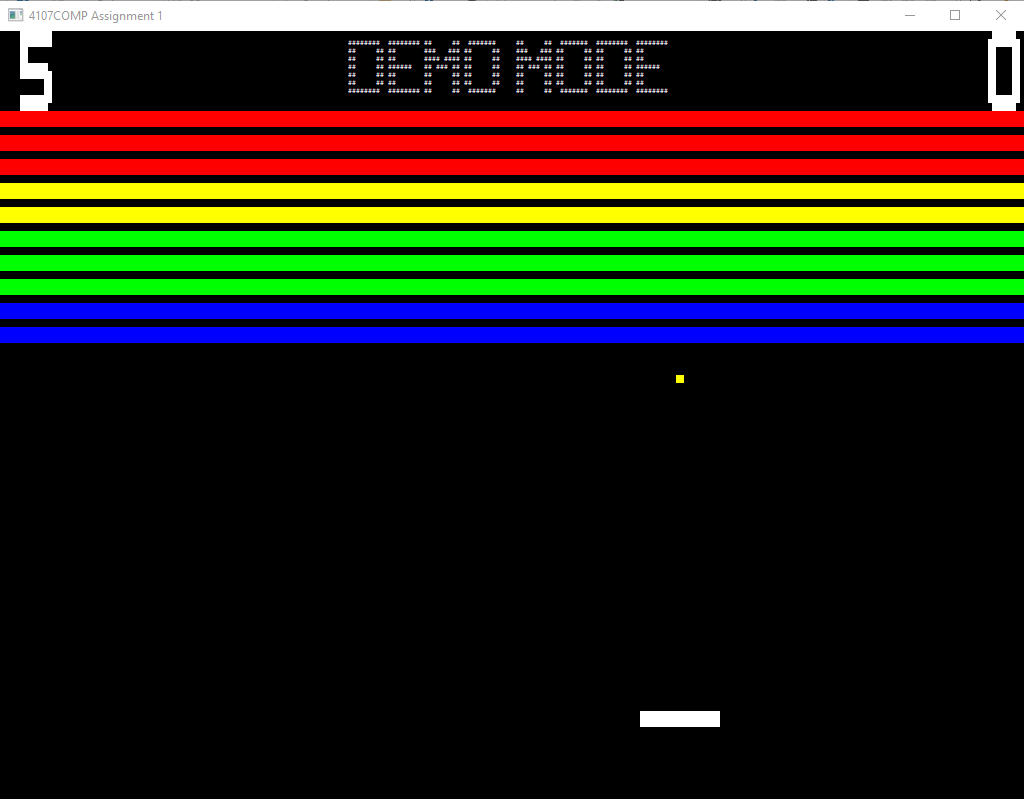
The bricks are different colors depending on point value

|  |  |  |  |
| --- | --- | --- | --- |
| Brick | Color | Color name | Points |
|  |  | Blue | 10 |
|  |  | Green | 20 |
|  |  | Yellow | 70 |
|  |  | Red | 100 |

Brick Matrix.cpp



## Demo mode



This was just used for testing but now appears when you wait on the main menu

Locations:

* PlayerPaddle.cpp:17, 54-61, 85-92,145-148
* ObjectBall.cpp:56-62
* Game.cpp:14, 31,142-153,244-249,268-271,281-288,293-302,412,421-424

## Pause menu

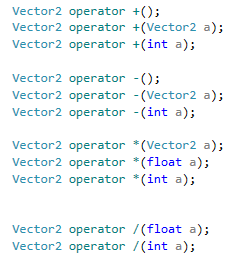
Added a pause menu for when you press escape, double pressing escape will bring you to the main menu

Location:

* Game.cpp:115-140,306,318,323,349-366,427-431

# Additions to Provided Framework

## Vector2::operator overloads

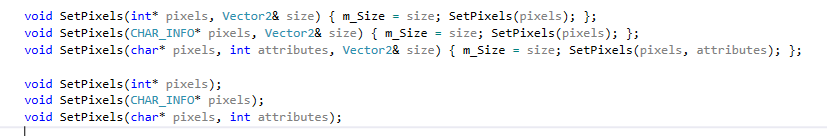


These where used to make things easier instead on manually expanding the vector and changing values.

By just adding two vectors the x and y are individually added together for you.

I use it to center things in relation to things and position the bricks and move the ball.

## Sprite::SetPixels



This was added to enable the dynamic changing of sprites. This is mainly used in player paddle to help with a changing sizes depending upon the health