

Headsoccer

Generated by Doxygen 1.9.4

1 File Index	1
1.1 File List	1
2 File Documentation	3
2.1 src/game.cpp File Reference	3
2.1.1 Typedef Documentation	4
2.1.1.1 Collision	4
2.1.2 Enumeration Type Documentation	4
2.1.2.1 Direction	4
2.1.3 Function Documentation	4
2.1.3.1 CheckCollision() [1/2]	4
2.1.3.2 CheckCollision() [2/2]	4
2.1.3.3 INITIAL_BALL_VELOCITY()	5
2.1.3.4 PLAYER_SIZE()	5
2.1.3.5 PLAYER_VELOCITY()	5
2.1.3.6 VectorDirection()	5
2.1.4 Variable Documentation	5
2.1.4.1 Ball	5
2.1.4.2 BALL_RADIUS	5
2.1.4.3 Player1	6
2.1.4.4 Player2	6
2.1.4.5 Renderer	6
2.2 src/game_object.cpp File Reference	6
2.3 src/main.cpp File Reference	6
2.3.1 Function Documentation	6
2.3.1.1 framebuffer_size_callback()	7
2.3.1.2 Headsoccer()	7
2.3.1.3 key_callback()	7
2.3.1.4 main()	7
2.3.2 Variable Documentation	7
2.3.2.1 SCREEN_HEIGHT	7
2.3.2.2 SCREEN_WIDTH	7
2.4 src/resource_manager.cpp File Reference	8
2.5 src/shader.cpp File Reference	8
2.6 src/sprite_renderer.cpp File Reference	8
2.7 src/stb_image.cpp File Reference	8
2.7.1 Macro Definition Documentation	8
2.7.1.1 STB_IMAGE_IMPLEMENTATION	8
2.8 src/texture.cpp File Reference	8
Index	9

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

src/ game.cpp	3
src/ game_object.cpp	6
src/ main.cpp	6
src/ resource_manager.cpp	8
src/ shader.cpp	8
src/ sprite_renderer.cpp	8
src/ stb_image.cpp	8
src/ texture.cpp	8

Chapter 2

File Documentation

2.1 src/game.cpp File Reference

```
#include "game.h"  
#include "resource_manager.h"  
#include "sprite_renderer.h"  
#include "game_object.h"
```

Typedefs

- typedef std::tuple< bool, [Direction](#), glm::vec2 > [Collision](#)

Enumerations

- enum [Direction](#) { [UP](#) , [RIGHT](#) , [DOWN](#) , [LEFT](#) }

Functions

- const glm::vec2 [INITIAL_BALL_VELOCITY](#) (100.0f, -350.0f)
- const float [PLAYER_VELOCITY](#) (500.0f)
- const glm::vec2 [PLAYER_SIZE](#) (100.0f, 200.0f)
- bool [CheckCollision](#) (GameObject &one, GameObject &two)
- [Collision](#) [CheckCollision](#) (BallObject &one, GameObject &two)
- [Direction](#) [VectorDirection](#) (glm::vec2 closest)

Variables

- SpriteRenderer * [Renderer](#)
- const float [BALL_RADIUS](#) = 25.0f
- BallObject * [Ball](#)
- GameObject * [Player1](#)
- GameObject * [Player2](#)

2.1.1 Typedef Documentation

2.1.1.1 Collision

```
typedef std::tuple<bool, Direction, glm::vec2> Collision
```

2.1.2 Enumeration Type Documentation

2.1.2.1 Direction

```
enum Direction
```

Enumerator

UP	
RIGHT	
DOWN	
LEFT	

2.1.3 Function Documentation

2.1.3.1 CheckCollision() [1/2]

```
Collision CheckCollision (  
    BallObject & one,  
    GameObject & two )
```

2.1.3.2 CheckCollision() [2/2]

```
bool CheckCollision (  
    GameObject & one,  
    GameObject & two )
```


2.1.3.3 INITIAL_BALL_VELOCITY()

```
const glm::vec2 INITIAL_BALL_VELOCITY (
    100.  0f,
    -350. 0f )
```

2.1.3.4 PLAYER_SIZE()

```
const glm::vec2 PLAYER_SIZE (
    100.  0f,
    200. 0f )
```

2.1.3.5 PLAYER_VELOCITY()

```
const float PLAYER_VELOCITY (
    500. 0f )
```

2.1.3.6 VectorDirection()

```
Direction VectorDirection (
    glm::vec2 closest )
```

2.1.4 Variable Documentation

2.1.4.1 Ball

```
BallObject* Ball
```

2.1.4.2 BALL_RADIUS

```
const float BALL_RADIUS = 25.0f
```

2.1.4.3 Player1

```
GameObject* Player1
```

2.1.4.4 Player2

```
GameObject* Player2
```

2.1.4.5 Renderer

```
SpriteRenderer* Renderer
```

2.2 src/game_object.cpp File Reference

```
#include "game_object.h"
```

2.3 src/main.cpp File Reference

```
#include <glad/glad.h>
#include <GLFW/glfw3.h>
#include "game.h"
#include "resource_manager.h"
#include <iostream>
```

Functions

- void [framebuffer_size_callback](#) (GLFWwindow *window, int width, int height)
- void [key_callback](#) (GLFWwindow *window, int key, int scancode, int action, int mode)
- Game [Headsoccer](#) ([SCREEN_WIDTH](#), [SCREEN_HEIGHT](#))
- int [main](#) (int argc, char *argv[])

Variables

- const unsigned int [SCREEN_WIDTH](#) = 800
- const unsigned int [SCREEN_HEIGHT](#) = 600

2.3.1 Function Documentation

2.3.1.1 framebuffer_size_callback()

```
void framebuffer_size_callback (
    GLFWwindow * window,
    int width,
    int height )
```

2.3.1.2 Headsoccer()

```
Game Headsoccer (
    SCREEN_WIDTH ,
    SCREEN_HEIGHT )
```

2.3.1.3 key_callback()

```
void key_callback (
    GLFWwindow * window,
    int key,
    int scancode,
    int action,
    int mode )
```

2.3.1.4 main()

```
int main (
    int argc,
    char * argv[] )
```

2.3.2 Variable Documentation

2.3.2.1 SCREEN_HEIGHT

```
const unsigned int SCREEN_HEIGHT = 600
```

2.3.2.2 SCREEN_WIDTH

```
const unsigned int SCREEN_WIDTH = 800
```

2.4 src/resource_manager.cpp File Reference

```
#include "resource_manager.h"
#include <iostream>
#include <sstream>
#include <fstream>
#include "stb_image.h"
```

2.5 src/shader.cpp File Reference

```
#include "shader.h"
#include <iostream>
```

2.6 src/sprite_renderer.cpp File Reference

```
#include "sprite_renderer.h"
```

2.7 src/stb_image.cpp File Reference

```
#include "stb_image.h"
```

Macros

- #define [STB_IMAGE_IMPLEMENTATION](#)

2.7.1 Macro Definition Documentation

2.7.1.1 STB_IMAGE_IMPLEMENTATION

```
#define STB_IMAGE_IMPLEMENTATION
```

2.8 src/texture.cpp File Reference

```
#include <iostream>
#include "texture.h"
```

Index

Ball
 game.cpp, 5
BALL_RADIUS
 game.cpp, 5

CheckCollision
 game.cpp, 4
Collision
 game.cpp, 4

Direction
 game.cpp, 4
DOWN
 game.cpp, 4

framebuffer_size_callback
 main.cpp, 6

game.cpp
 Ball, 5
 BALL_RADIUS, 5
 CheckCollision, 4
 Collision, 4
 Direction, 4
 DOWN, 4
 INITIAL_BALL_VELOCITY, 4
 LEFT, 4
 Player1, 5
 Player2, 6
 PLAYER_SIZE, 5
 PLAYER_VELOCITY, 5
 Renderer, 6
 RIGHT, 4
 UP, 4
 VectorDirection, 5

Headsoccer
 main.cpp, 7

INITIAL_BALL_VELOCITY
 game.cpp, 4

key_callback
 main.cpp, 7

LEFT
 game.cpp, 4

main
 main.cpp, 7
main.cpp
 framebuffer_size_callback, 6
 Headsoccer, 7
 key_callback, 7
 main, 7
 SCREEN_HEIGHT, 7
 SCREEN_WIDTH, 7

Player1
 game.cpp, 5
Player2
 game.cpp, 6
PLAYER_SIZE
 game.cpp, 5
PLAYER_VELOCITY
 game.cpp, 5

Renderer
 game.cpp, 6
RIGHT
 game.cpp, 4

SCREEN_HEIGHT
 main.cpp, 7
SCREEN_WIDTH
 main.cpp, 7
src/game.cpp, 3
src/game_object.cpp, 6
src/main.cpp, 6
src/resource_manager.cpp, 8
src/shader.cpp, 8
src/sprite_renderer.cpp, 8
src/stb_image.cpp, 8
src/texture.cpp, 8
stb_image.cpp
 STB_IMAGE_IMPLEMENTATION, 8
STB_IMAGE_IMPLEMENTATION
 stb_image.cpp, 8

UP
 game.cpp, 4

VectorDirection
 game.cpp, 5