PS0 Hello World with SFML

Assignment Summary:

The objective of this assignment is to get set up and running with the build environment. Make sure GCC and Make works on the machine that is being used. After a successful install use the SFML library to draw and image to the screen and control it using keystrokes or make a function for it to move.

What I accomplished :

Created a moving sprite of a smiley face, if none of the directions buttons are being pressed it will move to the lower right corner of the screen. If a directional button is pressed the image will go in the direction the button is being pressed.

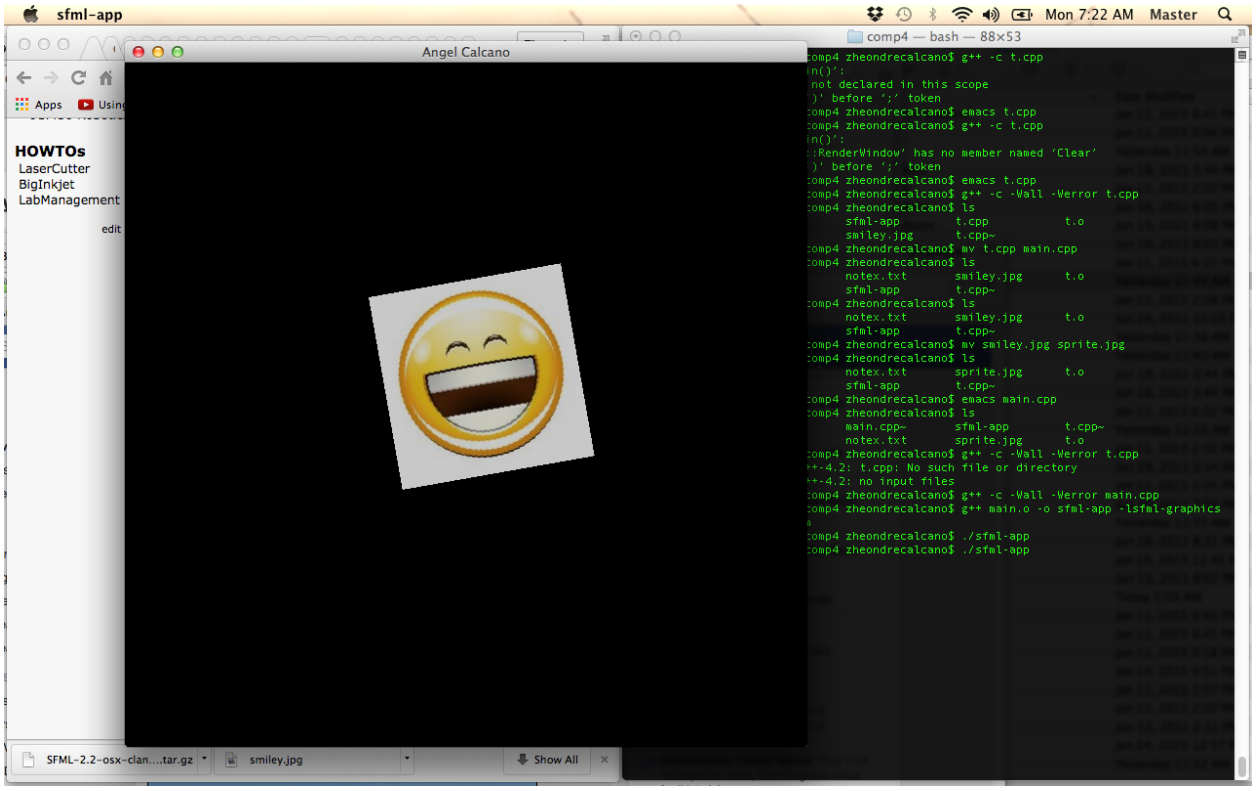
Design :

Used a lot of objects from the SFML library to enable manipulation of the image.

What I learned:

How to print images to the screen, how to control them, and how to implement the SFML library in c++.

Output:



```
all: ps0

ps0: main.o
	g++ main.o -o ps0 -lsfml-graphics -lsfml-window -lsfml-system

main.o: main.cpp
	g++ -c main.cpp -Wall -Werror -ansi -pedantic

clean:
	rm *.o ps0 *~ *#
```

```
 1: // Angel Zheondre Calcano ps0 Hello World with SFML
 2: #include <SFML/Graphics.hpp>
 3:
 4: int main(int argc, char *argv[]) {
 5:   sf::RenderWindow window(sf::VideoMode(700, 700), "Angel Calcano");
 6:   sf::Texture texture;
 7:   texture.loadFromFile("sprite.png");
 8:   sf::Sprite sprite;
 9:   sprite.setTexture(texture);
10:   sprite.setColor(sf::Color(255, 255, 255, 200));
11:   sprite.setPosition(100, 25);
12:   while (window.isOpen()) {
13:     if(sf::Mouse::isButtonPressed(sf::Mouse::Left))
14:       sprite.rotate(10);
15:     if(sf::Keyboard::isKeyPressed(sf::Keyboard::Left))
16:       sprite.move(-5,0);
17:     if(sf::Keyboard::isKeyPressed(sf::Keyboard::Up))
18:       sprite.move(0,-5);
19:     if(sf::Keyboard::isKeyPressed(sf::Keyboard::Down))
20:       sprite.move(0,5);
21:     if(sf::Keyboard::isKeyPressed(sf::Keyboard::Right))
22:       sprite.move(5,0);
23:     else
24:       sprite.move(0.5,0.5);
25:     sf::Event event;
26:     while (window.pollEvent(event)) {
27:       if (event.type == sf::Event::Closed)
28:         window.close();
29:     }
30:     window.clear();
31:     window.draw(sprite);
32:     /* window.draw(shape); */
33:     window.display();
34:   }
35:   return 0;
36: }
```