

```
1: //////////////////////////////////////
2: //
3: // Copyright (C) 2014 Maximilian Wagenbach (aka. Foaly) (foaly.f@web.de)
4: //
5: // This software is provided 'as-is', without any express or implied warr
anty.
6: // In no event will the authors be held liable for any damages
7: // arising from the use of this software.
8: //
9: // Permission is granted to anyone to use this software for any purpose,
10: // including commercial applications, and to alter it and redistribute it
freely,
11: // subject to the following restrictions:
12: //
13: // 1. The origin of this software must not be misrepresented;
14: // you must not claim that you wrote the original software.
15: // If you use this software in a product, an acknowledgment
16: // in the product documentation would be appreciated but is not required.
17: //
18: // 2. Altered source versions must be plainly marked as such,
19: // and must not be misrepresented as being the original software.
20: //
21: // 3. This notice may not be removed or altered from any source distribut
ion.
22: //
23: //////////////////////////////////////
24:
25: #ifndef ANIMATEDSPRITE_INCLUDE
26: #define ANIMATEDSPRITE_INCLUDE
27:
28: #include <SFML/Graphics/RenderTarget.hpp>
29: #include <SFML/System/Time.hpp>
30: #include <SFML/Graphics/Drawable.hpp>
31: #include <SFML/Graphics/Transformable.hpp>
32: #include <SFML/System/Vector2.hpp>
33:
34: #include "Animation.hpp"
35:
36: class AnimatedSprite : public sf::Drawable, public sf::Transformable {
37: public:
38:     explicit AnimatedSprite(sf::Time frameTime = sf::seconds(0.2f),
39:                             bool paused = false, bool looped = true);
40:
41:     void update(sf::Time deltaTime);
42:     void setAnimation(const Animation& animation);
43:     void setFrameTime(sf::Time time);
44:     void play();
45:     void play(const Animation& animation);
46:     void pause();
47:     void stop();
48:     void setLooped(bool looped);
49:     void setColor(const sf::Color& color);
50:     const Animation* getAnimation() const;
51:     sf::FloatRect getLocalBounds() const;
52:     sf::FloatRect getGlobalBounds() const;
53:     bool isLooped() const;
54:     bool isPlaying() const;
55:     sf::Time getFrameTime() const;
56:     void setFrame(std::size_t newFrame, bool resetTime = true);
57:
58: private:
59:     const Animation* m_animation;
60:     sf::Time m_frameTime;
61:     sf::Time m_currentTime;
62:     std::size_t m_currentFrame;
```

```
63:     bool m_isPaused;
64:     bool m_isLooped;
65:     const sf::Texture* m_texture;
66:     sf::Vertex m_vertices[4];
67:
68:     virtual void draw(sf::RenderTarget& target, sf::RenderStates states)
const;
69: };
70:
71: #endif
72: // ANIMATEDSPRITE_INCLUDE
```