```
Scene
# GameEngine * m_game
# EntityManager m_entityManager
# ActionMap m actionMap
# size t m currentFrame
# bool m_paused
# bool m hasEnded
+ void setPaused(bool)
+ Scene()
+ Scene(GameEngine *gameEngine)
+ virtual void update()=0
+ virtual void sDoAction
  (const Action &action)=0
+ virtual void sRender()=0
+ virtual void doAction
  (const Action &action)
+ void registerAction
  (int inputKey, const
  std::string &actionName)
+ size t width() const
+ size_t height() const
+ size t currentFrame
  () const
+ bool hasEnded() const
+ const ActionMap & getAction
  Map() const
# virtual void onEnd()=0
           Scene Menu
   # std::string m_title
   # sf::Text m titleText
   # std::vector< sf::Text
      > m texts
   # std::vector< std::string
      > m_menuStrings
   # std::vector< std::string
      > m levelPaths
   # sf::Text m menuText
   # int m selectedMenuIndex
   + Scene Menu(GameEngine
       gameEngine=nullptr)
```

+ void sRender()

void sDoAction(const
 Action &action)

void init()
void update()
void onEnd()