

GameEngine

+ GameEngine(const std::string &path, const std::string &path2)

+ void changeScene(const std::string &, std::shared_ptr< Scene > scene, bool endCurrentScene)

+ void run()

+ void quit()

+ Assets & getAssets()

+ sf::RenderWindow & window()

+ bool isRunning()

+ size_t getCoins()

+ size_t getVolume()

+ size_t getHighestScore()

+ bool getMusic()

+ bool getWASD()

+ void setCoins(size_t coins)

+ void setVolume(size_t volume)

+ void setHighestScore(size_t highestScore)

+ void setMusic(bool music)

+ void setWASD(bool wasd)

EntityManager

+ EntityManager()

+ void init()

+ void update()

+ std::shared_ptr< Entity > addEntity(const std::string &tag)

+ EntityVec & getEntities()

+ EntityVec & getEntities(const std::string &tag)

map< int, std::string >

size_t

bool

#m_game

#m_entityManager

#m_actionMap

#m_currentFrame

#m_paused

Scene

+ 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 currentFrame() const

+ const ActionMap & getActionMap() const

virtual void onEnd()=0