

There are 4 main types of class. Entities, Containers, Recording and Screens.

Entity	
id	ID
MAX_VELOCITY	float
VELOCITY_CHANGE	float
position	Vector2
velocity	Vector2
width	int
height	int
moveRandomCell(TiledMapTileLayer)	void
draw(Batch)	void
updatePosition(float, TiledMapTileLayer)	void
incrementTimeSinceLastUpdate(float)	void
resetTimeSinceLastUpdate()	void
applyMovementAction(Action)	void
XVelocity	float
YPosition	float
YVelocity	float
XPosition	float
position	Vector2
timeSinceLastUpdate	float

GameScreen	
TILE_SIZE	int
render(float)	void
keyUp(int)	boolean
keyDown(int)	boolean
keyTyped(char)	boolean
touchDown(int, int, int, int)	boolean
touchUp(int, int, int, int)	boolean
touchDragged(int, int, int)	boolean
mouseMoved(int, int)	boolean
scrolled(int)	boolean

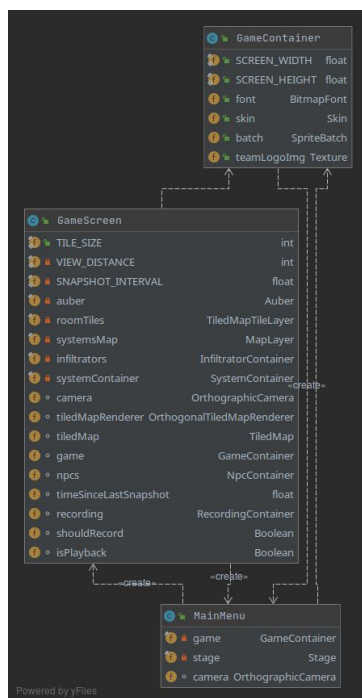
  

RecordingContainer	
newSnapshot()	void
addAction(Action)	void
addAllAction(List<Action>)	void
exportRecording()	void
snapshot	LinkedList<Action>

Powered by ymls

From the bottom up, the game starts with the GameContainer class, which starting the game and the loading screen.

Then it sets the “screen” to the MainMenu class.



The MainMenu has 4 options, Play Game, Record Game, Playback Game and quit.

Depending on what the user selects, it will instantiate a new GameScreen with the required parameters (A recordingContainer).

The GameScreen is the overarching class that links to 4 container classes, and the Auber class, for all aspects of the game.

It handles all aspects of updating and rendering entities, as well as recording snapshots of what has taken place, via function calls to the respective classes.

