Project Outcome Document

Build 840

Introduction:

My app is a Game called “STO” which stands for Space Treaty Organisation. As mentioned in the first scene this is a sci fi game whereby the player emulates the role of a United States Marine assigned to STO forces in a far way solar system which contains Australian and Chinese colonies.

The first few scenes of the game resemble a shooter game akin to Space Invaders. The player manoeuvres the ship left and right(in later scenes one can move the ship to others area of the screen) and fires missiles at the oncoming alien ships. The final scenes of the game are a platform shooter whereby the player emulates the pilot whom has landed on a planet and negotiate platform levels avoiding or destroying hostiles. There is a timer which starts at the beginning and is displayed at the end; one should try achieving the lowest time possible.

Main.lua

Nothing new here. This is the main that the simulator will kick off from. Four variables are listed here. “DamageFleetCounter” is a variable I use throughout the first few scenes. As the name suggests it counts the number of enemy ships that have been destroyed by the player. When his variable hits a certain value storyboard transitions to the next scene. I will explain this in more detail as I detail each scene. One will also notice that this variable is set to global as opposed to local as it is used in multiple scenes. The next three variables :” local minsText= 0 , secsText = 0 local timeText = 0 “ are used in the timescore fn which is common to all scenes. It starts a timer at the beginning and displays it throughout the game. Main transitions to the intro scene.

Intro.

This is the first scene of the game. I put up a nice sci fi background image and then some introductory text. A event listener toggles to the scene.

scene1a:

This is the second scene . Preceding this comment line , I include physics, sprite & storyboard. I also include two classes: missile & ship. The gravity is started. The No. of alien ships is determined by NoOfAliensX & NoOfAliensY. CreateScene starts with the calling of the Universe fn which creates four walls around the perimeter. Then the aliens fn is called. A table of ten sprites is created using a for loop. Each sprite has physics added. Initially I had thought to create my aliens as instances of a class however I had problems with collisions when the other object was also a instance of a class. The other object which these sprites inter act with are objects of the missile class which are obviously objects themselves. These sprites will transition to the bottom of the screen , whereby the GameOver fn is called. If the fleetstaus key

is still set to active, i.e not all the alien sprites have destroyed; then the game transitions to the GameEnd scene. The timeScore fn in then called. This fn has a display object which shows the time. A local timer constantly calls a local fn which increments the relevant objects and thus updates the display object. I then create the players ship. This a sprite with physics. I experimented with

Putting my ship in a class but I had difficulty with using sprites within a class so I simply left it in the scene. The ship is inserted in the group as is all the alien sprites. Enterscene has three timers. A tap listener is added to the ship which calls the missile fn. This fn takes the position of the ship and passes it to a instance of my missile class. I then add a collision listener to this instance which calls its own collision fn when necessary.

The missile class uses a image with physics and velocity added. It also plays a sound. It own collision fn makes the other object invisible, sensor = true, removes the other display object(display.remove); and finally remove itself if the other objects' type is set to 'destructible' which the alien sprites are. If the missile hits the wall, it removes itself.

I decided to play sounds for the collisions which I implemented in the class. The simulator has delay when it plays the sound after the event. I also experimented with playing them in the scenes but it made no difference. So in my opinion this is a bug in the app, unsure if it is a bug with corona or my setup.

The next listener is a runtime tap listener which calls the Moveship fn which moves the ship by simply changing the coordinates of the ship to that of the event.

The final listener is continuous timer which calls the DamageFleet fn every 1000ms. This fn accesses the DamageFleetCounter variable. If it is the value of ten, then the game moves onto the next scene. DamageFleetCounter is initially set to zero and will increment when a alien ship destroyed. This is implemented in the collision fn of the missile class.

The rest of the scene cancels the timer and purges the scene when app transitions to the next scene.

scene2a

This scene is the same as the previous scene except there are two rows of aliens which transition to the bottom of the screen.

scene3

This scene resembles the last two scene which introduces new features. Two new timers are introduced. EnemyMissileTimer calls the EnemyMissile fn. This fn picks one of the alien sprites at random and provided it is variable passes on it's coordinates with modification to a instance of the EnemyMissile class.

This class is very similar to the missile class with opposite slower velocity for the image object. However the collision listener is actually added in the constructor as opposed to the main. It's collision fn is self explanatory in that it makes the other invisible and removes itself as necessary.

The other new timer LifeOfViper continuously calls the LifeOfShip every 1000ms and accesses the visibility of the viper object. If it is invisible(which is toggled in the EnemyMissile class collision fn); then the app moves to GameEnd scene.

Also the aliens do not transition but are stationery.

scene4

This scene has a single row of alien sprites which fire missiles but the row is also in transition unlike in scene 3.

scene4a

Identical to Scene 4 except aliens are not in a row are randomly placed on the screen using math.random().

scene4b

Identical to scene4a except the alien sprites transition to the bottom of the screen at 40000ms instead of 80000ms.

scene4c

Again identical to the previous scene except transition is set to 20000ms for the alien table of sprites.

scene4d

A different approach in this scene. A timer AlienTimer is created in createScene which calls the alien fn every 2700 seconds. A alien sprite is generated and randomly placed on the screen and transitions to the bottom of the screen randomly firing missiles along the way.

scene4e

This scene is the same as the previous scene except the AlienTimer is activated every 1700ms plus the number of aliens to be destroyed before transition to the next scene is allowed is 20.

scene4f

Again a slight difference to scene4e in that the timer is changed to 700ms. There is a scene4g but it crashes corona for some reason so I have decided to leave it out of execution.

scene4i

This introduces the next phase of the game. It continues the story line and advises the player on the next platform scenes. Unfortunately when I transition form this scene to Scene 5 physics will not work properly. The main soldier sprite will not obey gravity properly and acts strangely to events. I do not why. Sometimes if one starts the game

from say scene 4a or 4c it works and maybe even from scene1a. But it is too inconsistent. I spend a few hours trying to find out why and trying various things but to no avail. I though it was scope in that in scene5 I originally set group to global within createScene. I then initiated physics from main.lua. I tried creating a transition from blank scene in between the problem levels. There is a flip command associated with storyboard.The reason I can come up with is that corona or storyboard cannot handle purging all the previous scenes with all the physics and then moving onto a totally different scene with different objects and physics. It could be that I am not removing objects properly or handling collisions; the simulator is getting bogged down and maybe half crashing. The subsequent levels will work fine if one starts the game at scene5 and progresses onwards.

scene5

This is a new phase to the app/game. Let's start with createScene. Five buttons are created at the bottom of the screen. Left, right, diagonal-left, diagonal-right and stop. Each button or image when tapped will cause the soldier sprite to turn & walk left; turn & walk right; jump up & left and jump up & right. The stop image causes the sprite to stop and will cause the sprite to fire a bullet in the direction it is facing. After these buttons comes the soldier sprite which is setup. I then call the Universe fn which as before creates a wall around the screen with physics. I then call the platform fn which creates two tables: swarm and platform. Each member of platform is a rectangle with physics placed around the screen. Each member of swarm is a sprite with physics placed on the platforms. Each alien has a collision event listener which ends the game if the soldier collides with a alien. The last member of the platform table is a image of a space door which has a listener so that when the soldier collides with the door it triggers a fn to transition to the next level.

Scene6,7,8,8a &8b

These next five levels of the game are essentially the same as scene5 but with the number and positioning of platforms different plus also the number/positioning of the aliens is different. The alien sprites sometimes have motion also in that their initial position causes momentum.

GameEnd

This is the final scene which shows the time you took to complete the whole game. Pressing the startagain image will allow a restart of the game.

Features

As outlined in the original document I intended creating levels akin to space invaders and a few platform levels. This was achieved. Unfortunately I did not achieve the levels whereby a player could target hostiles over a site. No work was done on creating a leader table although some research which I did conduct shows it is possible to implement.

Conclusion and Future work

An enjoyable and interesting project. I want always to create a video ever since I started playing them thirty years ago. Future work would entail more work using the os and system libraries of the corona api. My final year project is in steganography and I I hope to research creating a mobile app to run alongside the main user interface.

Bugs

The main bug that I mention was the transition form space invaders scenes to platform. So to experience the platform levels properly; I propose changing the gotoscene from scene1a to scene5 instead to run these levels properly. Or run as is to experience the error first hand.

I implemented the sound for collisions within the classes. There is a delay from when the collision happens to the actual sound been rendered? Even if I run from main this strange behaviour happens.

The clock acts weird in later levels in that it jumps 3 to 5 seconds at a time. This could be due to the stress I am maybe putting the simulator under. Or another reason is that I implemented the timeScore fn incorrectly. I have the time and the fn inside createScene in most cases. The proper way would be to rewrite the fn and have the timer event ran form enterScene. Hindsight is great but I will know for again. I decided at this late stage to leave as is.

During the final testing for reasons unknown when I start the game from scene5, the game will terminate to GameEnd at scene8 or 8a for some unknown reason even though I did not alter the code. So again to experience these levels properly one might half to setup intro to transition to these levels directly.