SOUTHEAST ASIAN JOURNAL OF SCIENCE AND TECHNOLOGY

Volume 2, Issue 1, 2017
P-ISSN: 2672-2984
E-ISSN: 2672-2992
www.sajst.org

Android Game Development: Space Maneuver

Jayson Sorio[1], Jo Ann Gutierrez[2], Alice Cervera[3], Jeffrey Tolentino[4,] Cristeta Tolentino[5]
Pangasinan State University, Lingayen Campus

Abstract – *Space Maneuver android game is a game where the player is an astronaut who's sent on a mission with his spaceship to collect clean energy sources called "Orbs" for his planet to be used as an alternative power source. The main goal of the player is to collect green orbs while avoiding asteroids until the player ran out of lives and set the highest score. The player spaceship has a weapon used for shooting asteroids. The mechanics of Space Maneuver game involves spaceship selection, map selection, game control, different orbs, and asteroids. The algorithms used in Space Maneuver game: Spaceship motion control, making the spaceship accelerate or slowdown, and random spawning of orbs and asteroids in the game scene. To measure the acceptability the developers conducted an acceptability survey to measure the Space Maneuver game's acceptability. The developers concluded that the Space Maneuver game is recommended to play in an android device that has accelerometer sensor for the tilting functionality of the game to work; a physics functions was made since the game is physics based; and lastly the game was highly accepted among players. There are some recommendations for the proper implementation and further development of the game, these are as follows: additional spaceship, so that the player have more choices and to make the game more enjoyable and exciting; time as score, to make the game more challenging it is good to include a time in calculating the score.*

Keywords – *Space Maneuver, spaceship, asteroids, orbs*

INTRODUCTION

Gaming today is widely recognized as part of our cultural landscape. Gaming itself is as old as history. Artifacts from ancient Sumeria and Egypt have shown that our ancestors enjoyed playing board games thousands of years ago. But electronic games required the invention of electronic computers. The earliest computers were slow, failure-prone monsters that took over entire rooms and had less power than a modern pocket calculator. Still, early programmers on these machines felt compelled to waste time by making these computers do things like playing tic-tac-toe. After World War II, electronic computers moved out of the realm of cutting-edge laboratories and into universities and large corporations. Many university students became the first game programmers, transforming their fantasy and sci-fi imaginations into digital adventures. (Reimer, 2005).

Videogames can change a person's brain and, as researchers are finding, often that change is for the better. A growing body of university research suggests that gaming improves creativity, decision-making, and perception. The specific benefits are wide ranging, from improved hand-eye coordination in surgeons to vision changes that boost night driving ability.

People who played action-based video and computer games made decisions 25% faster than others without sacrificing accuracy, according to a study. Indeed, the most adept gamers can make choices and act on them up to six times a second—four times faster than most people, other researchers found. (Hotz, 2012).

Android is the first open-source technology platform for mobile devices. Android is a mobile operating system based on the Linux kernel that is currently developed by Google and designed primarily for touchscreen mobile devices such as smartphones and tablets, as Android's user interface is mainly based on direct manipulation using touch gestures such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input.

Corona Labs Inc., formerly Ansca Mobile, is a California software company building a 2D game and app development platform. Its main offering is the Corona SDK, a cross-platform mobile development framework that builds native apps for iOS, Android, Kindle, Windows Phone, tvOS, Android TV and Mac and Windows desktop from a single code base. Corona uses integrated Lua layered on top of C++/OpenGL to build graphic applications.

The developed Space Maneuver android game is a game where the player acts as an astronaut who is sent on a mission with his spaceship to collect clean energy sources called “Orbs” for his planet to be used as an alternative power source instead of oil to avoid the increasing global carbon emissions which worsen global warming that threatens the existence of the astronaut’s planet. That is the story of the game.

The main goal of the player is to control the spaceship and to obtain score points and set the highest score. While collecting orbs the player must avoid obstacles or asteroids to progress in the game. The player spaceship’s health is determined by a health bar. Each time the player spaceship collides with an asteroid, the spaceship’s health is deducted by the asteroid’s damage which depends on the toughness level of the spaceship. For example, if the toughness level of the spaceship is level one (1) then the damage of asteroid is high, so every time the player upgrades the spaceship’s toughness level, the damage of asteroid is reduced.

In game there are six (6) types of orbs that the player can obtain: green, blue, red, yellow, and white. Green orb is the main orb that the player must collect to score points, each green orb is equivalent to one (1) point. Blue orb is the orb that makes the player spaceship slowdown for a certain time when obtained. Red orb, this orb adds one (1) additional HP to the player when obtained. Yellow orb, this orb when obtained multiplies the points obtained in each green orbs by two (2) points in a short period of time. Cyan orb is the orb that when obtained freezes all the asteroids for thirty (30) seconds. And lastly white orb, this clears all the asteriods for a short period of time when obtained.

Objectives Of The Study

The main aim of the study is to develop an Android game entitled Space Maneuver. In particular, the study sought (1) to identify the mechanics of the Space Maneuver android game, (2) to identify the algorithms that can be used in terms of spaceship motion controls of the game by touching the screen and by tilting the screen, making the spaceship slow down or accelerate; and randomly spawning of asteroids and orbs in the game scene and (3) to measure the level of acceptability of the game in terms of (a) gameplay, (b) graphics; and (c) sounds.

Methodology

Methodology is generally a guideline for solving problems with specific components such as phases, tasks, methods, techniques, and tools. The methodology that the developers used in the development of the system were presented and elaborated in this chapter.

The developers used the Scrum Methodology because of its procedures which allow them to focus on delivering the game in the shortest time. It allows the holistic product development where a development team works as a unit to reach a common goal. Scrum employs an iterative, incremental approach to optimize predictability and control risk. It is a process framework that has been used to manage complete product development since the early 1998s. Scrum is not just processes and techniques, scrum also makes clear the relative efficacy of the product management and development practices as the project improves.

The Scrum Methodology helped the developers to manage the best way to deliver the highest priority features of the project. It helped the developers to change the game structure and features at any point in time, having the final release to be in its fully functional state. Scrum resources that needed in the game development were well monitored and were ensured that these were available when needed. The game constantly changes to identify what it needs to be appropriate, competitive, and useful through continuous alterations and meets the requirements specified. In Scrum, as the game development progresses, anyone can see real working software and decide to release as is or continue to enhance for iteration.

Scrum Methodology aided the developers to use the seat resourceful solution in the whole development process. Each number was able to share ideas that contributed to the effectiveness of the project. Scrum has the following group of phrases:

Pre-Game

The pre-game phase concerns about two things: planning and high-level design.

Planning consists of conceptualization of the design and analysis of data structure of the game. The plan is created by the collaborative work of the developers. The team discussed the features that will encourage the gamer to become interested in the game being developed. The team worked together to forecast the functionality that will be developed. Based on these, the developers have the given importance of what shall be included in the game like graphics and textures, sound effects, game category and user interface.

High-level design covers reviewing of the planned concept, identifying the changes that will be made, and identifying any problems or issue that will be encountered in implementing the changes in building, enhancing and updating the concept of the study. This phase is where the developers discussed how they should adapt to the changes that might happen in the future. The term discussed what problems might be encountered and how they going to solve them.

During this phase, the developers identify the project goals, problems, and scope of required solution. The developers also had some research to identify and consider the concept of the game.

Mid-Game

The mid-game phase is usually called the sprint or development phase and it's an iterative cycle of development work. It consists of four phases: develop –coding and implementing the plans for the game, wrap– the executable version of the game being developed or the prototype, review – reviewing the game if it's good and ready for final release, and adjust – if the game is not yet good and has problems, adjust the game and resolve the problems encountered in review phase until it's ready for final release.

During this phase, the developers started coding the game based on the plans and concept made during the pre-game phase, hence, making the prototype. After the prototype was made, the developers tested and reviewed the prototype for further enhancement until the game is ready for release, in this phase, the concept and the design of the game might change.

Post-Game

After the game is fully developed and the developers felt that the requirements for the game were all satisfied. This phase prepares the developed game for general release and it is thoroughly tested ensuring that all changes made are working.

RESULTS AND DISCUSSION

Mechanics of the Developed Android Game

SPACESHIP SELECTION

The player must pick a spaceship in order to play. There are five (5) different spaceships available in the game, each with different design and features. At first the player can only access one of the spaceships, and the other spaceship can be unlocked by purchasing them through the game's currency called orb points which is equivalent to the total score of all the games played by the player. The spaceships can be also upgraded to enhance their capability in the game and each upgrade is also purchased.

MAP SELECTION

After the player chose a spaceship, next is to select a map or game scene where the player would play. There are five different maps that the player can choose. At first the player can only play on one map, the other maps can be unlocked by purchasing them also. Each map has its own feature that adds to the excitement of the game.

CONTROLS

There are two types of controls in the game; touch and tilt. Touch implies touching of the screen to control the spaceship, by touching the left half of the screen the spaceship rotates counterclockwise or goes to the left and it rotates clockwise or goes to the right if the player touches the right half of the screen. Tilt implies tilting of the phone to control the spaceship. By tilting perspective of the game scene or the environment rotates instead of the spaceship. Tilting to the left will cause the game perspective to rotate counterclockwise and clockwise if tilted to the right.

ORBS

These are the objects that the player must collect in order to progress in game. There are six (6) types of orbs in game:

- Green orb: the main orb that the player must collect. Getting this orb adds 1 point to the score. It is also needed to be able to use the weapon of the spaceship.
- Blue orb: Getting this orb will cause the spaceship to slow down temporarily.
- Red orb: Getting this orb gives additional HP to the player.
- Yellow orb: Getting this orb will doubles the points of every green orbs collected for a certain time.
- Cyan orb: Getting this orb will freezes all the asteroids for thirty (30) seconds.
- White orb: Getting this orb will clear out the asteroid which are visible in the screen.

Algorithms used in Space Maneuver android game

Spaceship motion controls of the game.

➢By touching the screen: To determine either the spaceship rotates clockwise or counter clockwise, the developers compares the value of the center of the screen in x-axis, _X = display.contentCenterX to the value of touch input in x-axis where the location of touch occurred, event.x. Ifevent.x is greater than _X then the spaceship rotates clockwise else the spaceship rotates counterclockwise.

➢By tilting the screen: In order to make tilting work the device should have an accelerometer sensor. Accelerometer in mobile phones measures the tilting motion and orientation of the phone using gravity. In Corona SDK, event.xGravity, event.yGravity, and event.zGravity provides the acceleration due to gravity relative to the x, y, and z axis of the device's screen in portrait orientation, this are the values that the developers used to manipulate the game's perspective rotation. The developers used the value of event.yGravity since the game is in landscape orientation. When tilting occurs, the value of event.yGravity becomes a positive value if the device is tilted to the right else it becomes negative if tilted to the left. In order to control the game's perspective rotation, the developers compares the value of event.yGravity to zero(0). If event.yGravity is greater than zero (positive value means tilting to the right) then the game perspective rotates to the right, else (negative value means tilting to the left) the game perspective rotates to the left.

Making the spaceship slow down or accelerate.

In order to make the spaceship accelerate or move, the developers made a function called shipAcceleration(), what it does is to calculate the velocity of the spaceship based on the value of its on-screen rotation angle -ship.rotation for its direction and the given value of how fast it will move on-screen for its speed -shipSpeed. On how to get the direction of the spaceship based on angles, the developers applied trigonometric functions to get the x-coordinate and y-coordinate where the spaceship should move.

The spaceship's rotation was set to 90 degrees because at game start the spaceship faces north. On line 3 and 4, the direction of the spaceship is determined by using trigonometric functions cos and sin to compute the direction's x-coordinate and y-coordinate based on the spaceship's rotation angle.

The spaceship won't accelerate unless it has speed. After the direction is determined, on line 5 and 6, the value of shipSpeed that is needed to cause the spaceship to move is multiplied to x and y coordinate or the direction where the spaceship should move. Here the xCoord and yCoord was set to negative value because in Corona SDK, positive y values extends downwards (not upwards as in Cartesian coordinate system) thus positive angles also goes clockwise and not counterclockwise.Finally, the on line 7, ship:setLinearVelocity() is called in order to make the spaceship move. The function object:setLinearVelocity() is a built-in Corona SDK function used for its built-in physics engine with its syntax, object:setLinearVelocity(xVelocity,yVelocity), it accepts x and y component for the physics object's linear velocity in pixel per second. For example, ship:setLinearVelocity(230,100) will cause the spaceship to move 230 pixel to the right and 100 pixel downwards per second. Randomly spawning of asteroids and orbs in the game scene.

In order to randomly spawn orbs and asteroids in the game scene, the developers made three functions each for orbs and asteroids namely: createOrbs(), createAsteroid(), deployOrbs(), deployAsteroid(), orbSpawnLoop(), and asteroidSpawnLoop(). In createOrbs() and createAsteroid() function, here the display objects which are orbs and asteroids are being initialized then moved inside a table named orbsTable and asteroidsTable. After the display objects are set, the deployOrbs() and deployAsteroid() function are called, what this functions does is to display or spawn the orbs and asteroids randomly to the position where it cannot be seen yet on the screen and is relative to the position of the spaceship since the spaceship is literally moving around the game scene causing the game scene's center coordinates to change everytime. And finally, to spawn many orbs and asteroids every time on the game scene, the orbSpawnLoop() and asteroidSpawnLoop() function was set to a timer namely orbSLTimer and asteroidSLTimer which serves as a game timer or game loop that always calls the orbSpawnLoop() and asteroidSpawnLoop() function every certain amount of time, this functions also traces the locations of orbs and asteroids which are drifted off the screen and removes them for the purpose of game memory clean up by accessing the tables, orbsTable and asteroidsTable where the display objects are stored and comparing each display object's coordinates to the spaceships coordinates, which determines the distance between the display object and the spaceship.

Even though the functions of spawning an orb is similar to the functions of spawning an asteroid, the developers purposely separate the functions of spawning orb and asteroid because the orb and asteroid display objects is different to each other and the delays and amount of spawning of this display objects is also different. Asteroid display objects requires a physics body in order to move

around in-game as if it is floating in space whereas the orb display objects are not. The timer for spawning asteroid is fast and the deployAsteroid() function is called twice every iteration or update of the timer whereas the timer for spawning orbs is slow and there's a limited number of orbs to be spawned in game.

Measure the level of acceptability of the game.

The level of acceptability of the Space Maneuver android game was established based on the following criteria: gameplay, graphics, and sounds.

Table 1. Overall Acceptability Level

Acceptability Variables	**Weighted Mean**	**Description**
Gameplay	4.52	Excellent
Graphics	4.56	Excellent
Sounds	4.54	Excellent
Overall Average	**4.54**	**Excellent**

Table 1 shows the weighted mean of the level of acceptability of the Space Maneuver android game is **4.54** with a descriptive rating of "**Excellent**". This means that the game has "excellent" gameplay, graphics, and sounds.

Conclusion and Recommendation

Based on the findings of the study, the researchers concluded that the Space Maneuver game is recommended to play in an android device that has accelerometer sensor to make the tilting functionality of the game works. Likewise, Space Maneuver game is a physics-based game. The developer made a function that would set the velocity of the spaceship and use it with the Corona SDK's built-in physics engine.

The developers recommends to the future developers to add more spaceship to Space

Maneuver android game so that the player have more choices and to make the game enjoyable and exciting. To make the game more challenging it is good to include a time in calculating the score for example the longer the player stays in the game, the higher the score of the player gets.

REFERENCES

Toledo R.G., Ventanilla J.D., Aquino M.M., March (2015), Brain Time Game, Pangasinan State University, Lingayen Campus

Fernandez J. R., Natoc J. A., Valenton D.V., March (2015), Whack-A-Mole android game, Pangasinan State University, Lingayen Campus
Jimenez KN. A., Sison Ma. C. F., Casupanan P. C., Pascual R. M., March (2015), Stealth Game, Pangasinan State University, Lingayen Campus

Fernandez J. F., Garcia BR. T., GumaposD.M., Paragas R.A., April (2016), City Trek 2D, Pangasinan State University, Lingayen Campus

Vargas ZM. C., Rosario SK. P., Marco RJ. P., April (2016), Rolling Ball android game, Pangasinan State University, Lingayen Campus

Flores JP. U., Alferez J.C., Hora MF. D., April (2016), aMAZEng Run android game Pangasinan State University, Lingayen Campus

Reimer, J. (2005, November 11). The evolution of gaming: computers, consoles, and arcade. Retrieve from https://arstechnica.com/features/2005/10/gaming-evolution/

Hotz, R.L. (2012, March 13). When gaming is good for you. Retrieve from https://www.wsj.com/articles/SB10001424052970203458604577263273943183932

Contact Information:

Name: JAYSON E. SORIO
Contact No: +639566033991
Email Address: sorio.jay@gmail.com

Name: ALICE C. CERVERA
Contact No: +639425595967
Email Address: cerveraalice@yahoo.com

Name: JO ANN N. GUTIERREZ
Contact No: +639105172884
Email Address: iam.joanngutierrez@gmail.com

Name: JEFFREY V. TOLENTINO
Contact No: +639064655724
Email Address: jvtolentino@gmail.com

Name: CRISTETA G. TOLENTINO
Contact No: +639425595967
Email Address: crisgarsin@yahoo.com