A 2d android game 'Balloon Blast'

Final Report
05.09.2019

Md Sehabub Zaman Pranta
1611251042
CSE299.12
North South University
Electrical & Computer Engineering
Dhaka, Bangladesh
sehabub.pranta@northsouth.edu

Hasanul Banna
1612536642
CSE299.12
North south University
Electrical & Computer Engineering
Dhaka, Bangladesh
hasanul.banna@northsouth.edu

***Abstract*— Android 2D games are very popular in today's world. One of the most popular arcade game is Fruit Ninja which has more than 100 million download in google play store [14]. Some others arcade games like Bubble Shooter, Pyramid Crush, Fruit Master are also very popular and have over millions of download. So now it is very clear why we choose this arcade game project. We have tried to make a unique 2D game in this segment for high popularity and android users demand. In this project we made an arcade game by using android support libraries and 2D graphics. Graphics were animated and moved in vertical position to run the game. This game needs players touch on screen to control and no need to move the device for objects direction changing as, the direction was fixed to the vertical position. Player will start from level 1, and each increasing level the game will be harder. This game has two chance to skip any two-levels and at the same time player will get two levels bonus point. Players need to cross 20 level successfully to win this game.**

Keywords— Android game, Video game, Arcade game, Offline game, Balloon game, 2D game.

I. INTRODUCTION

Android game development is now a trend for android app developer and java developer. According to Wikipedia, first android game published in October 28, 2008 in google play [1]. Then the popularity of the android game spread the whole world. Day by day google started supporting more android libraries for developers and the developer community started raising from that time. Now this platform is supporting many country and individuals economically. Many companies are also investing in this field. The google announced android online game as a service in 2013 developer conference [1]. For this purpose, the standalone mobile application was launched on July 2013[1].

Android game is an application of java programming. It focuses on the user's attraction by providing fantastic graphics and smooth-running path. Games are available for users through google play store for free or at a cost [1]. Play store verifies the games the developer submits to google play for user's reliability. Games have some documentation about user's hardware requirement, game feature, privacy policy and other security alerts which are managed by google. As this is a worldwide popular platform, it had 82 billion apps download by the year 2016 [1]. So, the main goal is to develop and spread applications throughout the world. As android is free for both developers and users, the community of developers are increasing day by day.

For game developments two languages are very popular nowadays, one is Java and another one is Kotlin. So, java programming was used to build this game for its simplicity. All the libraries were used from java library and Googles android support library. The interface design was conducted by using XML language for its resource availability and simplicity. Some external graphics were also integrated to make the game more attractive and accurate.

So from the next session we will discuss details of this project how it was implemented by maintaining some rules. The topics that we will discuss is given below:

- Background
- List of Components
- Experimental Result

- Task Distribution
- Game Description
- Result Analysis
- Conclusion
- Future Plan

So from here we will start our discussion.

II. BACKGROUND

In the category of Arcade game here have many popular and related games of our project. All these related projects are live now in google play store. Some related game like Balloon Archer, Balloon Smasher, Balloon Bow and other arcade games which have over millions of download [15]. As android is an open platform, people of the whole world is trying to contribute their role on this platform. Now it is a huge platform and people emphasizes on some innovative games. So after observing the huge response of the people throughout the whole world and resource availability we decided to build this Balloon Blast game.

This game is basically based on the core java programming which compiles JVM to bytecode for making machine readable format. Machine then implements all high-level language codes to generate a specific output [2]. The developer's machine should satisfy some hardware requirement to build android applications. Some requirements are minimum of a 3 GB RAM and 8 GB recommended, available disk space of 2 GB, minimum of a screen resolution 1280 x 800 [3]. In this environment games run smoothly automatically updates all support libraries through internet. Android environment makes the initial structure of the project by its build tools and no need for developers interfere. As this game have some external graphics support, some additional android libraries were also implemented to integrate all the projects resource. Android grade build tools also provides some services to test the application on a virtual device of developer's machine by the permission from BIOS.

The game interface mainly built by XML programming and the background of the XML Java program will control the whole game activities. In all resources background, java library was implemented to run the resources smoothly. When the game runs in the virtual device, the Android studio remains idle and it works in emulators background until the app successfully installs in the emulator. So, it is clear that

when the app successfully installs in a virtual or physical device, the android studio closes the connection with android devices. Then no thread or programming controls remains in background to interrupt the running game on android device.

III. LIST OF COMPONENTS

this project only software components have been used. The main software components have been listed below:

1. Android Studio.
2. Adobe Photoshop
3. Final Android Resizer
4. Android Virtual Device
5. Android Physical Device or Real Device

In The main tool for building the android application is Android Studio [4]. It generates the APK file to fun in android system. The core java and XML programming has been implemented here. It is a worldwide popular integrated development kit and people are using it to build some excellent project. It has also a version control system to work in a group or in a team to maintain a centralized system. One of the major advantage of Android Studio is, it has direct access to google libraries so that developers can use all the libraries efficiently.

For external graphics support the Adobe Photoshop were used [5]. The balloon design, background graphics, pin icon and some other graphics were designed by this tool. Some graphics which were designed by core XML needed to assign some external color to it [6]. The Photoshop also have many advanced feature to design graphics in 3D segment. But for this project we only used 2D features of Photoshop.

Another tool for formatting the graphics in the different formats, Final Android Resizer was used [7]. As the different phone has different size and all the phones don't have the same android version, it is necessary to create different graphics format to run in different hardware or software environment smoothly. So, here Final Android Resizer controlled this situation by providing different resolution of graphics from a core graphics to fit in different android screen.

After successfully building the APK file, it was tested in virtual device to observe the game performance. The virtual device behaves as same as a real physical device [8]. Virtual actually mimics all the idea from real physical device. Because google have tried to push all features from a real device to virtual device. It also protects app privacy and performance. Though it is virtual environment, it can evaluate apps like a real device.

To implement all resources in a correct manner an online tutor's support had also taken from Lynda.com [9].

Now time comes to test in real physical device. The physical device had connected with computer and game were launched in that device. The performance was almost same between two devices. Because virtual device is exactly the copy version of physical device.

IV. EXPERIMENTAL RESULT

The target output of this game was to run all functionalities which implemented in android studio. The targeted output didn't come directly as there were some internal bugs. The issues mainly came from the java file which were tried to run the XML designed interface. The interface reported some error because of misuse of some XML library. Then the error was detected by debugging the java files.

Still expected output didn't come. Some representation of graphics was not displayed properly on the game screen. Because the android animation library only understands the grayscale image. Then some graphics were re-designed with grayscale format and again integrated with the android environment. This time all graphics were successfully launched.

Another issue was with game score. As no cloud database were used, there raised a discrepancy with games current score. Because temporary memories are volatile. To overcome this problem, the local storage of the user's device then used. Then the score and other data were successfully stored. To test each function individually, a debug operation was applied in all functions and it didn't report any error.

As this game have three stages it was difficult to place the specific resource in a sequence with the interaction of increasing level. Later it was fixed by splitting the range of the game levels.

So all these issues were fixed after testing each features separately. Then the app had run smoothly.

V. TASK DISTRIBUTION

The whole projects task was divided into 8 subtasks and each task was completed sequentially. The output of Each task was the input of further tasks. To distribute the whole project in terms of time, the online tool TeamGantt was used [10]. Each subtask was tested separately to remove future bugs in project functionality. After completing all the 8 part, all these parts were merged to perform the integrated task. Then all parts became one unit and built an APK file after successive run.

Figure 1: Gantt Chart

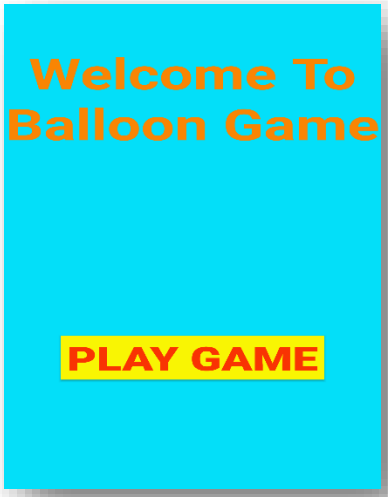
Then again, the test was required for this single unit to check that any error or bugs are raising or not.

As Figure 1 shows that each tasks took some specific period of time. So the two members of our group have worked simultaneously in each period of time. Each single week represents a portion of a whole task and it has finished sequentially.

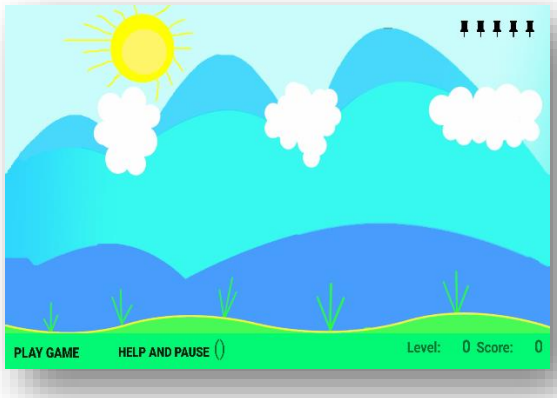
VI. GAME DESCRIPTION

In this session we will discuss the whole game in details. For this purpose will present some visual representation of our game.

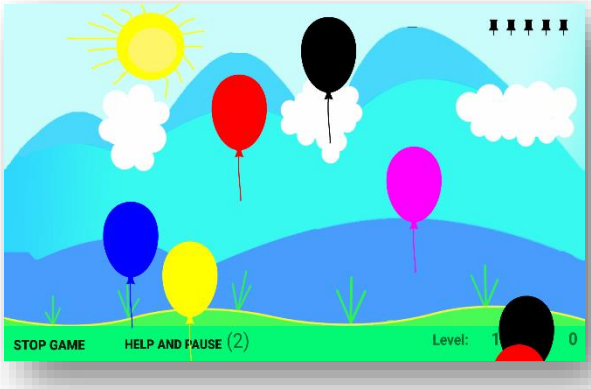
Step-1: First game starts and shows a welcome screen. The welcome Screen only have one button to start the game and if users wants to exit then needs to press back button in android phone.



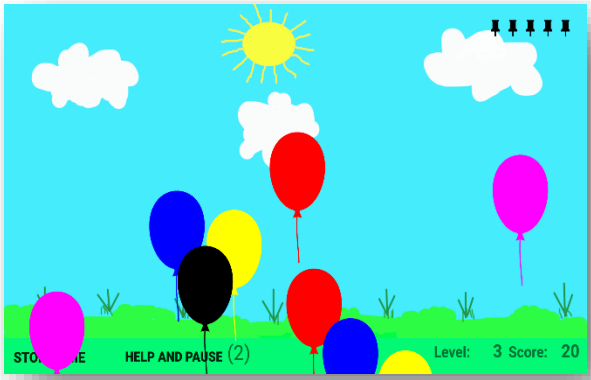
Step-2: When user press the PLAY GAME button then comes a new game screen. This screen shows five black icon that indicates life for player. The Start Game button is for start the game and some text view at the right bottom corner shows score and level.



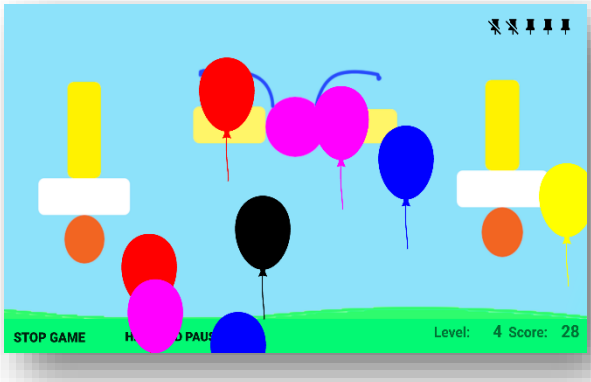
Step-3: When player touches on play game button then the game starts and Balloon starts flowing from bottom to top of the screen. The help and pause button gets value of 2. That means player can skip at most 2 level and can get bonus maximum 2 times. Then the level and score will increase for successive touch on balloon. If user misses any balloon to touch then the black icon will be crossed mark which indicates player's life.



Step-4: If player crosses level 4 and starts level 5 then the new stage will come to screen.



Step-5: Likewise if user crosses level 9 and starts level 10 then the final stage will appear. This stage will run until level 20 and if user can pass level 20 then the game will be over and the player will be win.



So these are the all process from start to end of this game that a player can follow.

VII. RESULT ANALYSIS

The output was meet with the expected result after some iteration. First the testing devices took some time to install the APK file. Once the APK had installed, the game ran successfully. The game was tested in different android devices and each device returned an excellent result. The running time in all android device was the same. The graphics had viewed as per devices resolution. The database which was managed locally worked perfectly in each device.

The support of background music was also fine. Music started when the game started, and balloon blasting sound also delivered properly. By this time some android library updated, and the overall performance had increased.

According to medium, it is also possible to increase any games performance in android device by following some rules. They are:

* Android Developer Option: Enabling Force 4x MSAA options increases games funning speed.

* Uninstall Unwanted Apps: By removing unwanted apps, RAM becomes free and games get more space to run.

* Update Android: Always keep the device up to date results in speed and performance increase.

* Turn Off Background Services: Background task can interrupt the running game. So, it is an efficient way to turn off all background tasks.

* Turn Off Animations: Screen animations can decrease the overall performance. When game runs, the android environment get interrupted when other animations play in background [11].

VIII. CONCLUSION

According to google, there are 2.5 billion android active devices have in today's world [12]. Android game is now an advanced technological service throughout the whole world. Because game is not limited only in 2D segment. Now many companies and organizations are integrating artificial intelligence to the game technology. Todays community game are also very popular. So, in this kind of situation, building skill in game developing would be a great idea. The support from Google is also taking it to next level. Because Google is very concern to android system and trying to improve the system regularly. As a result, google is publishing new updates and sharing new resources frequently. Not only the library and resources, but the android system also updates every year.

The thought of the game as a part of a bigger technological process is really in the core mindset that this project wants to reflect. Games can do many tasks very well, but they certainly cannot do everything at once. Especially not without solid supporting structures around the game environment. Throughout the project and the case studies, we built this was true. This project aimed too much at using different and innovative methods through coding modern games and playing games as part of learning, as it develops the skills of students in extending academic goals to understand, support and includes the whole career [13].

IX. FUTURE PLAN

In future we have plan to develop game with artificial intelligence in 3D segment. In this structure the game will be able to beat more human intelligence. We have also plan to build the game with the IOS environment so that, the game can reach both android and IOS users. Because the demand of the mobile game is increasing everyday as the users increasing.

We have also plan to build develop game for windows or macOS environment. Because games are also popular in computer environment. So, our main goal is to build game for any environment.

REFERENCES

[1] Google Play. [Online]. Available at: https://en.wikipedia.org/wiki/Google_Play#Games [Accessed: 28-sep-2018]

[2] Wikipedia. "Java Virtual Machine". Medium [online]. Available at: https://en.wikipedia.org/wiki/Java_virtual_machine

[3] Friesen J. (2019). "ANDROID STUDIO FOR BEGINNERS". Medium. [Online]. Available at: https://www.javaworld.com/article/3095406/android-

studio-for-beginners-part-1-installation-and-setup.html [Accessed: 12-Feb-2019]

[4] Android Studio. Medium. [online]. Available at: https://developer.android.com/studio

[5] Adobe. Medium. [Online]. Available at: https://www.adobe.com/products/photoshop.html

[6] HTML COLOR CODES. Medium. [Online]. Available at: https://htmlcolorcodes.com/

[7] RecordNotFound. Medium. [Online]. Available at: https://recordnotfound.com/Final-Android-Resizer-asystat-15318 [Accessed: 14-Mar-2018]

[8] Android Studio. Medium. [Online]. Available at: https://developer.android.com/studio/run/managing-avds

[9] Lynda. Medium. [Online]. Available at: https://www.lynda.com/

[10] Teamgantt. Medium. [Online]. Available at: https://www.teamgantt.com/h?utm_expid=.Il6xIwhZRhS6iptrhhXPFg.1&utm_referrer=https%3A%2F%2Fwww.google.com%2F

[11] A Medium Corporation. Medium. [Online]. Available at: https://medium.com/@systweak/mhow-to-boost-gaming-performance-on-android-f612fd9de9e0 [Accessed: 29-June-2018]

[12] Rossignol J. (2017) Medium. [Online]. Available at: https://www.macrumors.com/2017/05/17/2-billion-active-android-devices/ [Accessed: 17-May-2017]

[13] GAMES-for learning. Medium. [Online]. Available at: https://www.gamesforlearning.se/games-conclusions/

[14] Fruit Ninja. Medium. [Online]. Available at: https://play.google.com/store/apps/details?id=com.halfbrick.fruitninjafree [Accessed: 28-Aug-2019]

[15] Balloon Bow & Arrow. Medium. [Online]. Available at: https://play.google.com/store/apps/details?id=com.dexterltd.games.balloon_bow_arrow