

Android for Java Programmers

Documentation :

Note : Sorry about my english, really hard to explain problem, I hope you will understand what I meant in this project .

My presentation on YouTube : https://www.youtube.com/watch?v=o_eTA4ZBrO4
I'm sorry for the sound.

1) General project description, project goals and use cases.

My project is about a little game where you have to catch some balls to upgrade level and avoid dark ball. The player is represented by a small image of a bird and by clicking on the mouse, you can « moving » the player up and down. I also disable the keyback, (I will show you in the video), otherwise it provide some huge bug.

2) Features

As I said before, you can play with the mouse, you can also turn off the sound in the settings activity and change the background color, you can share your score once you loose, the highscore is saved with SharedPreferences. If you pass a level the speeds of the ball increase.

3) Problems faced during project implementation, how they were solved.

Well, there was some problem at the start like how the box will go up or down, so I used MotionEvent and some booleans (action_flg, start_flg), the box must not go outside of the screen (frameLayout), so I needed to get the size of the frameLayout (using WindowManager to get the size), about upgrade the level, it was a bit tricky because i wanted something like every 50 points you upgrade, so use modulo, but the player can have 40 points, « eat » a pink ball and then have 70 points so I didn't really know how I'm gonna make this work, so I decide that if the player eat 5 balls then the level is passed, I know it's not really a good solution but I didn't know how make it worked.

4) Problems left unsolved, ideas about possible solutions. / Possible future work, additional features.

As a possible future work ; play with the keyboard and then the player can also go in the « X » orientation (left to right), I think it's possible with the KeyEvent, maybe add some more balls thrown to the player with some bonus and special action (like invincible during x seconds), that means add a boolean when he eat the balls, if he touched a dark ball we don't stop the timer, maybe add more than one life, maybe also add a menu when you can directly choose your level, there is a lot of stuff than I can made but it tooked me too long to really understand how implement the « simple game ». I could also make multiple image player with different actions when they eat the bonus and the player can choose his image player before starting the game. As I said there is a lof of stuff to do.

5) Instructions on how to compile/run/use your application

Nothing particular.

6) References to external resources (documents, books, projects, source codes) used in the project.

A lot of the website StackOverflow, otherwise the android developer website (I put the website in comments in my code).

7) About what you asked to me :

- *There must be some sprite (or a 3D model, if you wish) representing the player :*

The player is represented by an bird image

- *There must be several levels in the game, which should differ with at least the background image (it would be better to have more surroundings, but I am not sure what you had in mind);*

There is several levels and the background image change every time the level is passed

- *There also must be several levels of difficulty, which would differ in the rate and speed of balls thrown at the player, etc. (see below);*

Every time the level is passed, the speed of ball thrown to the player is increase.

- *There must be a score system: it's up to you to decide on the details, but I would suggest adding a certain number of points for each gameplay second until the game over (the precise number should depend on the difficulty); also, the score must be saved somehow (SharedPreferences, a serialized file, etc.);*

There is different point according to the different ball, the score is saved and display at the end of the game, the highScore is saved with SharedPreferences.

- *The game must have at least the start screen, the game screen, the settings screen, and the high score screen;*

I think it's ok.

- *When the player dies and the game is over, there must be a message about that (including the resulting score), and then the player must be redirected to the start screen;*

There is a message saying game over and then display the score and the highScore, the player can clicked on try again and then redirected to the start screen.

- *Your game must handle the pause mode during the gameplay: if the user clicks the Home button, and then opens your game again (i.e., onPause() and then onResume are invoked()), the game should appear as paused and provide some button / click handler to continue, rather than get reset (or crash!);*

I made a demonstration in the video and you can click on the home button and then come back to the game, it won't crash.

- *The game must support sounds and background music;*

There is a background music (MediaPlayer) and also a sound when you « eat » or when it's game over (SoundPool).

- *The settings screen must contain access to SharedPreferences (either with a PreferenceFragment/PreferenceActivity or via a custom activity, if you want to keep it stylized as the rest of the game UI), including separate options to turn sounds and music off/on;*

In the settings screen, you can disable sound and also change the preferences background color start

- *The game must support sharing the game results / high score as plain text with other apps using an App Chooser dialog (similar to Assignment 3).*

When the game is over, there is a button share where you can share your score to your friend or whatever.