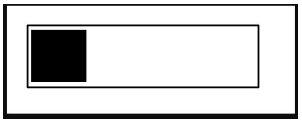
Unity Game Assignment

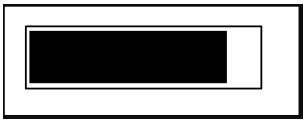
You have to create a game in Unity which can be played in Android (or iOS if you prefer). You need to provide a link to the repository (you can use bitbucket, or github, or any other) where you have been pushing your code and an apk file, or a testflight invite, or a device with the app installed.

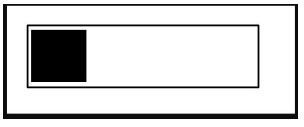
The game is a simple skee-ball simulator with multiple holes played on a table. The game will have a main menu with 2 buttons, one of them will go to the game, and another to a local leaderboard. The highest score will be displayed in the main menu.

The local leaderboard will show the 5 highest scores obtained by players in this device. The leaderboard will be persistent between sessions, so if the app is closed the leaderboard data is still there. Each entry will contain only the points obtained from the highest to lowest. This window will have a button to go back to main menu.

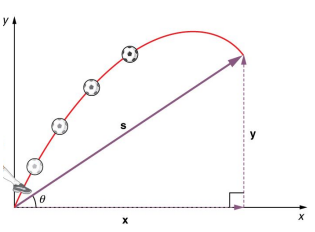
When the user enters the game they will be a countdown of 3 seconds. The game will have a time limit of 30 seconds. The player will have unlimited balls and each ball played is randomly generated with different weight and bonus points. The table will have different holes on it. Each hole will give certain points to the user. To be able to throw the ball the user will press on the screen. This will define position from where they will shoot the ball. The force to push the ball will be defined by how long the user holds the finger on the screen. This power will have minimum and a maximum power level. As soon as the power bar reaches the maximum it will go back to minimum, and when it reaches minimum again it will go back to maximum. Here there is a good example. The game will always show the current score.







Once the user releases the finger the ball will be shot following a parabolic path. The force applied is the value we get from the power bar. If the ball falls in any of the holes the user will get the



points for that hole. The user will be able to shoot again as soon as the ball reaches certain distance away of the camera, as long as there is some time left. If the ball falls outside the table the user gets no points. Every ball will disappear after 5 seconds. Every time the user scores points, the score will increase.

As soon as the timer has completed you will see a popup telling the user the amount of points they have achieved and the position they obtained in the leaderboard. If their points is too low the following message will be shown instead: "*Your score is low. Try again*". If their position in the leaderboard is the first the following message will show: "*Amazing! You got the highest score!*". If they got in first position the popup will have 1 button with the text "*Go back*" that will send them to the main menu. However, if they end up in another position the popup will have another extra button saying: "*Try again*.". This button will restart the game.

What will be evaluated:

- Performance of the game : that game works and there are no crashes, or visual glitches
- Use of design pattern
- The quality of the code
- The game follows the requirements
- Information in the repository and usage of the repository

What won't be evaluated:

- Graphics for the game. Simple unity 3d cubes are ok. If creating the walls around the holes is difficult, just make a simple plane with square holes. There is no penalization in any way if it doesn't look pretty.
- Audio in the game. There is no need to add any audio.

Extra things that are not required but would be nice to see:

- Using latest Android sdk, added icon and splash screen
- Make use of the Android back button.
- Make the game work for different aspect ratios, mobile and tablet.
- Make the power bar move a different speed if it is closer to the end of the bar.
- Be able to have some kind of game configuration where you can modify certain values for the game for the developer. These values could be:
    - Amount of points given for each hole.
    - The duration of the timer of each game.
    - Speed (or time to complete) of the Power bar.
- Possibility to choose from different maps (different number of holes). And have different leaderboards for each map.
- Possibility to choose different difficulties, where difficulty will move the table in harder difficulties.
- Support different level configurations based on config files.
- Handle loading/saving data on a server (for example: Firebase)