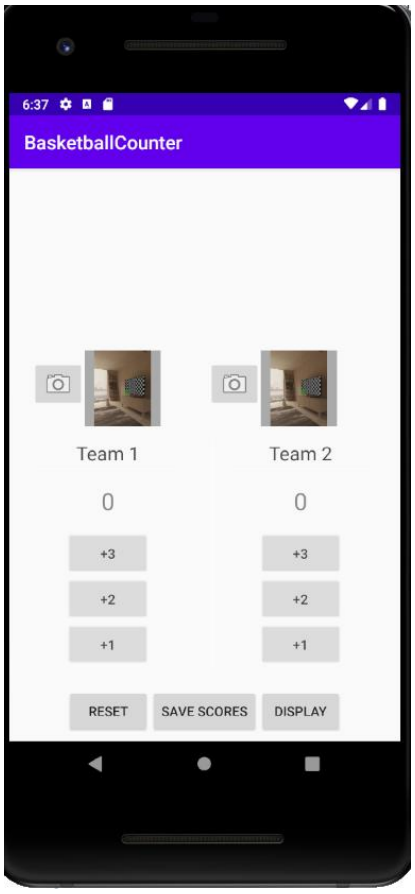
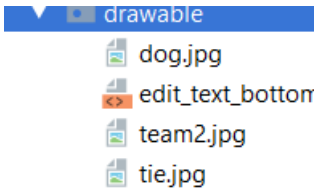
Project5 Report

Vinit Kothari

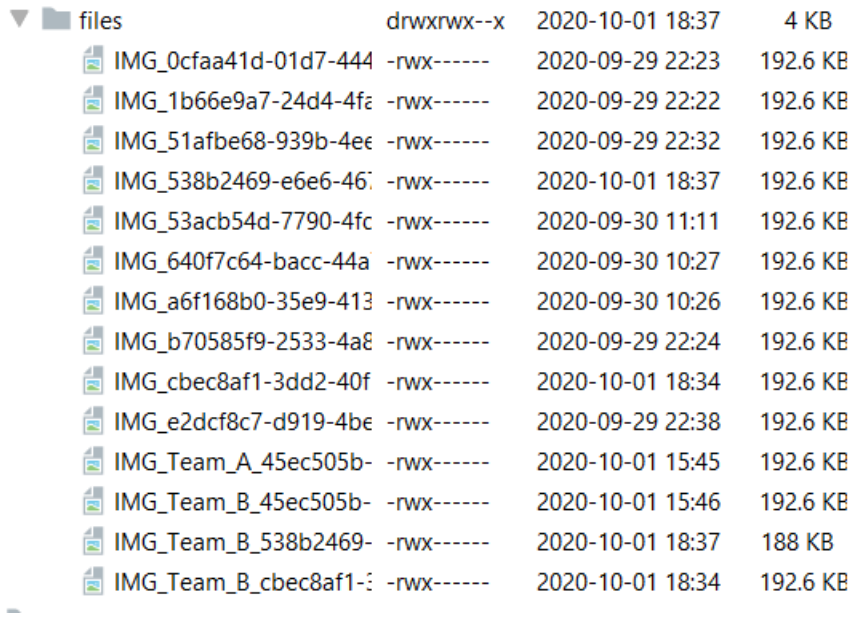
Part 1: Using Implicit Intents to upload team photos.

Part 1 involved giving each team a profile picture. This involved creating camera intents in the form of implicit intents that would enable clicking pictures and allow them to be saved for the right team. In order to achieve this a request code, URI and photofile were created for each image which then helped tell who the picture would be saved for. As seen in picture 1 an image is uploaded for both teams from camera. Bitmap was also used to give a scaled down image as an actual size would consume too much memory. A screenshot of the files of the images stored in the AVD can be seen in picture 3.





Picture 2: drawable resources

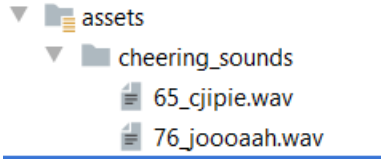


Picture 1: Showing images for both teams.

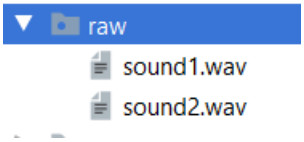
Picture 3: Files Screenshot from AVD.

Part 2:

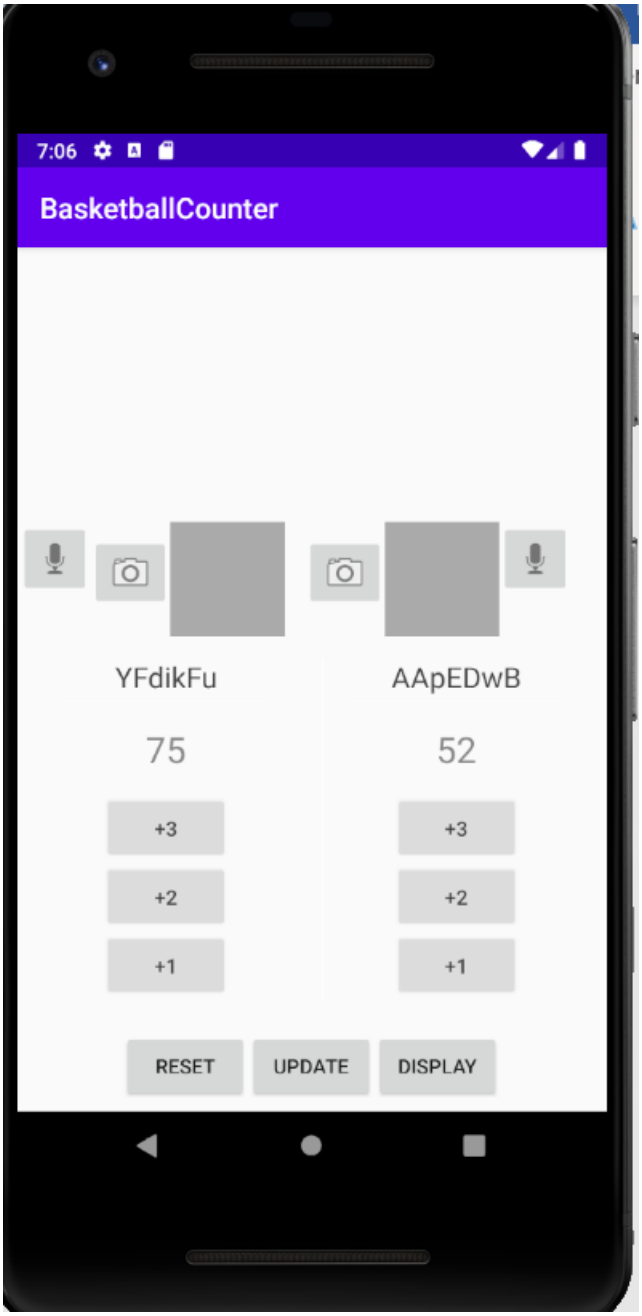
Part 2 of the project involved using soundpool and Media player to play sounds. Soundpool was implemented by first using an assets folder where the sound files were stored then a soundpool was created, after the soundpool was created it was used to play the files from the assets folder, on click of each of the cheer buttons. The sounds played had values dedicated for their volume to be looped and so on to make sure a sound is only played appropriately when needed. After completing this part I managed to implement soundpool also using a raw folder instead of an assets folder. The same raw folder was then used to create sounds through a mediaplayer. A mediaplayer is often used for larger sounds but in this case was used for cheers to make sure appropriate sounds were played a media player was created only on click of each of the two cheer buttons. And then through the mediaplayer the soundfiles were then accessed in the raw folder and then were given a value for their volume left, right, loop and so on. I also released mediaplayer files when the activity was paused or stopped or destroyed to ensure that resources were not wasted. Pictures 4 and 5 show the assets and the raw folder. (For more information please ask for a screen recording of the screen while playing the cheers) Picture 6 shows the layout of the screen with the cheer buttons.



Picture 4: The assets folder



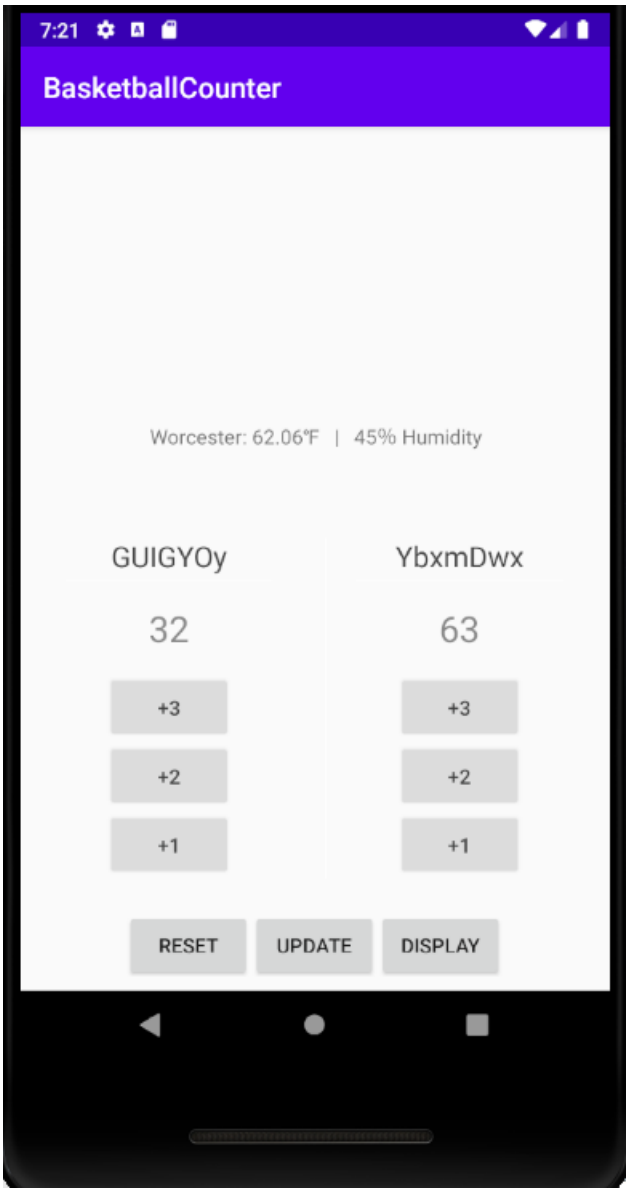
Picture 5: Raw Folder.

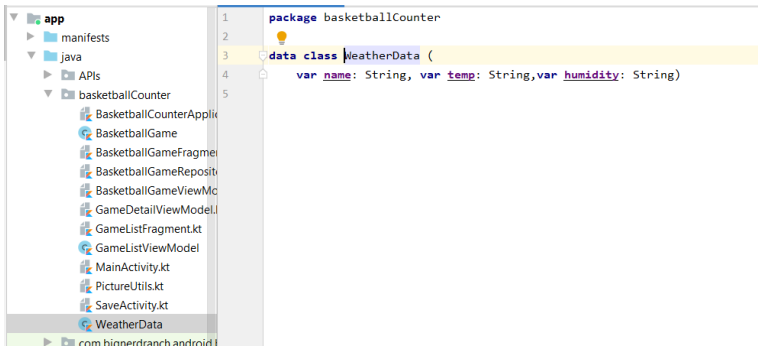


Picture 6: Layout with the cheer buttons.

Part 3: Getting weather Information

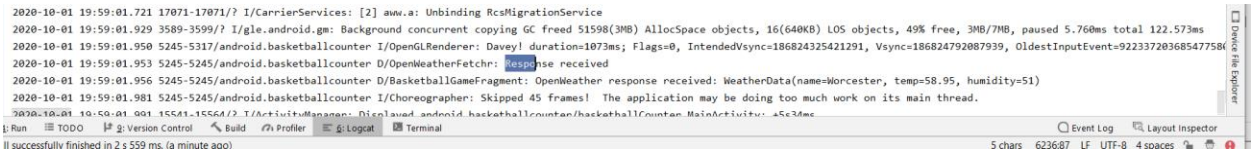
Part 3 involved implementing an API in order to receive weather data about where you were . The weather data included the temperature and humidity as a percent. To ensure that this data is stored a data class called Weather Data was created as shown in figure a class to deserialize the Json Using GSON was created also. This class was called weatherfetchr. Deserailizing using GSON is important to convert strings to JSON objects which is what helps show weather data. Picture 9 shows the logcat logs for the Weather Data.





Picture 7: Displaying the weather information

Picture 8: Creating a data class to hold API elements



Picture 9: Logcat logs.