**Drinking Game Application**

Minyoung Kim

CIT - Purdue University  
2506 Redfree Dr.  
765-413-9571

kim2292@purdue.edu

Jieun You

CIT - Purdue University  
430 W Wood St.  
765-775-0540

you44@purdue.edu

Ge Chen

CIT - Purdue University  
101 Andrew Place.

765-418-7351

chen1934@purdue.edu

**ABSTRACT**

Our project is about developing an application that consists of various drinking games. There are many people who are taking risks from drinking outside with friends. These can lead to injuries, arrest, etc. This application provides joy and cheerful environment to users/drinkers with no risks.

**Keywords**

Android Application; Entertainment; Games;

# INTRODUCTION

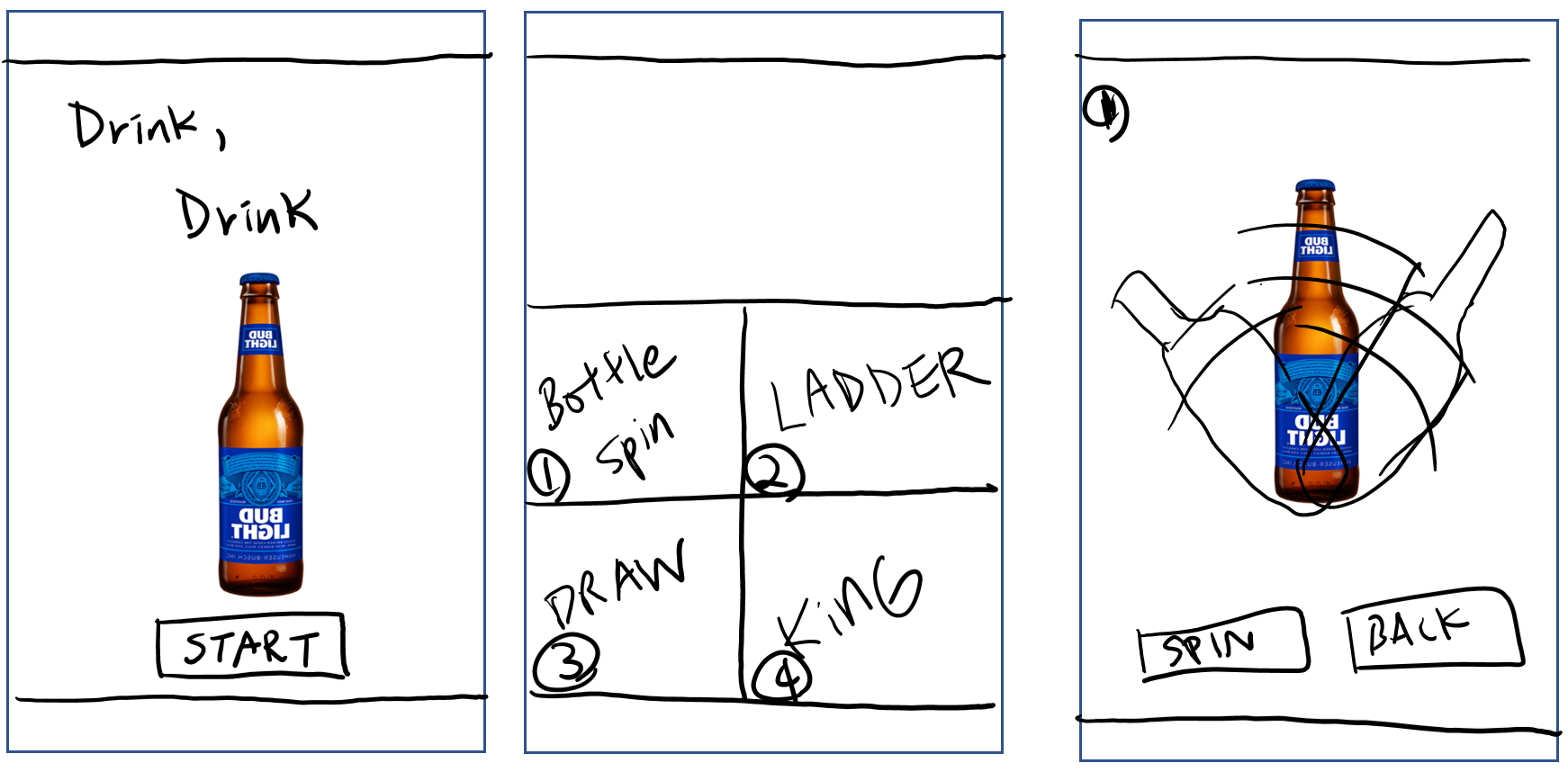
This application is intended to entertain users providing 4 different types of mini games to play, especially when they are drinking. Often, when people gather together and drink, they often play games to change into a friendly atmosphere. Playing board games are good too but these days people tend to use more of their smartphones. By creating this application, users can play the game more conveniently by just launching the application.

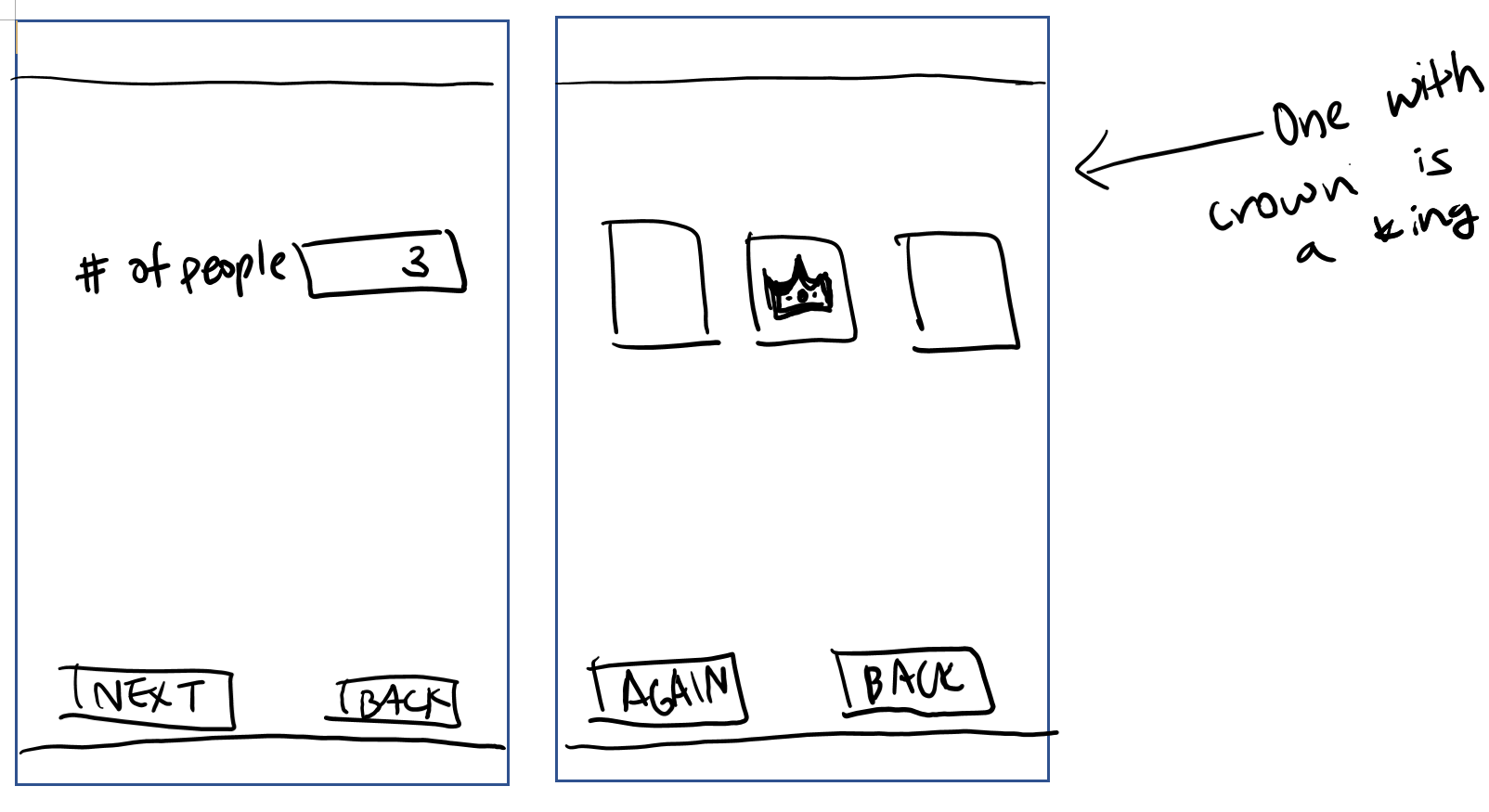
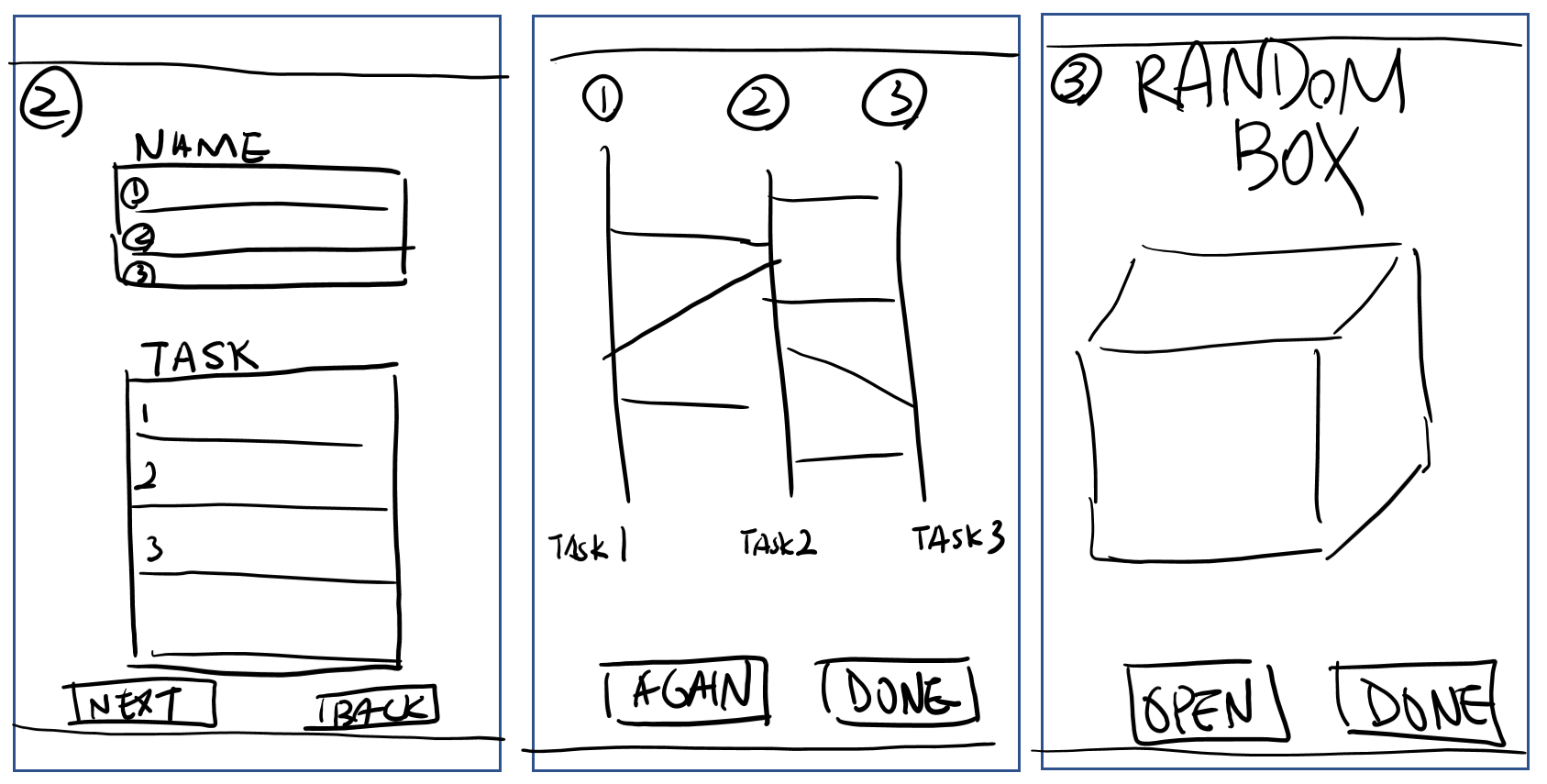
# BACKGROUND

Android app is a mobile [software](https://www.webopedia.com/TERM/S/software.html) [application](https://www.webopedia.com/TERM/A/application.html) developed for use on devices powered by Google's [Android](https://www.webopedia.com/TERM/A/Android_platform.html) platform. Android apps are available in the [Google Play](https://www.webopedia.com/TERM/G/google_play.html) Store (formerly known as the [Android Market](https://www.webopedia.com/TERM/A/android_market.html)), in the [Amazon Appstore](https://www.webopedia.com/TERM/A/appstore.html) and on various Android App-focused sites, and the apps can run on Android [smartphones](https://www.webopedia.com/TERM/S/smartphone.html), [tablets](https://www.webopedia.com/TERM/T/tablet_PC.html), [Google TV](https://www.webopedia.com/TERM/G/google_tv.html) and other devices.

This app is developed using Android Studio. The features include Intent, Activity java file(button listener), XML file, Widgets, . The intent we are using to create multiple activities in order to let four games and main page connected. Activity java file is used to write actions for each activities and inside Activity java file, Button “Onclick” listener is used to make actions once user click a button and use this function to jump to different activities as well. XML files are used to design the app. Widgets such as ImageView, button, TextView are used to design the app.

# SKETCHES





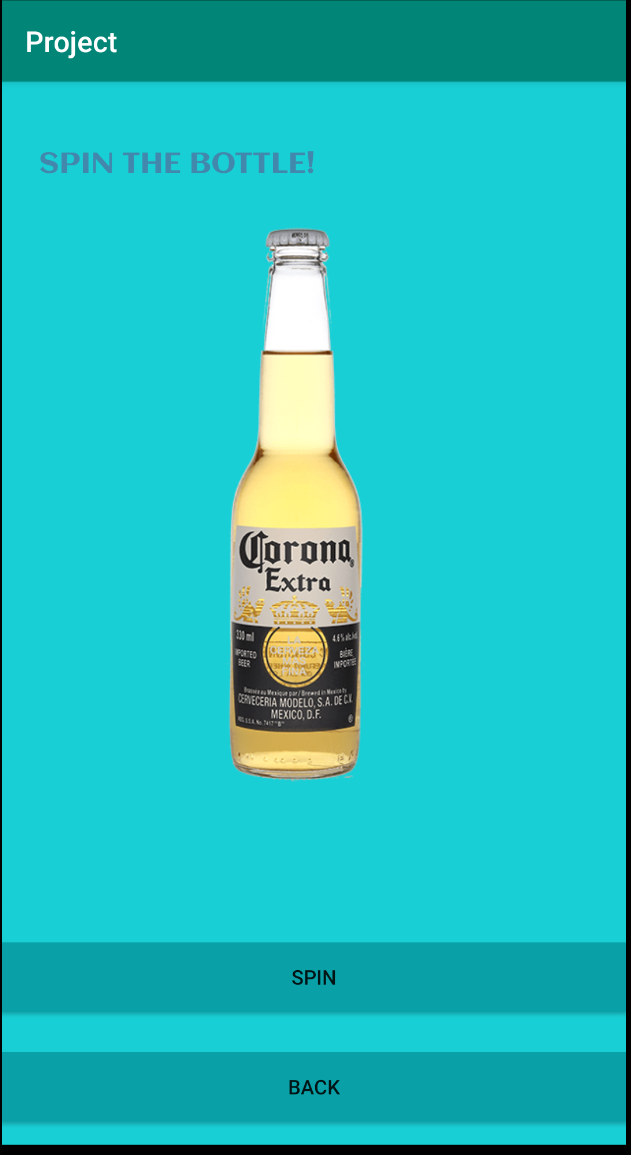
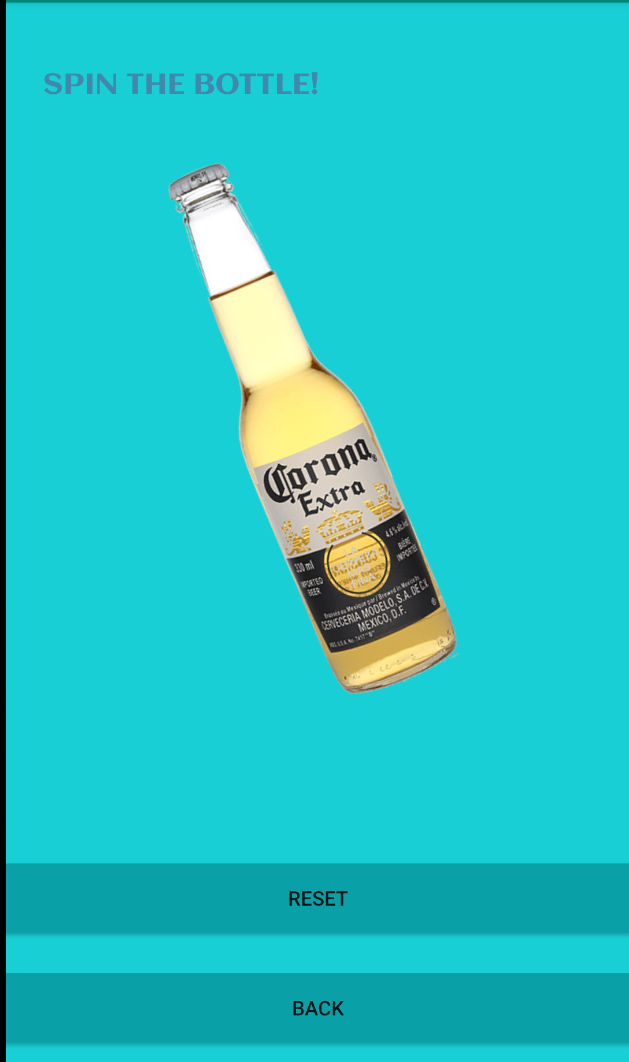
## 4. HOW TO USE

Open the application with a mobile device. On the Start screen, there are 4 selections of minigames you can choose from. Click on one of the games perfered. The game starts.

## 5. GAME DESCRIPTIONS



## 5.1 Bottle Spin

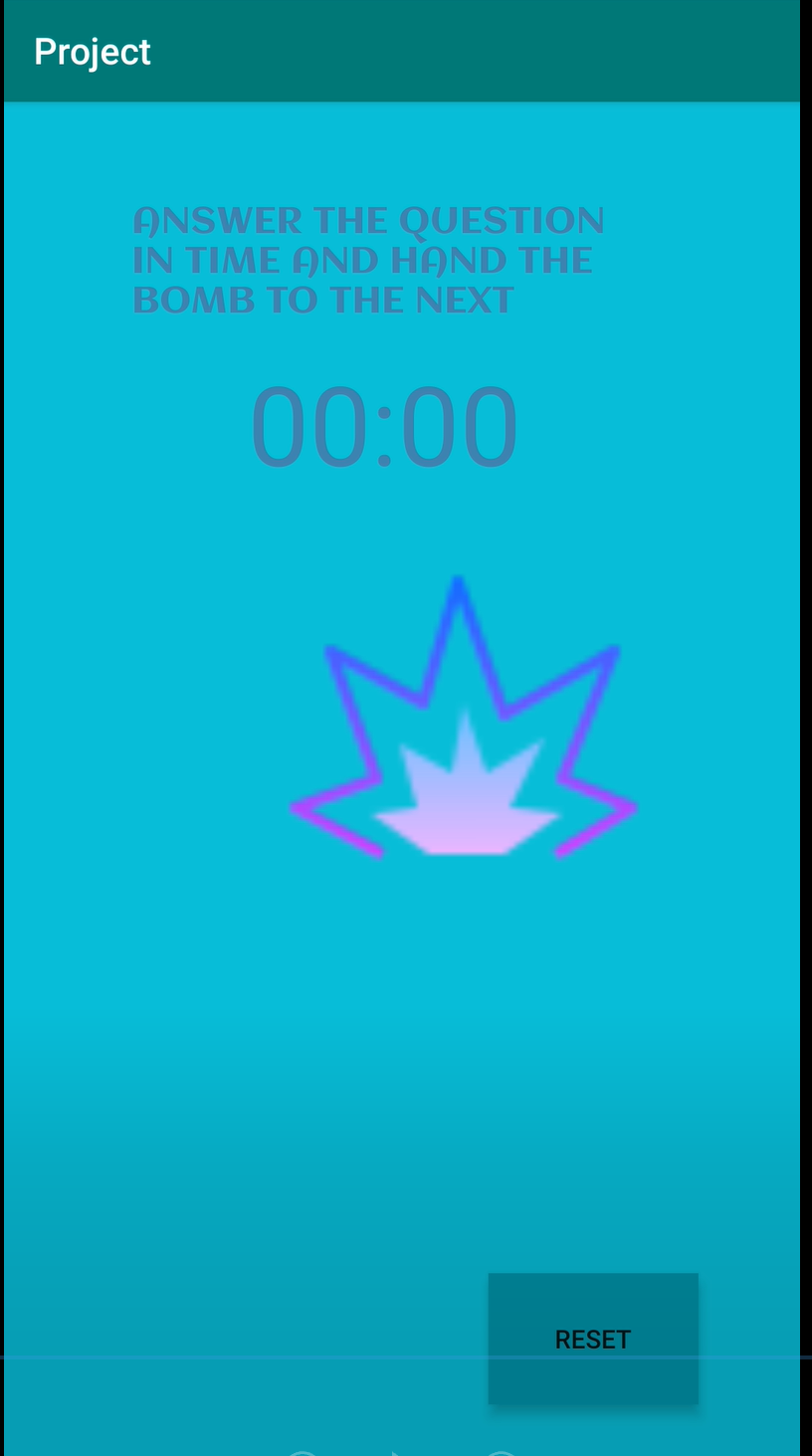
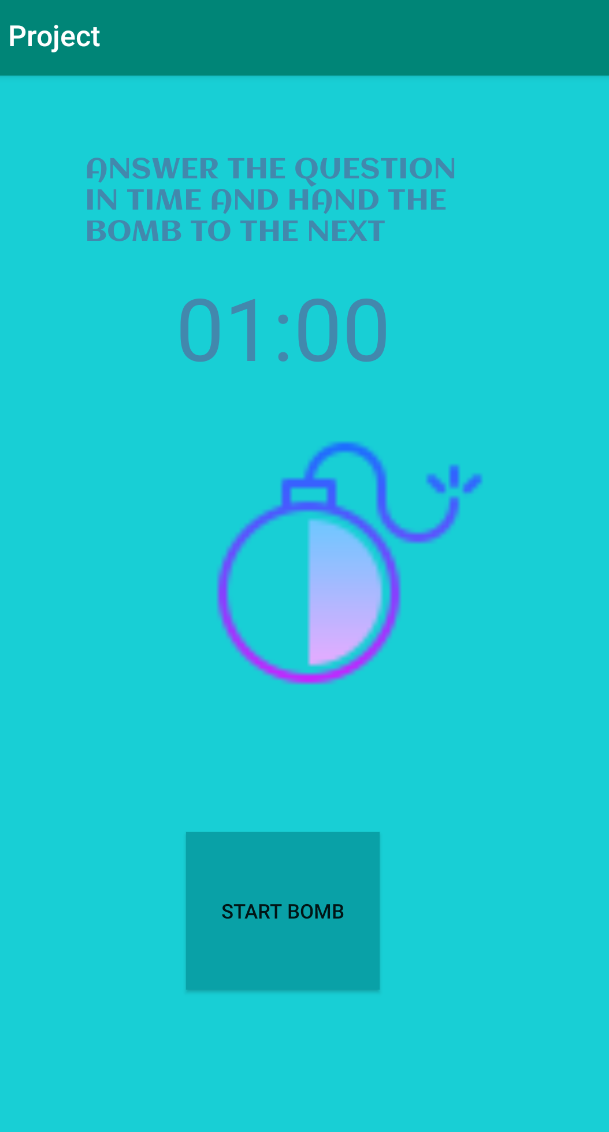
Bottle spinning game can be played by multiple players. It starts out by one player spinning the bottle. And when the bottle stops, the person is pointed drink. In Drinking Mini Games, one person clicks the button to spin the bottle then the bottle stops at a random location. The person that the bottle points must drink. 

Spin the Bottle game used RotateAnimation constructor and two buttons. One to spin the bottle and another button to go back to main menu. It contains six different parameters, starting from fromDegrees, toDegrees, pivotXtype, pivotXvalue, pivotYtype, and lastly, pivotYvalue. All the pivot types are relative to self because it is using the dimension of the object other than the parent. fromDegrees and toDegrees can be changed depends on where you want to start and how much.

The angle used a random number generator to get the random position or angle that bottle stops and by placing the angle into the constructor at toDegrees, the bottle stops at that random angle and leaving the fromDegree to 0 since the bottle starts at 0 degrees. It also used fillAfter(true) to apply the transformation to bottle after the animation is done.

## 5.2 60 Sec Time Attack

For this game, people have to choose a penalty such as “taking a shot”. Then choose a question to be answered. Then users can start the timer and pass on the phone. The person who has the phone has to answer the question and pass onto the next person. After 60 seconds, the timer ends and phone gives alert sound with vibration.



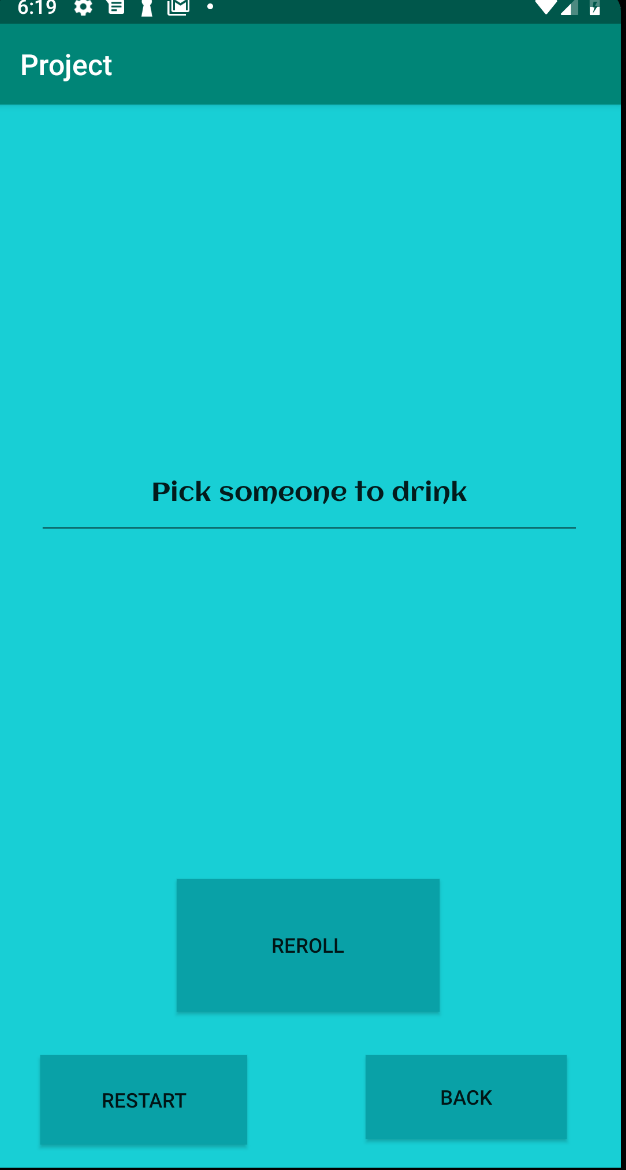
There is a Start Bomb button to start the bomb timer. When pressed start, there is “Pause”(not shown in the screenshot). When the time is up, only “Reset” button is shown to restart the game.

The time count down part, it uses CountDownTimer class, where it is first instantiated in startTimer() method. Count down time interval is set with 1000milisecond(which is 1 second) so it goes down by 1 second. Global variable START\_TIME\_IN\_MILIS was set as 60000 milliseconds(which is 60 second). The time was updated as it reduced. The currently left time was saved in mTimeLeftInMillis. It was formatted in “Minutes:Seconds”. The image of the bomb changes to an exploded one when the timer is finished.

For the alert sound, the MediaPlayer Class was used. It used to loop the sound and stop the sound. It also includes vibration when the time is over. For the vibration, added permission “android.permission.VIBRATE” to the manifest file for VIBRATOR\_SERVICE.

## 5.3 Radom Box

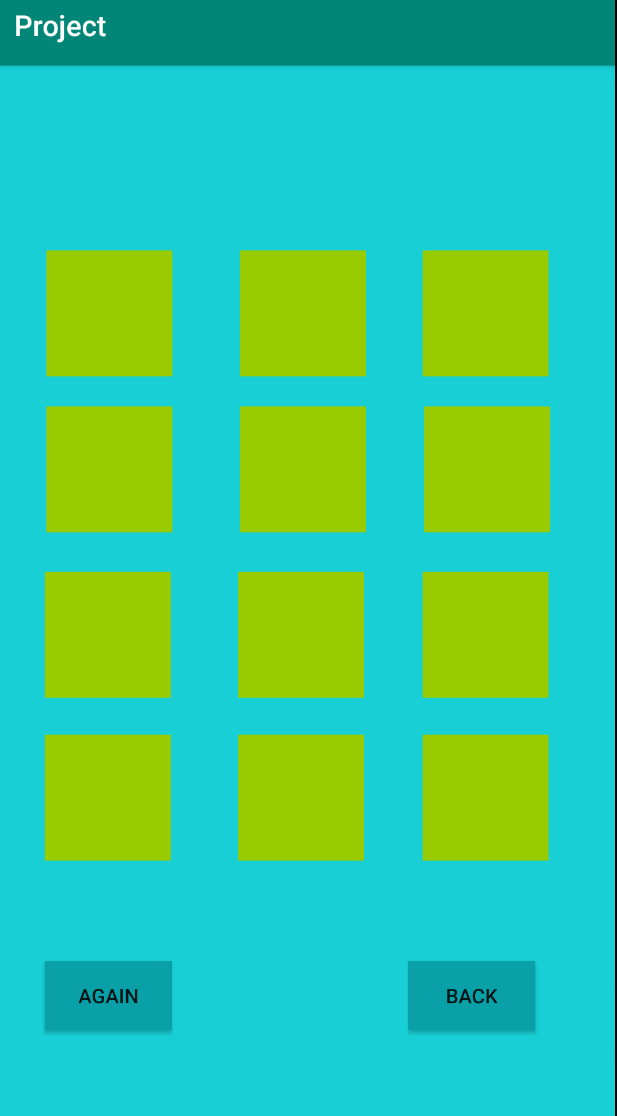
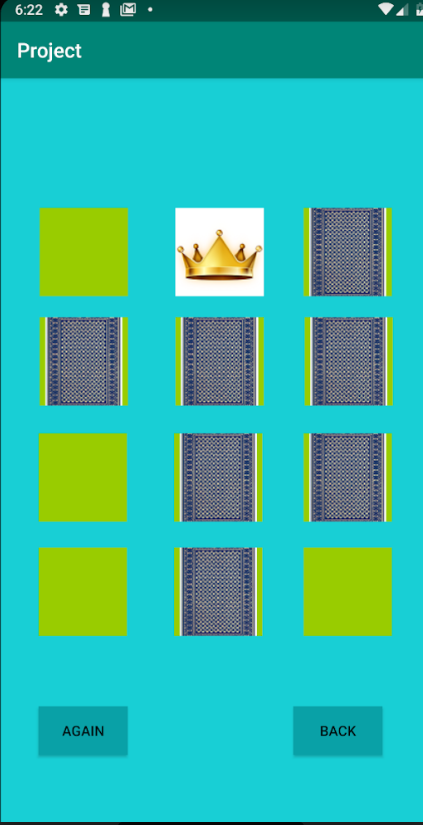
Random box game is drawing random tasks from the box. User can open the box by clicking a button and the box generates random quotes of instructions eg."take a shot","sing a song". Player must follow the instructions. If the player refuses to do the task, they must take extra shots.



Used String Array to save the possible penalties (ex.”Take a shot!”). Then, used Java’s Random class to use “rand.nextInt()” method to randomly select the position of the array and pick a quote when the button is pressed.

## 5.4 The Game of King

The game of king is a game that makes all the players to follow what King says. King can be chosen randomly in various ways. If players do not follow the King, they must be punished. So this game, chooses who to be the king by clicking cards.



The game uses 12 ImageButtons which are added into the constraint, so that these cards can display images and also have the button OnClick function. At the beginning, the default background is set to green which is in the Drawable color file. When users click one of the cards, the cards will have the opportunity to change into either card or the king sign by using ResourcesCompat.*getDrawable*

method to change the images in the ImageButton. Each time users click the “Again” button, they activity will be finished and restart again. The random number inside will be reassigned as well, placing the king sign in different spot.

# 6. REFERENCES

1. Beal, V. (n.d.). Android app. (Oct.2019) DOI=https://www.webopedia.com/TERM/A/android\_app.html.