**Simple TicTacToe Application**

**By Chong Wei Shen AKA LaggingNinja**

Contacts

Discord: 阿轩~ Wei Shen#8528

LinkedIn: Chong Wei Shen

**Table of content:**

[Program code 2](#_Toc69603966)

[MainActivity.java 2](#_Toc69603967)

[activitymain.xml 6](#_Toc69603968)

[Program Screenshots 12](#_Toc69603969)

# Program code

## MainActivity.java

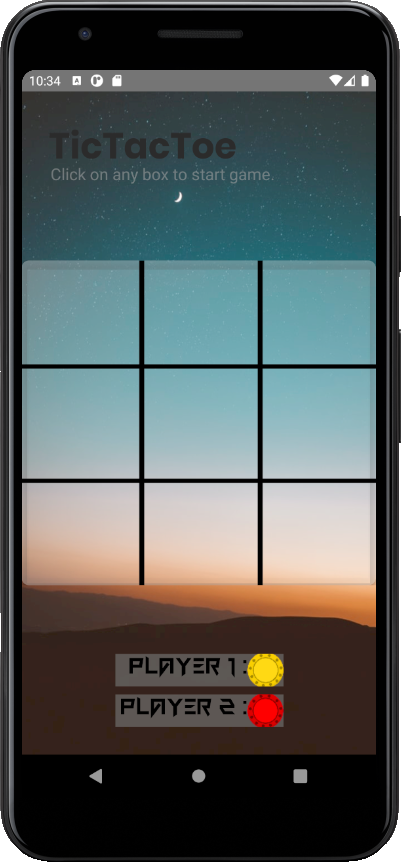
package com.example.tictactoe;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.GridLayout;  
import android.widget.ImageView;  
import android.widget.LinearLayout;  
import android.widget.TextView;  
  
public class MainActivity extends AppCompatActivity {  
  
 // 0 = yellow, 1 = red  
 int activePlayer = 0; // set active player to 0, means player 1  
 boolean gameIsActive = true; // yes, the game is active  
 // 2 means unplayed  
 int[] gameState = {2, 2, 2, 2, 2, 2, 2, 2, 2};  
 int[][] winningPositions = {{0,1,2}, {3,4,5}, {6,7,8}, {0,3,6}, {1,4,7}, {2,5,8}, {0,4,8}, {2,4,6}};  
 // tic tac toe game, which cell is marked  
  
 // what happen when the user click the image on the board  
 public void dropIn(View view) {  
  
 // here  
 ImageView counter = (ImageView) view;  
 // we need to know how many pictures click on the board, 9 pictures  
  
 int tappedCounter = Integer.*parseInt*(counter.getTag().toString());  
 // we need to know which picture click by the user on the board  
 // look at the tag value for each image view object  
 // tag 0 = first image to tag 8 9th image  
 // we have 9 images here for 3 rows x 3 columns cell  
  
 // if the user click the image within the board and  
 // the game status is still active  
 if (gameState[tappedCounter] == 2 && gameIsActive) {  
 gameState[tappedCounter] = activePlayer;  
  
 // if the user is active, player turn 1 or 2  
 counter.setTranslationY(-1000f); // run animation  
  
 //subtitle invisible once player start clicking on any box  
 TextView test = (TextView) findViewById(R.id.*subtitle*);  
 test.setVisibility(View.*INVISIBLE*);  
  
 //good luck text visible once player start clicking on any box  
 TextView test2 = (TextView) findViewById(R.id.*good\_luck*);  
 test2.setVisibility(View.*VISIBLE*);  
  
 if (activePlayer == 0) { // if player 1 turn  
 counter.setImageResource(R.drawable.*yellow*);  
 // get this image from res folder  
 activePlayer = 1;  
 // set active player to player 2 after player 1 finish  
 } else { // if player 2 turn  
 counter.setImageResource(R.drawable.*red*);  
 // get this image from res folder  
 activePlayer = 0; // set active player to player 2 finish  
 }  
  
  
 // here  
 counter.animate().translationYBy(1000f).rotation(360).setDuration(300);  
 // run animation again  
  
 // paste here  
 // use for loop here to find out the winners by checking the array elements of the winning position  
 // check if the user has clicked on the winning position or not  
 for (int[] winningPosition : winningPositions) {  
  
 if (gameState[winningPosition[0]] == gameState[winningPosition[1]] &&  
 gameState[winningPosition[1]] == gameState[winningPosition[2]] &&  
 gameState[winningPosition[0]] != 2) {  
  
 // Someone has won!, because they have click in the winning cells or positions  
 // written in the array  
 gameIsActive = false; // set game status to false, not active anymore, cannot play already  
 String winner = "Player 2"; // set winner to red, player 2  
  
 if (gameState[winningPosition[0]] == 0) { // if player 1 win the game  
 winner = "Player 1"; // if the winner is player 1  
 }  
  
 // here  
 TextView winnerMessage = (TextView) findViewById(R.id.*winnerMessage*);  
 // get the label from res folder, xml file  
 winnerMessage.setText(winner + " has won!"); // display the winner  
 GridLayout layout = (GridLayout) findViewById(R.id.*playAgainLayout*);  
 // get the layout manager from res folder, xml file  
 layout.setVisibility(View.*VISIBLE*); // set the layout to visible to display all the components  
  
 //make good luck invisible  
 TextView aa = (TextView) findViewById(R.id.*good\_luck*);  
 aa.setVisibility(View.*INVISIBLE*);  
 //if anyone wins, show congrats text  
 TextView test3 = (TextView) findViewById(R.id.*congrats*);  
 test3.setVisibility(View.*VISIBLE*);  
  
 } else {  
 boolean gameIsOver = true;  
 // game is over, no winner or losers  
  
 for (int counterState : gameState) {  
 // for loop to check no winner or losers  
 if (counterState == 2) gameIsOver = false;  
 } // end of for loop for draw condition, no winner or loser  
  
 if (gameIsOver) {  
 // if game is over already with no winner or losers  
 TextView winnerMessage = (TextView) findViewById(R.id.*winnerMessage*);  
 // get the textview label from res, xml file  
 winnerMessage.setText("It's a draw");  
 // display message  
 GridLayout layout = (GridLayout) findViewById(R.id.*playAgainLayout*);  
 // get the layout manager from res, xml file  
 layout.setVisibility(View.*VISIBLE*); // set it to visible to display all components  
  
 //make good luck invisible  
 TextView aa = (TextView) findViewById(R.id.*good\_luck*);  
 aa.setVisibility(View.*INVISIBLE*);  
 //if draw, show nice try text  
 TextView test3 = (TextView) findViewById(R.id.*nice\_try*);  
 test3.setVisibility(View.*VISIBLE*);  
  
 } // end if for game over  
  
 } // else if game over  
  
 } // end of for loop for checking winner  
 } // end if for game is active  
  
  
 }  
  
  
 // when the user click play again button  
 public void playAgain(View view) {  
  
 gameIsActive = true;  
 // change the status of the game to active again  
 GridLayout layout = (GridLayout) findViewById(R.id.*playAgainLayout*);  
 // get the linear layout manager from the res and xml file  
 layout.setVisibility(View.*INVISIBLE*);  
  
 //good luck text invisible once player click play again  
 TextView test2 = (TextView) findViewById(R.id.*good\_luck*);  
 test2.setVisibility(View.*INVISIBLE*);  
  
 //congrats text invisible once player click play again  
 TextView test3 = (TextView) findViewById(R.id.*congrats*);  
 test3.setVisibility(View.*INVISIBLE*);  
  
 //nice try text invisible once player click play again  
 TextView test4 = (TextView) findViewById(R.id.*nice\_try*);  
 test4.setVisibility(View.*INVISIBLE*);  
  
 //subtitle visible once user lick play again  
 TextView test = (TextView) findViewById(R.id.*subtitle*);  
 test.setVisibility(View.*VISIBLE*);  
  
 // turn it to become invisible again to hide all the components  
 activePlayer = 0;  
 // set player 1 as active, player 1 start the game first  
  
 for (int i = 0; i < gameState.length; i++) {  
 gameState[i] = 2; // why ? 3 cols and 3 rows  
 } // set all array elements value is 2, game is active, all 9 cells  
  
 GridLayout gridLayout = (GridLayout)findViewById(R.id.*gridLayout*);  
 // get the gridlayout from res folder, xml file  
  
 // use for loops to run 9 times, get 9 pictures from the res folder,  
 // display in the cell when the user click, i = 0 to 8  
 for (int i = 0; i< gridLayout.getChildCount(); i++) {  
 ((ImageView) gridLayout.getChildAt(i)).setImageResource(0);  
 }  
 }  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 }  
}

## activitymain.xml

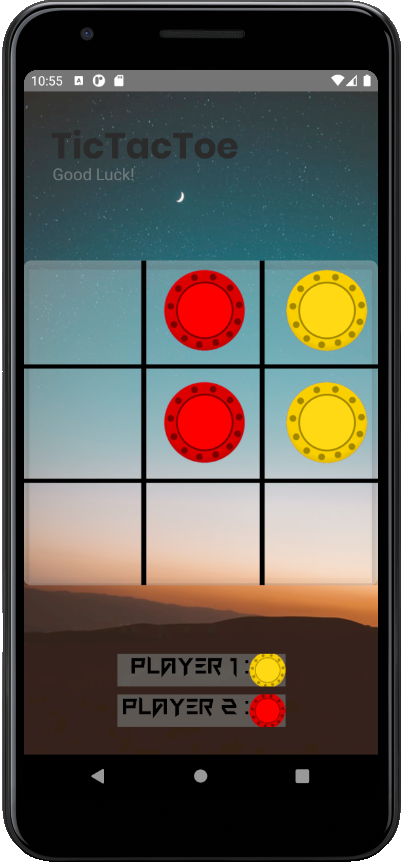
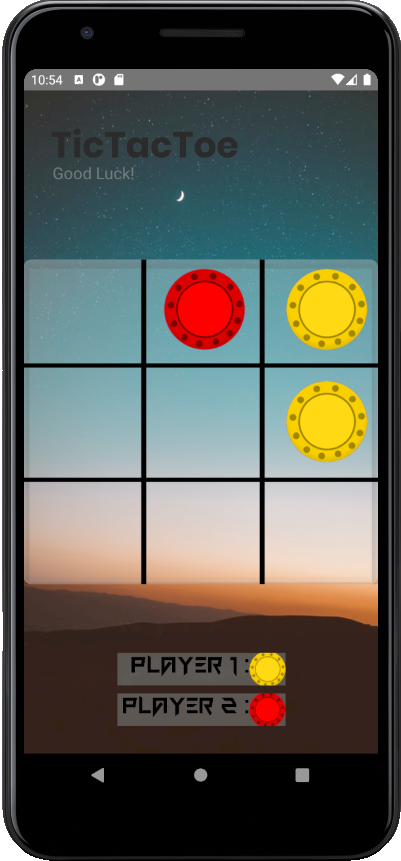
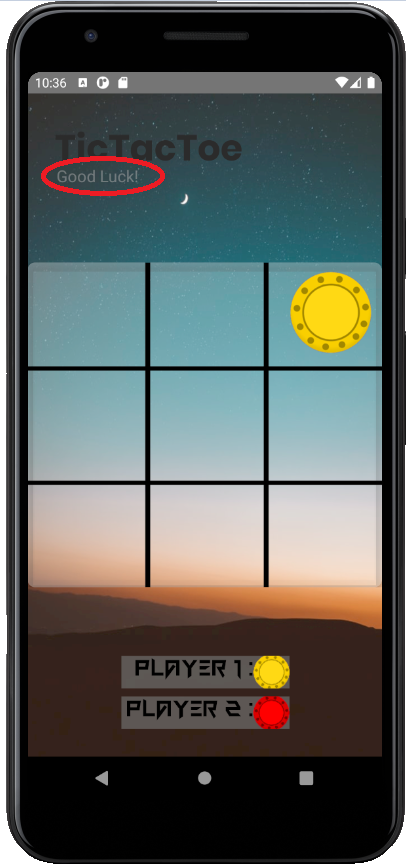
<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="@drawable/tttbg"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="295dp"  
 android:layout\_height="49dp"  
 android:layout\_margin="30dp"  
 android:fontFamily="@font/poppinsbold"  
 android:text="TicTacToe"  
 android:textColor="#2C2C2C"  
 android:textSize="40sp"  
 android:textStyle="bold"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <TextView  
 android:id="@+id/subtitle"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="32dp"  
 android:text="Click on any box to start game."  
 android:textColor="#8DA5A5A5"  
 android:textSize="18sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView" />  
  
 <TextView  
 android:id="@+id/good\_luck"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="32dp"  
 android:visibility="invisible"  
 android:text="Good Luck!"  
 android:textColor="#8DA5A5A5"  
 android:textSize="18sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView" />  
  
 <TextView  
 android:id="@+id/congrats"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="32dp"  
 android:visibility="invisible"  
 android:text="Wow, Congrats!"  
 android:textColor="#8DA5A5A5"  
 android:textSize="18sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView" />  
  
 <TextView  
 android:id="@+id/nice\_try"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="32dp"  
 android:visibility="invisible"  
 android:text="It was close! Nice Try."  
 android:textColor="#8DA5A5A5"  
 android:textSize="18sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView" />  
  
 <GridLayout  
 android:id="@+id/gridLayout2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="360dp"  
 android:layout\_alignParentEnd="true"  
 android:layout\_centerHorizontal="true"  
 android:layout\_centerVertical="true"  
 android:background="@drawable/board"  
 android:columnCount="3"  
 android:rowCount="3"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent">  
  
 <androidx.cardview.widget.CardView  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 app:cardBackgroundColor="#50FFFFFF"  
 app:cardCornerRadius="8dp"  
 app:cardElevation="6dp"></androidx.cardview.widget.CardView>  
  
 </GridLayout>  
  
 <GridLayout  
 android:id="@+id/gridLayout"  
 android:layout\_width="match\_parent"  
 android:layout\_height="360dp"  
 android:layout\_alignParentEnd="true"  
 android:layout\_centerHorizontal="true"  
 android:layout\_centerVertical="true"  
 android:background="@drawable/board"  
 android:columnCount="3"  
 android:rowCount="3"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent">  
  
  
 <ImageView  
 android:id="@+id/imageView"  
 android:layout\_width="90dp"  
 android:layout\_height="90dp"  
 android:layout\_row="0"  
 android:layout\_column="0"  
 android:layout\_marginLeft="10dp"  
 android:layout\_marginTop="10dp"  
 android:layout\_marginRight="10dp"  
 android:layout\_marginBottom="10dp"  
 android:onClick="dropIn"  
 android:tag="0" />  
  
 <ImageView  
 android:id="@+id/imageView2"  
 android:layout\_width="90dp"  
 android:layout\_height="90dp"  
 android:layout\_row="0"  
 android:layout\_column="1"  
 android:layout\_marginLeft="45dp"  
 android:layout\_marginTop="10dp"  
 android:onClick="dropIn"  
 android:tag="1" />  
  
  
 <ImageView  
 android:id="@+id/imageView3"  
 android:layout\_width="90dp"  
 android:layout\_height="90dp"  
 android:layout\_row="0"  
 android:layout\_column="2"  
 android:layout\_marginLeft="45dp"  
 android:layout\_marginTop="10dp"  
 android:onClick="dropIn"  
 android:tag="2" />  
  
 <ImageView  
 android:id="@+id/imageView4"  
 android:layout\_width="90dp"  
 android:layout\_height="90dp"  
 android:layout\_row="1"  
 android:layout\_column="0"  
 android:layout\_marginLeft="10dp"  
 android:layout\_marginTop="24dp"  
 android:onClick="dropIn"  
 android:tag="3" />  
  
 <ImageView  
 android:id="@+id/imageView5"  
 android:layout\_width="90dp"  
 android:layout\_height="90dp"  
 android:layout\_row="1"  
 android:layout\_column="1"  
 android:layout\_marginLeft="45dp"  
 android:layout\_marginTop="24dp"  
 android:onClick="dropIn"  
 android:tag="4" />  
  
 <ImageView  
 android:id="@+id/imageView6"  
 android:layout\_width="90dp"  
 android:layout\_height="90dp"  
 android:layout\_row="1"  
 android:layout\_column="2"  
 android:layout\_marginLeft="45dp"  
 android:layout\_marginTop="24dp"  
 android:onClick="dropIn"  
 android:tag="5" />  
  
 <ImageView  
 android:id="@+id/imageView7"  
 android:layout\_width="90dp"  
 android:layout\_height="90dp"  
 android:layout\_row="2"  
 android:layout\_column="0"  
 android:layout\_marginLeft="10dp"  
 android:layout\_marginTop="30dp"  
 android:onClick="dropIn"  
 android:tag="6" />  
  
 <ImageView  
 android:id="@+id/imageView8"  
 android:layout\_width="90dp"  
 android:layout\_height="90dp"  
 android:layout\_row="2"  
 android:layout\_column="1"  
 android:layout\_marginLeft="45dp"  
 android:layout\_marginTop="30dp"  
 android:onClick="dropIn"  
 android:tag="7" />  
  
 <ImageView  
 android:id="@+id/imageView9"  
 android:layout\_width="90dp"  
 android:layout\_height="90dp"  
 android:layout\_row="2"  
 android:layout\_column="2"  
 android:layout\_marginLeft="45dp"  
 android:layout\_marginTop="30dp"  
 android:onClick="dropIn"  
 android:tag="8" />  
  
 </GridLayout>  
  
 <GridLayout  
 android:id="@+id/playAgainLayout"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerHorizontal="true"  
 android:layout\_centerVertical="true"  
 android:background="#CB537CE3"  
 android:orientation="vertical"  
 android:padding="30dp"  
 android:visibility="invisible"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent">  
  
 <TextView  
 android:id="@+id/winnerMessage"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center\_horizontal"  
 android:fontFamily="@font/mo\_li"  
 android:text="New Text"  
 android:textSize="30sp" />  
  
 <Button  
 android:id="@+id/playAgainButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center\_horizontal"  
 android:fontFamily="@font/mo\_me"  
 android:onClick="playAgain"  
 android:text="Play Again" />  
  
  
 </GridLayout>  
  
 <androidx.constraintlayout.widget.ConstraintLayout  
 android:id="@+id/players"  
 android:layout\_width="400dp"  
 android:layout\_height="200dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/gridLayout2">  
  
 <TextView  
 android:id="@+id/player1"  
 style="@style/Smaller\_Drawable"  
 android:layout\_width="233dp"  
 android:layout\_height="45dp"  
 android:drawableRight="@drawable/yellow\_resized"  
 android:text="Player 1 :"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="@+id/players" />  
  
  
 <TextView  
 android:id="@+id/player2"  
 style="@style/Smaller\_Drawable"  
 android:layout\_width="233dp"  
 android:layout\_height="45dp"  
 android:drawableRight="@drawable/red\_resized"  
 android:text="Player 2 :"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/player1" />  
  
  
 </androidx.constraintlayout.widget.ConstraintLayout>  
  
</androidx.constraintlayout.widget.ConstraintLayout>

# Program Screenshots

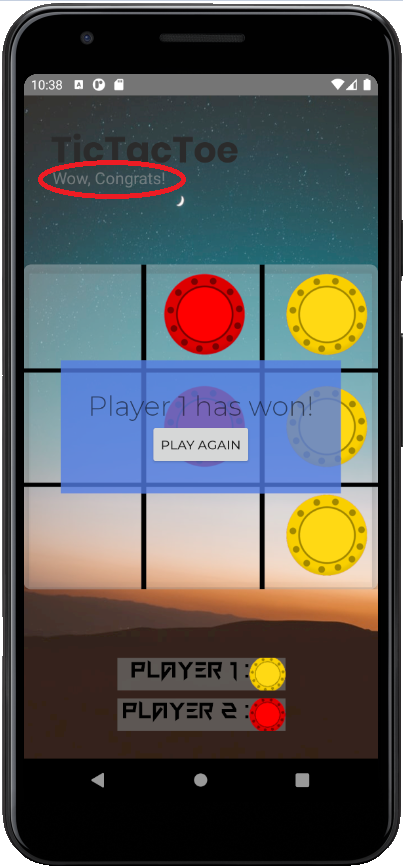
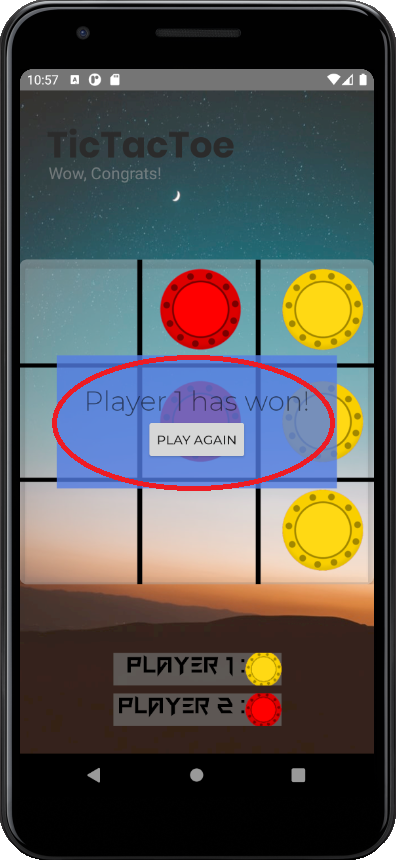
**Diagram 1.1**

As shown in **Diagram 1.1,** this is the user interface of the Tic-tac-toe game application. It has a simple good-looking background. A title “TicTacToe”, and simple instruction on how to start a game. There is a 3x3 grid layout for the game board and a card view background. The background color of the card view is white with 50% of transparency. Bellow the game board, there is two TextView to let the user know player 1 is yellow color, while player 2 is red color. The fonts used in this user interface are Poppinsbold, Courier New and FunGames.

**Diagram 1.2**

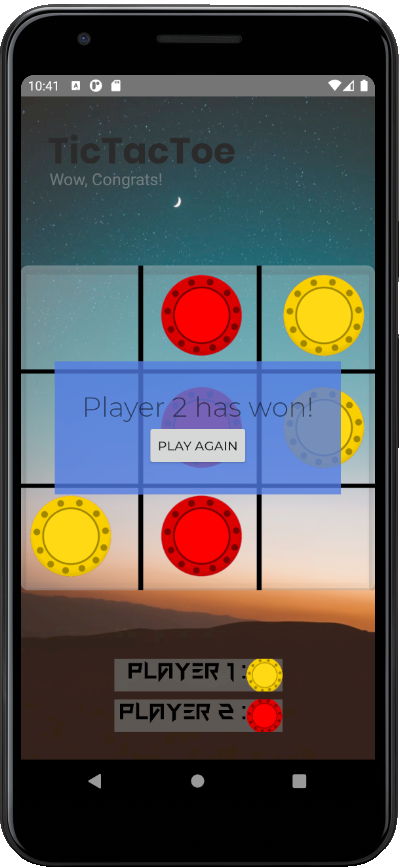
Once the user clicked on any box to start the game, the previous text will go invisible, while the text "Good Luck" will come visible **(Diagram 1.2)**. Player 1 which is yellow color will start first and then follow up by player 2. Pictures above shown an example of two players playing the game. The rules of tic-tac-toe game are: The first player to get 3 of her marks in a row (up, down, across or diagonally) is the winner. When all 9 squares are full without a winner, the game is over with a draw.

**Diagram 1.3 Diagram 1.4**

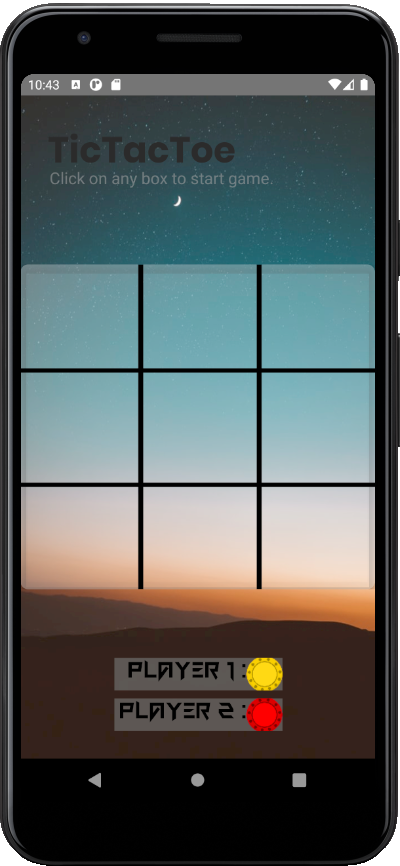
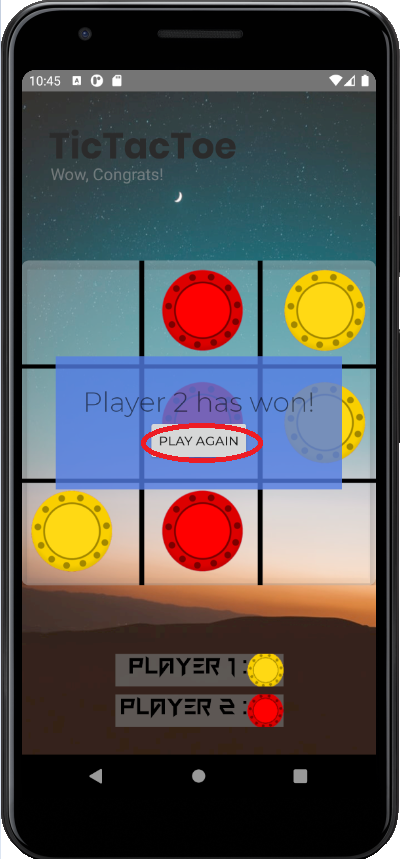


**Diagram 1.3** shows that when the yellow color player won the game. A message will pop up and say “Player 1 has won!” with a play again button. This is a grid layout, with the background of 60% transparency blue color. The font used in this layout are two different fonts for better design. At the same time, the “Good Luck” text will go invisible and the text “Wow, Congrats” will show whenever a player won the game (**Diagram 1.4)**.

**Diagram 1.5**

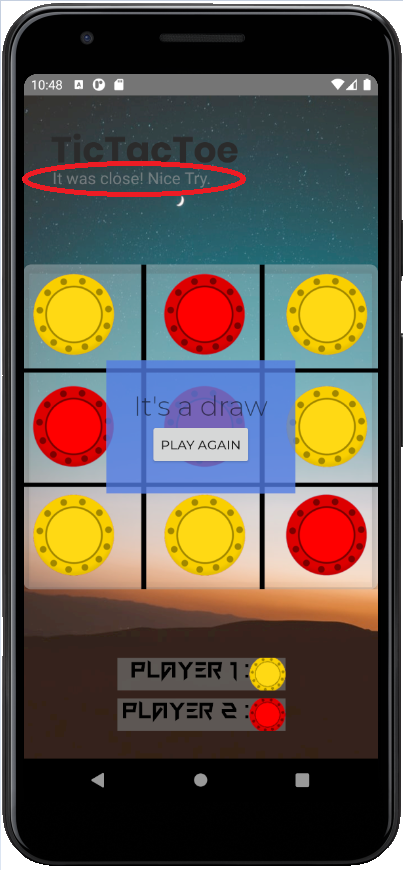
**Diagram 1.5** shows that same pop up will show once player 2 (red color) won the game. 

**Diagram 2.1 Diagram 2.2**



**Diagram 2.1** and **Diagram 2.2** shows what will happen when the user click on the “PLAY AGAIN” button. The system will reset the game and all the red and yellow marks will go back to invisible. The “Wow, Congrats” text will also back to the text that tell the user how to start a game. User will be able to play again by clicking at any box starting with player 1.

**Diagram 2.3**

Lastly, this screenshot is to show that what will happen when the game is drawn. If the game ends without a winner, the system will indicate the game result as a draw. The text “Good Luck” will go invisible and a text of “It was close! Nice Try” will be visible **(Diagram 2.3)**. In the grid layout, it will tell the user It’s a draw and the same play again button is shown.

**END OF REPORT**