

Aim: To make a stopwatch app in android studio.

Algorithm:

1. Open your Android Studio . Click on Start a New Android Studio Project .Give Application Name Login Form and leave other fields as it is, then click NEXT.

2. Select the Empty Activity and click NEXT.

3. Now we have activity_main.xml in which we have to make the layout of our app.

4. Add three button for starting , pausing and resetting the stopwatch respectively , one textview for showing the countdown. Thus our layout is finished.

5. Now in our MainActivity.java file add variable {

```
private Chronometer chronometer;  
private boolean running;  
private long pauseOffSet;
```

```
}
```

for the respective field we have selected in xml file and connect them to the xml file by the help of declared id.

6. Make a startChronometer function which will start the chronometer

7. Make a pauseChronometer function which will pause the chronometer

8. Make a resetChronometer function which will reset the chronometer

9. Thus our app is finished.

Activity_main.xml:-

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
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:orientation="vertical"
    android:gravity="center"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <!-- Setting the Chronometer -->
    <Chronometer
        android:id="@+id/chronometer"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="30sp" />

    <!-- Setting the Start button -->
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="startChronometer"
        android:text="Start" />

    <!-- Setting the Pause Button -->
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Pause"
        android:onClick="pauseChronometer"/>

    <!-- Setting the Reset Button -->
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Reset"
        android:onClick="resetChronometer"/>

</LinearLayout>

```

MainActivity.java:

```

package com.example.stopwatch;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.os.SystemClock;
import android.view.View;
import android.widget.Chronometer;

```

```

public class MainActivity extends AppCompatActivity {

    //Declaring the variable for the program.
    private Chronometer chronometer;
    private boolean running;
    private long pauseOffset;

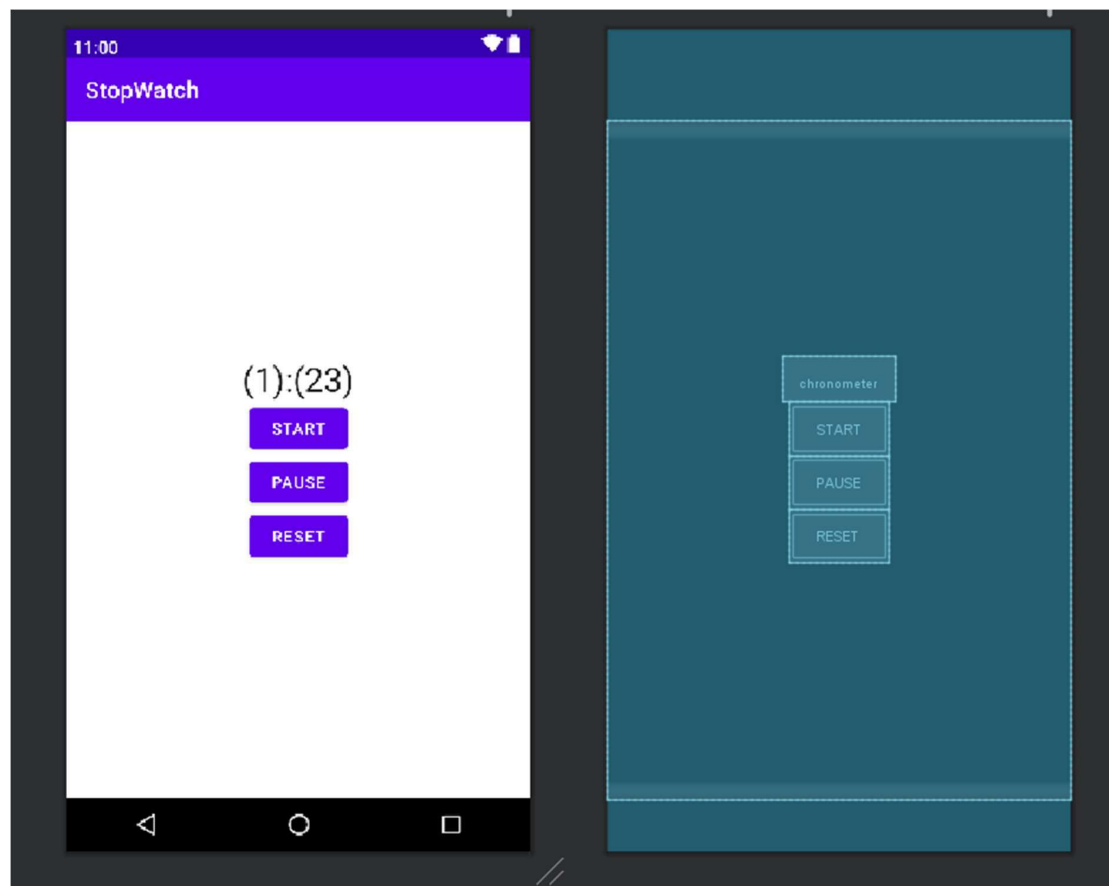
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        chronometer = findViewById(R.id.chronometer);
        chronometer.setFormat("Time: %s");
    }

    //Creating the start Chronometer function for starting the
    Timer.
    public void startChronometer(View view) {
        if(!running){
            chronometer.setBase(SystemClock.elapsedRealtime() -
pauseOffset);
            chronometer.start();
            running = true;
        }
    }
    //Creating the pause Chronometer function for Pausing the Timer.
    public void pauseChronometer(View view) {
        if (running){
            chronometer.stop();
            pauseOffset = SystemClock.elapsedRealtime() -
chronometer.getBase();
            running = false;
        }
    }

    //Creating the reset Chronometer function for reseing the Timer.
    public void resetChronometer(View view) {
        chronometer.setBase(SystemClock.elapsedRealtime());
        pauseOffset = 0;
    }
}

```

Output1(Design and Blueprint):-



Output2:-

