ASSIGNMENT NO - 5

Aim : write a simple android program for a functioning stopwatch and display running time (button as Start , Reset and Stop).

- **Description:**

  I have to create an android application which will show the stopwatch which consist of timer in the form of second , minute and hour with some basic functionality like start , reset, stop . the steps are as follows:

  - Create a new project
  - Name them as stopwatch app
  - It will generate activity_main.xml and mainactivity.java file.
  - Then initialise three button as start, stop , reset.

- **Expected input:**

  My expected input is clicking the start, stop, reset button.

- **Expected output:**

  My expected output will be the stopwatch which will show the time elapsed from the start time as well as show the other functionality.

- **Discussion:**

  In this program I have created three button which is start , reset , pause. Whose functionality is decided based on the defined function like when I press the start button it will start the stopwatch , like show when I will press the reset button it will reset the timer to zero value and then again start the stopwatch from the zero value, stop button stop the stopwatch at the time when it has been pressed.

- **Error checking :**

  I have done the error checking there is no any error the program is running well and fine.

- **Assumption taken:**

There must be system clock installed in the system.

- **Scope for improvement:**

  Its user interface as well as option should be improved.

- **Additional feature :**

  It will show the real stopwatch with some functionality like start stop as well as reset .

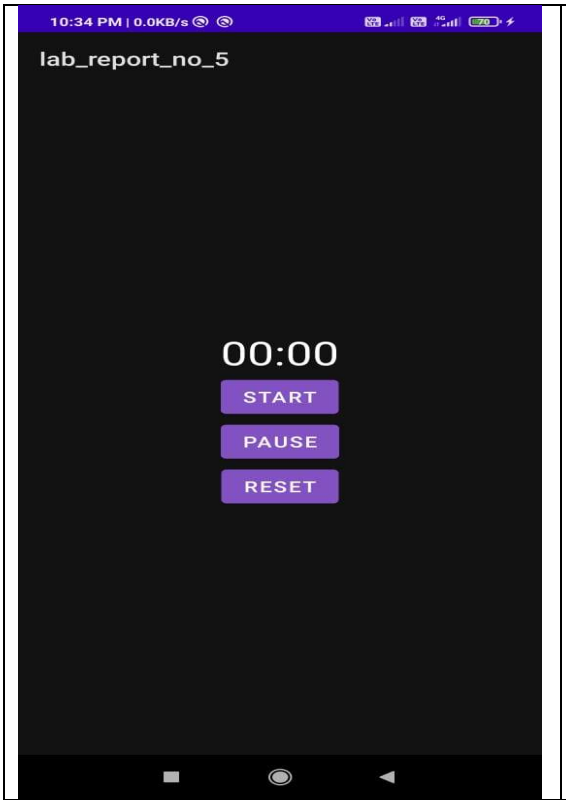
Activity_main.xml

```
<?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:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    tools:context=".MainActivity">

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

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Start"
        android:onClick="startChronometer"/>

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Pause"
        android:onClick="pauseChronometer"/>
<Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Reset"
        android:onClick="resetChronometer"/>
</LinearLayout>
```

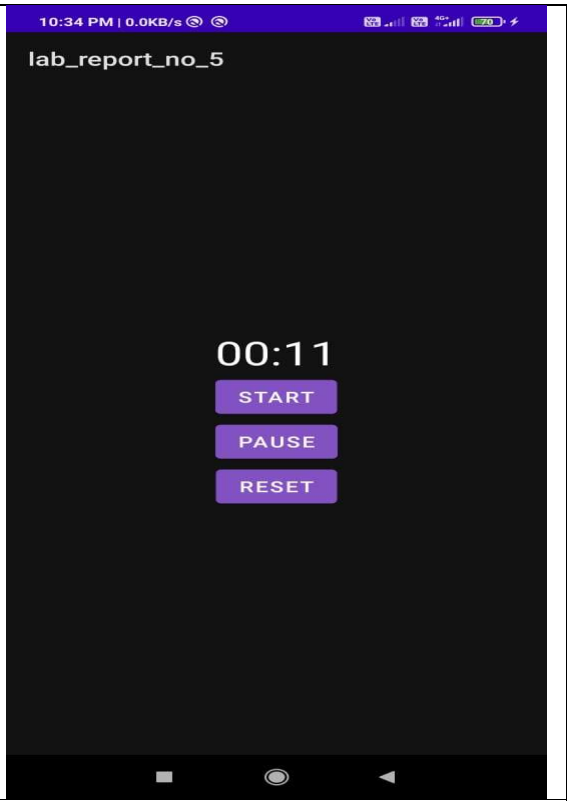
Mainactivity.java

```
package com.example.lab_report_no_5;

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 {

    private Chronometer chronometer;
    private long pauseOffset;
    private boolean running;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        chronometer = findViewById(R.id.chronometer);
    }
    public void startChronometer(View v) {
        //first check if the chronometer is running or not
        if(!running){
            chronometer.setBase(SystemClock.elapsedRealtime()
- pauseOffset);
            chronometer.start();
            running = true;
        }

    }

    public void pauseChronometer(View v){
        if(running){
            chronometer.stop();
            pauseOffset = SystemClock.elapsedRealtime() -
chronometer.getBase();
            running = false;
        }

    }
    public void resetChronometer(View v){
        chronometer.setBase(SystemClock.elapsedRealtime());
        pauseOffset = 0;
    }
}
```

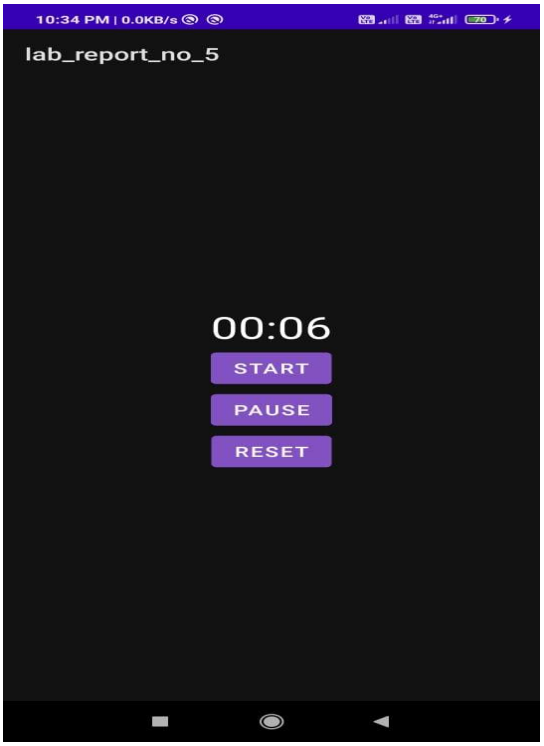
Output



Before start



After start



On clicking pause