

Department of CSE

SSN College of Engineering

Vishakan Subramanian - 18 5001 196 - Semester VII

13 September 2021

UCS 1711 - Mobile Application Development Lab

Exercise 5: Progress Bar and Alarm Clock using Multithreading

Aim:

To develop Android applications for:

- Progress Bar Simulation
- Simulation for Suspending the UI
- Alarm Clock

Use **multithreading** concepts to implement the progress bar.

Code - Progress Bar: Main Activity:

```
1 package com.example.progressbar;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.app.ProgressDialog;
6 import android.os.Bundle;
7 import android.os.Handler;
8 import android.view.View;
9 import android.widget.Button;
10 import android.widget.EditText;
11 import android.widget.ProgressBar;
12
13 public class MainActivity extends AppCompatActivity {
14
15     @Override
16     protected void onCreate(Bundle savedInstanceState) {
17         super.onCreate(savedInstanceState);
18         setContentView(R.layout.activity_main);
19
20         final EditText durationText = findViewById(R.id.durationText);
21         final Button startButton = findViewById(R.id.startButton);
22         final Button sleepButton = findViewById(R.id.sleepButton);
23         final ProgressBar progressBar = findViewById(R.id.progressBar);
24         final Handler handler = new Handler();
25
26         sleepButton.setOnClickListener(new View.OnClickListener() {
27             @Override
28             public void onClick(View v) {
29                 final ProgressDialog progressDialog = new ProgressDialog(
30                     MainActivity.this);
31                 progressDialog.setTitle("\uD83D\uDD15 Sleep in Progress...");
32                 progressDialog.setMessage("Sleeping \uD83D\uDCA4");
33                 progressDialog.show();
34
35                 int duration = Integer.parseInt(durationText.getText().
36                     toString());
37
38                 //Dismiss the progressDialog after waiting for the sleep
39                 duration
40                     new Handler().postDelayed(new Runnable() {
41                         @Override
42                         public void run() {
43                             progressDialog.dismiss();
44                         }
45                     }, duration * 1000);
46             }
47         });
48     }
49 }
```

```

44     });
45
46     startButton.setOnClickListener(new View.OnClickListener() {
47         @Override
48         public void onClick(View v) {
49             startButton.setEnabled(false);
50             startButton.setText("IN PROGRESS...");
51
52             //Runs in a separate thread
53             new Thread(new Runnable() {
54                 @Override
55                 public void run() {
56                     for(int i = 0; i <= 100; i+=5){
57                         if(i == 100){
58                             //Re-enable the button once the loop is
over
59                             handler.post(new Runnable() {
60                                 @Override
61                                 public void run() {
62                                     startButton.setEnabled(true);
63                                     startButton.setText("START");
64                                 }
65                             });
66                         }
67
68                         progressBar.setProgress(i);
69
70                         //Wait for 100ms before the next iteration
71                         try{
72                             Thread.sleep(100);
73                         } catch (Exception e){
74                             e.printStackTrace();
75                         }
76                     }
77                 }
78             }).start();
79         }
80     });
81 }
82 }

```

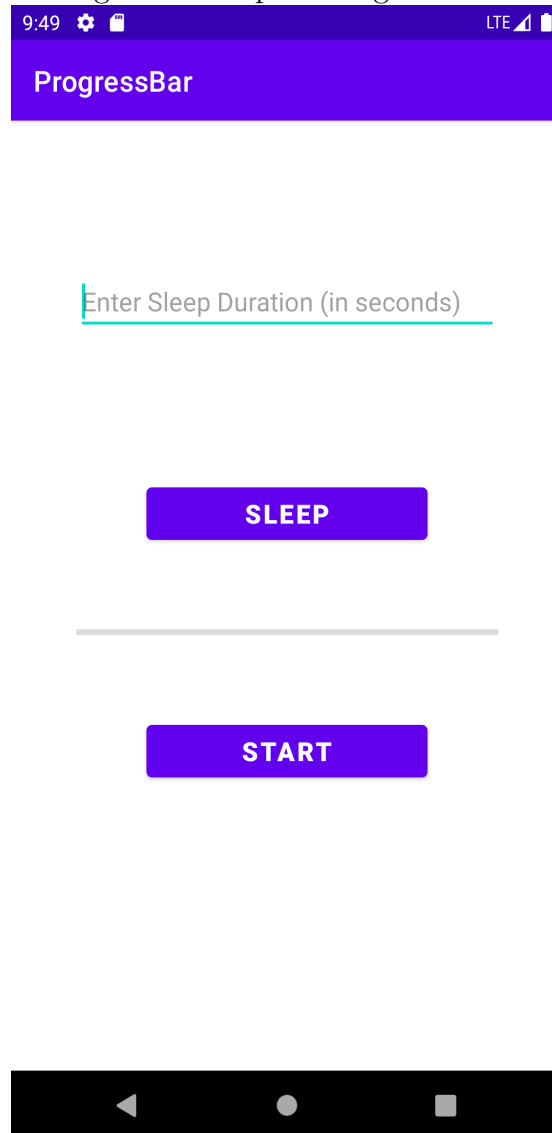
Code - Progress Bar: Main Activity Layout:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://
  schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   tools:context=".MainActivity">
8
9   <EditText
10      android:id="@+id/durationText"
11      android:layout_width="300dp"
12      android:layout_height="wrap_content"
13      android:layout_marginTop="100dp"
14      android:ems="10"
15      android:gravity="bottom"
16      android:hint="Enter Sleep Duration (in seconds)"
17      android:inputType="number"
18      android:minHeight="48dp"
19      app:layout_constraintEnd_toEndOf="parent"
20      app:layout_constraintStart_toStartOf="parent"
21      app:layout_constraintTop_toTopOf="parent" />
22
23   <Button
24      android:id="@+id/sleepButton"
25      android:layout_width="200dp"
26      android:layout_height="wrap_content"
27      android:layout_marginTop="100dp"
28      android:text="SLEEP"
29      android:textSize="18sp"
30      android:textStyle="bold"
31      app:layout_constraintEnd_toEndOf="parent"
32      app:layout_constraintStart_toStartOf="parent"
33      app:layout_constraintTop_toBottomOf="@+id/durationText" />
34
35   <ProgressBar
36      android:id="@+id/progressBar"
37      style="?android:attr/progressBarStyleHorizontal"
38      android:layout_width="300dp"
39      android:layout_height="wrap_content"
40      android:layout_marginTop="50dp"
41      app:layout_constraintEnd_toEndOf="parent"
42      app:layout_constraintStart_toStartOf="parent"
43      app:layout_constraintTop_toBottomOf="@+id/sleepButton" />
44
45   <Button
46      android:id="@+id/startButton"
```

```
47     android:layout_width="200dp"
48     android:layout_height="wrap_content"
49     android:layout_marginTop="50dp"
50     android:text="START"
51     android:textSize="18sp"
52     android:textStyle="bold"
53     app:layout_constraintEnd_toEndOf="parent"
54     app:layout_constraintStart_toStartOf="parent"
55     app:layout_constraintTop_toBottomOf="@+id/progressBar" />
56 </androidx.constraintlayout.widget.ConstraintLayout>
```

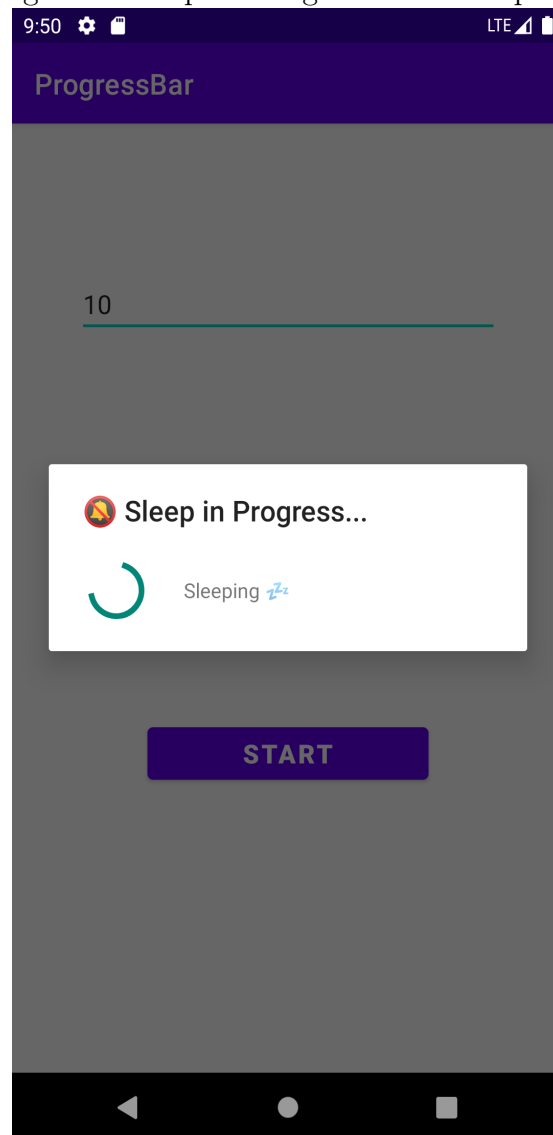
Output: Progress Bar:

Figure 1: Output: Progress Bar.



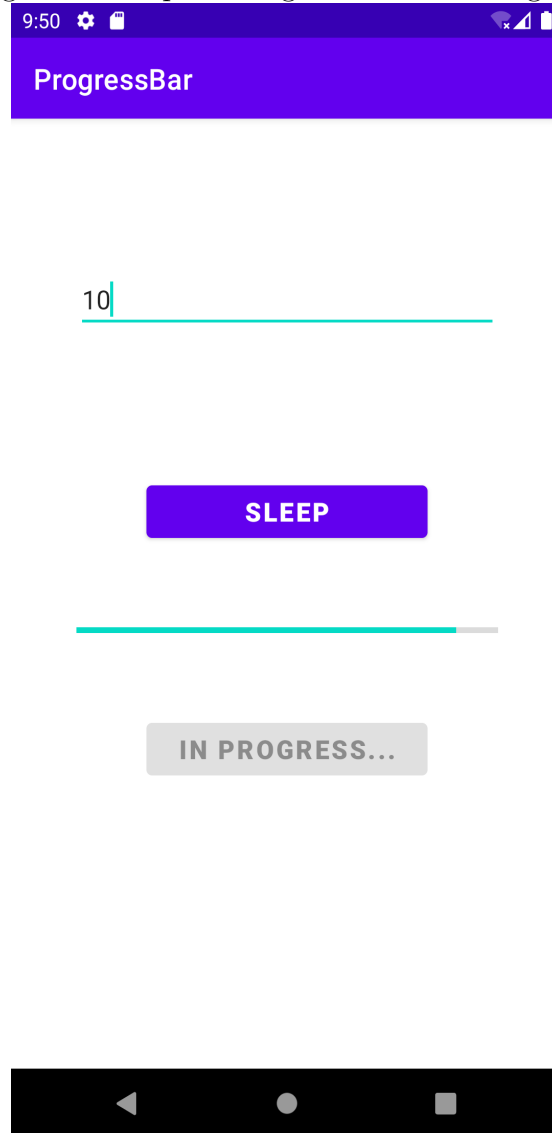
Output: Progress Bar - Sleeping:

Figure 2: Output: Progress Bar - Sleeping.



Output: Progress Bar - In Progress:

Figure 3: Output: Progress Bar - In Progress.



Code - Alarm Clock: Main Activity:

```
1 package com.example.alarmclock;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.app.AlarmManager;
6 import android.app.PendingIntent;
7 import android.content.Context;
8 import android.content.Intent;
9 import android.os.Bundle;
10 import android.util.Log;
11 import android.view.View;
12 import android.widget.TextView;
13 import android.widget.TimePicker;
14 import android.widget.Toast;
15 import android.widget.ToggleButton;
16
17 import java.text.DateFormat;
18 import java.util.Calendar;
19
20
21 public class MainActivity extends AppCompatActivity {
22
23     @Override
24     protected void onCreate(Bundle savedInstanceState) {
25         super.onCreate(savedInstanceState);
26         setContentView(R.layout.activity_main);
27
28         final TextView alarmText = findViewById(R.id.alarmText);
29         final TimePicker timePicker = findViewById(R.id.timePicker);
30         final AlarmManager alarmManager = (AlarmManager) getSystemService(
Context.ALARM_SERVICE);
31         final ToggleButton alarmToggle = findViewById(R.id.alarmToggle);
32         final Intent intent = new Intent(getApplicationContext(),
AlarmReceiver.class);
33         final PendingIntent pendingIntent = PendingIntent.getBroadcast(
MainActivity.this, 1, intent, PendingIntent.FLAG_UPDATE_CURRENT);
34
35         alarmToggle.setOnClickListener(new View.OnClickListener() {
36             @Override
37             public void onClick(View v) {
38                 if(((ToggleButton) v).isChecked()){
39                     //Turn the alarm on
40
41                     Calendar calendar = Calendar.getInstance();
42
43                     calendar.set(Calendar.HOUR_OF_DAY, timePicker.getHour
());
44                 }
45             }
46         });
47     }
48 }
```

```

44         calendar.set(Calendar.MINUTE, timePicker.getMinute());
45         calendar.set(Calendar.SECOND, 0);
46
47         //If time in the past was chosen, move alarm to the
next day
48         if(calendar.before(Calendar.getInstance())) {
49             calendar.add(Calendar.DATE, 1);
50         }
51
52         Log.i("Time ", calendar.getTime().toString());
53         alarmText.setText("Alarm Set For: " + DateFormat.
getTimeInstance(DateFormat.SHORT).format(calendar.getTime()) + " \uD83D
\uDD14");
54
55
56         alarmManager.setRepeating(AlarmManager.RTC_WAKEUP,
calendar.getTimeInMillis(), 10000, pendingIntent);
57         Toast.makeText(MainActivity.this, "ALARM SET!", Toast.
LENGTH_SHORT).show();
58
59         } else{
60             //Turn the alarm off
61
62             alarmManager.cancel(pendingIntent);           //Cancel the
pending alarm intent
63
64             if(AlarmReceiver.ringtone != null && AlarmReceiver.
ringtone.isPlaying()){
65                 //If the ringtone is playing i.e. the alarm is
ringing and the ringtone object has been set
66                 AlarmReceiver.ringtone.stop(); //Stop the
ringtone started from the alarm receiver
67             }
68
69             Toast.makeText(MainActivity.this, "ALARM OFF!", Toast.
LENGTH_SHORT).show();
70             alarmText.setText("Alarm Not Set \uD83D\uDEAB");
71         }
72     }
73 });
74 }
75 }

```

Code - Alarm Clock: Alarm Receiver Class:

```
1 package com.example.alarmclock;
2
3
4 import android.content.BroadcastReceiver;
5 import android.content.Context;
6 import android.content.Intent;
7 import android.media.Ringtone;
8 import android.media.RingtoneManager;
9 import android.widget.Toast;
10
11
12
13 public class AlarmReceiver extends BroadcastReceiver {
14     public static Ringtone ringtone = null;    //Important - Required to
        stop the ringtone from the main activity
15
16     @Override
17     public void onReceive(Context context, Intent intent) {
18         Toast.makeText(context, "Alarm Time!", Toast.LENGTH_LONG).show();
19
20         ringtone = RingtoneManager.getRingtone(context, RingtoneManager.
            getDefaultUri(RingtoneManager.TYPE_ALARM));
21         ringtone.play();
22     }
23 }
```

Code - Alarm Clock: Main Activity Layout:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://
  schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   tools:context=".MainActivity">
8
9   <TextView
10     android:id="@+id/textView"
11     android:layout_width="wrap_content"
12     android:layout_height="wrap_content"
13     android:layout_marginTop="20dp"
14     android:text="ALARM CLOCK"
15     android:textColor="?android:attr/textColorPrimary"
16     android:textSize="24sp"
17     android:textStyle="bold"
18     app:layout_constraintEnd_toEndOf="parent"
19     app:layout_constraintStart_toStartOf="parent"
20     app:layout_constraintTop_toTopOf="parent" />
21
22   <TimePicker
23     android:id="@+id/timePicker"
24     android:layout_width="wrap_content"
25     android:layout_height="wrap_content"
26     android:layout_marginTop="50dp"
27     android:headerBackground="@android:color/holo_purple"
28     android:numbersSelectorColor="@android:color/holo_purple"
29     app:layout_constraintEnd_toEndOf="parent"
30     app:layout_constraintHorizontal_bias="0.0"
31     app:layout_constraintStart_toStartOf="parent"
32     app:layout_constraintTop_toBottomOf="@+id/textView" />
33
34   <ToggleButton
35     android:id="@+id/alarmToggle"
36     android:layout_width="150dp"
37     android:layout_height="wrap_content"
38     android:backgroundTint="@android:color/holo_purple"
39     android:checked="false"
40     android:text="Off"
41     android:textColor="?android:attr/textColorPrimary"
42     android:textOff="On"
43     android:textOn="Off"
44     android:textSize="18sp"
45     android:textStyle="bold"
46     app:layout_constraintBottom_toBottomOf="parent"
```

```

47         app:layout_constraintEnd_toEndOf="parent "
48         app:layout_constraintStart_toStartOf="parent "
49         app:layout_constraintTop_toBottomOf="@+id/timePicker" />
50
51     <TextView
52         android:id="@+id/alarmText "
53         android:layout_width="wrap_content "
54         android:layout_height="wrap_content "
55         android:text="Alarm Not Set"
56         android:textColor="?android:attr/textColorPrimary"
57         android:textSize="18sp"
58         android:textStyle="bold"
59         app:layout_constraintBottom_toTopOf="@+id/alarmToggle "
60         app:layout_constraintEnd_toEndOf="parent "
61         app:layout_constraintStart_toStartOf="parent "
62         app:layout_constraintTop_toBottomOf="@+id/timePicker" />
63
64 </androidx.constraintlayout.widget.ConstraintLayout>

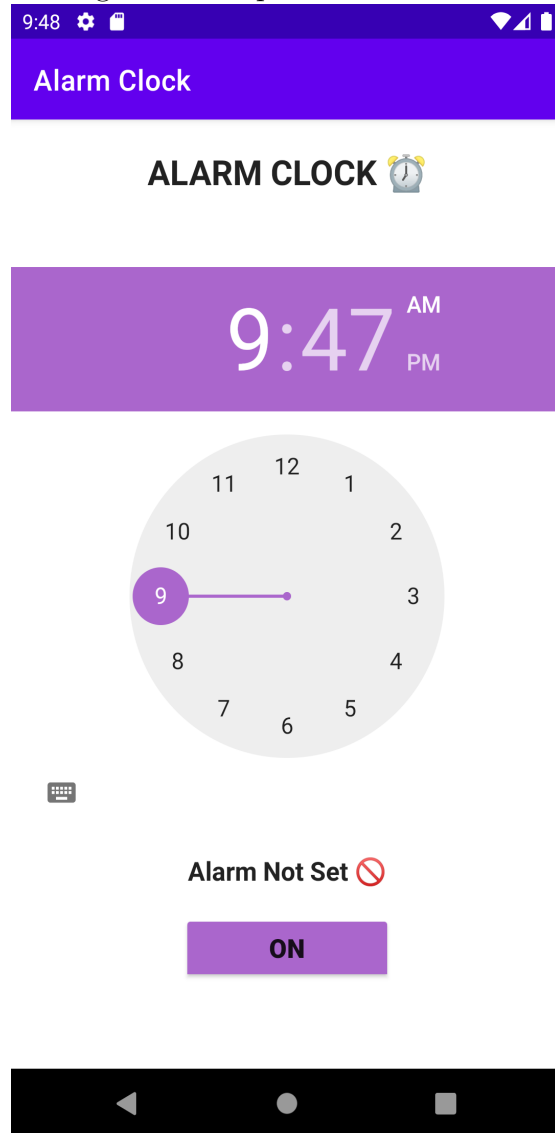
```

Code - Alarm Clock: Android Manifest XML:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="com.example.alarmclock">
4
5     <application
6         android:allowBackup="true"
7         android:icon="@mipmap/ic_launcher"
8         android:label="@string/app_name"
9         android:roundIcon="@mipmap/ic_launcher_round"
10        android:supportsRtl="true"
11        android:theme="@style/Theme.AlarmClock">
12        <activity
13            android:name=".MainActivity"
14            android:exported="true">
15            <intent-filter>
16                <action android:name="android.intent.action.MAIN" />
17
18                <category android:name="android.intent.category.LAUNCHER"
19            />
20            </intent-filter>
21        </activity>
22
23        <receiver android:name=".AlarmReceiver"/> <!-- added here for
24        alarm -->
25    </application>
26</manifest>
```

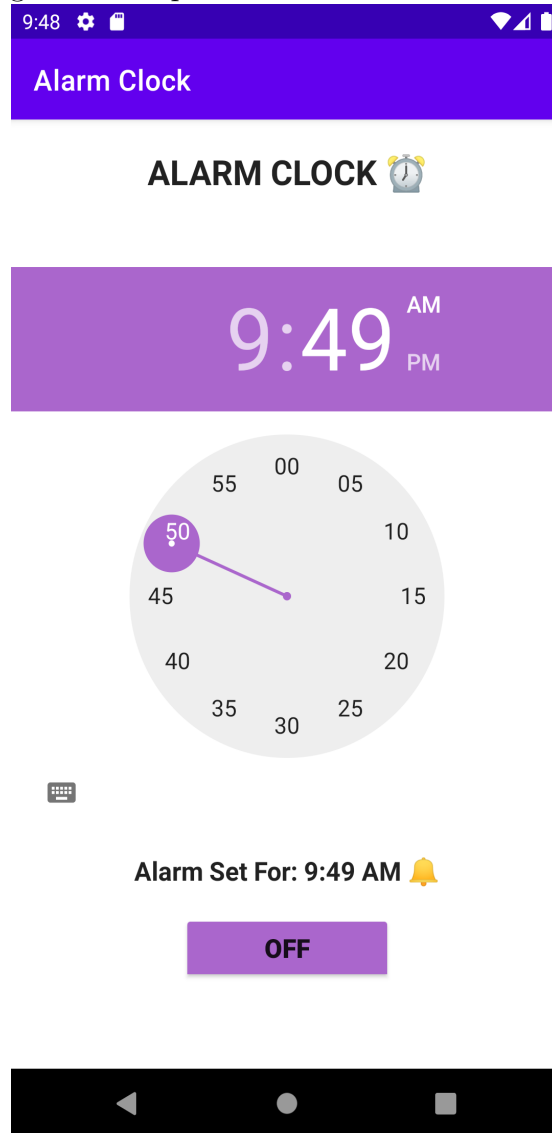
Output: Alarm Clock:

Figure 4: Output: Alarm Clock.



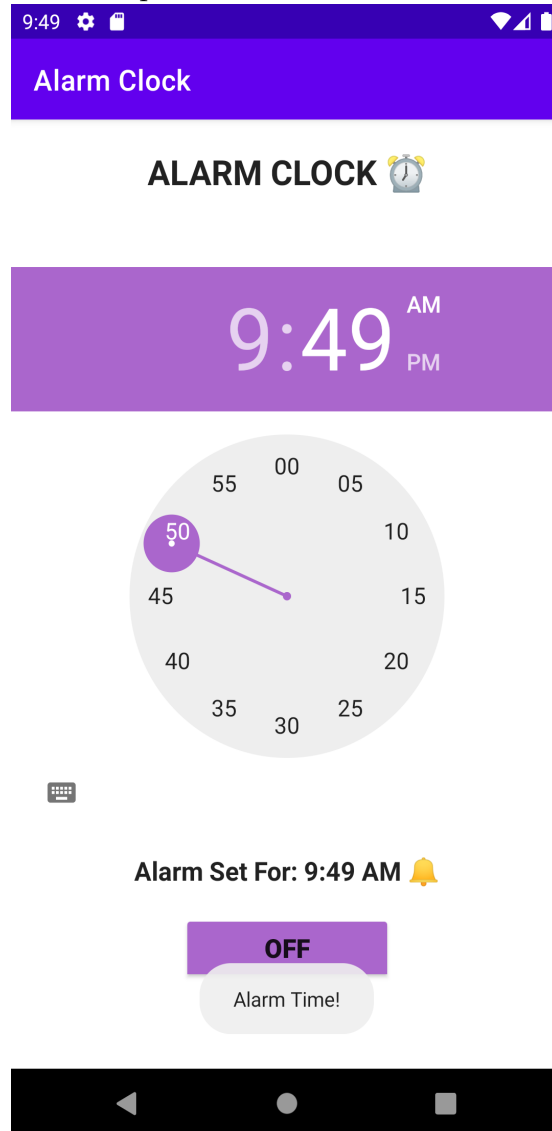
Output: Alarm Clock - Alarm Set:

Figure 5: Output: Alarm Clock - Alarm Set.



Output: Alarm Clock - Alarm Ringing:

Figure 6: Output: Alarm Clock - Alarm Ringing.



Learning Outcome:

- I understood how to design a **Progress Bar** in Android.
- I understood how to use **Thread** in Android to handle and update the progress bar's progress value.
- I understood how Threads work in Android by implementing a **Runnable** Object and how concurrent execution is handled in the UI using threads.
- I learnt how to display a **Progress Dialog** in Android and dismiss it while the main UI was sleeping.
- I learnt to use a **ToggleButton** and **TimePicker** to implement an Alarm Clock in Android.
- I learnt to use the inbuilt **AlarmManager** object to set alarms and play a ringtone using the **Ringtone** object for reminding the user during the time of alarm.
- I learnt to use **PendingIntent** and **Intent** to set the alarm in AlarmManager.
- I understood how to use **Log** object for debugging purposes and printing status messages.