SSN COLLEGE OF ENGINEERING, KALAVAKKAM

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

UCS1711 - MOBILE APPLICATION DEVELOPMENT LAB

Assignment 9

Name: Jayannthan P T *Dept: CSE 'A'* *Roll No.: 205001049*

Alarm Clock Application

Ex. No:9

Title of the Program:

Develop an alarm clock application for scheduling the task. The application should start an activity to set a new alarm or timer that reminds the user by alerting at the scheduled time. It should also have the option for snooze and stop the alarm.

Objective:

The objective of the Alarm Android App project is to create an application that allows users to set alarms using a TimePicker, trigger notifications, and play an alarm ringtone. The app provides a user interface to set and cancel alarms, and it includes functionalities such as creating a notification channel and handling alarm reception using a BroadcastReceiver.

Algorithm:

1. Create the MainActivity with a TimePicker for selecting the alarm time and a ToggleButton to set/cancel the alarm.
2. Implement methods to set and cancel alarms using the AlarmManager.
3. Create a NotificationChannel for managing notifications.
4. Implement AlarmReceiver as a BroadcastReceiver to handle the alarm trigger.
5. In AlarmReceiver, display a notification with relevant information and play the alarm ringtone.
6. Implement an additional activity (In) to display a message when the alarm is triggered.

Features used:

1. AlarmManager for scheduling alarms.
2. NotificationManagerCompat for displaying notifications.
3. RingtoneManager for managing alarm ringtones.
4. TimePicker for selecting alarm time.
5. ToggleButton for setting/canceling alarms.
6. BroadcastReceiver to handle alarm reception.
7. Intent for passing data between activities.

Source code:

- MainActivity.java

```
package com.example.alarm;

import androidx.appcompat.app.AppCompatActivity;
import android.app.AlarmManager;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Intent;
import android.os.Build;
import android.os.Bundle;
import android.text.format.Time;
import android.view.View;
import android.widget.Button;
import android.widget.TimePicker;
import android.widget.Toast;
import android.widget.ToggleButton;
import com.example.alarm.databinding.ActivityMainBinding;
import java.util.Calendar;

public class MainActivity extends AppCompatActivity {
    private ActivityMainBinding binding;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        binding = ActivityMainBinding.inflate(getLayoutInflater());
        setContentView(binding.getRoot());
        createNotificationChannel();
        ToggleButton b1 = findViewById(R.id.set);
        TimePicker t1 = findViewById(R.id.time);
        t1.setIs24HourView(true);
        b1.setOnCheckedChangeListener((buttonView, isChecked) -> {
            if (isChecked) {
                setAlarm();
            } else {
                cancelAlarm();
            }
        });
    }

    private void createNotificationChannel() {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
            CharSequence name = "alarm";
            String desc = "Channel for alarm";
            int importance = NotificationManager.IMPORTANCE_HIGH;
            NotificationChannel channel = new NotificationChannel("alarm", name,
importance);
            channel.setDescription(desc);
            NotificationManager notifs = getSystemService(NotificationManager.class);
            notifs.createNotificationChannel(channel);
```

```
        }
    }

    public void setAlarm() {
        TimePicker timePicker = findViewById(R.id.time);
        AlarmManager alarm = (AlarmManager) getSystemService(ALARM_SERVICE);
        Intent intent = new Intent(this, AlarmReceiver.class);
        PendingIntent pendingIntent =
PendingIntent.getBroadcast(this.getApplicationContext(), 234, intent,
                PendingIntent.FLAG_IMMUTABLE);
        int hour = timePicker.getHour();
        int minute = timePicker.getMinute();
        Calendar calendar = Calendar.getInstance();
        calendar.set(Calendar.HOUR_OF_DAY, hour);
        calendar.set(Calendar.MINUTE, minute);
        calendar.set(Calendar.SECOND, 0);
        long triggerTime = calendar.getTimeInMillis();
        int timeInSec = 1;
        alarm.set(AlarmManager.RTC_WAKEUP, triggerTime, pendingIntent);
        Toast.makeText(this, "Alarm set for " + hour + ":" + minute,
Toast.LENGTH_SHORT).show();
    }

    public void cancelAlarm() {
        AlarmManager alarm = (AlarmManager) getSystemService(ALARM_SERVICE);
        Intent intent = new Intent(this, AlarmReceiver.class);
        PendingIntent pendingIntent =
PendingIntent.getBroadcast(this.getApplicationContext(), 234, intent,
                PendingIntent.FLAG_IMMUTABLE);
        if (alarm != null) {
            alarm.cancel(pendingIntent);
        }
        Toast.makeText(this, "Alarm unset!", Toast.LENGTH_SHORT).show();
    }
}
```

- activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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" tools:context=".MainActivity">
    <TimePicker android:layout_width="wrap_content" android:layout_height="wrap_content"
android:id="@+id/time" />
    <ToggleButton android:id="@+id/set" android:layout_width="150dp"
android:layout_height="50dp" android:layout_below="@+id/time"
android:layout_marginLeft="130dp" />
</RelativeLayout>
```

- AlarmReceiver.java

```java
package com.example.alarm;

import static androidx.core.content.ContextCompat.startActivity;
import android.app.AlarmManager;
import android.app.Notification;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.media.Ringtone;
import android.media.RingtoneManager;
import android.net.Uri;
import android.os.Build;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.app.NotificationCompat;
import androidx.core.app.NotificationManagerCompat;

public class AlarmReceiver extends BroadcastReceiver {
    static Uri alarmrt1 = RingtoneManager.getDefaultUri(RingtoneManager.TYPE_ALARM);

    @Override
    public void onReceive(Context context, Intent intent) {
        Toast.makeText(context, "INSIDE WOHOOO", Toast.LENGTH_LONG).show();
        Intent i = new Intent(context, In.class);
        intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
                Intent.FLAG_ACTIVITY_CLEAR_TASK);
        PendingIntent p = PendingIntent.getActivity(context, 0, i,
PendingIntent.FLAG_IMMUTABLE);
        NotificationCompat.Builder builder = new NotificationCompat.Builder(context,
"alarm")
                .setSmallIcon(R.drawable.ic_launcher_foreground)
                .setContentTitle("Your Alarm is going off!!!")
                .setContentInfo("You set this alarm!")
                .setAutoCancel(true)
                .setDefaults(NotificationCompat.DEFAULT_ALL)
                .setPriority(NotificationCompat.PRIORITY_HIGH)
                .setContentIntent(p);
        NotificationManagerCompat notifications = NotificationManagerCompat.from(context);
        if (ActivityCompat.checkSelfPermission(context,
                android.Manifest.permission.POST_NOTIFICATIONS) !=
PackageManager.PERMISSION_GRANTED) {
            return;
        } else {
            notifications.notify(123, builder.build());
        }
        Ringtone ringtone = RingtoneManager.getRingtone(context.getApplicationContext(),
alarmrt1);
        // Toast.makeText(context, ringtone.toString(), Toast.LENGTH_SHORT).show();
        intent.putExtra("RINGTONE_URI", alarmrt1);
```

```
        ringtone.play();
    }

    public static Uri getInstant() {
        return alarmrt1;
    }
}
```

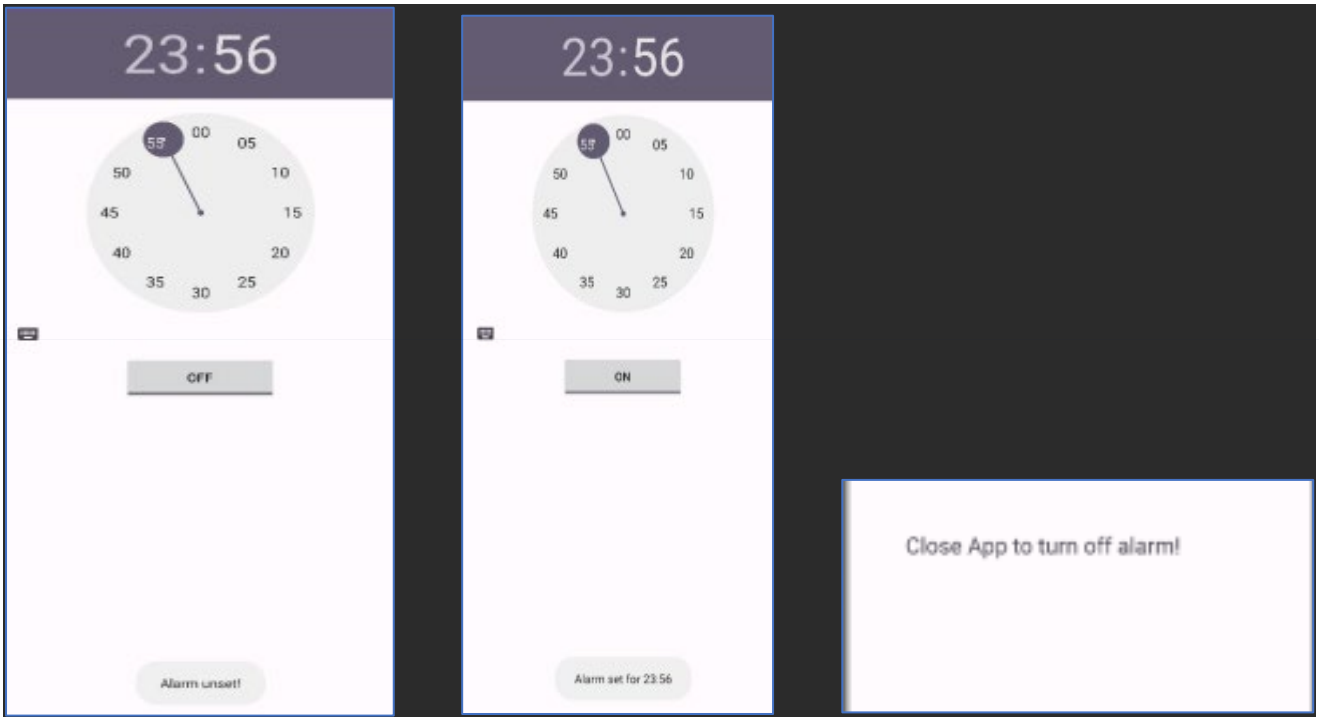
- In.java

```
package com.example.alarm;

import android.content.Intent;
import android.media.Ringtone;
import android.media.RingtoneManager;
import android.net.Uri;
import android.os.Bundle;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class In extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.display);
        Intent intent = getIntent();
        Uri ringtoneUri = intent.getParcelableExtra("RINGTONE_URI");
        Ringtone ringtone = RingtoneManager.getRingtone(In.this, ringtoneUri);
        // Toast.makeText(In.this, ringtone.toString(), Toast.LENGTH_SHORT).show();
        if (ringtone.isPlaying()) {
            // Toast.makeText(In.this, "yes", Toast.LENGTH_SHORT).show(); ringtone.stop();
        } else {
            // Toast.makeText(In.this, "no", Toast.LENGTH_SHORT).show();
        }
    }
}
```

- display.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"
android:layout_width="match_parent" android:layout_height="match_parent">
    <TextView android:id="@+id/textView" android:layout_width="wrap_content"
android:layout_height="wrap_content" android:text="Close App to turn off alarm!"
android:padding="50dp" android:textSize="20dp"
app:layout_constraintStart_toStartOf="parent" app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Output:



Result:

The mobile application was completed successfully

Best Practices:

1. Employ PendingIntent.FLAG_IMMUTABLE for PendingIntent to prevent modifications.
2. Provide meaningful constant values for notification IDs and other flags.
3. Request necessary permissions (e.g., POST_NOTIFICATIONS) explicitly.
4. Utilize separate activities for distinct functionalities (MainActivity, In).
5. Implement error handling for potential exceptions during alarm operations.

Learning Outcomes:

1. Understanding and implementing alarms using AlarmManager.
2. Handling notifications with NotificationManagerCompat and NotificationCompat.Builder.
3. Working with BroadcastReceiver to capture broadcasted alarms.
4. Using intents to transfer data between activities (passing ringtone URI).
5. Managing and playing alarm ringtones with RingtoneManager.