Exercise 9 - Alarm Clock

Objective

Develop an Alarm Clock Android Application.
1. Use permission for WAKE_LOCK.
2. Have a TimePicker component followed by a ToggleButton to select time and Alarm On / Off.
3. Use the AlarmManager to set the alarm and send notification on alarm trigger. 4. Perform 3 different notifications
a. Show a message to user in the activity UI
b. Play the alarm ringtone
c. Send an Android notification message

Algorithm

1. Initialize the Project:
   - Open Android Studio and create a new Android Project.

2. Request Permission:
   - Request the `WAKE_LOCK` permission in the manifest file.

3. Design the UI:
   - Design the UI with a `TimePicker` component and a `ToggleButton` to set the alarm.

4. Set Alarm Using AlarmManager:
   - Utilize the `AlarmManager` to set the alarm based on the selected time.

5. Show Message in UI:
   - Display a message in the activity UI when the alarm is triggered.

6. Play Alarm Ringtone:
   - Play the alarm ringtone when the alarm is triggered.

7. Send Android Notification:
   - Send a notification message using the `NotificationManager` when the alarm triggers.

Features used

Main Features:
- Permission handling for `WAKE_LOCK`.
- UI components for selecting time and toggling the alarm.
- Utilization of `AlarmManager` for setting the alarm.
- Displaying a message in the UI upon alarm trigger.
- Playing the alarm ringtone.
- Sending an Android notification message.

Source Code

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>
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>
```

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(); }
}
```

AlarmReceiver.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;
```

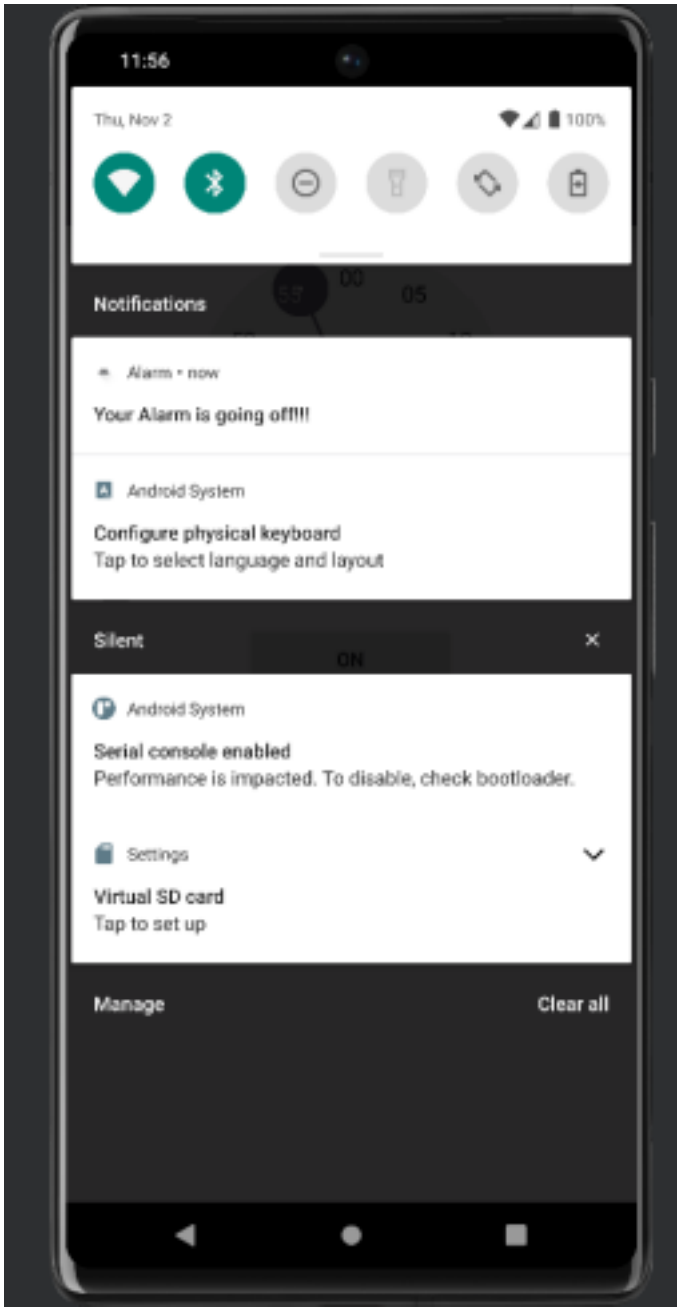
```
import android.os.Bundle;
import android.widget.Toast;
//ce3c069
import androidx.appcompat.app.AppCompatActivity;
public class In extends AppCompatActivity {
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.display);
Intent 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(); }
}
}
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools">
<uses-permission android:name="android.permission.POST_NOTIFICATIONS" />
<uses-permission android:name="android.permission.WAKE_LOCK" />
<uses-permission android:name="android.permission.VIBRATE" />
<application
android:allowBackup="true"
android:dataExtractionRules="@xml/data_extraction_rules"
android:fullBackupContent="@xml/backup_rules"
android:icon="@mipmap/ic_launcher"
android:label="@string/app_name"
android:roundIcon="@mipmap/ic_launcher_round"
android:supportsRtl="true"
android:theme="@style/Theme.Alarm"
tools:targetApi="31">
<activity
android:name=".MainActivity"
```

```
android:exported="true">
<intent-filter>
<action android:name="android.intent.action.MAIN" />
<category android:name="android.intent.category.LAUNCHER" /> </intent-filter>
</activity>
<activity android:name=".In"
android:exported="false"
/>
<receiver android:name=".AlarmReceiver"
/>
</application>
</manifest>
```

Output Screenshots



Result

Thus an alarm clock was implemented

Best Practices

1. User friendly design

2. Readable layouts
3. Modularity
4. Used apt names for xml and java files.
5. Set padding and margins for dynamically added elements
6. Use default ringtone as it is known by everyone.

Learning outcomes

- Learnt how to use UI Elements for Android App development.
- Learnt how to use Phone shared storage.