

SOURCE CODE

MainActivity.java

```
package com.example.medicationreminder;

import android.app.AlarmManager;

import android.app.PendingIntent;

import android.content.Context;

import android.content.Intent;

import android.media.MediaPlayer;

import android.os.Build;

import android.os.Bundle;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.EditText;

import android.widget.ListView;

import android.widget.Spinner;

import android.widget.Toast;


import androidx.appcompat.app.AppCompatActivity;

import java.text.SimpleDateFormat;
```

```

import java.util.ArrayList;

import java.util.Calendar;

import java.util.Locale;

public class MainActivity extends AppCompatActivity {

    private EditText medicationName, reminderTime, guardianPhone;

    private Spinner repeatOption;

    private Button addReminderButton, stopAlarmButton;

    private ListView remindersListView;

    private ArrayList<String> remindersList;

    private ArrayAdapter<String> adapter;

    private MediaPlayer mediaPlayer;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        medicationName = findViewById(R.id.medicationName);

        reminderTime = findViewById(R.id.reminderTime);

        guardianPhone = findViewById(R.id.guardianPhone);

        repeatOption = findViewById(R.id.repeatOption);

```

```

addReminderButton = findViewById(R.id.addReminderButton);

stopAlarmButton = findViewById(R.id.stopAlarmButton);

remindersListView = findViewById(R.id.remindersListView);

remindersList = new ArrayList<>();

adapter = new ArrayAdapter<>(this, android.R.layout.simple_list_item_1,
remindersList);

remindersListView.setAdapter(adapter);

ArrayAdapter<CharSequence> spinnerAdapter =
ArrayAdapter.createFromResource(this,
    R.array.repeat_options, android.R.layout.simple_spinner_item);

spinnerAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropd
own_item);

repeatOption.setAdapter(spinnerAdapter);

addReminderButton.setOnClickListener(v -> addReminder());

stopAlarmButton.setOnClickListener(v -> stopAlarm());

}

private void addReminder() {

    String name = medicationName.getText().toString();

```

```

String time = reminderTime.getText().toString();

String guardian = guardianPhone.getText().toString();

String repeat = repeatOption.getSelectedItem().toString();

if (name.isEmpty() || time.isEmpty() || guardian.isEmpty()) {

    Toast.makeText(this, "Please fill in all fields",

Toast.LENGTH_SHORT).show();

    return;

}

try {

    SimpleDateFormat sdf = new SimpleDateFormat("HH:mm",

Locale.getDefault());

    Calendar calendar = Calendar.getInstance();

    // Parse the user-provided time

    String[] timeParts = time.split(":");

    int hour = Integer.parseInt(timeParts[0]);

    int minute = Integer.parseInt(timeParts[1]);

    // Set the reminder time

    calendar.set(Calendar.HOUR_OF_DAY, hour);

```

```

calendar.set(Calendar.MINUTE, minute);

calendar.set(Calendar.SECOND, 0);

// If the time is in the past, schedule it for the next day

if (calendar.getTimeInMillis() <= System.currentTimeMillis()) {

    calendar.add(Calendar.DAY_OF_YEAR, 1);

}

// Log the scheduled time for debugging

System.out.println("Scheduled reminder for: " + calendar.getTime());

// Schedule the alarm

scheduleAlarm(name, guardian, calendar, repeat);

remindersList.add(name + " at " + new SimpleDateFormat("HH:mm",
Locale.getDefault()).format(calendar.getTime()));

adapter.notifyDataSetChanged();

medicationName.setText("");

reminderTime.setText("");

guardianPhone.setText("");

Toast.makeText(this, "Reminder added", Toast.LENGTH_SHORT).show();

} catch (Exception e) {

```

```

        Toast.makeText(this, "Invalid time format", Toast.LENGTH_SHORT).show();

    }

}

private void scheduleAlarm(String medicationName, String guardianPhone,
Calendar calendar, String repeat) {

    AlarmManager alarmManager = (AlarmManager)
getSystemService(Context.ALARM_SERVICE);

    Intent intent = new Intent(this, ReminderReceiver.class);

    intent.putExtra("medicationName", medicationName);

    intent.putExtra("guardianPhone", guardianPhone);

    PendingIntent pendingIntent = PendingIntent.getBroadcast(this, 0, intent,
PendingIntent.FLAG_UPDATE_CURRENT);

    try {

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {

            alarmManager.setExactAndAllowWhileIdle(AlarmManager.RTC_WAKEUP,
calendar.getTimeInMillis(), pendingIntent);

        } else {

            alarmManager.setExact(AlarmManager.RTC_WAKEUP,
calendar.getTimeInMillis(), pendingIntent);

```

```

    }

    Toast.makeText(this, "Alarm scheduled for: " + calendar.getTime().toString(),
Toast.LENGTH_SHORT).show();

    } catch (SecurityException e) {

        Toast.makeText(this, "Failed to schedule alarm: " + e.getMessage(),
Toast.LENGTH_LONG).show();

    }

    // Handle repeating alarms

    long interval = 0;

    switch (repeat) {

        case "Daily":

            interval = AlarmManager.INTERVAL_DAY;

            break;

        case "Weekly":

            interval = AlarmManager.INTERVAL_DAY * 7;

            break;

        case "Monthly":

            interval = AlarmManager.INTERVAL_DAY * 30;

            break;

```

```

    }

    if (interval > 0) {

        alarmManager.setRepeating(AlarmManager.RTC_WAKEUP,

calendar.getTimeInMillis(), interval, pendingIntent);

    }

}

private void stopAlarm() {

    AlarmManager alarmManager = (AlarmManager)

getSystemService(Context.ALARM_SERVICE);

    Intent intent = new Intent(this, ReminderReceiver.class);

    PendingIntent pendingIntent = PendingIntent.getBroadcast(this, 0, intent,

PendingIntent.FLAG_UPDATE_CURRENT);

    if (mediaPlayer != null && mediaPlayer.isPlaying()) {

        mediaPlayer.stop();

        mediaPlayer.release();

        mediaPlayer = null;

    }

    alarmManager.cancel(pendingIntent);

    pendingIntent.cancel();

```



```

        ReminderReceiver.stopAlarmSound();

        Toast.makeText(this, "Alarm stopped", Toast.LENGTH_SHORT).show();

    }

    @Override

    protected void onDestroy() {

        super.onDestroy();

        if (mediaPlayer != null) {

            mediaPlayer.release();

        }

    }

}

```

activity_main.xml

```

<LinearLayout

    xmlns:android="http://schemas.android.com/apk/res/android"

    android:layout_width="match_parent"

    android:layout_height="match_parent"

    android:orientation="vertical"

```

```
android:padding="16dp">
```

```
<EditText
```

```
    android:id="@+id/medicationName"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:autofillHints=""
```

```
    android:hint="@string/Add_medication_name"
```

```
    android:inputType="text" />
```

```
<EditText
```

```
    android:id="@+id/reminderTime"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:autofillHints=""
```

```
    android:hint="@string/Add_Reminder_Time"
```

```
    android:inputType="time" />
```

```
<EditText
```

```
    android:id="@+id/guardianPhone"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

android:autofillHints=""

android:hint="@string/Add_Guardian_Phone_Number"

android:inputType="phone" />

<Spinner

android:id="@+id/repeatOption"

android:layout_width="match_parent"

android:layout_height="wrap_content"

android:layout_marginTop="8dp" />

<Button

android:id="@+id/addReminderButton"

android:layout_width="match_parent"

android:layout_height="wrap_content"

android:text="@string/Add_Reminder" />

<Button

android:id="@+id/stopAlarmButton"

android:layout_width="match_parent"

android:layout_height="wrap_content"

android:text="@string/Stop_alarm" />

<ListView

android:id="@+id/remindersListView"

android:layout_width="match_parent"

android:layout_height="0dp"

android:layout_weight="1"

android:layout_marginTop="16dp"

android:divider="@android:color/darker_gray"

android:dividerHeight="1dp" />

</LinearLayout>

Strings.xml

<resources>

<string name="app_name">MedicationReminder</string>

<string name="action_settings">Settings</string>

<!-- Strings used for fragments for navigation -->

<string name="Add_medication_name">Enter Medication Name</string>

<string name="Add_Reminder_Time">Enter Reminder Time (HH:mm)</string>

<string name="Add_Guardian_Phone_Number">Enter Guardian Phone

Number</string>

<string name="Add_Reminder">Add Reminder</string>

<string name="Stop_alarm">Stop Alarm</string>

<string name="first_fragment_label">First Fragment</string>

<string name="second_fragment_label">Second Fragment</string>

<string name="next">Next</string>

<string name="previous">Previous</string>

<string-array name="repeat_options">

<item>Once</item>

<item>Daily</item>

<item>Weekly</item>

<item>Monthly</item>

</string-array>

<string name="lorem_ipsum">

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam in scelerisque sem.

Mauris volutpat, dolor id interdum ullamcorper, risus dolor egestas lectus, sit amet

mattis purus

dui nec risus. Maecenas non sodales nisi, vel dictum dolor. Class aptent taciti

sociosqu ad

litora torquent per conubia nostra, per inceptos himenaeos. Suspendisse blandit

eleifend

diam, vel rutrum tellus vulputate quis. Aliquam eget libero aliquet, imperdiet nisl
a,

ornare ex. Sed rhoncus est ut libero porta lobortis. Fusce in dictum tellus.\n\n
Suspendisse interdum ornare ante. Aliquam nec cursus lorem. Morbi id magna
felis. Vivamus

egestas, est a condimentum egestas, turpis nisl iaculis ipsum, in dictum tellus
dolor sed

neque. Morbi tellus erat, dapibus ut sem a, iaculis tincidunt dui. Interdum et
malesuada

fames ac ante ipsum primis in faucibus. Curabitur et eros porttitor, ultricies urna
vitae,

molestie nibh. Phasellus at commodo eros, non aliquet metus. Sed maximus nisl
nec dolor

bibendum, vel congue leo egestas.\n\n

Sed interdum tortor nibh, in sagittis risus mollis quis. Curabitur mi odio,
condimentum sit

amet auctor at, mollis non turpis. Nullam pretium libero vestibulum, finibus orci
vel,

molestie quam. Fusce blandit tincidunt nulla, quis sollicitudin libero facilisis et.

Integer

interdum nunc ligula, et fermentum metus hendrerit id. Vestibulum lectus felis,
dictum at

lacinia sit amet, tristique id quam. Cras eu consequat dui. Suspendisse sodales
nunc ligula,

in lobortis sem porta sed. Integer id ultrices magna, in luctus elit. Sed a
pellentesque

est.\n\n

Aenean nunc velit, lacinia sed dolor sed, ultrices viverra nulla. Etiam a venenatis
nibh.

Morbi laoreet, tortor sed facilisis varius, nibh orci rhoncus nulla, id elementum
leo dui

non lorem. Nam mollis ipsum quis auctor varius. Quisque elementum eu libero
sed commodo. In

eros nisl, imperdiet vel imperdiet et, scelerisque a mauris. Pellentesque varius ex
nunc,

quis imperdiet eros placerat ac. Duis finibus orci et est auctor tincidunt. Sed non
viverra

ipsum. Nunc quis augue egestas, cursus lorem at, molestie sem. Morbi a

consectetur ipsum, a

placerat diam. Etiam vulputate dignissim convallis. Integer faucibus mauris sit amet finibus

convallis.\n\n

Phasellus in aliquet mi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. In volutpat arcu ut felis sagittis, in finibus massa

gravida. Pellentesque id tellus orci. Integer dictum, lorem sed efficitur ullamcorper,

libero justo consectetur ipsum, in mollis nisl ex sed nisl. Donec maximus ullamcorper

sodales. Praesent bibendum rhoncus tellus nec feugiat. In a ornare nulla. Donec rhoncus

libero vel nunc consequat, quis tincidunt nisl eleifend. Cras bibendum enim a justo luctus

vestibulum. Fusce dictum libero quis erat maximus, vitae volutpat diam dignissim.

</string>

</resources>

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.SEND_SMS" />
```

```
    <uses-permission android:name="android.permission.CALL_PHONE" />
```

```
    <uses-feature android:name="android.hardware.telephony" required="false"
```

```
        tools:ignore="MissingPrefix" />
```

```
    <uses-permission
```

```
        android:name="android.permission.SCHEDULE_EXACT_ALARM"
```

```
        android:permissionGroup="android.permission-group.SCHEDULE" />
```

```
    <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.MedicationReminder"

        tools:targetApi="31">

        <activity

            android:name=".MainActivity"

            android:exported="true"

            android:theme="@style/Theme.MedicationReminder">

            <intent-filter>

                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />

            </intent-filter>

        </activity>

        <receiver android:name=".ReminderReceiver" />

    </application>

</manifest>

```

ReminderReceiver.java

```

package com.example.medicationreminder;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.Intent;

import android.media.MediaPlayer;

import android.media.Ringtone;

import android.media.RingtoneManager;

import android.net.Uri;

import android.os.Handler;

import android.telephony.SmsManager;

import android.widget.Toast;


public class ReminderReceiver extends BroadcastReceiver {


    private static MediaPlayer mediaPlayer; // Use static to persist state across
broadcasts


    private static Ringtone ringtone;    // Static to manage a single instance


    private static Handler handler = new Handler(); // For delayed actions


    private static Runnable stopAlarmRunnable;    // Runnable for delayed stop

```

```

@Override

public void onReceive(Context context, Intent intent) {

    String action = intent.getAction();

    // Handle stop alarm action

    if ("STOP_ALARM".equals(action)) {

        stopAlarmSound();

        return; // Exit immediately if it's a stop alarm action

    }

    // Get extras from the intent

    String medicationName = intent.getStringExtra("medicationName");

    String guardianPhone = intent.getStringExtra("guardianPhone");

    // Notify the user

    Toast.makeText(context, "Time to take: " + medicationName,
Toast.LENGTH_LONG).show();

```

```

// Play ringtone

Uri alarmUri =

RingtoneManager.getDefaultUri(RingtoneManager.TYPE_ALARM);

if (alarmUri == null) {

    alarmUri =

RingtoneManager.getDefaultUri(RingtoneManager.TYPE_NOTIFICATION);

}

    ringtone = RingtoneManager.getRingtone(context, alarmUri);

    ringtone.play();


// Play additional alarm sound using MediaPlayer

playAlarmSound(context);


// Schedule notification to guardian after 1 minute

stopAlarmRunnable = () -> {

    stopAlarmSound(); // Stop the alarm sound

    notifyGuardian(context, guardianPhone, medicationName);

};

handler.postDelayed(stopAlarmRunnable, 60 * 1000); // 1-minute delay

```

```
}
```

```
private void playAlarmSound(Context context) {
```

```
    if (mediaPlayer == null) {
```

```
        mediaPlayer = MediaPlayer.create(context, R.raw.music_05);
```

```
        mediaPlayer.setLooping(true);
```

```
        mediaPlayer.start();
```

```
    }
```

```
}
```

```
public static void stopAlarmSound() {
```

```
    // Stop MediaPlayer if it's playing
```

```
    if (mediaPlayer != null && mediaPlayer.isPlaying()) {
```

```
        mediaPlayer.stop();
```

```
        mediaPlayer.release();
```

```
        mediaPlayer = null;
```

```
    }
```

```
    // Stop Ringtone if it's playing
```

```
    if (ringtone != null && ringtone.isPlaying()) {
```

```
        ringtone.stop();
```

```

        ringtone = null;

    }

    // Cancel any pending stop alarm runnable

    if (handler != null && stopAlarmRunnable != null) {

        handler.removeCallbacks(stopAlarmRunnable);

    }

}

private void notifyGuardian(Context context, String phoneNumber, String
medicationName) {

    sendSMS(context, phoneNumber, medicationName);

    makeCall(context, phoneNumber);

}

private void sendSMS(Context context, String phoneNumber, String
medicationName) {

    try {

        SmsManager smsManager = SmsManager.getDefault();

        String message = "Reminder Alert: Medication (" + medicationName + ") has
not been taken. Please check.";

```

```

        smsManager.sendTextMessage(phoneNumber, null, message, null, null);

        Toast.makeText(context, "SMS sent to guardian: " + phoneNumber,
Toast.LENGTH_LONG).show();

    } catch (Exception e) {

        Toast.makeText(context, "Failed to send SMS to guardian.",
Toast.LENGTH_LONG).show();

        e.printStackTrace();

    }

}

private void makeCall(Context context, String phoneNumber) {

    try {

        Intent callIntent = new Intent(Intent.ACTION_CALL);

        callIntent.setData(Uri.parse("tel:" + phoneNumber));

        callIntent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);

        context.startActivity(callIntent);

        Toast.makeText(context, "Calling guardian: " + phoneNumber,
Toast.LENGTH_LONG).show();

    } catch (SecurityException e) {

```



```
        Toast.makeText(context, "Unable to make the call. Missing permission.",  
        Toast.LENGTH_LONG).show();  
  
        e.printStackTrace();  
    }  
}
```

APPENDIX – B

SCREENSHOTS

Sample Output

The screenshot shows a mobile application interface for adding a medication reminder. At the top, the status bar displays the time 10:39, signal strength, 4G network, and 72% battery. The app interface has a light pink background. It features three input fields with placeholder text: "Enter Medication Name", "Enter Reminder Time (HH:mm)", and "Enter Guardian Phone Number". Below these fields is a dropdown menu currently set to "Once". At the bottom of the form are two purple buttons: "Add Reminder" and "Stop Alarm". The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps icons.

Figure 6.B.1: Add medication Reminder



Figure 6.B.2: Reminding Caretaker



Figure 6.B.3: Alert caretaker via phone call

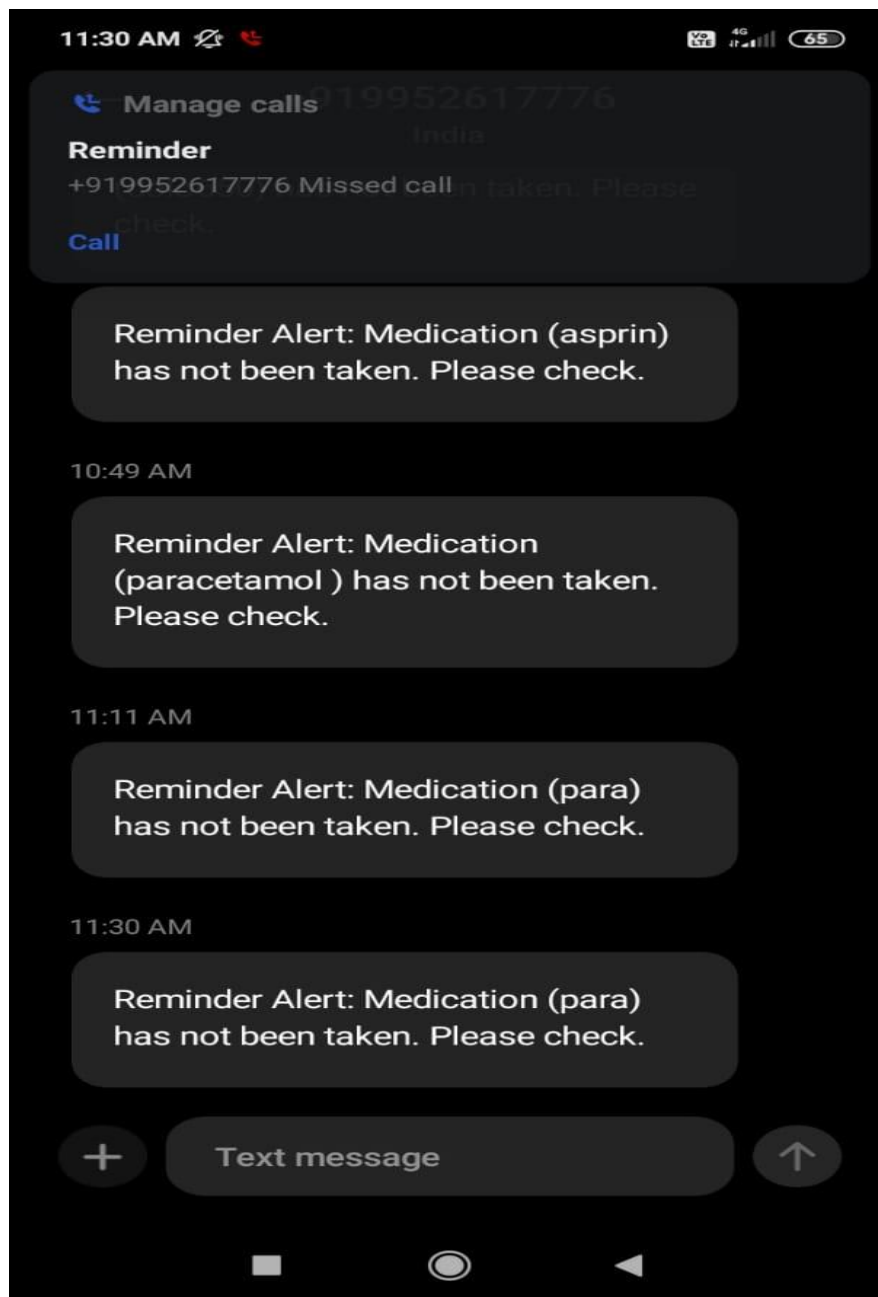


Figure 6.B.4:Alert caretaker via SMS