Department of
Computer Science and Engineering

Title: Implementation of Broadcast Intent and Broadcast Receiver

Mobile Application Development

CSE 402

Green University of Bangladesh

1 Objective(s)

- Broadcast receiver is an Android component which allows you to send or receive Android system or application events.These messages are sometime called events or intents. This experiment is designed to implement android broadcast component by creating an alarm clock in android development.

2 Problem analysis

Broadcast Receivers simply respond to broadcast messages from other applications or from the system itself.Broadcast intents are a mechanism by which an intent can be issued for consumption by multiple components on an Android system. Broadcasts are detected by registering a Broadcast Receiver which, in turn, is configured to listen for intents that match particular action strings. In general, broadcast receivers remain dormant until woken up by the system when a matching intent is detected. In this experiment we will create an alarm clock which will hold a broadcast intent to be implemented by Broadcast receiver.

3 Implementation of Broadcast for making an alarm clock

3.1 XML File of Alarm Clock

```
<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"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="20dp"
        android:text="Enter the time in seconds"
        android:textSize="30dp"
        android:textStyle="bold" />

    <EditText
        android:id="@+id/eTime"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp" />

    <Button
        android:id="@+id/bClick"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="20dp"
        android:text="Click" />
</LinearLayout>
```

3.2 Adding Broadcast Receiver java class to the Project

- Right click app -> New -> Other -> Broadcast Receiver
- Give class name Alarm -> Finish
- A Broadcast Receiver Class named Alarm will be created which will have onReceive function.

3.3 MainActivity Java File implementing an Alarm Clock

```java
package com.example.alarmclock;

import androidx.appcompat.app.AppCompatActivity;

import android.app.AlarmManager;
import android.app.PendingIntent;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    EditText eTime;
    Button bClick;

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

        eTime=findViewById(R.id.eTime);
        bClick=findViewById(R.id.bClick);

        bClick.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                int i= Integer.parseInt(eTime.getText().toString());

                Intent intent=new Intent(getApplicationContext(),Alarm.class);
                PendingIntent pendingIntent=PendingIntent.getBroadcast(
                   getApplicationContext(),
                        800000, intent,0);

                AlarmManager alarmManager=(AlarmManager) getSystemService(
                   ALARM_SERVICE);
                alarmManager.set(AlarmManager.RTC_WAKEUP,
                        System.currentTimeMillis()+(i*1000),
                        pendingIntent);

                Toast.makeText(getApplicationContext(),"Alarm set after "+i+"
                   seconds.",
                        Toast.LENGTH_LONG).show();
            }
        });

    }
}
```

3.4 Alarm Java File implementing Broadcast Receiver for an Alarm Clock

```java
package com.example.alarmclock;

import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.widget.Toast;

public class Alarm extends BroadcastReceiver {

    @Override
    public void onReceive(Context context, Intent intent) {

        Toast.makeText(context,"ALARM!!!!", Toast.LENGTH_LONG).show();
    }
}
```

4 Input/Output

Run the code and observe the output in the virtual device.

5 Discussion & Conclusion

From this experiments we learn about how intent, pending intent, broadcast intent work. This experiment is designed in a way to teach the students about implementing android Broadcast Component.

6 Lab Task (Please implement yourself and show the output to the instructor)

1. Improve the design of the Alarm clock.
2. Include the feature of playing an audio file when it is time for the alarm.

6.1 Problem analysis

Implement the lab tasks with the help of Broadcast Receiver as taught in the experiment. To implement this, one needs to add raw resource directory in the res, and add an audio file in that directory. After that on needs to create MediaPlayer object in the Alarm java file. This object will read the audio file from the res and start that media inside the onReceive function.

7 Lab Exercise (Submit as a report)

- Design a Broadcast Receiver which will show whenever battery percentage is changed.

8 Policy

Copying from internet, classmate, seniors, or from any other source is strongly prohibited. 100% marks will be *deducted* if any such copying is detected.