

Lab bài 5 – Service

1. MỤC TIÊU

Sau khi hoàn thành bài thực hành, bạn có thể nắm được các vấn đề sau:

- Hiểu được hoạt động của Started Service và BoundService
- Biết cách triển khai các trạng thái của Service
- Nắm rõ hơn về vòng đời của một Service

2. THỜI GIAN THỰC HIỆN: 20'

3. YÊU CẦU

Viết một ứng dụng Android đơn giản với những yêu cầu cụ thể như sau:

- Sử dụng Started Service để có thể phát và tắt một bản nhạc
- Sử dụng Bound Service để hiển thị thông tin thời gian hiện tại của hệ thống

4. CÁC BƯỚC THỰC HIỆN

Bước 1

Tạo project với các thông tin sau:

- Project Name: myservice
- Package Name: myservice.ténnv.lab

Bước 2

Tạo một activity có tên MainActivity, là activity cho phép người dùng lựa chọn sử dụng trạng thái service nào.

```

package funix.prm.lab;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity extends Activity implements View.OnClickListener {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        Button btnStart = (Button) findViewById(R.id.btnStartService);
        btnStart.setOnClickListener(this);

        Button btnStop = (Button) findViewById(R.id.btnBoundService);
        btnStop.setOnClickListener(this);
    }

    @Override
    public void onClick(View view) {
        switch (view.getId()) {
            case R.id.btnStartService:
                startActivity(new Intent( packageContext: MainActivity.this, StartServiceActivity.class));
                break;
            case R.id.btnBoundService:
                startActivity(new Intent( packageContext: MainActivity.this, BoundServiceActivity.class));
                break;
        }
    }
}

```

Bước 3

Tạo một file layout có tên activity_main.xml, là giao diện cho MainActivity

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="20dp"
    android:paddingRight="20dp"
    tools:context=".MainActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_centerVertical="true"
        android:orientation="vertical">
        <Button
            android:id="@+id/btnStartService"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Start Service" />
        <Button
            android:id="@+id/btnBoundService"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Bound Service" />
    </LinearLayout>
</RelativeLayout>

```

Bước 4

Tạo activity có tên StartServiceActivity để cho phép người dùng start/stop một bản nhạc

```

package funix.prm.lab;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class StartServiceActivity extends Activity implements View.OnClickListener {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_start_service);

        Button btnStart = (Button) findViewById(R.id.btnStart);
        btnStart.setOnClickListener(this);

        Button btnStop = (Button) findViewById(R.id.btnStop);
        btnStop.setOnClickListener(this);
    }

    @Override
    public void onClick(View view) {
        switch (view.getId()) {
            case R.id.btnStart:
                startService(new Intent( packageContext: this, MyStartService.class));
                break;
            case R.id.btnStop:
                stopService(new Intent( packageContext: this, MyStartService.class));
                break;
        }
    }
}

```

Bước 5

Tạo file layout có tên activity_start_service.xml để làm giao diện cho StartServiceActivity

```

        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:paddingLeft="20dp"
        android:paddingRight="20dp"
        tools:context=".StartServiceActivity">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_centerVertical="true"
            android:orientation="vertical">

            <Button
                android:id="@+id/btnStart"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:text="Start Service" />

            <Button
                android:id="@+id/btnStop"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:text="Stop Service" />

        </LinearLayout>
    </RelativeLayout>

```

Bước 6

Tạo một Service có tên MyStartService để xử lý nghiệp vụ start/stop một bản nhạc

```

package funix.prm.lab;

import android.app.Service;
import android.content.Intent;
import android.media.MediaPlayer;
import android.os.IBinder;
import android.provider.Settings;
import android.support.annotation.Nullable;

public class MyStartService extends Service {

    private MediaPlayer mMediaPlayer;

    @Nullable
    @Override
    public IBinder onBind(Intent intent) { return null; }

    @Override
    public int onStartCommand(Intent intent, int flags, int startId) {
        mMediaPlayer = MediaPlayer.create( context: this,
            Settings.System.DEFAULT_RINGTONE_URI);
        mMediaPlayer.setLooping(true);
        mMediaPlayer.start();
        return START_STICKY;
    }

    @Override
    public void onDestroy() {
        super.onDestroy();
        mMediaPlayer.stop();
    }
}

```

Bước 7

Tạo activity có tên BoundServiceActivity để cho phép người dùng hiển thị thông tin thời gian hiện tại của hệ thống

```

package funix.prm.lab;
import android.app.Activity;
import android.content.ComponentName;
import android.content.Context;
import android.content.Intent;
import android.content.ServiceConnection;
import android.os.Bundle;
import android.os.IBinder;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;
import java.util.Date;

public class BoundServiceActivity extends Activity {

    private MyBoundService myBoundService;

    private boolean mIsBound = false;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_bound_service);
        Button btnShowCurrentDate = (Button) findViewById(R.id.btnShowCurrentDate);
        btnShowCurrentDate.setOnClickListener((view) -> {
            if (mIsBound) {
                Date date = myBoundService.getCurrentDate();
                Toast.makeText(context: BoundServiceActivity.this,
                    String.valueOf(date), Toast.LENGTH_SHORT).show();
            }
        });
    }
}

```

```

    @Override
    protected void onStart() {
        super.onStart();
        Intent intent = new Intent(context: this, MyBoundService.class);
        bindService(intent, connection, Context.BIND_AUTO_CREATE);
    }

    @Override
    protected void onStop() {
        super.onStop();
        if (mIsBound) {
            unbindService(connection);
            mIsBound = false;
        }
    }

    private ServiceConnection connection = new ServiceConnection() {
        @Override
        public void onServiceConnected(ComponentName className, IBinder service) {
            MyBoundService.LocalBinder binder = (MyBoundService.LocalBinder) service;
            myBoundService = binder.getService();
            mIsBound = true;
        }

        @Override
        public void onServiceDisconnected(ComponentName arg0) { mIsBound = false; }
    };
}

```

Bước 8

Tạo file layout có tên activity_bound_service.xml để làm giao diện cho BoundServiceActivity

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#C98C00"
    android:gravity="center"
    android:paddingLeft="20dp"
    android:paddingRight="20dp"
    tools:context=".BoundServiceActivity">

    <Button
        android:id="@+id/btnShowCurrentDate"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="Show Current Date" />

</RelativeLayout>
```

Bước 9

Tạo file service có tên MyBoundService để xử lý lấy thông tin thời gian của hệ thống

```
package funix.prm.lab;

import android.app.Service;
import android.content.Intent;
import android.os.Binder;
import android.os.IBinder;

import java.util.Calendar;
import java.util.Date;

public class MyBoundService extends Service {

    private final IBinder binder = new LocalBinder();

    public class LocalBinder extends Binder {
        MyBoundService getService() {
            return MyBoundService.this;
        }
    }

    @Override
    public IBinder onBind(Intent intent) {
        return binder;
    }

    public Date getCurrentDate() {
        return Calendar.getInstance().getTime();
    }
}
```

Bước 10

Cấu hình cho phép ứng dụng sử dụng activity, service trên AndroidManifest.xml

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="funix.prm.lab">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="PRM391xLabs"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".StartServiceActivity" />
        <activity android:name=".BoundServiceActivity" />

        <service android:name=".MyStartService" />
        <service android:name=".MyBoundService" />
    </application>
</manifest>
```

5. KIỂM TRA KẾT QUẢ

Chạy ứng dụng trên AVD hoặc bất cứ thiết bị Android nào mà bạn có. Thực hiện các thao tác sau để kiểm tra kết quả

Started Service

Chọn vào nút "START SERVICE", màn hình sau xuất hiện



START SERVICE

STOP SERVICE

Case 1

Chọn nút "START SERVICE" ==> Thiết bị của bạn sẽ phát ra một bản nhạc chuông

Case 2

Chọn nút "STOP SERVICE" ==> Thiết bị của bạn sẽ tự động tắt bản nhạc chuông đang phát

Bound Service

Chọn vào nút "BOUND SERVICE", màn hình sau xuất hiện



85% 22:27

SHOW CURRENT DATE

Chọn nút "SHOW CURRENT DATE" ==> Thông tin thời gian của hệ thống xuất hiện thông qua Toast



SHOW CURRENT DATE

Mon May 07 22:27:32 GMT+07:00 2018