**Ngô thành hữu 1150070015**

**1) Mã nguồn**

**B1, AndroidManifest.xml**

<?xml version="1.0" encoding="utf-8"?>

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

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

<uses-permission android:name="android.permission.FOREGROUND\_SERVICE"/>

<uses-permission android:name="android.permission.FOREGROUND\_SERVICE\_DATA\_SYNC"/>

<uses-permission android:name="android.permission.POST\_NOTIFICATIONS"/>

<application

android:allowBackup="true"

android:icon="@mipmap/ic\_launcher"

android:label="@string/app\_name"

android:supportsRtl="true"

android:theme="@style/Theme.Bt\_7"

android:usesCleartextTraffic="true">

<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>

<service

android:name=".DownloadService"

android:exported="false"

android:foregroundServiceType="dataSync"/>

<receiver

android:name=".ControlReceiver"

android:exported="false"/>

</application>

</manifest>

**B2, res/values/strings.xml**

<resources>

<string name="app\_name">bt\_7</string>

<string name="link\_download">Link download</string>

<string name="download">Download</string>

</resources>

**B3, res/layout/activity\_main.xml**

<?xml version="1.0" encoding="utf-8"?>

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

android:id="@+id/main"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

android:padding="16dp">

<EditText

android:id="@+id/edtUrl"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:minHeight="48dp"

android:hint="@string/link\_download"

android:inputType="textUri"/>

<Button

android:id="@+id/btnDownload"

android:layout\_width="match\_parent"

android:layout\_height="56dp"

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

</LinearLayout>

**B4, res/layout/noti\_download.xml**

<?xml version="1.0" encoding="utf-8"?>

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

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

android:padding="8dp">

<TextView

android:id="@+id/tvUrl"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:maxLines="1"

android:ellipsize="end"

android:text="Link:" />

<TextView

android:id="@+id/tvProgress"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Complete: 0%" />

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal">

<Button

android:id="@+id/btnPrimary"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:text="Pause" />

<Button

android:id="@+id/btnSecondary"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:text="Cancel" />

</LinearLayout>

</LinearLayout>

**B5, com/example/bt\_7/DL.java**

package com.example.bt\_7;

public final class DL {

public static final String CH\_ID = "dl\_channel\_v3";

public static final int NOTI\_ID = 1001;

public static final String ACTION\_PAUSE = "dl.PAUSE";

public static final String ACTION\_RESUME = "dl.RESUME";

public static final String ACTION\_CANCEL = "dl.CANCEL";

public static final String EXTRA\_URL = "url";

private DL() {}

}

**B6, com/example/bt\_7/ControlReceiver.java**

package com.example.bt\_7;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.Intent;

public class ControlReceiver extends BroadcastReceiver {

@Override public void onReceive(Context c, Intent i) {

String a = i.getAction();

if (DL.ACTION\_PAUSE.equals(a)) DownloadService.signal(DownloadService.Cmd.PAUSE);

if (DL.ACTION\_RESUME.equals(a)) DownloadService.signal(DownloadService.Cmd.RESUME);

if (DL.ACTION\_CANCEL.equals(a)) DownloadService.signal(DownloadService.Cmd.CANCEL);

}

}

**B7, com/example/bt\_7/MainActivity.java**

package com.example.bt\_7;

import android.Manifest;

import android.content.Intent;

import android.content.pm.PackageManager;

import android.os.Build;

import android.os.Bundle;

import android.util.Log;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import androidx.activity.result.ActivityResultLauncher;

import androidx.activity.result.contract.ActivityResultContracts;

import androidx.appcompat.app.AppCompatActivity;

import androidx.core.content.ContextCompat;

public class MainActivity extends AppCompatActivity {

private ActivityResultLauncher<String> notifPerm;

@Override protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

if (Build.VERSION.SDK\_INT >= 33) {

notifPerm = registerForActivityResult(

new ActivityResultContracts.RequestPermission(), g -> {});

notifPerm.launch(Manifest.permission.POST\_NOTIFICATIONS);

}

Button btn = findViewById(R.id.btnDownload);

EditText edt = findViewById(R.id.edtUrl);

btn.setOnClickListener(v -> {

String url = edt.getText().toString().trim();

if (url.isEmpty()) return;

if (Build.VERSION.SDK\_INT >= 33 &&

checkSelfPermission(Manifest.permission.POST\_NOTIFICATIONS)

!= PackageManager.PERMISSION\_GRANTED) {

Toast.makeText(this, "Allow Notifications", Toast.LENGTH\_SHORT).show();

if (notifPerm != null) notifPerm.launch(Manifest.permission.POST\_NOTIFICATIONS);

return;

}

Log.d("DL","CLICK: " + url);

Intent i = new Intent(this, DownloadService.class);

i.putExtra(DL.EXTRA\_URL, url);

ContextCompat.startForegroundService(this, i);

});

}

}

**B8, com/example/bt\_7/DownloadService.java (throttle để thấy % rõ)**

package com.example.bt\_7;

import android.app.\*;

import android.content.Intent;

import android.os.Build;

import android.os.IBinder;

import android.util.Log;

import android.widget.RemoteViews;

import androidx.core.app.NotificationCompat;

import java.io.\*;

import java.net.HttpURLConnection;

import java.net.URL;

import java.util.concurrent.ArrayBlockingQueue;

public class DownloadService extends Service {

public enum Cmd { PAUSE, RESUME, CANCEL }

private static final ArrayBlockingQueue<Cmd> q = new ArrayBlockingQueue<>(1);

public static void signal(Cmd c){ q.offer(c); }

private volatile boolean paused=false, canceled=false;

private String url;

private File file;

private long downloaded=0, total=-1;

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

@Override public int onStartCommand(Intent intent,int flags,int startId){

url = intent.getStringExtra(DL.EXTRA\_URL);

Log.d("DL","onStartCommand url=" + url);

if (url==null || url.isEmpty()) return START\_NOT\_STICKY;

createChannel();

startForeground(DL.NOTI\_ID, buildNoti(0,false));

String name="download.bin";

try {

String p=new URL(url).getPath(); int i=p.lastIndexOf('/');

if(i>=0 && i+1<p.length()) name=p.substring(i+1);

} catch(Exception ignore){}

File dir = getExternalFilesDir(null);

if (dir==null) dir=getFilesDir();

file = new File(dir, name);

new Thread(this::loop).start();

new Thread(this::listen).start();

return START\_NOT\_STICKY;

}

private void listen(){

try{

while(true){

Cmd c=q.take();

if(c==Cmd.PAUSE){ paused=true; notifyProgress(percent(),true); }

if(c==Cmd.RESUME){ paused=false; notifyProgress(percent(),false); }

if(c==Cmd.CANCEL){ canceled=true; stopSelf(); return; }

}

}catch(InterruptedException ignored){}

}

private void loop(){

try{

if(file.exists()) downloaded=file.length();

while(true){

if(canceled){ file.delete(); return; }

if(paused){ Thread.sleep(200); continue; }

HttpURLConnection conn=(HttpURLConnection)new URL(url).openConnection();

if(downloaded>0) conn.setRequestProperty("Range","bytes="+downloaded+"-");

conn.connect();

Log.d("DL","HTTP " + conn.getResponseCode());

if(total<=0){

String cr=conn.getHeaderField("Content-Range");

String cl=conn.getHeaderField("Content-Length");

if(cr!=null && cr.contains("/")) total=parseLongSafe(cr.substring(cr.lastIndexOf('/')+1));

if(total<=0) total=parseLongSafe(cl);

}

try(InputStream in=conn.getInputStream();

RandomAccessFile raf=new RandomAccessFile(file,"rw")){

if(downloaded>0) raf.seek(downloaded);

byte[] buf=new byte[4\*1024]; // throttle: 4KB

while(true){

if(canceled){ file.delete(); return; }

if(paused){ break; }

int r=in.read(buf);

if(r==-1){

notifyProgress(100,false);

try { Thread.sleep(5000); } catch (InterruptedException ignored) {}

Log.d("DL","DONE");

return;

}

raf.write(buf,0,r);

downloaded+=r;

notifyProgress(percent(),false);

try { Thread.sleep(300); } catch (InterruptedException ignored) {} // throttle

}

} finally { conn.disconnect(); }

}

}catch(Exception e){

Log.e("DL","ERROR", e);

notifyFail();

} finally { stopSelf(); }

}

private void notifyProgress(int pct, boolean isPaused){

NotificationManager nm=(NotificationManager)getSystemService(NOTIFICATION\_SERVICE);

nm.notify(DL.NOTI\_ID, buildNoti(pct,isPaused));

}

private Notification buildNoti(int pct, boolean isPaused){

RemoteViews rv = new RemoteViews(getPackageName(), R.layout.noti\_download);

rv.setTextViewText(R.id.tvUrl, "Link: " + url);

rv.setTextViewText(R.id.tvProgress, pct>=0 ? ("Complete: " + pct + "%") : "Starting…");

rv.setTextViewText(R.id.btnPrimary, isPaused ? "Resume" : "Pause");

PendingIntent piPrimary = PendingIntent.getBroadcast(

this, 1, new Intent(this, ControlReceiver.class)

.setAction(isPaused ? DL.ACTION\_RESUME : DL.ACTION\_PAUSE),

PendingIntent.FLAG\_UPDATE\_CURRENT | PendingIntent.FLAG\_IMMUTABLE);

PendingIntent piCancel = PendingIntent.getBroadcast(

this, 2, new Intent(this, ControlReceiver.class).setAction(DL.ACTION\_CANCEL),

PendingIntent.FLAG\_UPDATE\_CURRENT | PendingIntent.FLAG\_IMMUTABLE);

rv.setOnClickPendingIntent(R.id.btnPrimary, piPrimary);

rv.setOnClickPendingIntent(R.id.btnSecondary, piCancel);

NotificationCompat.Builder b = new NotificationCompat.Builder(this, DL.CH\_ID)

.setSmallIcon(android.R.drawable.stat\_sys\_download)

.setStyle(new NotificationCompat.DecoratedCustomViewStyle())

.setCustomContentView(rv)

.setOnlyAlertOnce(true)

.setOngoing(pct>=0 && pct<100);

if (Build.VERSION.SDK\_INT >= 34) {

b.setForegroundServiceBehavior(

NotificationCompat.FOREGROUND\_SERVICE\_IMMEDIATE);

}

return b.build();

}

private void createChannel(){

if(Build.VERSION.SDK\_INT>=26){

NotificationChannel ch=new NotificationChannel(

DL.CH\_ID,"Downloads",NotificationManager.IMPORTANCE\_HIGH);

ch.setDescription("Foreground downloads");

((NotificationManager)getSystemService(NOTIFICATION\_SERVICE))

.createNotificationChannel(ch);

}

}

private int percent(){ return total>0 ? (int)Math.max(0, Math.min(100,(downloaded\*100L)/total)) : 0; }

private static long parseLongSafe(String s){ try{ return Long.parseLong(s); }catch(Exception e){ return -1; } }

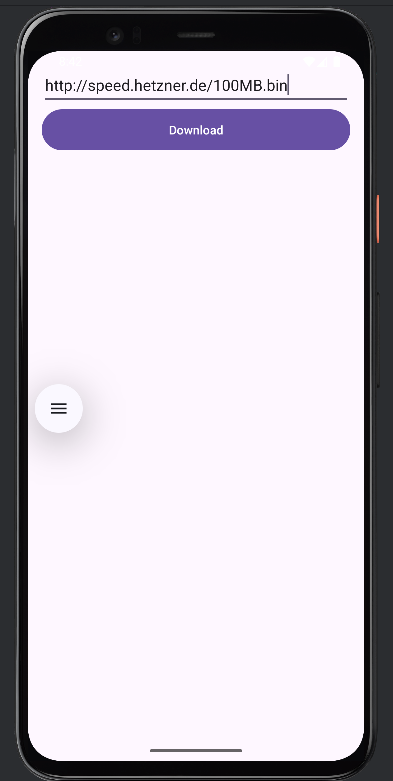
private void notifyFail(){ notifyProgress(-1,false); }

}

**2) Ảnh kết quả cần chụp**

**Link test:** <http://speed.hetzner.de/100MB.bin>

Trang chủ :



Download:

A screenshot of a phone

Description automatically generated