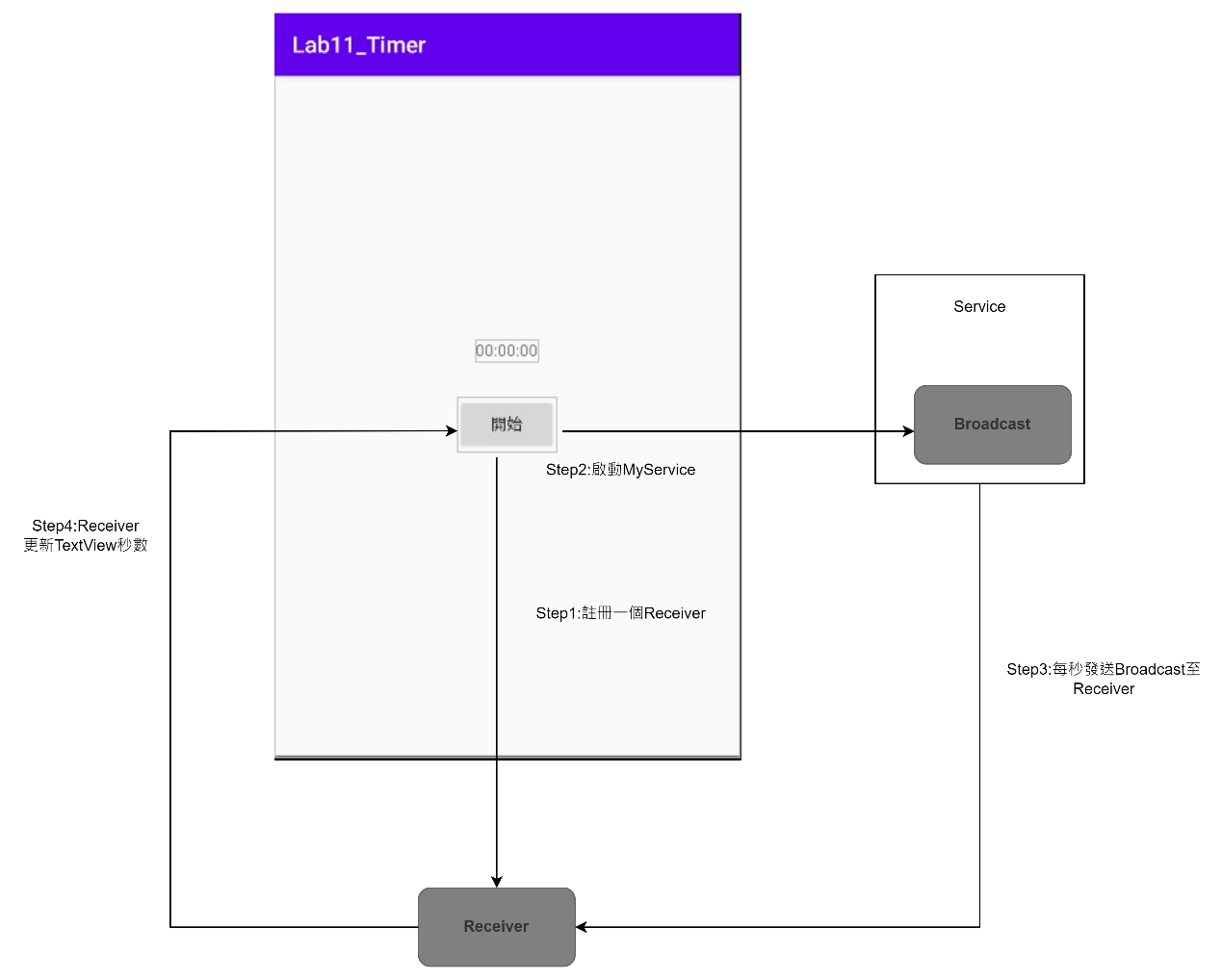
**計時器**

MainActivity 會註冊一個Receiver，接收廣播後會得到秒數{整數值}，並將秒數呈現於TextView。

MainActivity按下開始「開始SERVICE」按鈕後啟動MyService。

後台的MyService會建立一個Thead開始讀秒。

每秒MyService都會透過Broadcast Receiver ，送一個訊息給前台的MainActivity，來讓MainActivity更新秒數。



**計時器畫面設計**

新建專案，以及如圖1-1所示對應的class和 xml檔。

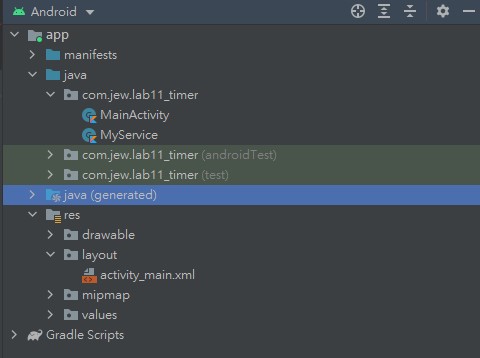


圖1-1計時器專案架構

繪製activity\_main.xml，如圖1-2所示。

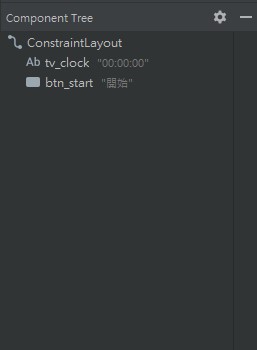
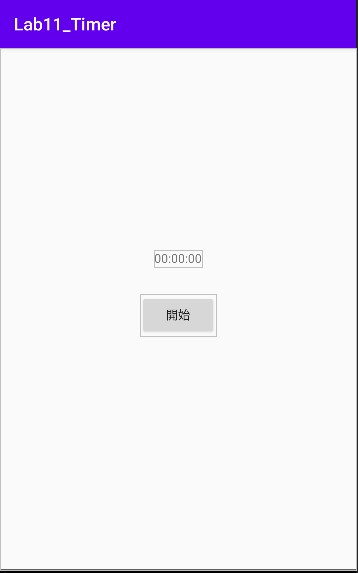


圖1-2 計時器預覽畫面與布局元件樹

對應的xml如下:

<?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"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity">

<TextView

android:id="@+id/tv\_clock"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="00:00:00"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintLeft\_toLeftOf="parent"

app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.4" />

<Button

android:id="@+id/btn\_start"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="32dp"

android:text="開始"

app:layout\_constraintEnd\_toEndOf="@id/tv\_clock"

app:layout\_constraintStart\_toStartOf="@id/tv\_clock"

app:layout\_constraintTop\_toBottomOf="@id/tv\_clock" />

</androidx.constraintlayout.widget.ConstraintLayout>

**接收廣播**

Step1撰寫MainActivtiy，一個啟動Service 的按鈕，與註冊一個BroadcastReceiver，當BroadcastReceiver收到識別字串為「MyMessage」的Intent時，會取出Intent中夾帶的秒數資訊，並使用TextView做顯示。

package com.jew.lab11\_timer

import android.content.BroadcastReceiver

import android.content.Context

import android.content.Intent

import android.content.IntentFilter

import androidx.appcompat.app.AppCompatActivity

import android.os.Bundle

import android.widget.Toast

import kotlinx.android.synthetic.main.activity\_main.\*

class MainActivity : AppCompatActivity() {

//計數器狀態

private var flag = false

//建立BroadcastReceiver物件

private val receiver: BroadcastReceiver = object : BroadcastReceiver() {

//在onReceive()中加入接收廣播後要執行的動作

override fun onReceive(context: Context, intent: Intent) {

//解析Intent取得計時資訊

val b = intent.extras?: return

tv\_clock?.text = String.format("%02d:%02d:%02d",

b.getInt("H"), b.getInt("M"), b.getInt("S"))

}

}

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

//註冊Receiver來接收有『MyMessage』識別字串的廣播

registerReceiver(receiver, IntentFilter("MyMessage"))

//取得Service狀態

flag = MyService.flag

btn\_start.text = if(flag) "暫停" else "開始"

btn\_start.setOnClickListener {

flag = !flag

btn\_start.text = if (flag) "暫停" else "開始"

//啟動Service

startService(Intent(this, MyService::class.java).putExtra("flag", flag))

Toast.makeText(this, if(flag)"計時開始" else "計時暫停", Toast.LENGTH\_SHORT).show()

}

}

override fun onDestroy() {

super.onDestroy()

//註銷Receiver

unregisterReceiver(receiver)

}

}

Step2撰寫Myservice，建立一個Thread每秒發送一次廣播，並把累加的秒數發送到Receiver

package com.jew.lab11\_timer

import android.app.Service

import android.content.Intent

import android.os.Bundle

import android.os.IBinder

class MyService : Service() {

companion object {

//計數器狀態

var flag: Boolean = false

}

//計數器數值（時、分、秒）

private var hour = 0

private var minute = 0

private var second = 0

override fun onBind(intent: Intent): IBinder {

TODO("Return the communication channel to the service.")

}

override fun onStartCommand(intent: Intent, flags: Int, startID: Int): Int {

flag = intent.getBooleanExtra("flag", false)

object : Thread() {

override fun run() {

while (flag) {

try {

//延遲1s

sleep(1000)

} catch (e: InterruptedException) {

e.printStackTrace()

}

//計數器加一

second++

//秒數大於60進位

if (second >= 60) {

second = 0

minute++

//分鐘數大於60進位

if (minute >= 60) {

minute = 0

hour++

}

}

//發送帶有『MyMessage』識別字串的廣播

val intent = Intent("MyMessage")

//將時間放入Bundle

val bundle = Bundle()

bundle.putInt("H", hour)

bundle.putInt("M", minute)

bundle.putInt("S", second)

//發送廣播

sendBroadcast(intent.putExtras(bundle))

}

}

}.start()

//自動重啟，但不會保留Intent

return START\_STICKY

}

}