

BÀI 2. PHIẾU BÀI TẬP

BÀI TẬP TRƯỚC KHI LÊN LỚP: Bài 2.1 đến 2.4

BÀI TẬP THỰC HIỆN TRÊN LỚP: Bài 2.5

BÀI TẬP SAU KHI LÊN LỚP: Bài 2.6

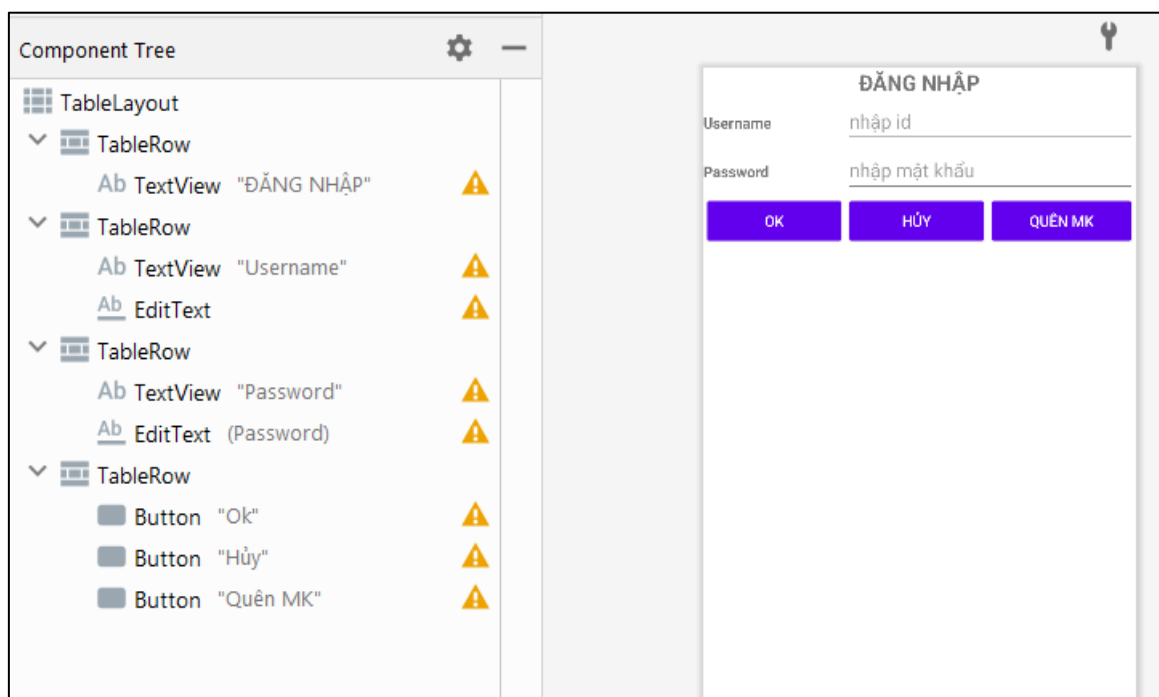
MỤC LỤC

Bài tập 2.1.	Thiết kế giao diện màn hình login với TableLayout	1
Bài tập 2.2.	Thiết kế giao diện màn hình login với LinearLayout	2
Bài tập 2.3.	Thiết kế giao diện màn hình login với RelativeLayout	1
Bài tập 2.4.	Thiết kế giao diện không có gợi ý	2
Bài tập 2.5.	Vòng đời ứng dụng Android.....	6
Bài tập 2.6.	Lưu thông tin trạng thái ứng dụng	3

Bài tập 2.1. Thiết kế giao diện màn hình login với TableLayout

- TableLayout có độ rộng các cột bằng nhau và lấp đầy màn hình như minh họa.
- Giao diện **có 4 hàng và 3 cột**
- Hàng 1: Chứa 1 ô TextView chiếm độ **rộng 3 ô**. Chữ hiển thị In đậm, kích thước chữ 20sp, nội dung căn giữa.
- Hàng 2: Chứa TextView (text:username) và 1 EditText.
 - o TextView chiếm độ rộng 1 ô, chữ in đậm, kích thước 14sp;
 - o EditText (hint: nhập ID) chiếm **độ rộng 2 ô** trong dòng
- Hàng 3 tương tự hàng 2: 1 TextView (text:password) và 1 EditText (hint: nhập mật khẩu)
- Hàng 4 có 3 nút lệnh độ rộng đều nhau. Chữ căn giữa trong không gian hiển thị nút lệnh, kiểu chữ đậm, nền chữ màu tím (có thể đặt màu nền khác mẫu)

Hướng dẫn thực hiện: Gợi ý cấu trúc câu thiết kế như sau:

**Phác thảo thiết kế:**

Thiết lập các cột có độ rộng bằng nhau:

`android:stretchColumns="*"`. Không cần chỉ định chiều cao hay độ rộng của các View. Sử dụng các thuộc tính phù hợp để điều chỉnh các hiển thị theo chiều rộng và chiều cao. Minh họa gợi ý thiết kế như sau:

TextView: ĐĂNG NHẬP	
<code>layout_span="3"</code>	
TextView:Username	EditText: username <code>layout_span="2"</code>

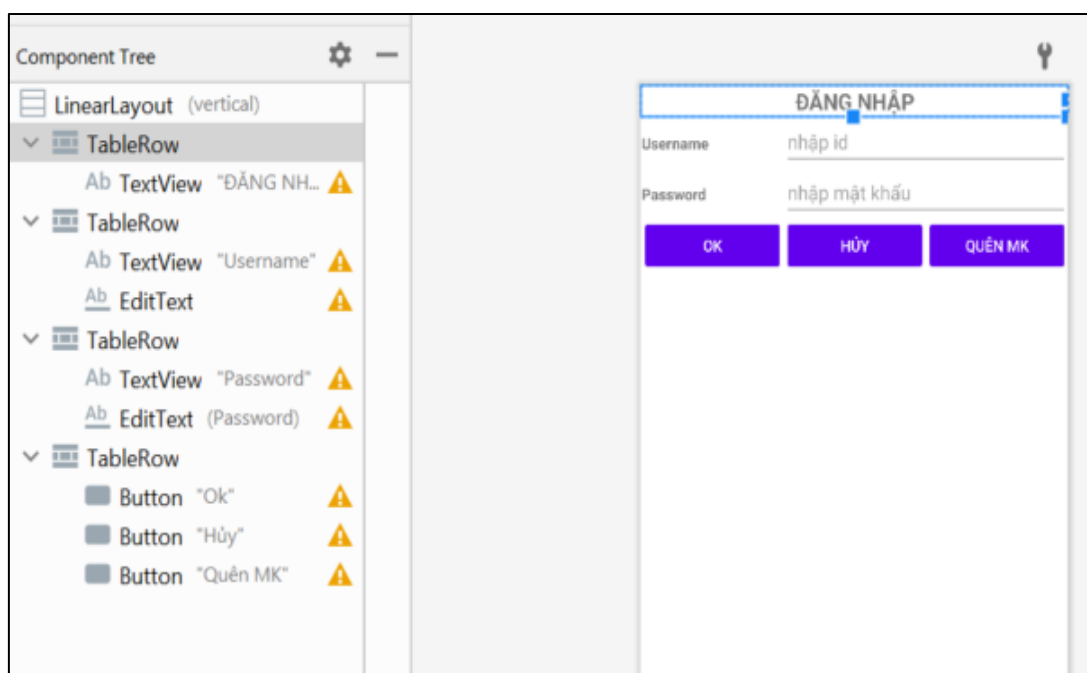
TextView: Password	EditText: Password <code>layout span="2"</code>	
OK	HỦY	Quên MK

- Các nội dung in đậm: `android:textStyle="bold"`
- Kích thước chữ : `android:textSize="14sp"`. Thay đổi kích thước theo yêu cầu bài toán
- Căn giữa không gian hiển thị: `android:gravity="center"`

Bài tập 2.2. Thiết kế giao diện màn hình login với LinearLayout

- LinearLayout có độ rộng các cột bằng nhau và lấp đầy màn hình như minh họa.
- Giao diện có 4 hàng và được bố trí hài hoà như hình minh họa.
- Hàng 1: Chứa 1 ô TextView chiếm độ **toàn không gian chiều rộng**. Chữ hiển thị In đậm, kích thước chữ 20sp, nội dung căn giữa.
- Hàng 2: Chứa TextView (text:username) và 1 EditText theo tỷ lệ 1/3
 - o TextView chiếm độ rộng 1 ô, chữ in đậm, kích thước 14sp chiếm 1 phần
 - o EditText (hint: nhập ID) chiếm **2 phần** trong dòng
- Hàng 3 tương tự hàng 2: 1 TextView (text:password) và 1 EditText (hint: nhập mật khẩu)
- Hàng 4 có 3 nút lệnh độ rộng đều nhau. Chữ căn giữa trong không gian hiển thị nút lệnh, kiểu chữ đậm, nền chữ màu tím (có thể đặt màu nền khác mẫu)

Hướng dẫn thực hiện: Gợi ý cấu trúc câu thiết kế như sau:



Phác thảo thiết kế

Ứng dụng có không gian toàn màn hình cần thiết lập chiều cao và chiều rộng của Layout thông qua các thuộc tính sau:

```
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="vertical"
```

Dòng cũng cần thiết lập chiều cao và chiều rộng và chỉ định không gian chiếm tổng thể

```
android:layout_height="wrap_content"
android:layout_width="match_parent"
android:weightSum="3"
```

Các View con cũng chỉ định kích thước và không gian chiếm chia theo tỷ lệ

```
android:layout_weight="1"
```

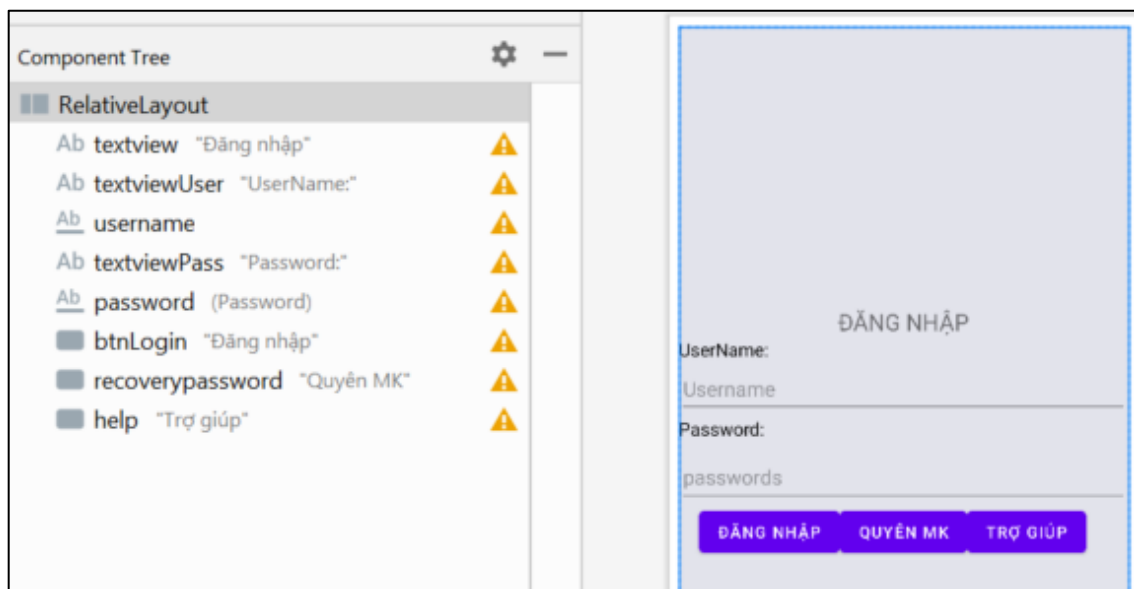
Phác thảo thiết kế:

TextView : ĐĂNG NHẬP		
<pre>ndroid:layout_width="match_parent" android:layout_height="wrap_content" android:layout_gravity="center" android:layout_weight="1"</pre>		
TextView: Username <pre>android:layout_height="wrap_content" android:layout_width="0dp" android:layout_weight="1"</pre>	EditText: Username <pre>android:layout_height="wrap_content" android:layout_width="0dp" android:layout_weight="2"</pre>	
TextView :Password <pre>android:layout_height="wrap_content" android:layout_width="0dp" android:layout_weight="1"</pre>	EditText: Password <pre>android:layout_height="wrap_content" android:layout_width="0dp" android:layout_weight="2"</pre>	
Button: OK <pre>android:layout_height="wrap_content" android:layout_width="0dp" android:layout_weight="1"</pre>	Button: HUY <pre>android:layout_height="wrap_content" android:layout_width="0dp" android:layout_weight="1"</pre>	Button: Quên MK <pre>android:layout_height="wrap_content" android:layout_width="0dp" android:layout_weight="1"</pre>

Bài tập 2.3. Thiết kế giao diện màn hình login với RelativeLayout

- RelativeLayout có các điều khiển sắp xếp theo các vị trí tương đối nhau theo chiều từ trên xuống dưới, từ trái qua phải.
- Giao diện xuất hiện giữa màn hình được sắp xếp và được bố trí hài hoà như hình minh hoạ.
- Trên cùng giao diện là 1 TextView (text: Đăng nhập) chiều rộng **toàn không gian rộng thiết bị**. Chữ hiển thị In đậm, kích thước chữ 20sp, nội dung căn giữa.
- TextView (text: username) ở vị trí bên dưới TextView (text: Đăng nhập). Chiều cao/rộng vừa nội dung hiển thị; cỡ chữ 16sp, màu chữ tùy chọn.
- EditText (hint: Username) chiều cao 48dp; chiều rộng toàn màn hình. Phía dưới TextView (text: username) .
- TextView (text: Password) ở vị trí bên dưới EditText (hint: Username). Chiều cao/rộng vừa nội dung hiển thị; cỡ chữ 16sp, màu chữ tùy chọn.
- EditText (hint: passwords) chiều cao 48dp; chiều rộng toàn màn hình. Phía dưới TextView (text: Password) .
- Tiếp theo 3 nút lệnh thẳng hàng. Chữ căn giữa trong không gian hiển thị nút lệnh, kiểu chữ đậm, nền chữ màu tím (có thể đặt màu nền khác mẫu)

Hướng dẫn thực hiện: Gợi ý cấu trúc câu thiết kế như sau:



Phác thảo thiết kế:

ĐĂNG NHẬP		
TextView: Đăng nhập <code>android:id="@+id/textview"</code> <code>android:text="Đăng nhập"</code> <code>android:layout_centerHorizontal="true"</code> <code>android:textSize="20sp"</code> <code>android:textAllCaps="true"</code> <code>android:layout_width="match_parent"</code> <code>android:layout_height="wrap_content"</code>		
TextView: Username <code>android:id="@+id/textviewUser"</code> <code>android:layout_width="wrap_content"</code> <code>android:layout_height="wrap_content"</code> <code>android:text="UserName:"</code> <code>android:layout_below="@id/textview"</code> <code>android:textColor="@color/material_dynamic_neutral0"</code> <code>android:textSize="16sp"</code>		
EditText:Nhập Username <code>android:id="@+id/username"</code> <code>android:layout_width="match_parent"</code> <code>android:layout_height="48dp"</code> <code>android:layout_below="@id/textviewUser"</code> <code>android:hint="Username"</code>		
TextView: Password <code>android:id="@+id/textviewPass"</code> <code>android:layout_width="wrap_content"</code> <code>android:layout_height="wrap_content"</code> <code>android:text="Password:"</code> <code>android:layout_below="@id/username"</code> <code>android:textSize="16sp"</code> <code>android:textColor="@color/material_dynamic_neutral0"</code>		
EditText:Nhập password <code>android:id="@+id/password"</code> <code>android:layout_width="match_parent"</code> <code>android:layout_height="48dp"</code> <code>android:layout_below="@+id/textviewPass"</code> <code>android:layout_marginTop="8dp"</code> <code>android:hint="passwords"</code> <code>android:inputType="textPassword"</code>		
Button: OK <code>android:id="@+id/btnLogin"</code> <code>android:layout_below="@id/password"</code> <code>android:layout_toLeftOf=</code>	Button: HỦY <code>android:id="@+id/recoverypassword"</code> <code>android:text="Quyên MK"</code> <code>android:layout_centerHorizontal="true"</code>	Button: Quên MK <code>android:id="@+id/help"</code> <code>android:text="Trợ giúp"</code> <code>android:layout_below="@id/password"</code> <code>android:layout_toRightOf="@id/recoverypa</code>

<pre>"@id/recoverypassword" android:text="Đăng nhập" android:layout_width="wrap_content" android:layout_height="wrap_content" android:layout_width="wrap_content" android:layout_height="wrap_content"</pre>	<pre>android:layout_below="@id/password" android:layout_width="wrap_content" android:layout_height="wrap_content"</pre>	<pre>password" android:layout_width="wrap_content" android:layout_height="wrap_content"</pre>
--	---	---

Bài tập 2.4. Thiết kế giao diện không có gợi ý

Yêu cầu:

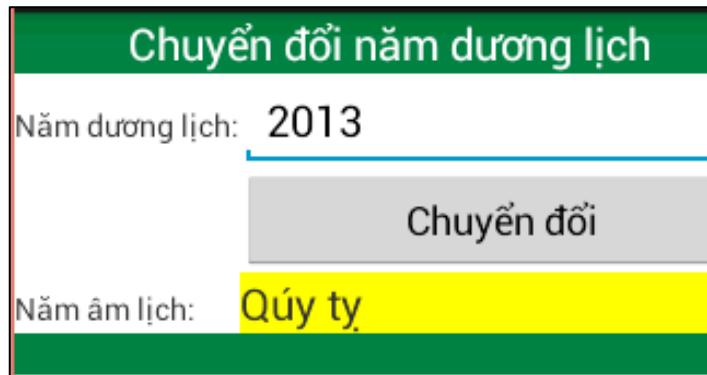
- Phác thảo giao diện
- Lựa chọn layout
- Chọn các thuộc tính mô tả phù hợp cho các View

a. Thiết kế màn hình Login có giao diện sau:

Màn hình gồm:

- 2 ô EditText nhập liệu tên đăng nhập và mật khẩu.
- 1 Checkbox lưu thông tin
- 2 Button nhập và thoát

b. Thiết kế phần mềm chuyển đổi năm Âm lịch sang năm dương lịch có mẫu thiết kế sau:

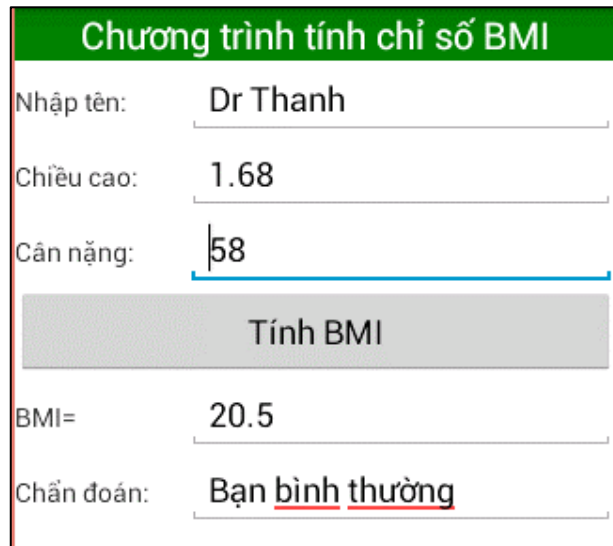


The screenshot shows a mobile application titled "Chuyển đổi năm dương lịch" (Convert solar year). It features a text input field for "Năm dương lịch:" (Solar year) containing the value "2013". Below this is a grey button labeled "Chuyển đổi" (Convert). Under the button is another text input field for "Năm âm lịch:" (Lunar year) containing the value "Quý ty". The entire interface is set against a green background with white text.

Màn hình gồm:

- 1 ô EditText nhập năm dương lịch
- 1 TextView hiển thị kết quả chuyển đổi năm dương lịch sang năm âm lịch
- 1 Button chuyển đổi

c. Thiết kế màn hình chương trình tính chỉ số cơ thể BMI



The screenshot shows a mobile application titled "Chương trình tính chỉ số BMI" (BMI Calculation Program). It contains several text input fields: "Nhập tên:" (Enter name) with "Dr Thanh", "Chiều cao:" (Height) with "1.68", and "Cân nặng:" (Weight) with "58". Below these is a grey button labeled "Tính BMI" (Calculate BMI). Under the button, there are two more text input fields: "BMI=" with "20.5" and "Chẩn đoán:" (Diagnosis) with "Bạn bình thường" (You are normal). The interface has a green header and a white body.

Màn hình gồm:

- Các EditText nhập tên, chiều cao, cân nặng. 2 EditText nhận kết quả chỉ số BMI và chuẩn đoán tình trạng cơ thể
- 1 Button tính BMI

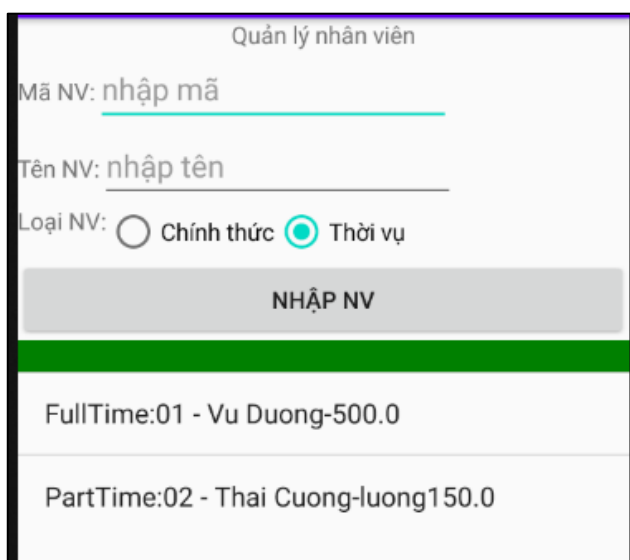
d. Thiết kế phần mềm hiển thị thông tin cá nhân



Màn hình bao gồm:

- 2 EditText nhập họ tên và CMND
- 3 RadioButton lựa chọn Bằng cấp
- 3 CheckBox cho phép lựa chọn sở thích
- 1 EditText cho phép nhập thông tin bổ sung
- 1 Button gửi thông tin

e. Thiết kế màn hình quản lý thông tin cá nhân:



Màn hình bao gồm:

- 2 ô EditText nhập liệu tên đăng nhập và mật khẩu.
- 1 Checkbox lưu thông tin

- 2 Button nhập và thoát

f. Thiết kế màn hình quản lý sản phẩm

Quản lý sản phẩm

Danh mục: 2 - Iphone

Mã Sp: IP5

Tên Sp: iPhone 5

Nhập SP

Danh sách sản phẩm theo danh mục

IP3 - Iphone 3

IP4 - Iphone 4

IP5 - Iphone 5

Màn hình bao gồm:

- 1 Spinner lưu trữ danh mục sản phẩm
- 2 ô EditText nhập liệu mã sản phẩm và tên sản phẩm
- 1 Button nhập sản phẩm

g. Thiết kế màn hình quản lý công việc

Ex_Module2_Bai23

Quản lý công việc hàng tuần

Công việc:

Nội dung:

Ngày HT: 22/02/2013 Date

Giờ HT: 8:30 AM Time

Thêm CV

Danh sách công việc:

Rửa chén - 23/2/2013 - 9:30 AM

Lau nhà - 23/2/2013 - 10:00 AM

Học Android - 23/2/2013 - 17:00 PM

My Datetime picker

Feb 21 2012

Mar 22 2013

Apr 23 2014

Done

Set time

16:32

17:33

18:34

Done

Màn hình bao gồm:

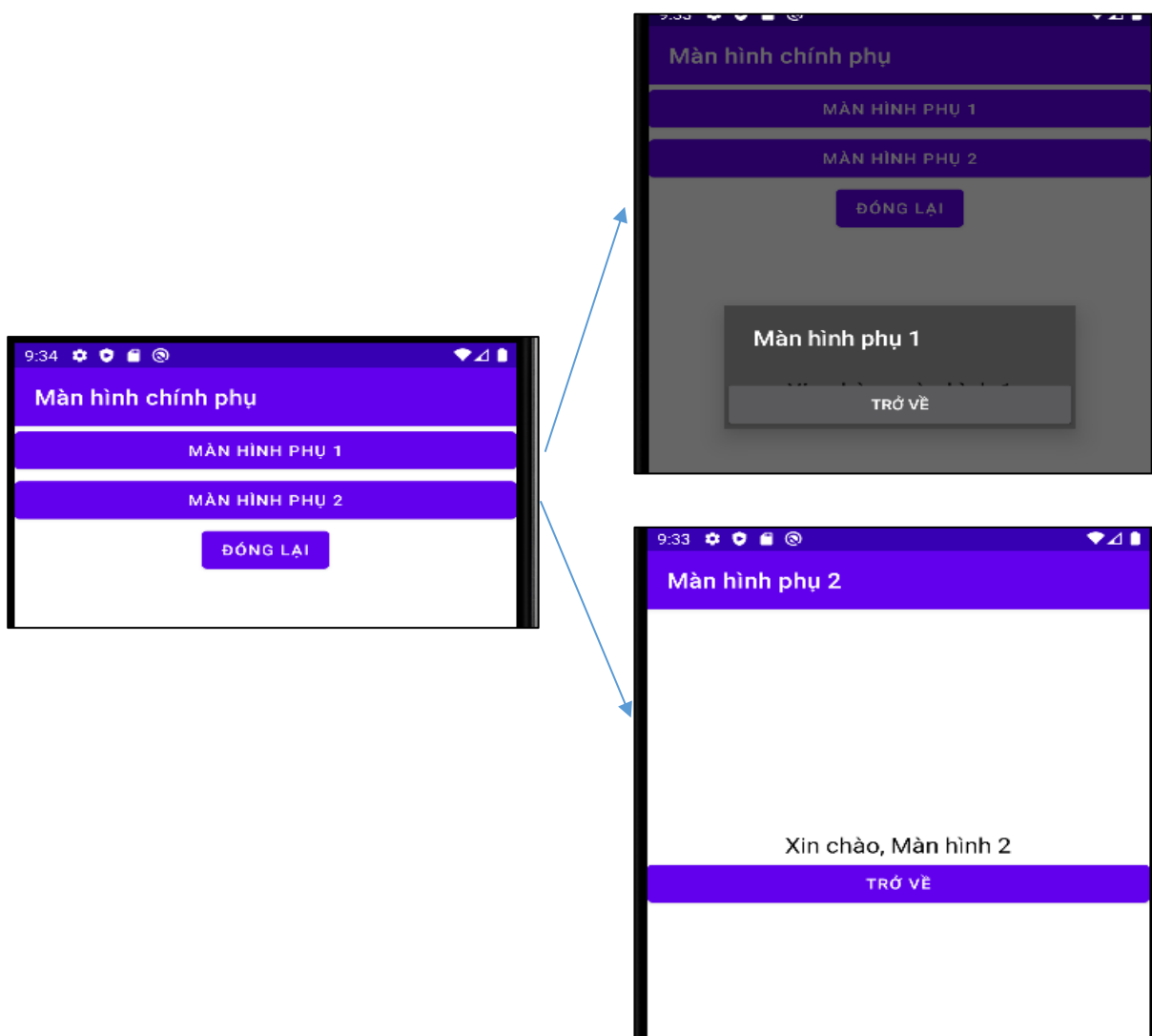
- 2 ô EditText nhập liệu công việc và nội dung công việc

- 2 EditText nhập kết quả ngày và giờ làm việc. Kế tiếp 2 nút lệnh Date và Time
- 1 nút lệnh Thêm CV để thêm công việc
- 1 TextView có nền màu xanh, chữ trắng hiển thị nội dung danh sách công việc
- 1 ListView hiển thị nội dung công việc.

Bài tập 2.5. Vòng đời ứng dụng Android.

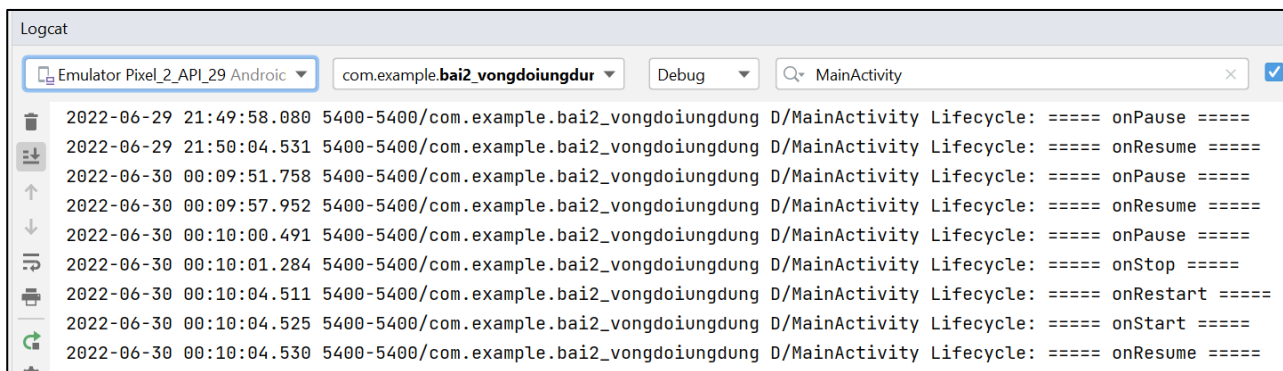
Xây dựng ứng dụng kiểm tra các trạng thái của Activity Màn hình chính.

- Kiểm tra xem trạng thái **Tiếp tục** khi Activity ở đỉnh của ngăn xếp và nhận tương tác.
- Kiểm tra trạng **thái dừng** khi gọi màn hình phụ 1, màn hình Phụ 1 sẽ che mất 1 phần màn hình chính
- Kiểm tra trạng thái **Dừng** khi màn hình chính mất tương tác do gọi màn hình phụ 2 hiển thị toàn màn hình và không cho phép tương tác với màn hình chính.



Yêu cầu:

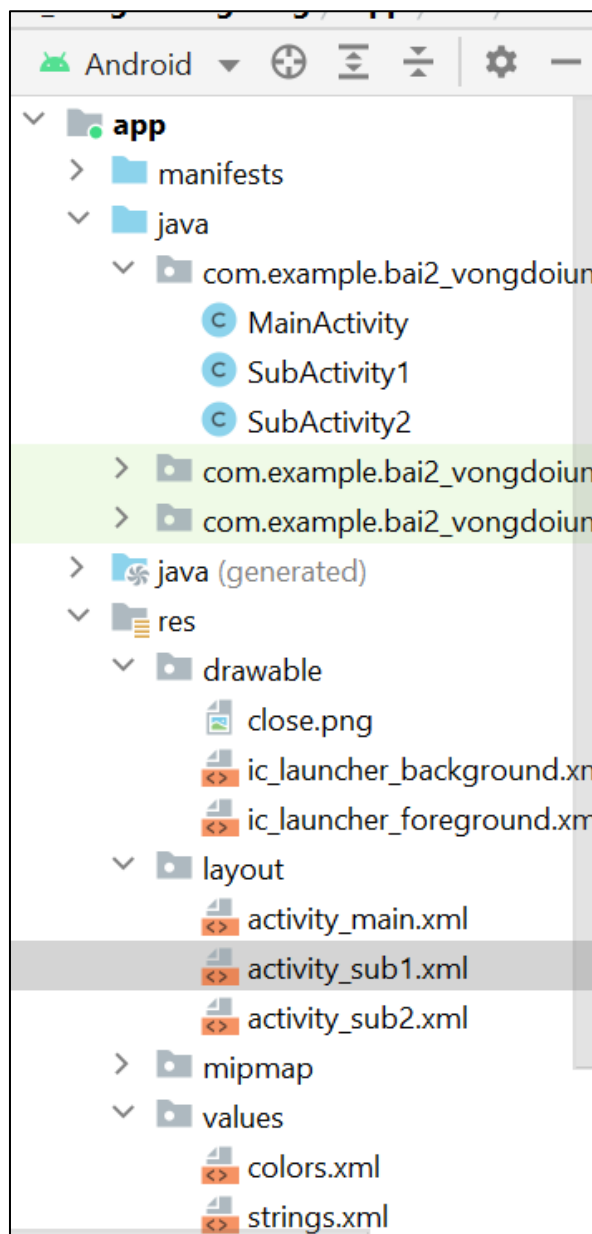
1. Thiết kế giao diện: theo mẫu gợi ý, phần màu nền và màu chữ có thể thay đổi. cụ thể như sau
 - a. Màn hình chính phụ: có 3 nút lệnh: màn hình phụ 1, màn hình phụ 2 và đóng lại
 - b. Màn hình Phụ 1 có 1 nút lệnh trở về
 - c. Màn hình Phụ 2 có 1 Text hiển thị lời chào và 1 nút lệnh trở về.
2. Thực hiện viết mã lệnh
 - a. Ghi đè các sự kiện trong vòng đời hoạt động của Activity màn hình chính hiển thị thông báo ra cửa sổ Logcat hoặc hiển thị thông tin xuất hiện nhân trên màn hình (dùng Toast)
 - b. Quan sát và ghi nhớ vòng đời Activity. Ví dụ cửa sổ **logcat** lọc theo Tag: MainActivity LifeCircle



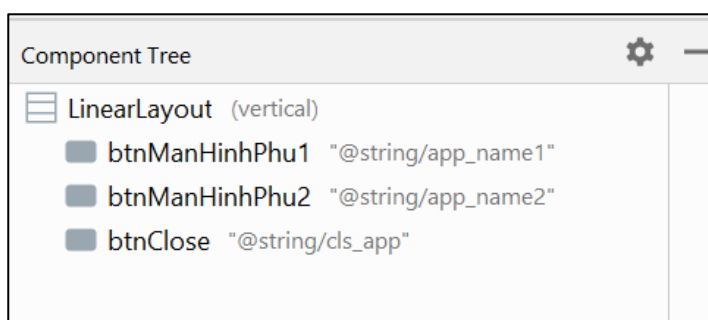
```
Logcat
Emulator Pixel_2_API_29 Android  com.example.bai2_vongdoiungdung  Debug  MainActivity
2022-06-29 21:49:58.080 5400-5400/com.example.bai2_vongdoiungdung D/MainActivity Lifecycle: ===== onPause =====
2022-06-29 21:50:04.531 5400-5400/com.example.bai2_vongdoiungdung D/MainActivity Lifecycle: ===== onResume =====
2022-06-30 00:09:51.758 5400-5400/com.example.bai2_vongdoiungdung D/MainActivity Lifecycle: ===== onPause =====
2022-06-30 00:09:57.952 5400-5400/com.example.bai2_vongdoiungdung D/MainActivity Lifecycle: ===== onResume =====
2022-06-30 00:10:00.491 5400-5400/com.example.bai2_vongdoiungdung D/MainActivity Lifecycle: ===== onPause =====
2022-06-30 00:10:01.284 5400-5400/com.example.bai2_vongdoiungdung D/MainActivity Lifecycle: ===== onStop =====
2022-06-30 00:10:04.511 5400-5400/com.example.bai2_vongdoiungdung D/MainActivity Lifecycle: ===== onRestart =====
2022-06-30 00:10:04.525 5400-5400/com.example.bai2_vongdoiungdung D/MainActivity Lifecycle: ===== onStart =====
2022-06-30 00:10:04.530 5400-5400/com.example.bai2_vongdoiungdung D/MainActivity Lifecycle: ===== onResume =====
```

Hướng dẫn thực hiện.

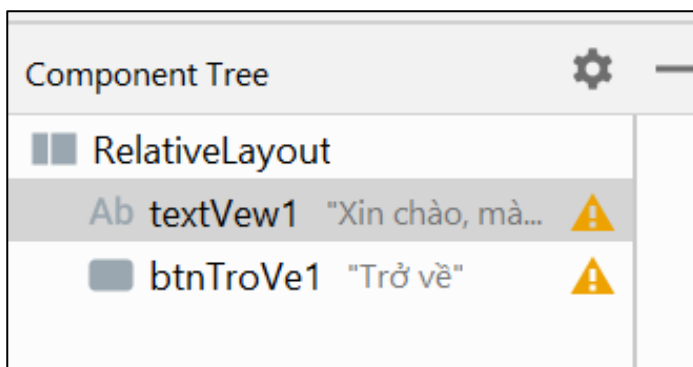
1. Thiết kế giao diện sinh viên tự làm. Để tiện theo dõi các em có thể đặt tên các điều khiển theo gợi ý sau
 - a. Cấu trúc ứng dụng



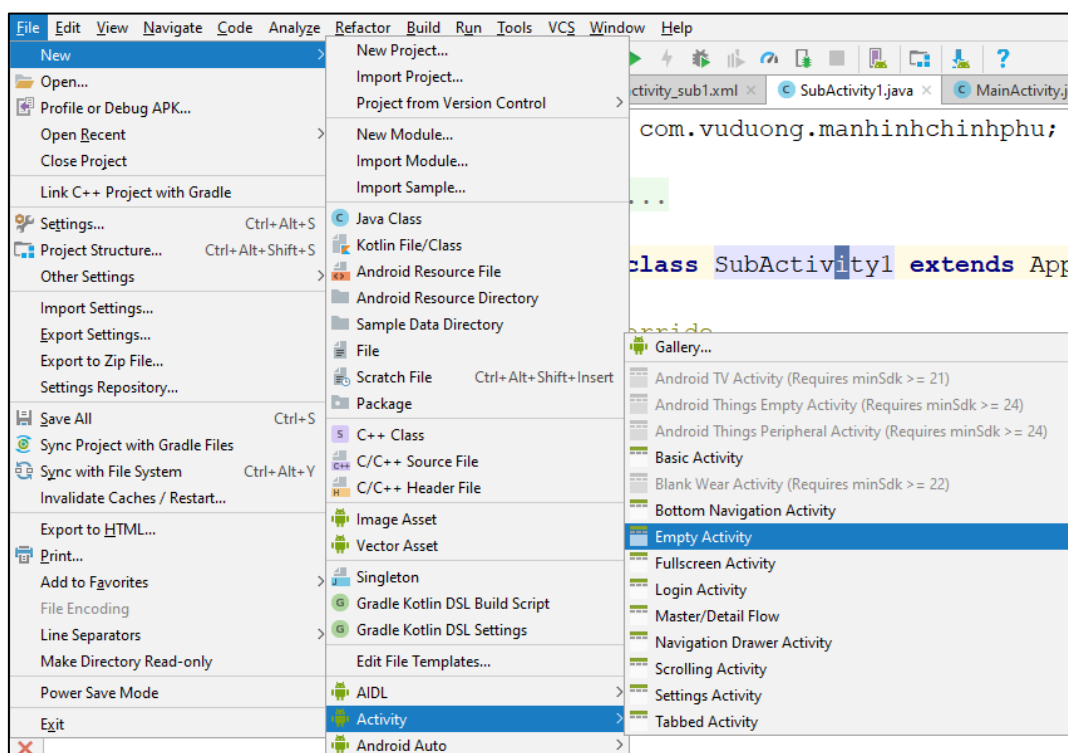
b. Màn hình chính:



Màn hình SubActivity1: **activity_sub1.xml**



Cách tạo: Để thêm 1 hoạt động mới vào dự án thực hiện chọn File\New\Activity\Empty Activity như hình vẽ dưới.



Hình 2.1 Hình mô tả các bước thêm một Hoạt động mới vào ứng dụng

- Đặt tên cho hoạt động trong lựa chọn activity name: SubActivity 1 và **layout name** tương ứng là activity_sub1 như biểu diễn hình sau:

Creates a new empty activity

Activity Name:

☒ Generate Layout File

Layout Name:

☐ Launcher Activity

☒ Backwards Compatibility (AppCompat)

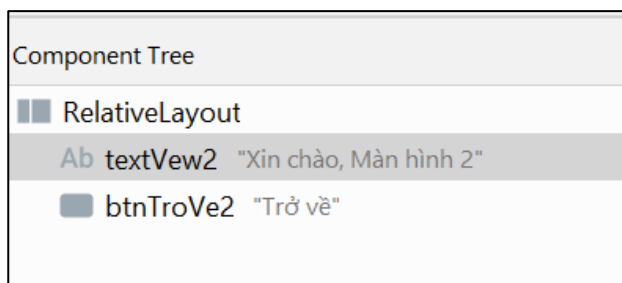
Package name:

Source Language:

Hình 2.2 Hình mô tả bước đặt tên hoạt động phụ 1

- Thực hiện thêm các điều khiển và thiết kế giao diện cho “Màn hình phụ 1”. Lặp lại các bước như hoạt động trên để thiết kế giao diện hoạt động phụ 2 SubActivity2. File\New\Activity\Empty Activity\SubActivity2.

c. Màn hình SubActivity2: **activity_sub1.xml**



- d. Strings.xml: Thêm các thẻ lưu giá trị hiển thị cho tiêu đề các màn hình:

Res\value\strings.xml

```
<resources>
    <string name="app_name" >Màn hình chính phụ</string>
    <string name="app_name1">Màn hình phụ 1</string>
    <string name="app_name2">Màn hình phụ 2</string>
    <string name="cls_app">Đóng lại </string>
</resources>
```

- e. Manifest.xml. Kiểm tra file cấu hình đã khai báo ứng dụng đã có đủ 3 Activity.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
package="com.example.bai2_vongdoiungdung">
    <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.Bai2_VongDoiUngDung"
tools:targetApi="31">
<activity
    android:name=".SubActivity1"
    android:label="@string/app_name1"
    android:theme="@style/Theme.AppCompat.Dialog" />
<activity
    android:name=".SubActivity2"
    android:label="@string/app_name2" />
<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>
</application>
</manifest>
```

Ghi chú AndroidManifest.xml:

- Dòng lệnh android: **theme="@style/Theme.AppCompat.Dialog"** quy định hiển thị nổi dạng hội thoại.
- Dòng lệnh **<intent-filter>** quy định các bộ lọc ý định khác nhau hay nói cách khác đó là cách mà các thành phần khác của ứng dụng có thể kích hoạt hoạt động nào
- **android:label="@string/app_name1"**. Giá trị chuỗi hiển thị được mô tả trong tệp tài nguyên **res\values\strings.xml**. Trong tệp **strings.xml** như mô tả trên đã định nghĩa thêm 2 giá trị chuỗi **app_name1** và **app_name2** là các giá trị gán tiêu đề cho 2 Activity được gọi. Đó cũng là cách tham chiếu sử dụng tài nguyên trong ứng dụng.

MainActivity.java

Ghi đè sự kiện thực hiện: Code\override method. Chọn phương thức ghi đè cần tìm/ ok

```
public class MainActivity extends AppCompatActivity {
    final String TAG="MainActivity Lifecycle";
    Button btnSub1,btnSub2,btnCls;
    @Override

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        btnSub1=findViewById(R.id.btnManHinhPhu1);
        btnSub2=findViewById(R.id.btnManHinhPhu2);
        btnCls=findViewById(R.id.btnClose);
        btnCls.setOnClickListener(new View.OnClickListener()
        {
            @Override
            public void onClick(View v) {
                finish();
            }
        });

        btnSub1.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent1= new
Intent(MainActivity.this,SubActivity1.class);
                startActivity(intent1);
            }
        });
        btnSub2.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent2= new
Intent(MainActivity.this,SubActivity2.class);
                startActivity(intent2);
            }
        });
    }

    protected void onRestart() {
        super.onRestart();
        Toast.makeText(this,"đang gọi tới
MainActivity.onRestart",
        Toast.LENGTH_SHORT).show();
        Log.d(TAG, "===== onRestart =====");
    }
}
```

```
}

@Override
protected void onStart() {
    super.onStart();
    //ghi chú thông tin hiển thị và log sinh viên tự làm
}
@Override
protected void onResume() {
    super.onResume();
    //ghi chú thông tin hiển thị và log sinh viên tự làm
}
@Override
protected void onPause() {
    super.onPause();
    //ghi chú thông tin hiển thị và log sinh viên tự làm
}
@Override
protected void onStop() {
    super.onStop();
    //ghi chú thông tin hiển thị và log sinh viên tự làm
}
@Override
protected void onDestroy() {
    super.onDestroy();
    //ghi chú thông tin hiển thị và log sinh viên tự làm
}
}
```

Ghi chú:

- `findViewById` được hiểu như sau: tất cả các control trên giao diện cần được ánh xạ sang đối tượng khai báo trong phần nghiệp vụ thông qua lệnh `findViewById(R.id....)`. Ví dụ `btnSub1=findViewById(R.id.btnManHinhPhu1)`;
- `btnCls.setOnClickListener(...)` Hiểu là: Gọi hành động của nút lệnh thực thi nhiệm vụ khi nhận chuột. Sinh viên làm theo mẫu minh họa
- `Toast.makeText(this, String, Toast.LENGTH_SHORT).show();`.
Hiển thị xâu String ra màn hình hiện thời (this) trong khoảng thời gian ngắn 2.5s
- **`Intent intent1= new Intent(MainActivity.this, SubActivity1.class);`** và **`startActivity(intent1);`** được hiểu là khai báo 1 đối tượng Intent có nhiệm vụ gọi hiển thị 1 Activity khác
 - o This là ngữ cảnh hiện tại
 - o `SubActivity1.class` là tên Activity nghiệp vụ đã biên dịch.

Bài tập 2.6. Lưu thông tin trạng thái ứng dụng

Tìm hiểu mở rộng lưu thông tin trạng thái ứng dụng qua sự kiện **onSaveInstanceState()**.

Và **onRestoreInstanceState()** theo các nội dung sau:

- Tác dụng của thao tác lưu thông tin trạng thái ứng dụng
- Sự kiện lưu thông tin trạng thái ứng dụng xảy ra trước sự hay có kết nối với sự kiện nào trong vòng đời ứng dụng Android

Hướng dẫn thực hiện:

- Tham khảo giáo trình chương 2, mục 2.4 và bài tập minh hoạ cuối chương.