

BÀI 4: MENU - DIALOG

I. Tóm tắt nội dung thực hành

1. Yêu cầu lý thuyết

Trong lập trình Java Android, menu và dialog là hai khái niệm quan trọng để tạo ra giao diện người dùng (UI) thân thiện và tương tác.

- **Menu:** là một lớp trừu tượng đại diện cho một menu trong ứng dụng Android. Nó định nghĩa cấu trúc của menu và các phương thức để thêm, xóa hoặc thay đổi các mục menu. Trong đó:
 - **OptionsMenu:** là một menu xuất hiện trong thanh công cụ của ứng dụng. Nó thường được sử dụng để hiển thị các tùy chọn chung cho hoạt động (Activity) hiện tại.
 - **ContextMenu:** là một menu ngữ cảnh, xuất hiện khi người dùng nhấn giữ lâu trên một phần tử trong giao diện người dùng. Nó cung cấp các tùy chọn liên quan đến phần tử đó.
 - **PopupMenu:** là một menu bật lên từ một phần tử giao diện người dùng cụ thể, thường được sử dụng để hiển thị các tùy chọn liên quan đến phần tử đó.
- **Dialog:** là một lớp trừu tượng đại diện cho một hộp thoại trong ứng dụng Android. Nó được sử dụng để hiển thị thông tin hoặc yêu cầu người dùng nhập dữ liệu.
 - **AlertDialog:** là một loại hộp thoại thường được sử dụng để hiển thị thông báo hoặc yêu cầu người dùng lựa chọn giữa một số tùy chọn.
 - **DatePickerDialog:** là một hộp thoại cho phép người dùng chọn ngày từ một lịch.
 - **TimePickerDialog:** là một hộp thoại cho phép người dùng chọn thời gian.
 - **ProgressDialog:** Đây là một hộp thoại hiển thị tiến trình của một tác vụ đang được thực hiện.

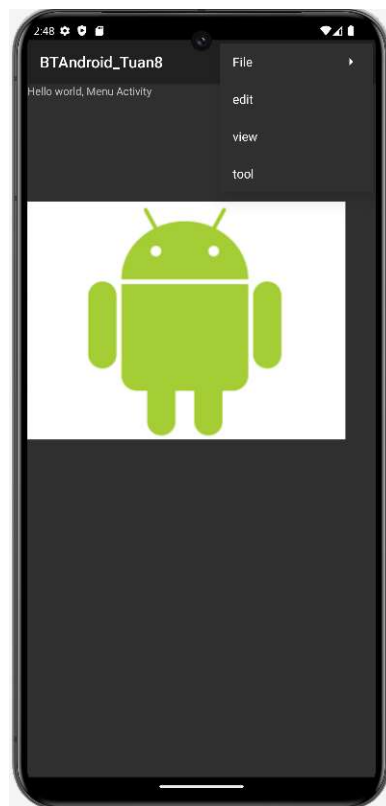
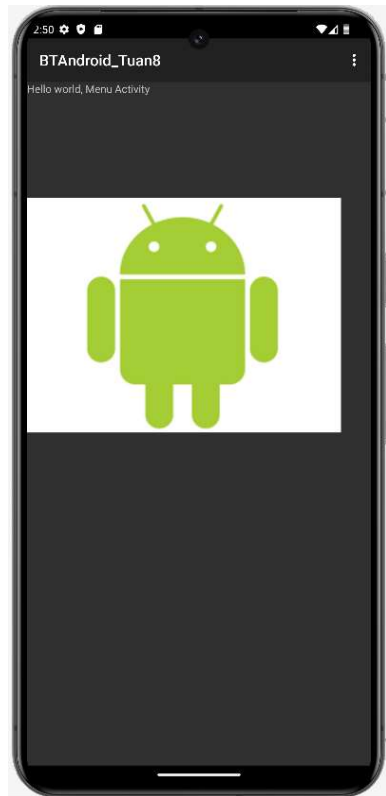
Các khái niệm trên đều có lớp và phương thức tương ứng để tạo, hiển thị và xử lý sự kiện của chúng trong ứng dụng Android. Việc sử dụng menu và dialog giúp cải thiện trải nghiệm người dùng và tăng tính tương tác của ứng dụng.

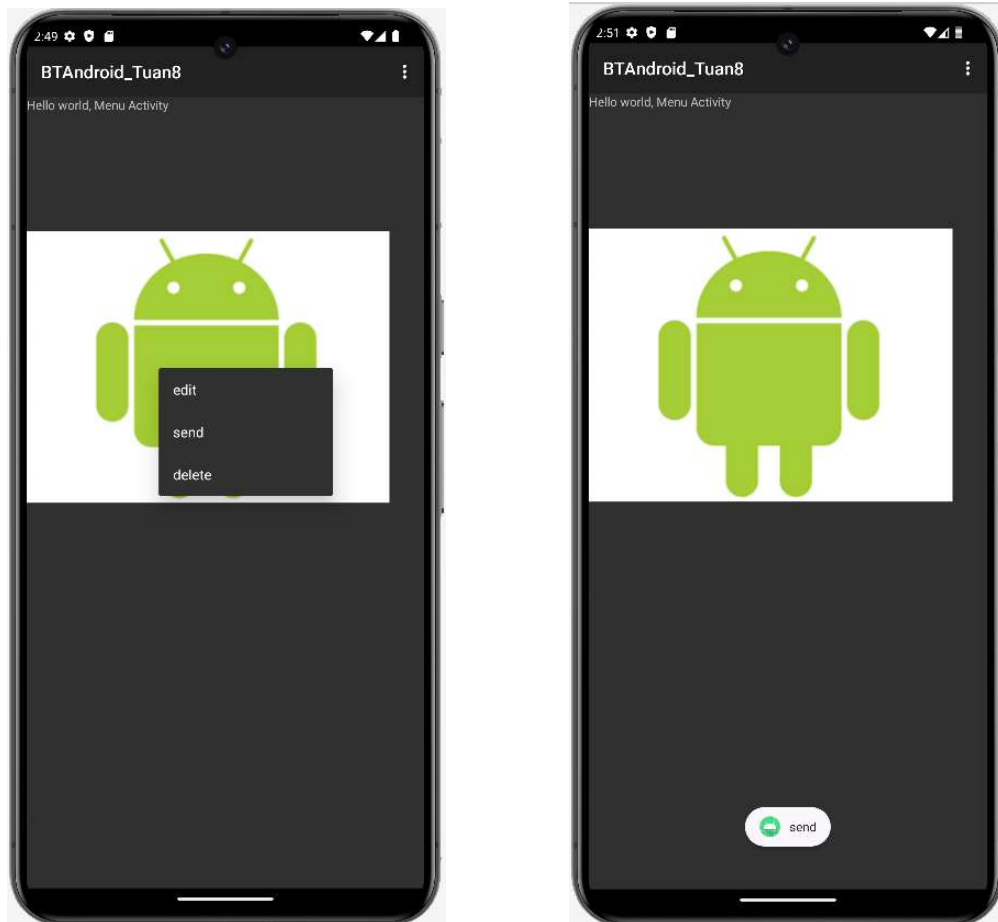
2. Nội dung

Thực hiện ứng dụng Android với các xử lý liên quan đến menu và dialog theo từng bước của hướng dẫn.

II. Bài tập yêu cầu

Bài 1: Xây dựng ứng dụng minh họa các loại menu như sau:

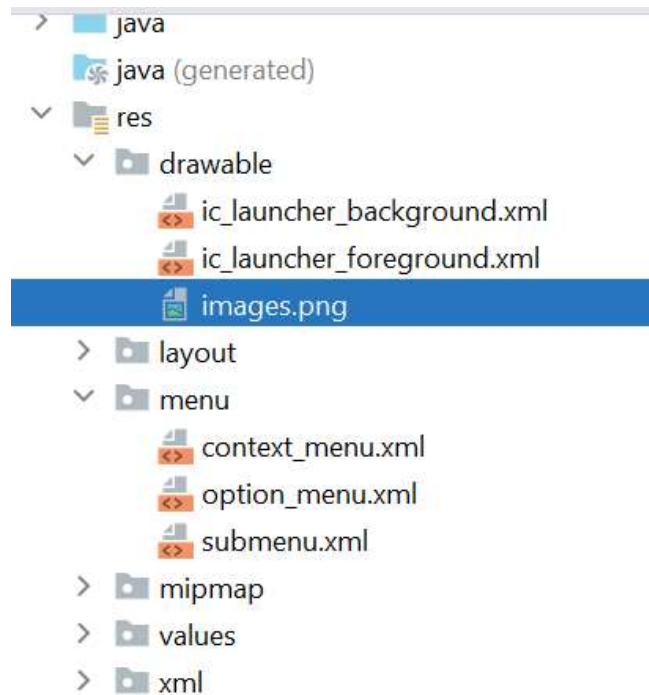




Hình 37 Giao diện bài tập 4.1

Hướng dẫn:

-Phân cấp file xml như hình



-File activity_main.xml tạo giao diện chính hiển thị 1 image view

```
<?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">
    <LinearLayout
        android:layout_width="409dp"
        android:layout_height="729dp"
        android:orientation="vertical"
        tools:layout_editor_absoluteX="1dp"
        tools:layout_editor_absoluteY="1dp">
        <TextView...>
        <ImageView
            android:id="@+id/imageView"
            android:layout_width="match_parent"
            android:layout_height="573dp"
            app:srcCompat="@drawable/images" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

-option_menu.xml

```

<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android" >
    <item android:id="@+id/sm_file" android:title="File">
        <menu>
            <item android:id="@+id/sm_new" android:title="new"></item>
            <item android:id="@+id/sm_save" android:title="save"></item>
            <item android:id="@+id/sm_load" android:title="load"></item>
            <item android:id="@+id/sm_exit" android:title="exit"></item>
        </menu>
    </item>
    <item android:id="@+id/sm_edit" android:title="edit"></item>
    <item android:id="@+id/sm_view" android:title="view"></item>
    <item android:id="@+id/sm_tool" android:title="tool"></item>
</menu>

```

-Code xử lý cho OptionsMenu

```

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.option_menu, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    Toast.makeText(context: this, item.getTitle(), Toast.LENGTH_SHORT).show();
    return true;
}

```

-Submenu.xml (submenu của menu item File)

```

<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/sm_file" android:title="File">
        <menu>
            <item android:id="@+id/sm_new" android:title="new"></item>
            <item android:id="@+id/sm_save" android:title="save"></item>
            <item android:id="@+id/sm_load" android:title="load"></item>
            <item android:id="@+id/sm_exit" android:title="exit"></item>
        </menu>
    </item>
    <item android:id="@+id/sm_edit" android:title="edit"></item>
    <item android:id="@+id/sm_view" android:title="view"></item>
    <item android:id="@+id/sm_tool" android:title="tool"></item>
</menu>

```

-context_menu.xml

```

<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android" >
    <item android:id="@+id/ct_edit" android:title="edit"></item>
    <item android:id="@+id/ct_send" android:title="send"></item>
    <item android:id="@+id/ct_delete" android:title="delete"></item>
</menu>

```

-Code xử lý context menu

```

@Override
public void onCreateContextMenu(ContextMenu menu, View v, ContextMenu.ContextMenuInfo menuInfo) {
    getMenuInflater().inflate(R.menu.context_menu, menu);
}

@Override
public boolean onContextItemSelected(MenuItem item) {
    Toast.makeText(context, this, item.getTitle(), Toast.LENGTH_SHORT).show();
    return true;
}

```

-Trong onCreate, đăng ký với hệ thống

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    ImageView im = (ImageView) findViewById(R.id.imageView);
    registerForContextMenu(im);
}

```

Chạy chương trình và xem kết quả

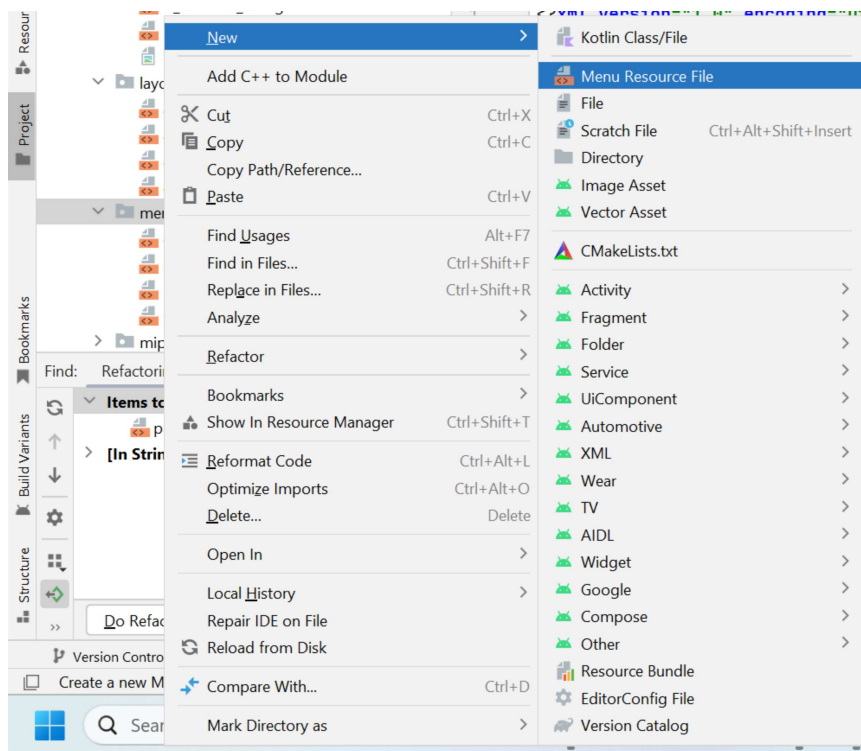
Thêm 1 button vào giao diện màn hình

```

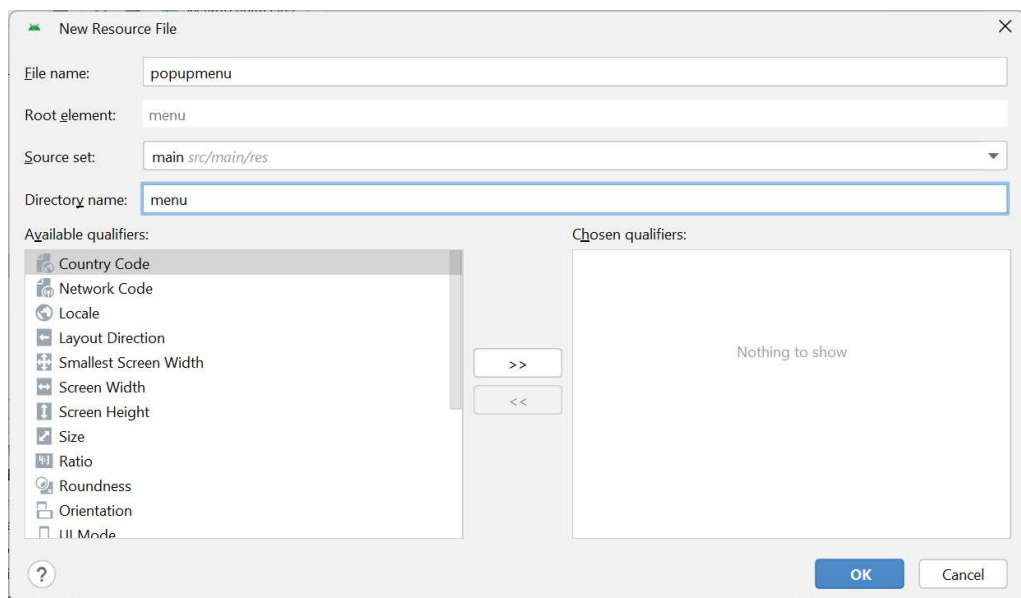
<Button
    android:id="@+id/btn"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Check PopupMenu" />

```

Thêm 1 file xml trong res/menu bằng cách thêm một resource như sau, click chuột phải vào thư mục menu trong cửa sổ project, chọn New → Menu Resource File



Hình 38 Tạo file menu



Hình 39 Giao diện Resource File

```

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

<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/pmOne"
        android:title="One"
        android:orderInCategory="100" />
    <item
        android:id="@+id/pmTwo"
        android:title="Two"
        android:orderInCategory="100" />

    <item
        android:id="@+id/pmThree"
        android:title="Three"
        android:orderInCategory="100" />
</menu>

```

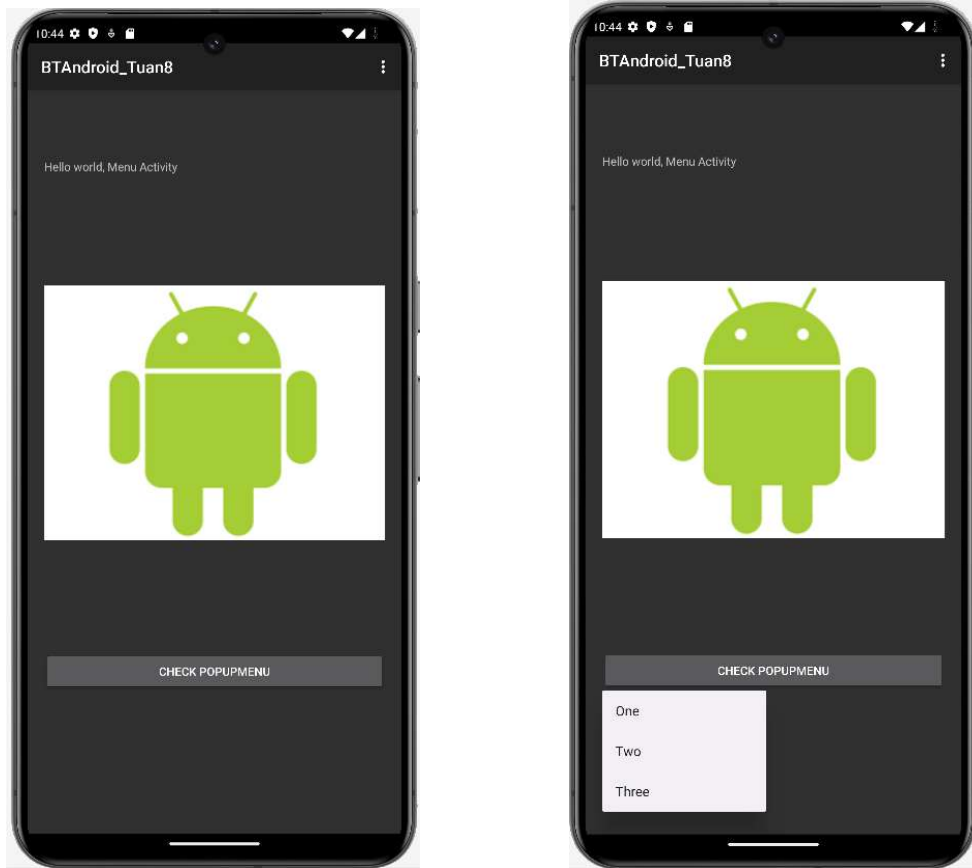
Thêm xử lý cho Button và gắn Popupmenu cho button

⚠️ 2 ✅ 1

```

View btnPopupMenu = (Button) findViewById(R.id.btn);
//bắt sự kiện cho button
btnPopupMenu.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        //Khởi tạo 1 popupmenu
        PopupMenu popupMenu;
        popupMenu= new PopupMenu(getApplicationContext(),btnPopupMenu);
        //đẩy layout tạo trong xml vào ứng dụng
        popupMenu.getMenuInflater().inflate(R.menu.popupmenu, popupMenu.getMenu());
        //Sự kiện click vào item của menu
        popupMenu.setOnMenuItemClickListener(new PopupMenu.OnMenuItemClickListener() {
            @Override
            public boolean onMenuItemClick(MenuItem item) {
                Toast.makeText( context: MainActivity.this, text: "Bạn vừa chọn popup menu",
                    Toast.LENGTH_SHORT).show();
                return false;
            }
        });
        popupMenu.show();
    }
}

```



Hình 40 Giao diện Popup Menu



Hình 41 Giao diện bài tập 4.1

Bài 2: Xây dựng ứng dụng xem thời gian hiện hành



Hình 42 Giao diện bài tập 4.2

Hướng dẫn:

-Tạo một Button trong activity_main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/an
    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">

    <Button
        android:id="@+id/button"
        android:layout_width="379dp"
        android:layout_height="71dp"
        android:layout_marginStart="18dp"
        android:layout_marginEnd="18dp"
        android:text="Xem thời gian"
        android:textSize="24sp"
        app:cornerRadius="0px"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        tools:layout_editor_absoluteY="16dp" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

-Trong file java, lấy id của Button

-Tạo một AlertDialog

```

final Button btn = (Button) findViewById(R.id.button);
final AlertDialog ad = new AlertDialog.Builder(context: this).create();

```

-Viết xử lý cho button

```

btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Date t = new Date();
        String message = "Thời gian hiện hành " + t.toString();
        ad.setMessage(message);
        ad.show();
    }
});

```

Lưu và chạy chương trình.

III. Câu hỏi ôn tập:

1. Phương thức nào được sử dụng để tạo một Menu trong Android?
 - A. `createMenu()`
 - B. `onOptionsItemSelected()`
 - C. `setMenu()`
2. Để xử lý sự kiện khi người dùng chọn một menu item, chúng ta cần override phương thức nào?
 - A. `onMenuItemSelected()`
 - B. `onOptionsItemSelected()`
 - C. `onItemClick()`
3. Phương thức nào được sử dụng để tạo một PopupMenu?
 - A. `createPopupMenu()`
 - B. `showPopupMenu()`
 - C. Cả A và B đều đúng
4. Phương thức nào được sử dụng để hiển thị một PopupMenu?
 - A. `show()`
 - B. `setVisibility(View.VISIBLE)`
 - C. Cả A và B đều đúng
5. Phương thức nào được sử dụng để thiết lập tiêu đề (title) cho AlertDialog?
 - A. `setTitle()`
 - B. `setHeader()`
 - C. `setDialogTitle()`

6. Để thêm các item vào Menu trong Android, cần sử dụng phương thức nào?

- A. addItem()
- B. addItemToMenu()
- C. getMenuInflater().inflate()

7. Để hiển thị một Dialog trong Android, phương thức nào được sử dụng?

- A. showAlertDialog()
- B. showDialog()
- C. displayDialog()

8. Làm thế nào để thiết lập nội dung cho một AlertDialog trong Android?

- A. setDialogContent()
- B. setMessage()
- C. setContent()