

Lập trình trên thiết bị di động

THÀNH PHẦN INTENT

GV: Nguyễn Huy Cường

Email: nh.cuong@hutech.edu.vn

Nội dung

1. Intent và Các loại Intent
2. Truyền dữ liệu giữa activity
3. Truyền dữ liệu giữa activity và fragment
4. Truyền dữ liệu các fragment
5. Intent Filter

1. INTENT

- **Intent** (Ý định): Kích hoạt thông qua thông điệp giữa các thành phần chính của ứng dụng (activity, service, broadcast receiver)

INTENT có thể

- Khởi chạy một activity, service, broadcast...

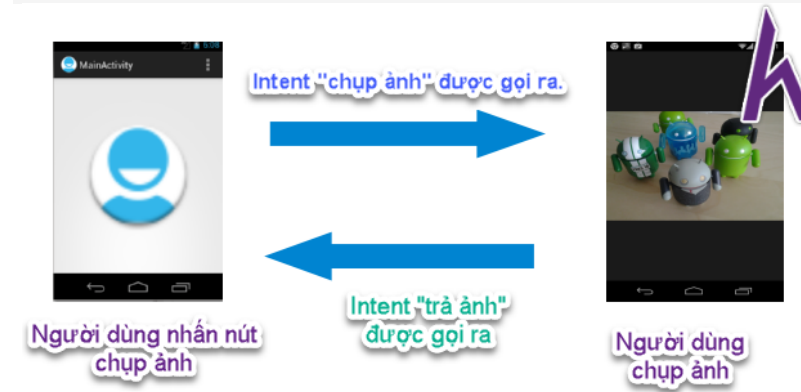
```
Intent intent = new Intent(MainActivity.this, ReceiverActivity.class);
startActivity(intent);
```

```
Intent firstIntent = new Intent(this, FirstService.class);
startService(firstIntent);
```

- Truyền/ nhận dữ liệu giữa các activity, ứng dụng

```
intent.putExtra("KEY_MSSV", mssv);
intent.putExtra("KEY_HOTEN", hoten);
intent.putExtra("KEY_DTB", dtb);
```

```
Intent intent = getIntent();
String mssv= intent.getExtras().getString("KEY_MSSV");
String hoten= intent.getExtras().getString("KEY_HOTEN");
float dtb = intent.getExtras().getFloat("KEY_DTB");
```



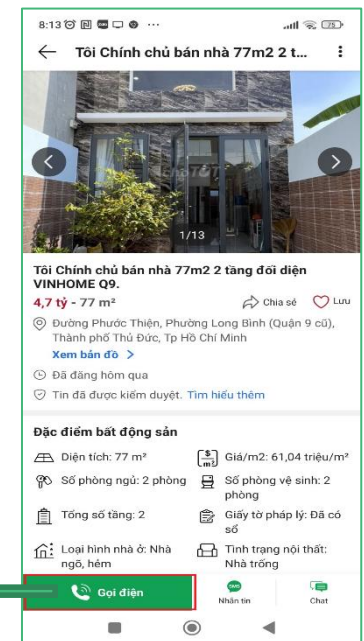
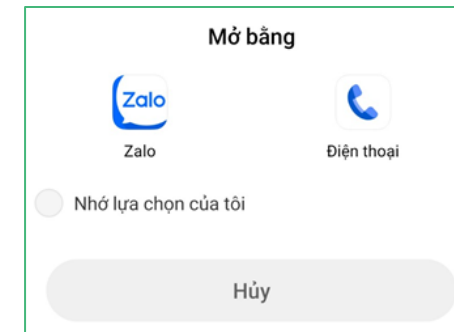
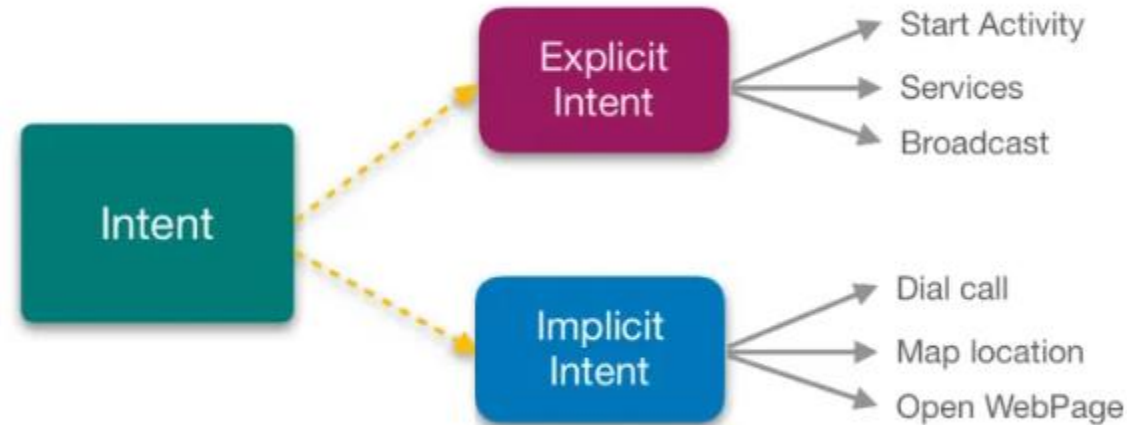
Các loại Intent

- Có 2 loại:

Explicit Intent (Trường minh, công khai): Là những intent chỉ định rõ ràng tên của các thành phần mục tiêu để xử lý

Implicit intents (Không tường minh, ẩn): Là loại intent không cần chỉ rõ thành phần xử lý mà chỉ cung cấp các thông tin về hành động (Action), Dữ liệu (Data), hay phân loại (category)

Ví dụ: Mở 1 trang web, mở bản đồ, gọi số điện thoại, mở camera, gallery...



Truyền dữ liệu có y/c phản hồi: **ActivityResultLauncher**

DEMO

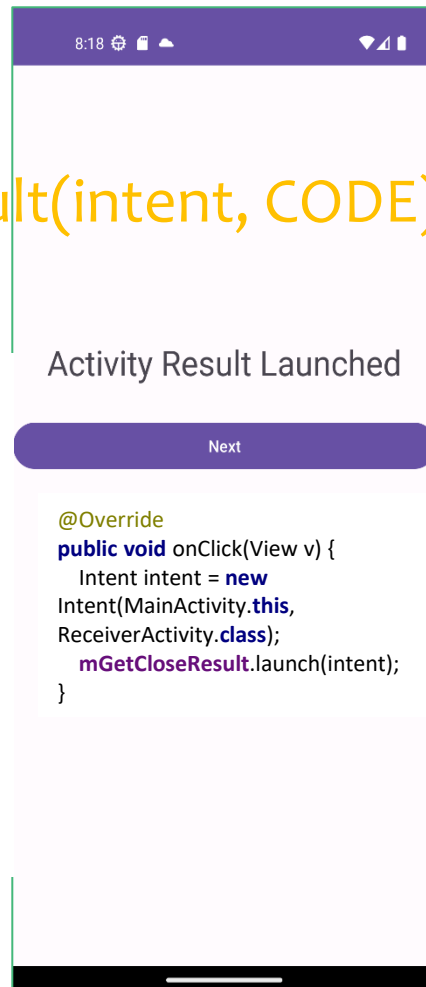
1. Activity Gửi (MainActivity) có Button gọi ReceiverActivity (có 2 Button để trở về)
2. Đợi kết quả trả về từ ReceiverActivity? Cho biết đã nhận Button nào?

❖ Activity Gửi chờ y/c phản hồi
(MainActivity)

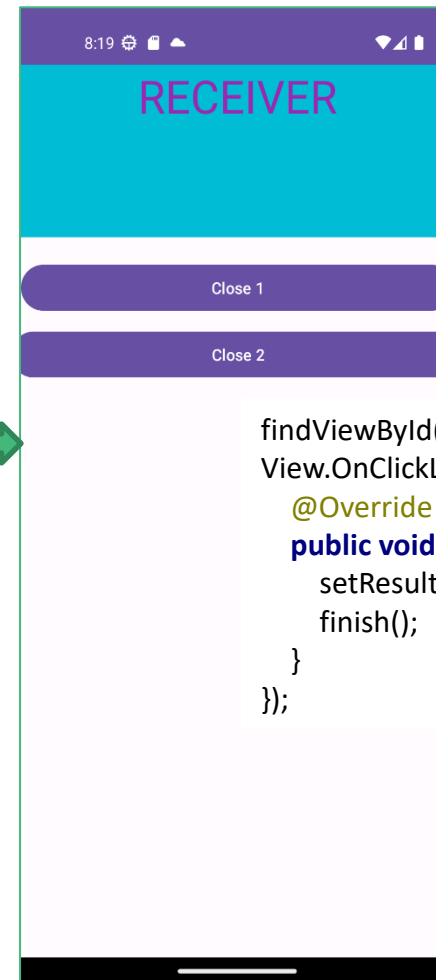
!!![deprecated]
`startActivityForResult(intent, CODE);`

- Sử dụng **ActivityResultLauncher**.

```
ActivityResultLauncher<Intent> mGetCloseResult =
registerForActivityResult(
    new ActivityResultContracts.StartActivityForResult(),
    new ActivityResultCallback<ActivityResult>() {
        @Override
        public void onActivityResult(ActivityResult result) {
            int resultCode = result.getResultCode();
            Toast.makeText(MainActivity.this,
                "Đã nhấn vào nút CLOSE " + resultCode,
                Toast.LENGTH_SHORT).show();
        }
    })
});
```



❖ **ReceiverActivity**: `setResult(statuscode).`



Hướng dẫn Demo

❖ Activity Gửi chờ y/c phản hồi (MainActivity)

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    findViewById(R.id.btnSend).setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent intent = new Intent( packageContext: MainActivity.this, ReceiverActivity.class);
            mGetCloseResult.launch(intent);
        }
    });
}
```

1 usage

```
ActivityResultLauncher<Intent> mGetCloseResult = registerForActivityResult(
    new ActivityResultContracts.StartActivityForResult(),
    new ActivityResultCallback<ActivityResult>() {
        @Override
        public void onActivityResult(ActivityResult result) {
            int resultCode = result.getResultCode();
            Toast.makeText( context: MainActivity.this,
                text: "Đã nhấn vào nút CLOSE " + resultCode,
                Toast.LENGTH_SHORT).show();
        }
    });
```

❖ Activity mục tiêu (ReceiverActivity)

```
findViewById(R.id.btnClose1).setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        setResult(1);
        finish();
    }
});
```

```
findViewById(R.id.btnClose2).setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        setResult(2);
        finish();
    }
});
```



Lấy ảnh lên ImageView từ camera/ Bộ sưu tập

Có **ImageView** và **ImageButton**

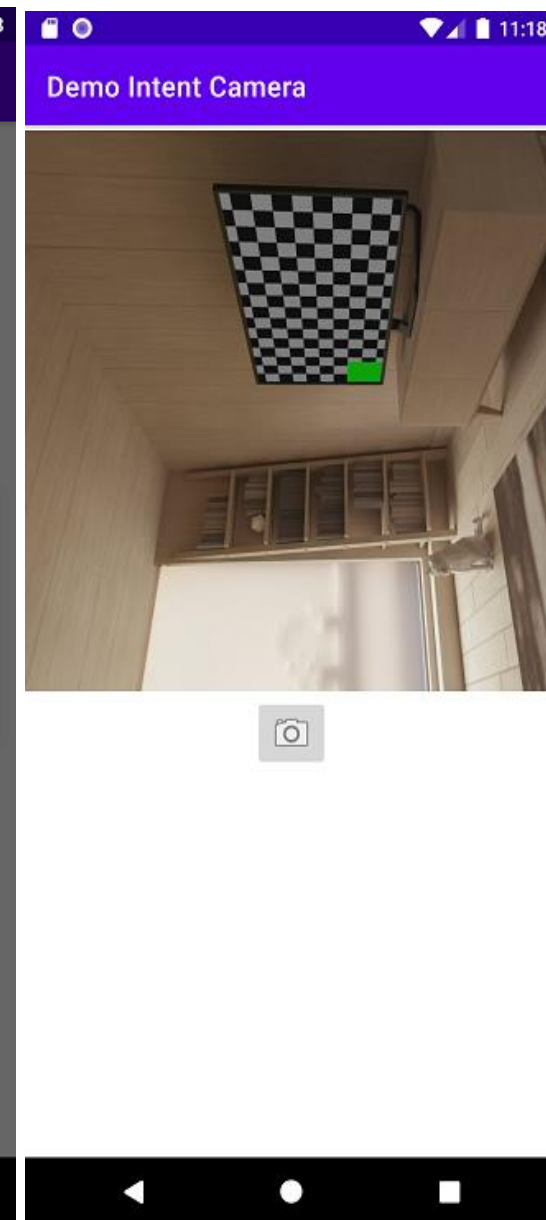
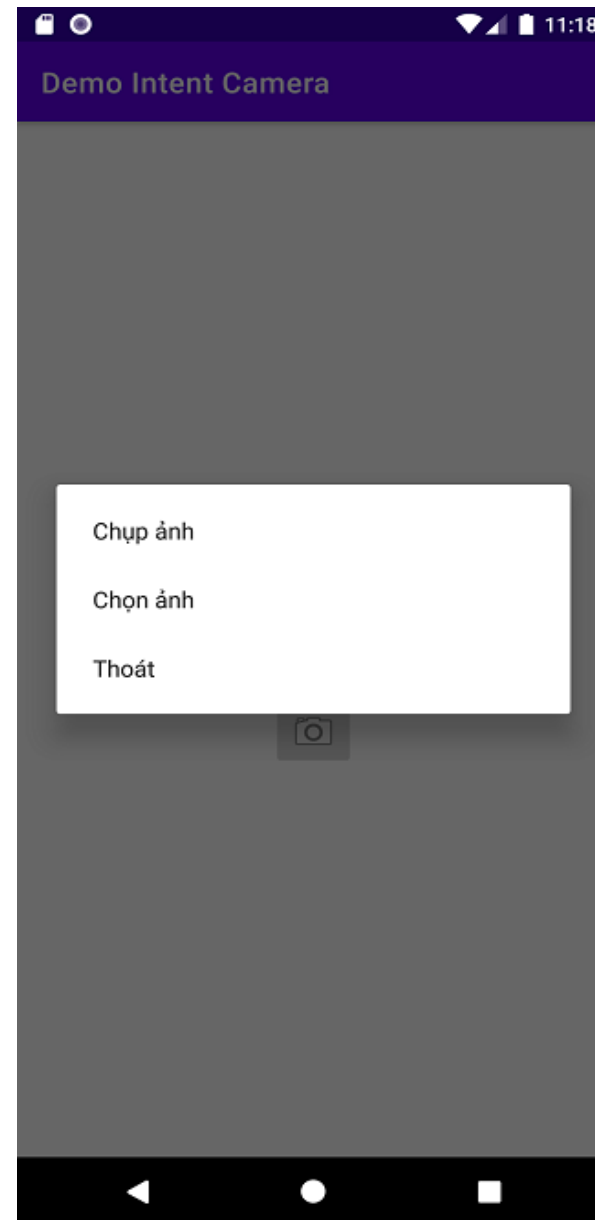
+ Click vào Button sẽ mở máy ảnh/ lấy dữ liệu từ Gallery đưa vào **imageView**

```
<LinearLayout 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"
    android:orientation="vertical"
    tools:context=".MainActivity">
```

```
<ImageView
    android:layout_width="match_parent"
    android:layout_height="400dp"
    android:id="@+id/imageView"
/>
```

```
<ImageButton
    android:id="@+id/btnTakePhoto"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:src="@android:drawable/ic_menu_camera"
    android:text="Ảnh" />
```

```
</LinearLayout >
```





Lấy ảnh lên ImageView từ camera/ Bộ sưu tập

```
public class MainActivity extends AppCompatActivity {
    public int CHON_ANH = 0;
    private ImageView imageView;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        imageView = findViewById(R.id.imageView);
        findViewById(R.id.btnTakePhoto).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                CharSequence[] optionsMenu = {"Chụp ảnh", "Chọn ảnh", "Thoát"};
                AlertDialog.Builder builder = new AlertDialog.Builder(context: MainActivity.this);
                builder.setItems(optionsMenu, new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {
                        if (optionsMenu[i].equals("Chụp ảnh")) { // Mở camera để chụp ảnh
                            Intent takePicture = new Intent(android.provider.MediaStore.ACTION_IMAGE_CAPTURE);
                            setResult(RESULT_OK, takePicture);
                            CHON_ANH = 1;
                            getData.launch(takePicture);
                        } else if (optionsMenu[i].equals("Chọn ảnh")) { // Chọn ảnh từ bộ sưu tập
                            Intent pickPhoto = new Intent();
                            pickPhoto.setType("image/*");
                            pickPhoto.setAction(Intent.ACTION_GET_CONTENT);
                            setResult(RESULT_OK, pickPhoto);
                            CHON_ANH = 2;
                            getData.launch(pickPhoto);
                        } else if (optionsMenu[i].equals("Thoát")) {
                            dialogInterface.dismiss();
                        }
                    }
                });
            }
        });
        builder.show();
    }
}
```

```
ActivityResultLauncher<Intent> getData = registerForActivityResult(
    new ActivityResultContracts.StartActivityForResult(),
    result -> {
        if (result.getResultCode() == Activity.RESULT_OK) {
            Intent data = result.getData();
            Bitmap selectedImage = null;
            if(CHON_ANH == 1) {
                selectedImage = (Bitmap) data.getExtras().get("data");
            }
            else if(CHON_ANH == 2)
            {
                Uri selectedImageUri = data.getData();
                try {
                    selectedImage = MediaStore.Images.Media.getBitmap(
                        this.getContentResolver(),
                        selectedImageUri);
                } catch (IOException e) {
                    e.printStackTrace();
                }
            }
            imageView.setImageBitmap(selectedImage);
        }
    });
```


2. Truyền dữ liệu giữa các activity: 4 cách

❖ Sender

```
Intent intent = new Intent(MainActivity.this,
                        ReceiverActivity.class);
// Gửi dữ liệu với putExtra/ putExtras
startActivity(intent);
```

C1: Gửi dữ liệu đơn putExtra(KEY,VALUE)

```
intent.putExtra("KEY_MSSV", mssv);
intent.putExtra("KEY_HOTEN", hoten);
intent.putExtra("KEY_DTB", dtb);
```

C2: gửi qua bundle với putExtra/ putExtras

```
Bundle bundle = new Bundle();
bundle.putString("KEY_MSSV", mssv);
bundle.putString("KEY_HOTEN", hoten);
bundle.putFloat("KEY_DTB", dtb);
//C21 intent.putExtra("KEY_BUNDLE", bundle);
intent.putExtras(bundle);
```

❖ Receiver

```
Intent intent = getIntent();
// Nhận dữ liệu với getExtras
```

C1: Nhận dữ liệu đơn với getExtras()

```
String mssv= intent.getExtras().getString("KEY_MSSV");
String hoten= intent.getExtras().getString("KEY_HOTEN");
float dtb = intent.getExtras().getFloat("KEY_DTB");
```

C2: Nhận giá trị qua bundle với getExtras()

```
//C21 Bundle bundle =
intent.getExtras().getBundle("KEY_BUNDLE");
Bundle bundle = intent.getExtras();
String mssv= bundle.getString("KEY_MSSV");
String hoten= bundle.getString("KEY_HOTEN");
float dtb = bundle.getFloat("KEY_DTB");
```

Truyền dữ liệu giữa các activity

C3: gửi qua bundle, object được putSerializable

// SinhVien được implements Serializable

```
Bundle bundle = new Bundle();
SinhVien temp = new SinhVien(mssv, hoten, dtb);
bundle.putSerializable("KEY_SER_SV", temp);
// intent.putExtra("KEY_SV", bundle);
intent.putExtras(bundle);
```

C4: gửi qua bundle, object được putParcelable

// SinhVien được implements Parcelable

```
Bundle bundle = new Bundle();
SinhVienPar temp = new SinhVienPar(mssv, hoten, dtb);
bundle.putParcelable("KEY_PAR_SV", temp);
// intent.putExtra("KEY_SV", bundle);
intent.putExtras(bundle);
```

C3: nhận bundle, object được getSerializable

```
// Bundle bundle=intent.getExtras().getBundle("KEY_SV");
Bundle bundle = intent.getExtras();
SinhVien obj = (SinhVien)bundle.getSerializable("KEY_SER_SV");
```

C4: nhận bundle, object được getParcelable

```
//Bundle bundle = intent.getExtras().getBundle("KEY_SV");
Bundle bundle = intent.getExtras();
SinhVienPar obj = (SinhVienPar)
bundle.getParcelable("KEY_PAR_SV");
```



Truyền dữ liệu giữa các Activity

```
Intent intent = new Intent(MainActivity.this, ReceiverActivity.class);
```

```
case R.id.btnPutExtra:
```

```
{
    intent.putExtra("KEY_MSSV", mssv);
    intent.putExtra("KEY_HOTEN", hoten);
    intent.putExtra("KEY_DTB", dtb);
    ReceiverActivity.iCase = 1;
    break;
}
```

```
case R.id.btnPutBundle:
```

```
{
    Bundle bundle = new Bundle();
    bundle.putString("KEY_MSSV", mssv);
    bundle.putString("KEY_HOTEN", hoten);
    bundle.putFloat("KEY_DTB", dtb);
    intent.putExtras(bundle);
    ReceiverActivity.iCase = 2;
    break;
}
```

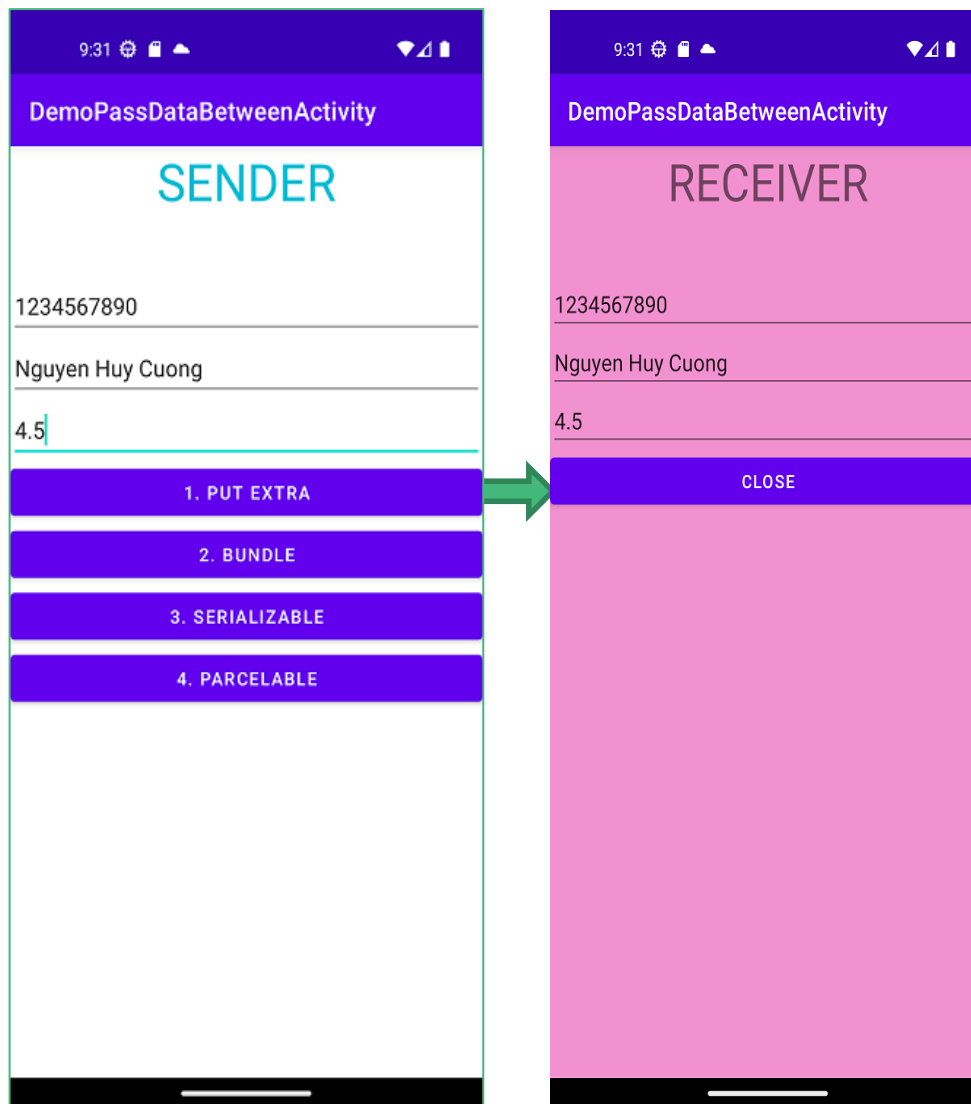
```
case R.id.btnSerializable:
```

```
{
    Bundle bundle = new Bundle();
    SinhVien temp = new SinhVien(mssv, hoten, dtb);
    bundle.putSerializable("KEY_SER_SV", temp);
    intent.putExtras(bundle);
    ReceiverActivity.iCase = 3;
    break;
}
```

```
case R.id.btnParcelable:
```

```
{
    Bundle bundle = new Bundle();
    SinhVienPar temp = new SinhVienPar(mssv, hoten, dtb);
    bundle.putParcelable("KEY_PAR_SV", temp);
    intent.putExtras(bundle);
    ReceiverActivity.iCase = 4;
    break;
}
```

```
startActivity(intent);
```



```
@Override
```

```
protected void onCreate(Bundle savedInstanceState) {
    Intent intent = getIntent();
}
```

```
private void GetStudentExtra(Intent intent) {
    String mssv= intent.getExtras().getString("KEY_MSSV");
    String hoten= intent.getExtras().getString("KEY_HOTEN");
    float dtb = intent.getExtras().getFloat("KEY_DTB");
    SetSV(mssv, hoten, dtb);
}
```

```
private void GetStudentBundle(Intent intent) {
    Bundle bundle = intent.getExtras();
    String mssv= bundle.getString("KEY_MSSV");
    String hoten= bundle.getString("KEY_HOTEN");
    float dtb = bundle.getFloat("KEY_DTB");
    SetSV(mssv, hoten, dtb);
}
```

```
private void GetStudentSerializable(Intent intent) {
    Bundle bundle = intent.getExtras();
    SinhVien obj = (SinhVien)
    bundle.getSerializable("KEY_SER_SV");
    SetSV(obj.MSSV, obj.HoTen, obj.DTB);
}
```

```
private void GetStudentParcelizable(Intent intent) {
    Bundle bundle = intent.getExtras();
    SinhVienPar obj = (SinhVienPar)
    bundle.getParcelable("KEY_PAR_SV");
    SetSV(obj.MSSV, obj.HoTen, obj.DTB);
}
```

Truyền dữ liệu giữa các Activity có y/c phản hồi



1. Activity Gửi (MainActivity)
2. Sau khi nhấn Save AND Close hãy lưu lại dữ liệu sinh viên cho MainActivity ?

❖ Activity Gửi chờ y/c phản hồi (MainActivity)

DemoPassDataBetweenActivity

SENDER

212203045

Nguyen Huy Cuong

8.5

1. PUT EXTRA

2. BUNDLE

3. SERIALIZABLE

4. PARCELABLE

- Sử dụng **ActivityResultLauncher**.
- Nhận lại intent từ ReceiverActivity

❖ Activity mục tiêu (ReceiverActivity)

DemoPassDataBetweenActivity

RECEIVER

212203045

Nguyen Huy Cuong (update)

8.5

CLOSE

SAVE AND CLOSE

- Cần phải gọi hàm **setResult(statuscode, intent)**.
- Dữ liệu trả về khi gọi hàm **finish()**.

3. Truyền dữ liệu giữa fragment và Activity

❖ Fragment send

Gửi qua Intent

```
Intent intent = new
Intent(getActivity().getBaseContext()
, TargetActivity.class);
intent.putExtra("message", message);
getActivity().startActivity(intent);
```

❖ Activity send

Gửi qua bundle, sử dụng **setArguments**

```
Bundle bundle = new Bundle();
bundle.putString("message", "From
Activity");
//set Fragmentclass Arguments
Fragmenfragobj.setArguments(bundle);
```

❖ Activity Received

Nhận qua Intent

```
Intent intent = getIntent();
String message =
intent.getStringExtra("message");
```

❖ Fragment Received

Nhận qua bundle, sử dụng **getArguments**

VD: Trong **onCreateView/onViewCreated** của fragment

```
String strtext =
getArguments().getString("message");
return inflater.inflate(R.layout.fragment,
container, false);
```



Activity -> Fragment

(1) Fragment trên cùng Activity

- ❑ Gửi **bundle** từ Activity sang Fragment bằng **setArguments**

```
SinhVien sv = ....
//truyen qua bundle
Bundle bundle = new Bundle();
bundle.putSerializable("KEY_SER_SV", sv);
SinhVienFragment svInfo = new SinhVienFragment();
svInfo.setArguments(bundle);
FragmentManager trans =
getSupportFragmentManager().beginTransaction();
trans.replace(R.id.fragmentContainerView, svInfo);
trans.commit();
```



- ❑ Nhận từ bundle trong Fragment bằng **getArguments**

```
Bundle bundle = getArguments();
if(bundle != null)
{
    SinhVien sv = (SinhVien)
bundle.getSerializable("KEY_SER_SV");
    txtMSSV.setText(sv.MSSV);
    txtHoten.setText(sv.HoTen);
    txtDTB.setText(Float.toString(sv.DTB));
}
```



Fragment -> Activity

(1) Fragment trên cùng Activity

```
view.findViewById(R.id.btnFrgPutExtra).setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        MainActivity mainActivity = (MainActivity) getActivity();
        mainActivity.txtMSSV.setText(txtMSSV.getText().toString());
        mainActivity.txtHoten.setText(txtHoten.getText().toString());
        mainActivity.txtDTB.setText(txtDTB.getText().toString());
    }
});
```





Truyền dữ liệu giữa các fragment và Activity

(2) Fragment khác Activity

❑ Gửi từ Activity sang Fragment

Gửi bundle sang ReceiverActivity

Intent intent = ...

Bundle bundle = **new** Bundle();

SinhVien sv = **new** SinhVien (...)

bundle.putSerializable("KEY_SER_SV", sv);

intent.putExtras(bundle);

startActivity(intent);

❑ Từ Fragment sang Activity ???

Gửi qua **intent**

Intent intent = **new**

Intent(getActivity().getBaseContext(),

ReceiverActivity.class);

Bundle bundle = **new** Bundle();

SinhVien sv = **new** SinhVien(...)

bundle.putSerializable("KEY_SER_SV", sv);

intent.putExtras(bundle);

startActivity(intent);



❑ Nhận từ bundle trong Fragment bằng **getArguments**

✓ ReceiverActivity nhận đưa vào fragment qua **setArguments**

SinhVienFragment svInfo = **new** SinhVienFragment();

svInfo.setArguments(getIntent().getExtras());

FragmentManager trans =

getSupportFragmentManager().beginTransaction();

trans.replace(R.id.fragmentContainerView, svInfo);

trans.commit();

✓ Fragment nhận qua **getArguments**

❑ Từ Fragment sang Activity ???

Nhận trong **intent**

4. Truyền dữ liệu giữa fragment – trên cùng activity

Step 1: Tạo **interface** để thực hiện Action truyền dữ liệu

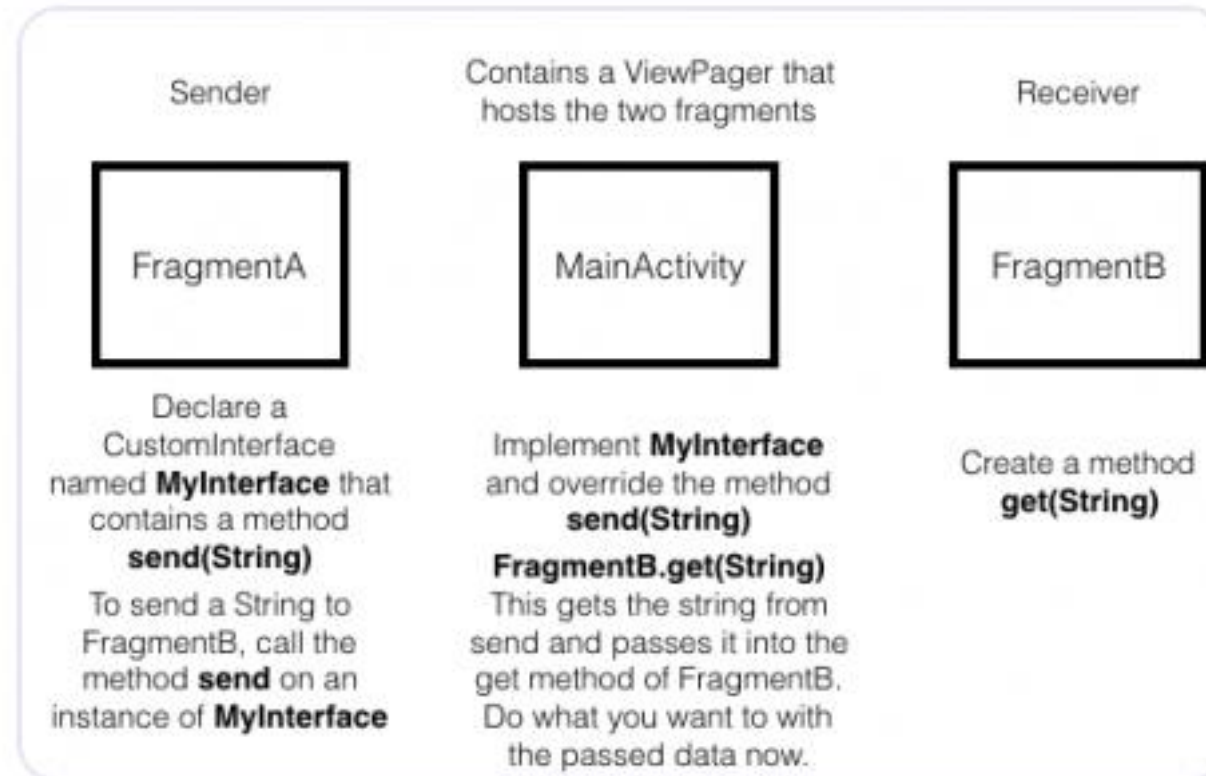
Step 2: Implements **Interface** ở Activity

Step 3:

Fragment A - Gửi dữ liệu ở interface

Fragment B – Nhận dữ liệu

❖ Fragment send

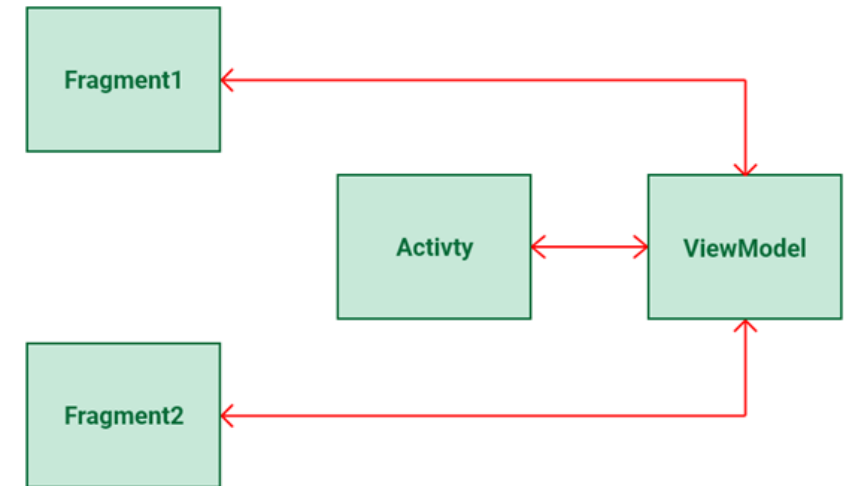


❖ Fragment Received

Truyền dữ liệu giữa fragment – Sử dụng LiveData trong ViewModel

❑ Sử dụng chung **ViewModel**: Dữ liệu được lưu trong ViewModel là LiveData

- (1) Tạo lớp kế thừa **ViewModel** để chia sẻ dữ liệu
- (2) Khởi tạo đối tượng ViewModel ở Activity
- (3) Thực hiện quan sát dữ liệu ở fragment1, fragment2, fragment3...





Sử dụng LiveData trong ViewModel

❑ Sử dụng chung **ViewModel**: Dữ liệu được lưu trong ViewModel là LiveData

(1) Tạo lớp kế thừa **ViewModel** để chia sẻ dữ liệu

- Kiểu dữ liệu **MutableLiveData** để quản lý và quan sát các dữ liệu có sự thay đổi
- thay đổi giá trị của LiveData bằng: **setValue()** hoặc **postValue()**

```
public class DataViewModel extends ViewModel {
    private MutableLiveData<String> myLiveData = new MutableLiveData<>();
    public LiveData<String> getMyLiveData() {
        return myLiveData;
    }
    public void updateData(String newData) {
        myLiveData.setValue(newData);
    }
}
```

(2) Khởi tạo đối tượng ViewModel ở Activity

```
sharedViewModel = new ViewModelProvider(this).get(DataViewModel.class);
```

(3) Thực hiện quan sát dữ liệu ở các fragments: fragment1, fragment2, fragment3...

```
DataViewModel viewModel = new ViewModelProvider(getActivity()).get(DataViewModel.class);
final TextView textView = binding.textHome;
viewModel.getMyLiveData().observe(getViewLifecycleOwner(), textView::setText);
```



Sử dụng LiveData trong ViewModel

(1) Tạo lớp **DataViewModel** kế thừa **ViewModel** để chia sẻ dữ liệu

```
public class DataViewModel extends ViewModel {
    private MutableLiveData<String> myLiveData = new
MutableLiveData<>();
    public LiveData<String> getMyLiveData() {
        return myLiveData;
    }
    public void updateData(String newData) {
        myLiveData.setValue(newData);
    }
}
```

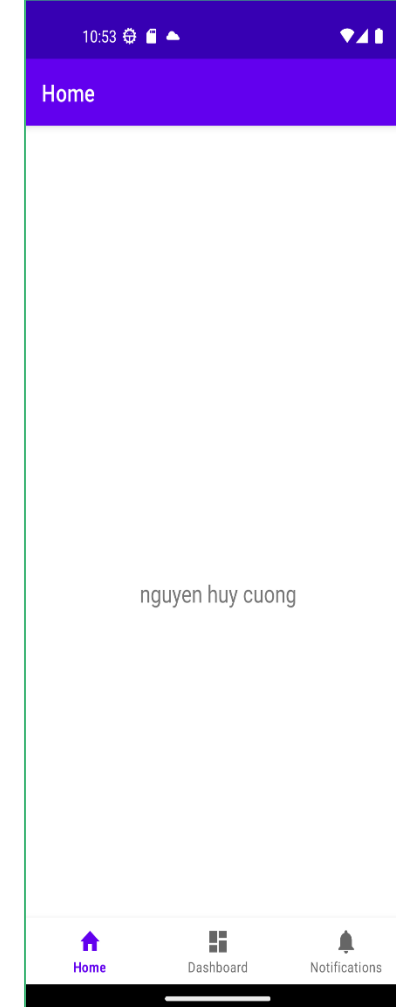
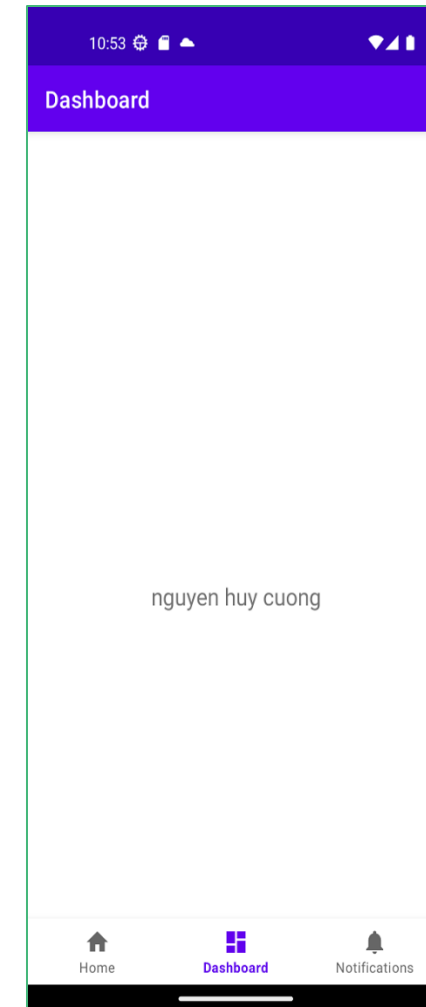
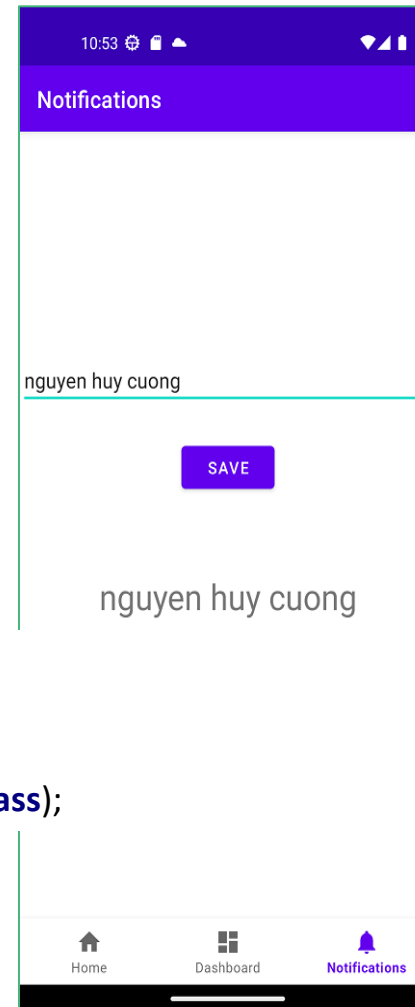
(2) Ở Activity

```
private DataViewModel sharedViewModel;
```

@Override

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    sharedViewModel = new ViewModelProvider(this).get(DataViewModel.class);
}
```

(3) Ở Fragments (notifications)





Sử dụng LiveData trong ViewModel

(3) Ở Fragments (notifications)

@Override

```
public void onCreateView(@NonNull View view, @Nullable Bundle savedInstanceState) {
    super.onCreateView(view, savedInstanceState);
```

```
    DataViewModel viewModel = new ViewModelProvider(getActivity()).get(DataViewModel.class);
    EditText editText = view.findViewById(R.id.txtText);
    final TextView textView = binding.textNotifications;
    viewModel.getMyLiveData().observe(getViewLifecycleOwner(), textView::setText);
```

```
    view.findViewById(R.id.btnSave).setOnClickListener(new View.OnClickListener() {
```

@Override

```
    public void onClick(View v) {
        viewModel.updateData(editText.getText().toString());
```

```
    }
```

```
});
```

```
}
```

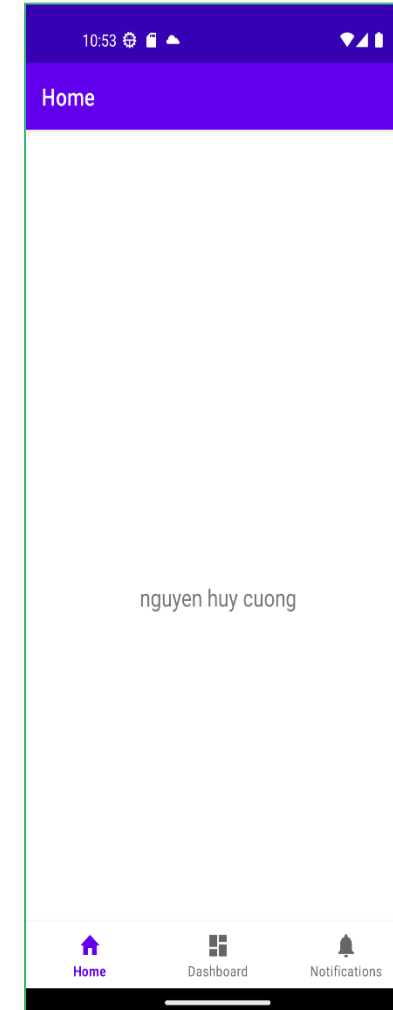
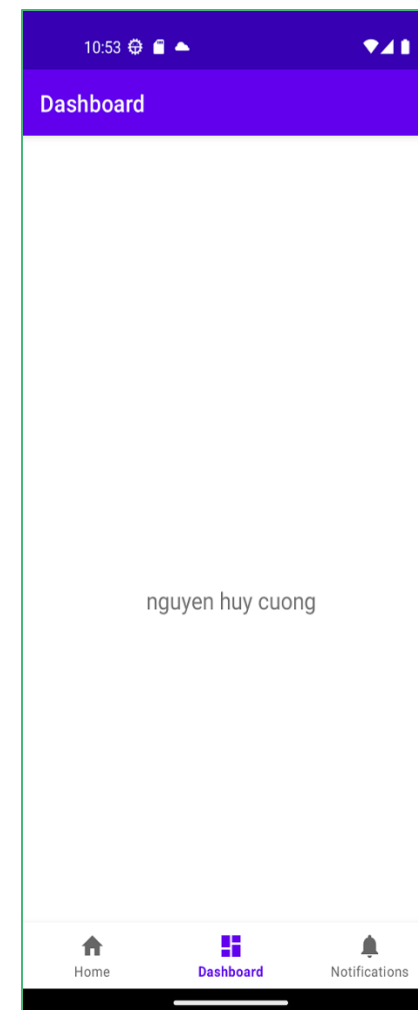
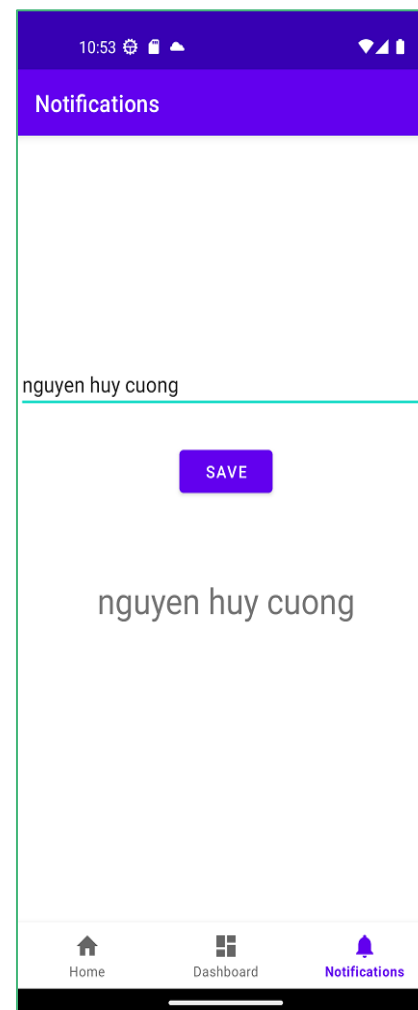
(3) Ở Dashboard

```
final TextView textView = binding.textDashboard;
```

```
DataViewModel viewModel = new
```

```
ViewModelProvider(getActivity()).get(DataViewModel.class);
```

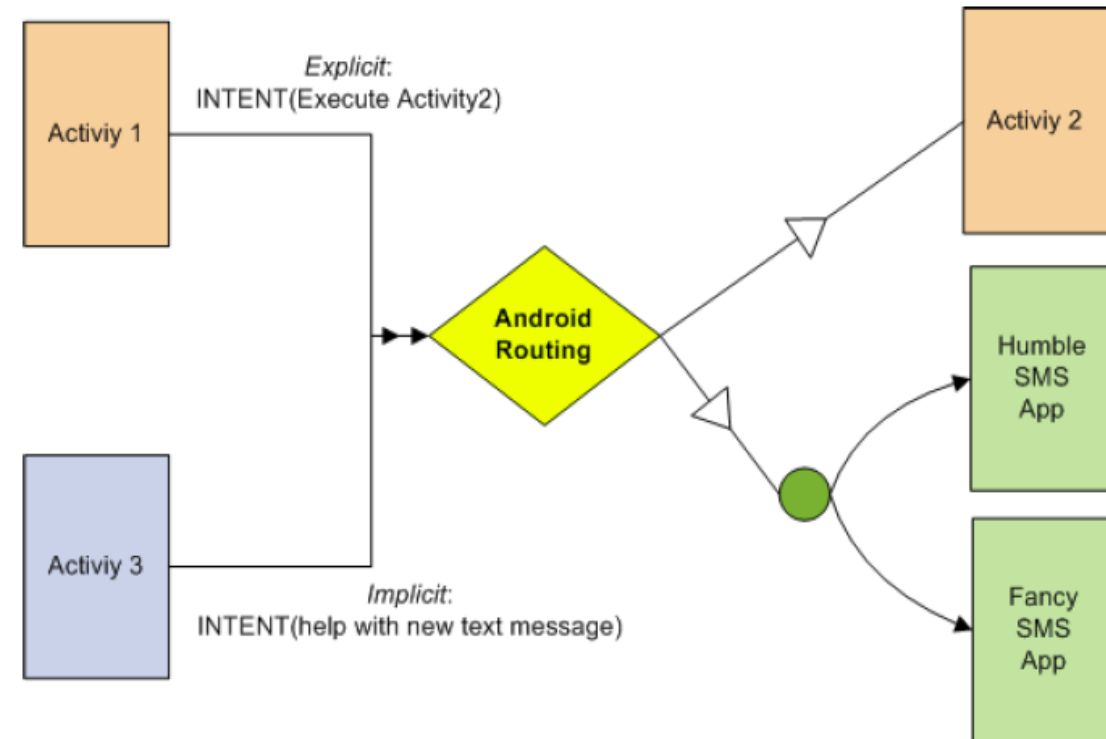
```
viewModel.getMyLiveData().observe(getViewLifecycleOwner(),
    textView::setText);
```



5. Intent Filter

- Là bộ lọc Intent, chỉ cho những Intent phù hợp đi qua.
- Thẻ **<intent-filter>** trong manifest nằm trong thẻ các component tương ứng. Xác định loại intent phù hợp thông qua 3 yếu tố: **action, category, data**
 - ❑ action*: chuỗi
 - ❑ [category]: chuỗi
 - ❑ [data]: loại dữ liệu phù hợp

Intent Resolution



Intent Filter component

❑ action

ACTION_CALL

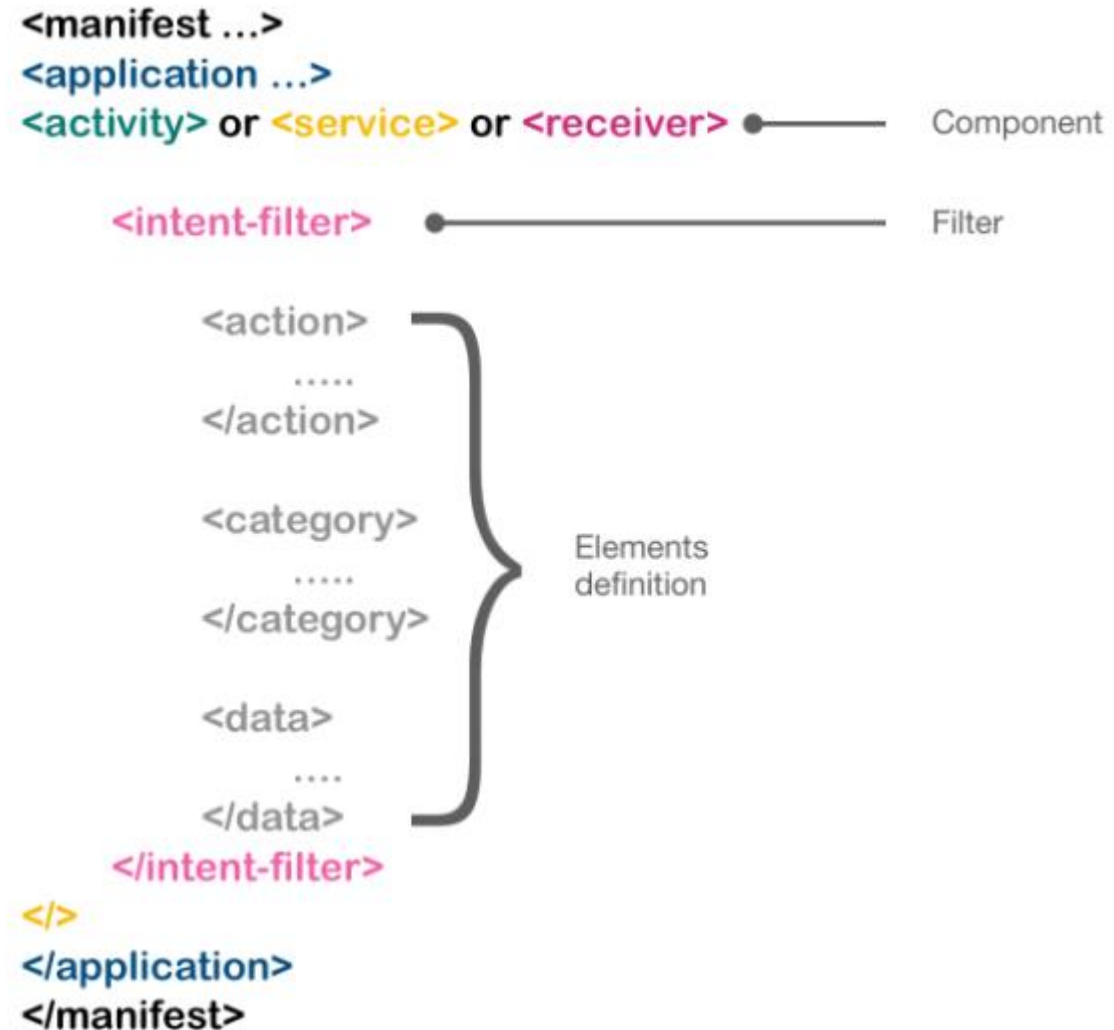
ACTION_MAIN

ACTION_BATTERY_LOW

❑ category [optional]

CATEGORY_BROWSABLE,
CATEGORY_LAUNCHER

❑ data



Ví dụ về Intent Filter

✓ Ví dụ 1: Khởi tạo màn hình chính

activity sử dụng:

- action = **MAIN**: khởi chạy activity
- category = **LAUNCHER**: biểu tượng activity này được đặt trong danh sách ứng dụng trên màn hình chính

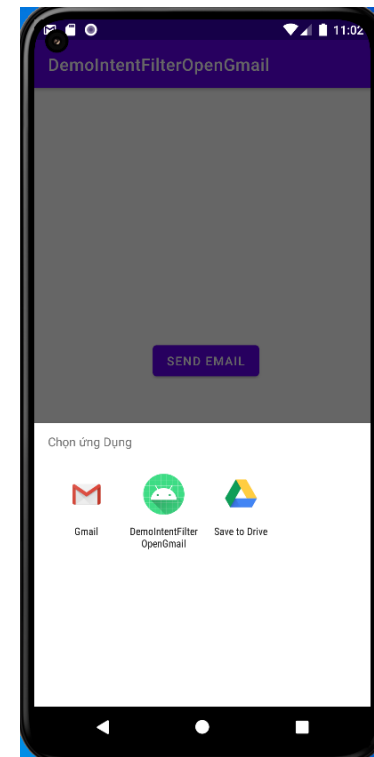
```
<activity
    android:name=".MainActivity"
    android:exported="true">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
```

✓ Ví dụ 2: Gọi ứng dụng Gmail

```
findViewById(R.id.btnSendEmail).setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent si = new Intent(Intent.ACTION_SEND);
        si.setType("message/rfc822");
        si.putExtra(Intent.EXTRA_EMAIL, new String[]{"nh.cuong@hutech.edu.vn"});
        si.putExtra(Intent.EXTRA_SUBJECT, value: "Nộp bài tập môn LTĐĐ");
        si.putExtra(Intent.EXTRA_TEXT, value: "Chào Thầy\n, Em xin nộp bài tập LTĐĐ...");
        startActivity(Intent.createChooser(si, title: "Chọn Ứng Dụng"));
    }
});
```

```
<activity
    android:name=".MainActivity"
    android:exported="true">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
        <action android:name="android.intent.action.SEND"/>
        <category android:name="android.intent.category.DEFAULT"/>
        <data android:mimeType="message/rfc822"/>
    </intent-filter>

    <meta-data
        android:name="android.app.lib_name"
        android:value="" />
</activity>
```



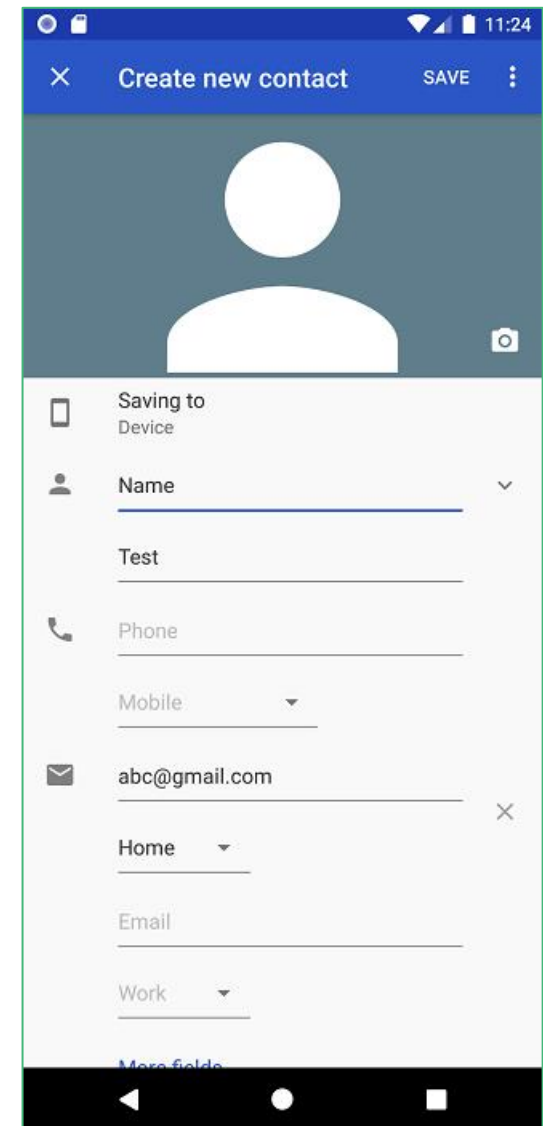
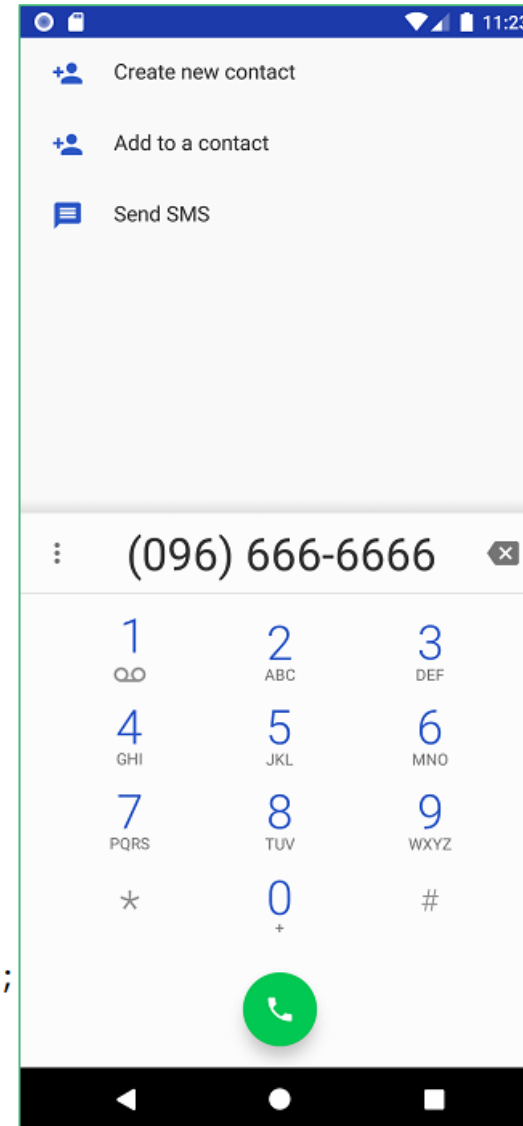
Ví dụ về Intent Filter

✓ Ví dụ 3: Gọi cuộc gọi

```
findViewById(R.id.btnCall).setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        String PhoneNumber = "0966.666666";
        Intent intent = new Intent(Intent.ACTION_DIAL);
        intent.setData(Uri.parse("tel:" + PhoneNumber));
        if (intent.resolveActivity(getPackageManager()) != null) {
            startActivity(intent);
        }
    }
});
```

✓ Ví dụ 4: Thêm liên hệ mới

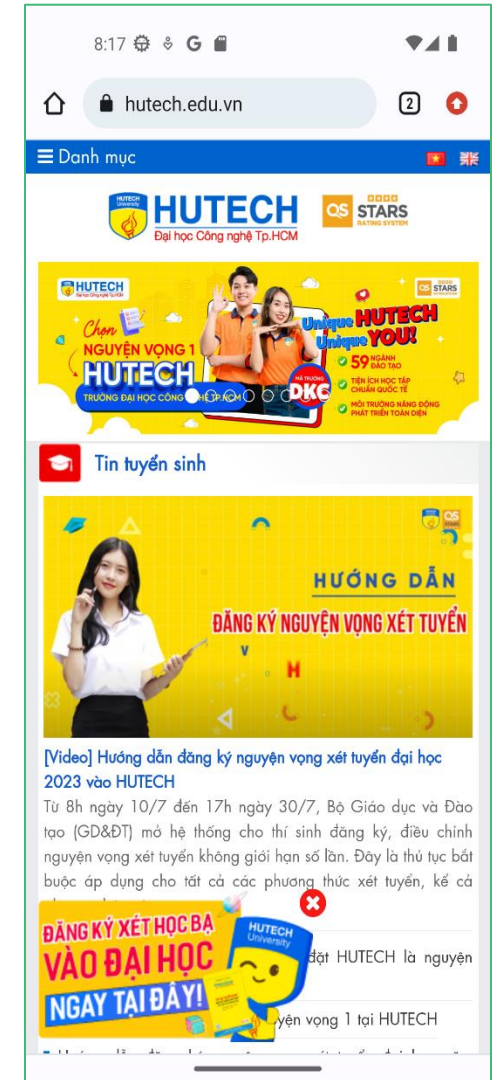
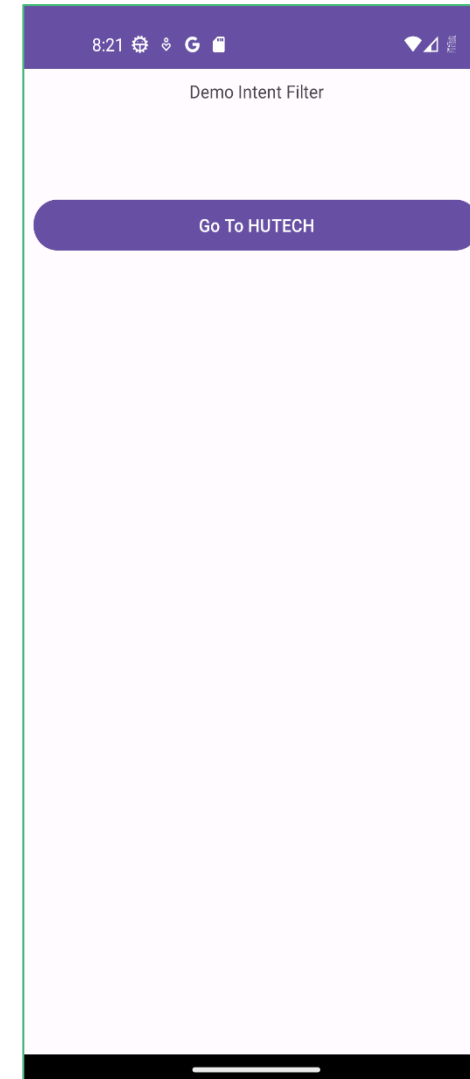
```
findViewById(R.id.btnSaveContact).setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent(Intent.ACTION_INSERT);
        intent.setType(ContactsContract.Contacts.CONTENT_TYPE);
        intent.putExtra(ContactsContract.Intents.Insert.NAME, value: "Name Test");
        intent.putExtra(ContactsContract.Intents.Insert.EMAIL, value: "abc@gmail.com");
        if (intent.resolveActivity(getPackageManager()) != null) {
            startActivity(intent);
        }
    }
});
```



Ví dụ về Intent Filter

✓ Ví dụ 5: Mở web

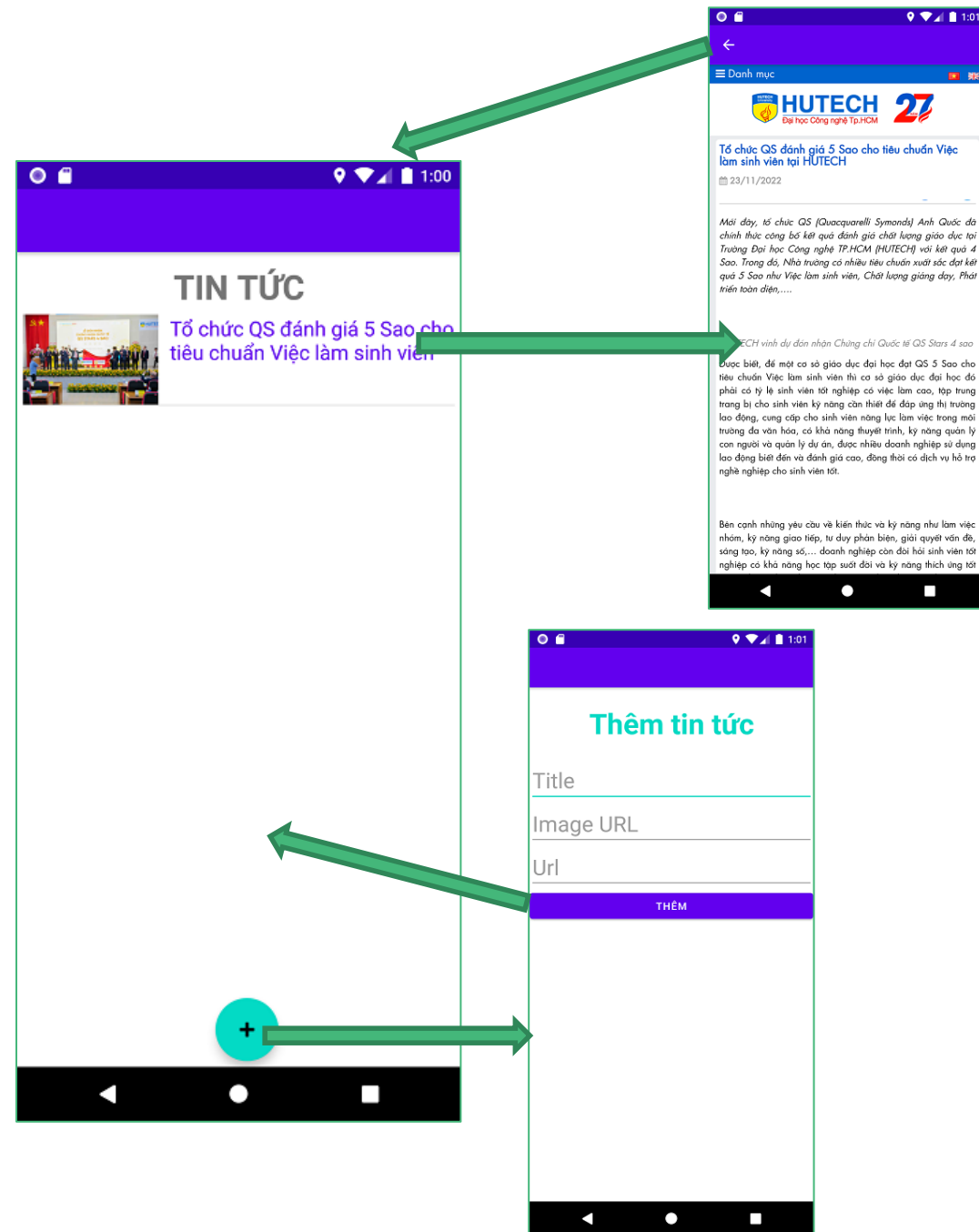
```
Button button = (Button)findViewById(R.id.button);
button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String url = "https://www.hutech.edu.vn/";
        //action
        Intent i = new Intent(Intent.ACTION_VIEW);
        //data
        i.setData(Uri.parse(url));
        startActivity(i);
    }
});
```



ASM 5.1 Ứng dụng xem tin tức (6đ)

Chương trình ứng dụng xem tin tức

1. Sử dụng **RecyclerView** Thiết kế chương trình xem tin tức. Mỗi tin có: Hình ảnh, Title , Url (3đ)
2. Khi vào **FloatActionButton** sẽ cho phép thêm mới 1 tin tức (các hình ảnh được lấy từ internet) (2đ)
3. Khi OnClick trên item sẽ mở “DetailActivity” có **WebView** để hiển thị nội dung tin tức và có thể back về. (1đ)



ASM5.2 E-Hutech (10đ)

Chương trình ứng dụng E-Hutech

Thiết kế ứng dụng BottomNavigationView có “Home”, “Thời khóa biểu” và “Cài Đặt”

1. Thiết kế “Home” và thực hiện:

1.1 Lấy tin tức mới nhất cho phần Home

Mỗi tin tức sẽ có: URL, URLImage, Title, Date

1.2 Phần Tin HUTECH

Thực hiện RecyclerView để lấy các tin tức được sắp xếp theo ngày giảm dần.

1.3 Khi OnClick vào các tin tức hoặc hình ảnh
Sẽ mở trang đọc tin tức và có thể back trở về.



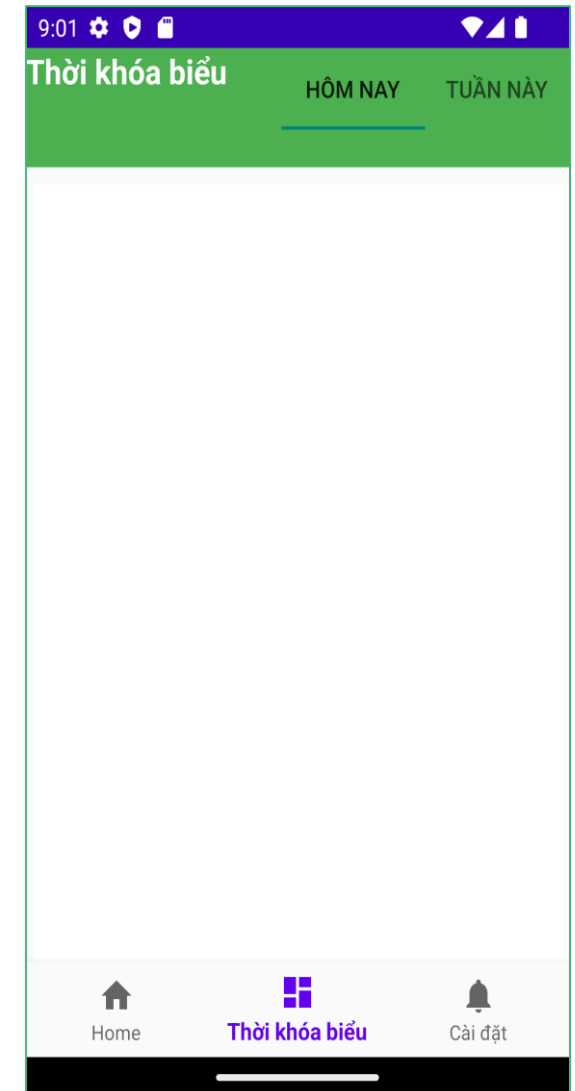
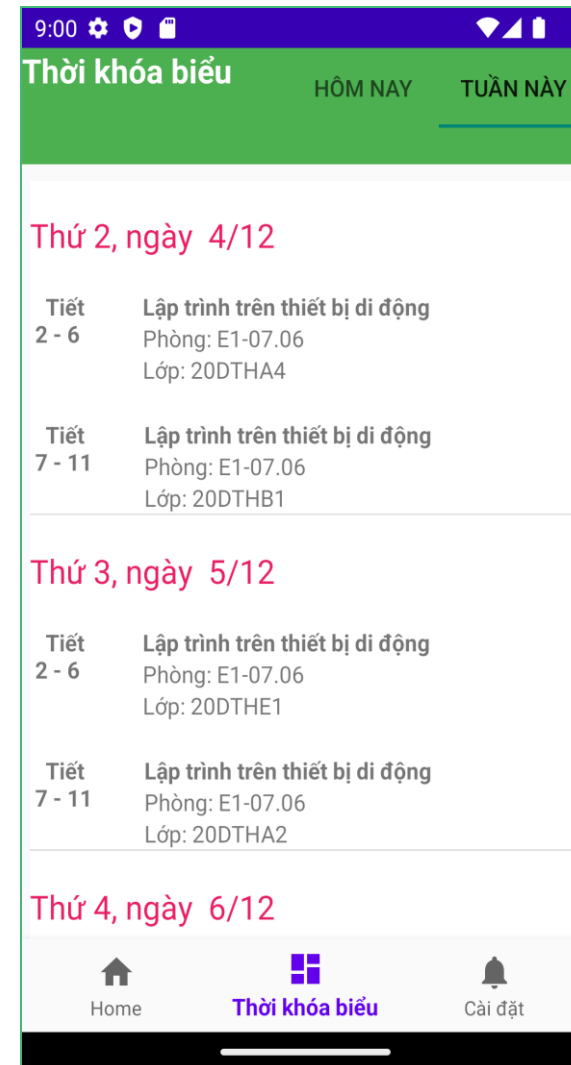
ASM5.2 E-Hutech (10đ)

2. Thiết kế “TKB” và thực hiện:

Dữ liệu thời khóa biểu có: Môn học, phòng, Lớp, tiết bắt đầu và kết thúc

2.1 Thực hiện **Tab Tuần này**: Lấy dữ liệu học Trong tuần hiện tại

2.2 Thực hiện Tab Hôm nay: Lấy dữ liệu học Của ngày hiện tại



ASM5.2 E-Hutech (10đ)

3. Thiết kế “Cài đặt” và thực hiện:

3.1 Thêm tin tức sẽ mở **Dialog** cho phép nhập các tin tức

3.2 Khi lưu tin tức ở Dialog, tin tức sẽ được cập Nhật ở Home

