

# HƯỚNG DẪN ASM ANDROID NÂNG CAO (MOB201) \*

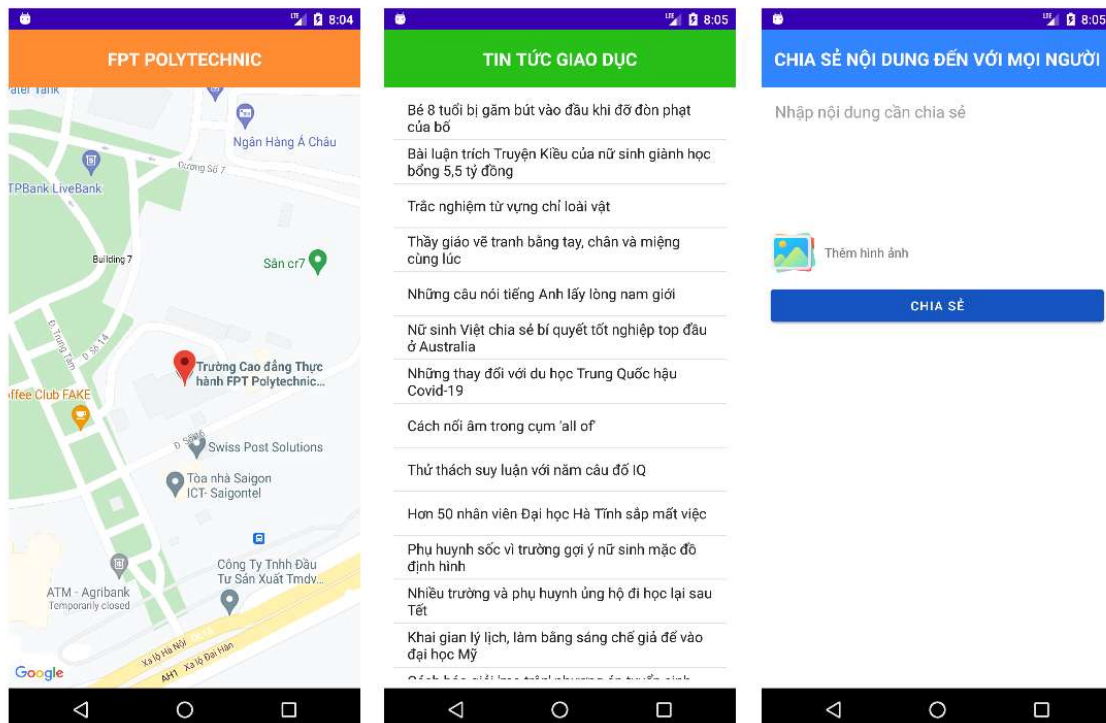
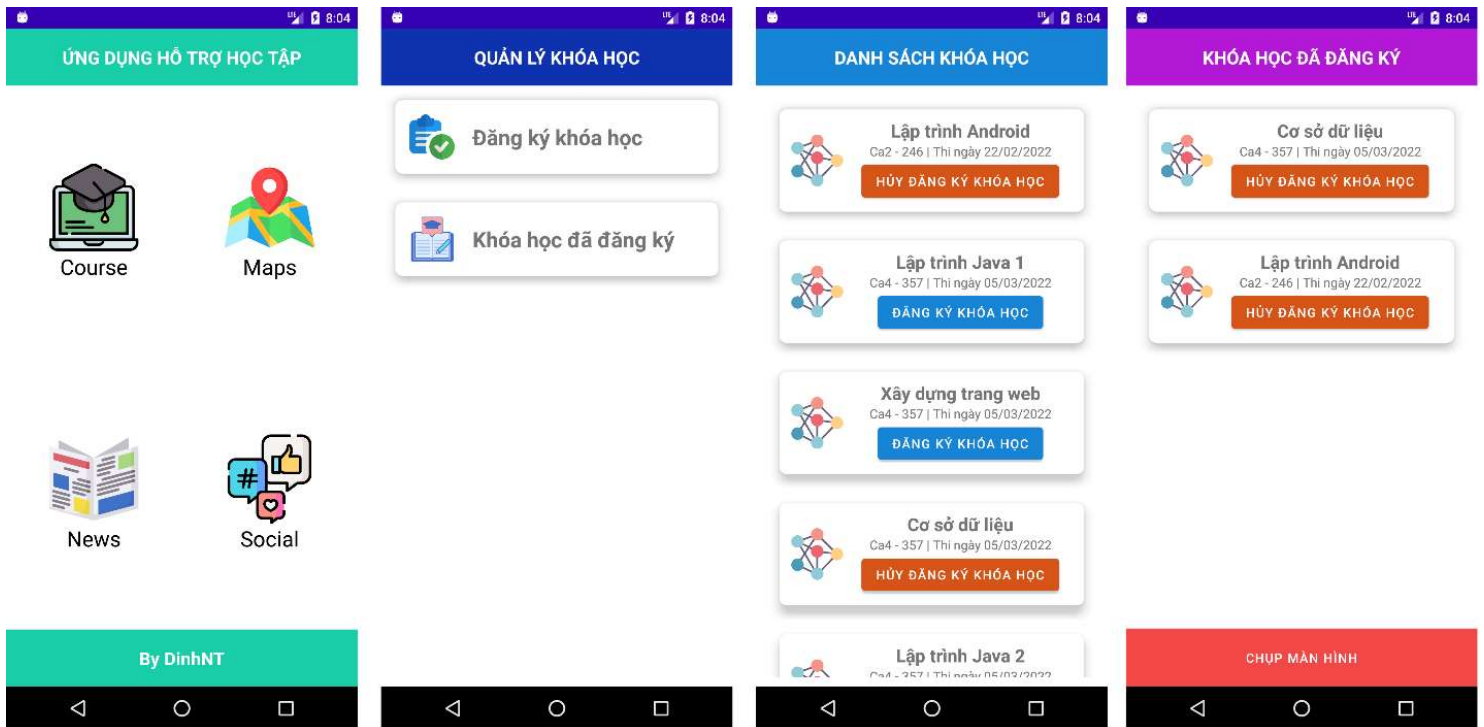
\*Sinh viên đọc kỹ ASM trước khi làm. Các chức năng còn lại được yêu cầu trong ASM

sinh viên tự thiết kế giao diện và thực hiện chức năng

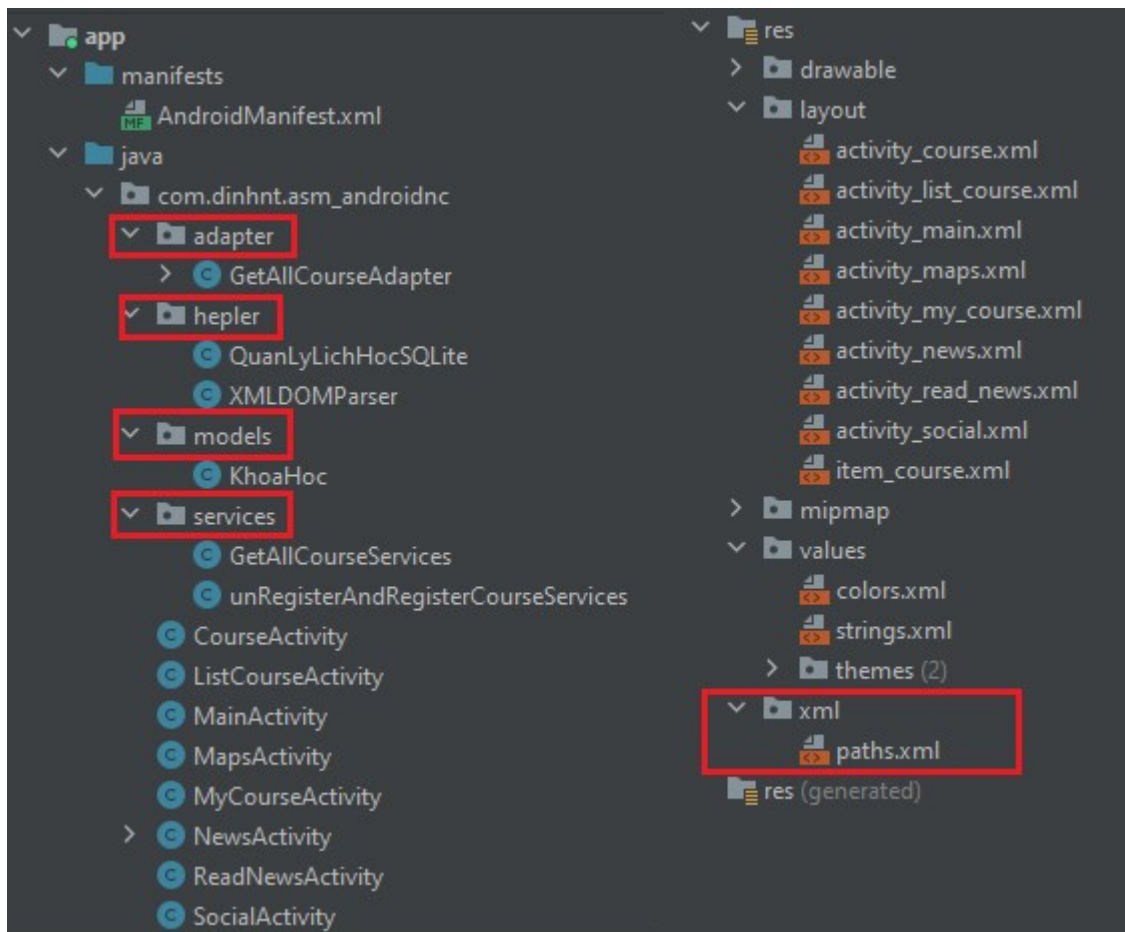
Project tham khảo: [https://github.com/dinhnguyenn/FPL\\_ASM\\_AndroidNC](https://github.com/dinhnguyenn/FPL_ASM_AndroidNC)

Demo: <https://www.youtube.com/watch?v=W4BOKhYEOBQ>

Một số hình ảnh giao diện trong hướng dẫn



Bố cục project như sau:



## 1. Chức năng **ĐĂNG KÝ KHÓA HỌC**

### a. Tạo SQLite cho chức năng

Tạo thêm 1 class tên **QuanLyLichHocSQLite** kế thừa từ **SQLiteOpenHelper**. Khai báo một số hằng số. Override 2 phương thức **onCreate**, **onUpgrade** và hàm tạo bằng cách chọn sửa lỗi nhanh. Chú ý, sau khi có hàm tạo nên sửa lại hàm tạo chỉ nhận 1 đối số là Context và trong hàm super của nó truyền tên database và version luôn để tí làm cho dễ.

```
public class QuanLyLichHocSQLite extends SQLiteOpenHelper {

    SQLiteDatabase db = this.getReadableDatabase();

    private static final String DATABASE_NAME = "quanlylichhoc";
    private static final int DATABASE_VERSION = 1;

    private static final String TABLE_SINHVIEN = "sinhvien";
    private static final String KEY_MASV = "masv";
    private static final String KEY_HOTEN = "hoten";
    private static final String KEY_NGAYSINH = "ngaysinh";
    private static final String KEY_DIACHI = "diachi";
    private static final String KEY_SDT = "sdt";

    private static final String TABLE_KHOAHOC = "khoahoc";
    private static final String KEY_MAKH = "makh";
    private static final String KEY_TENKH = "tenkh";
    private static final String KEY_LICHHOC = "lichhoc";
    private static final String KEY_LICHTHI = "lichthi";

    private static final String TABLE_THONGTINDANGKY = "thongtindangky";
    private static final String KEY_ID = "id";

    public QuanLyLichHocSQLite(Context context) {
        super(context, DATABASE_NAME, factory: null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {...}

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {...}
```

Trong hàm **onCreate** tạo các bảng sinh viên, khóa học, thông tin khóa học đã đăng ký:

```
@Override
public void onCreate(SQLiteDatabase db) {
    //tạo bảng sinh viên
    String sqlSinhVien = "create table " + TABLE_SINHVIEN +
        "(" +
        KEY_MASV + " text primary key, " +
        KEY_HOTEN + " text, " +
        KEY_NGAYSINH + " text, " +
        KEY_DIACHI + " text, " +
        KEY_SDT + " text" +
        ")";
    db.execSQL(sqlSinhVien);

    //tạo bảng khóa học
    String sqlKhoaHoc = "create table " + TABLE_KHOAHOC +
        "(" +
        KEY_MAKH + " integer primary key autoincrement, " +
        KEY_TENKH + " text, " +
        KEY_LICHHOC + " text, " +
        KEY_LICHTHI + " text" +
        ")";
    db.execSQL(sqlKhoaHoc);

    //tạo bảng thông tin đăng ký môn học
    String sqlTTDangKy = "create table " + TABLE_THONGTINDANGKY +
        "(" +
        KEY_ID + " integer primary key autoincrement, " +
        KEY_MASV + " text, " +
        KEY_MAKH + " integer" +
        ")";
    db.execSQL(sqlTTDangKy);

    //data khóa học
    String phanLoai = "INSERT INTO " + TABLE_KHOAHOC + " (" + KEY_MAKH + ", " + KEY_TENKH + ", " +
        + KEY_LICHHOC + ", " + KEY_LICHTHI + ") " +
        "Values ('1', 'Lập trình Android', 'Ca2 - 246', '22/02/2022'), " +
        "('2', 'Lập trình Java 1', 'Ca4 - 357', '05/03/2022'), " +
        "('3', 'Xây dựng trang web', 'Ca4 - 246', '05/03/2022'), " +
        "('8', 'CTDL và GT', 'Ca5 - 357', '05/03/2022'), " +
        "('9', 'Lập trình Javascript', 'Ca4 - 246', '17/03/2022'), " +
        "('10', 'Nhập môn lập trình', 'Ca1 - 246', '01/03/2022')";
    db.execSQL(phanLoai);
}
```



Trong onUpgrade:

```
@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    db.execSQL("drop table if exists " + TABLE_SINHVIEN);
    onCreate(db);

    db.execSQL("drop table if exists " + TABLE_KHOAHOC);
    onCreate(db);

    db.execSQL("drop table if exists " + TABLE_THONGTINDANGKY);
    onCreate(db);
}
```

Hàm lấy danh sách khóa học

```
//lấy danh sách khóa học
public ArrayList<KhoaHoc> getAllCourse() {
    Cursor cursor = db.rawQuery("select * from " + TABLE_KHOAHOC, null);

    ArrayList<KhoaHoc> list = new ArrayList<>();
    if (cursor.getCount() != 0) {
        cursor.moveToFirst();
        do {
            list.add(new KhoaHoc(cursor.getInt(0), cursor.getString(1), cursor.getString(2), cursor.getString(3)));
        } while (cursor.moveToNext());
    }
    return list;
}
```

Hàm hủy đăng ký khóa học

```
//Hủy khóa học đã đăng ký
public void unRegisterCourse(String masv, int makh) {
    db.delete(TABLE_THONGTINDANGKY, "masv=? and makh=?", new String[]{masv, String.valueOf(makh)});
}
```

Hàm xem danh sách khóa học đã đăng ký

```
//xem danh sách khóa học & lịch thi đã đăng ký
public ArrayList<KhoaHoc> getAllCourseRegister(String masv) {
    Cursor cursor = db.rawQuery("select * from " + TABLE_KHOAHOC + " kh, " + TABLE_THONGTINDANGKY
        + " tt where kh.makh = tt.makh and tt.masv like '" + masv + "'", null);

    ArrayList<KhoaHoc> list = new ArrayList<>();
    if (cursor.getCount() != 0) {
        cursor.moveToFirst();
        do {
            list.add(new KhoaHoc(cursor.getInt(0), cursor.getString(1), cursor.getString(2), cursor.getString(3)));
        } while (cursor.moveToNext());
    }
    return list;
}
```

## b. Chức năng ĐĂNG KÝ/HỦY KHÓA HỌC

Tạo service **GetAllCourseServices** **extends IntentService** để lấy danh sách khóa học (bao gồm khóa học chưa/đã đăng ký)

```
public class GetAllCourseServices extends IntentService {

    public GetAllCourseServices() { super(name: "GetAllCourseServices"); }

    @Override
    protected void onHandleIntent(@Nullable Intent intent) {
        if (intent != null) {
            QuanLyLichHocSQLite quanLyLichHocSQLite = new QuanLyLichHocSQLite(getApplicationContext());

            Intent i = new Intent(action: "GetAllCourseServices");
            final String action = intent.getAction();
            String masv = intent.getStringExtra(name: "masv");
            boolean isMine = intent.getBooleanExtra(name: "isMine", defaultValue: false);

            if (!isMine) {
                ArrayList<KhoaHoc> allCourse = quanLyLichHocSQLite.getAllCourse();
                i.putExtra(name: "allCourse", allCourse);
            }

            ArrayList<KhoaHoc> allCourseRegister = quanLyLichHocSQLite.getAllCourseRegister(masv);
            i.putExtra(name: "allCourseRegister", allCourseRegister);

            quanLyLichHocSQLite.close();

            i.putExtra(name: "action", action);
            i.putExtra(name: "resultCode", Activity.RESULT_OK);
            LocalBroadcastManager.getInstance(this).sendBroadcast(i);
        }
    }
}
```

Tạo thêm service **unRegisterAndRegisterCourseServices** tương tự như service **GetAllCourseServices** để thực hiện chức năng đăng ký/hủy khóa học

```
public class unRegisterAndRegisterCourseServices extends IntentService {

    public unRegisterAndRegisterCourseServices() { super( name: "unRegisterAndRegisterCourseServices"); }

    @Override
    protected void onHandleIntent(@Nullable Intent intent) {
        if (intent != null) {
            QuanLyLichHocSQLite quanLyLichHocSQLite = new QuanLyLichHocSQLite(getApplicationContext());

            Intent i = new Intent( action: "unRegisterAndRegisterCourseServices");
            final String action = intent.getAction();
            String masv = intent.getStringExtra( name: "masv");
            int makh = intent.getIntExtra( name: "makh", defaultValue: -1);
            boolean isRegister = intent.getBooleanExtra( name: "isRegister", defaultValue: false);

            if(isRegister){
                quanLyLichHocSQLite.registerCourse(masv, makh); //đăng ký
            }else {
                quanLyLichHocSQLite.unRegisterCourse(masv, makh); //hủy đăng ký
            }

            ArrayList<KhoaHoc> allCourse = quanLyLichHocSQLite.getAllCourse();
            ArrayList<KhoaHoc> allCourseRegister = quanLyLichHocSQLite.getAllCourseRegister(masv);
            quanLyLichHocSQLite.close();
            i.putExtra( name: "allCourse", allCourse);
            i.putExtra( name: "allCourseRegister", allCourseRegister);
            i.putExtra( name: "action", action);
            i.putExtra( name: "resultCode", Activity.RESULT_OK);
            LocalBroadcastManager.getInstance(this).sendBroadcast(i);
        }
    }
}
```

Đăng ký service **GetAllCourseServices** và **unRegisterAndRegisterCourseServices** trong **AndroidManifest.xml**

```
<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="ASM_AndroidNC"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.ASM_AndroidNC">
    <activity...>
    <activity...>

    <meta-data...>

    <provider...>

    <service
        android:name=".services.GetAllCourseServices"
        android:exported="false" />
    <service
        android:name=".services.UnRegisterAndRegisterCourseServices"
        android:exported="false" />
</application>
```

Trong **ListCourseActivity**, tạo **BroadcastReceiver** để nhận dữ liệu gửi về từ service khi load toàn bộ danh sách khóa học và thực hiện đăng ký/hủy khóa học

```
private final BroadcastReceiver getAllCourseReceiver = (context, intent) -> {
    int resultCode = intent.getIntExtra( name: "resultCode", RESULT_CANCELED);
    if (resultCode == RESULT_OK) {
        String action = intent.getStringExtra( name: "action");
        switch (action){
            case "GetAllCourseServices":
            case "unRegisterAndRegisterCourseServices":
                alKhoaHoc.clear();
                ArrayList<KhoaHoc> alCourse = (ArrayList<KhoaHoc>) intent.getSerializableExtra( name: "allCourse");
                ArrayList<KhoaHoc> alCourseRegister = (ArrayList<KhoaHoc>) intent.getSerializableExtra( name: "allCourseRegister");
                for(KhoaHoc kh : alCourse){
                    for(KhoaHoc khdk : alCourseRegister){
                        if(kh.getId() == khdk.getId()){
                            kh.setCheck(true);
                            break;
                        }
                    }
                    alKhoaHoc.add(kh);
                }

                GetAllCourseAdapter adapter = new GetAllCourseAdapter(alKhoaHoc, context: ListCourseActivity.this, quanLyLichHocSQLite);
                listViewAllCourse.setAdapter(adapter);
                adapter.notifyDataSetChanged();
                break;
            default:
                break;
        }
    }
};

@Override
protected void onPause() {
    super.onPause();
    LocalBroadcastManager.getInstance(this).unregisterReceiver(getAllCourseReceiver);
}
```



Trong **onCreate** thực hiện đăng ký 2 services vừa tạo ở trên thông qua **LocalBroadcastManager**

```
public class ListCourseActivity extends AppCompatActivity {
    ArrayList<KhoaHoc> alKhoaHoc = new ArrayList<>();
    QuanLyLichHocSQLite quanLyLichHocSQLite;
    ListView listViewAllCourse;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_list_course);

        listViewAllCourse = findViewById(R.id.listViewAllCourse);
        quanLyLichHocSQLite = new QuanLyLichHocSQLite(context: this);

        IntentFilter filterGetAllCourse = new IntentFilter(action: "GetAllCourseServices");
        LocalBroadcastManager.getInstance(this).registerReceiver(getAllCourseReceiver, filterGetAllCourse);

        IntentFilter filterRegisterCourse = new IntentFilter(action: "unRegisterAndRegisterCourseServices");
        LocalBroadcastManager.getInstance(this).registerReceiver(getAllCourseReceiver, filterRegisterCourse);

        Intent intent = new Intent(packageContext: this, GetAllCourseServices.class);
        //thay đổi mssv phù hợp với chức năng đăng nhập/đăng ký
        intent.putExtra(name: "masv", value: "ps00709");
        intent.putExtra(name: "isMine", value: false);
        intent.setAction("GetAllCourseServices");
        startService(intent);
    }
}
```

**Lưu ý:** Data mã số sinh viên các bạn lấy từ chức năng đăng nhập/đăng ký mà ASM yêu cầu

Tạo layout cho từng item khóa học (**item\_course**):

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <androidx.cardview.widget.CardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_rowWeight="1"
        android:layout_columnWeight="1"
        android:layout_margin="16dp"
        app:cardCornerRadius="8dp"
        app:cardElevation="8dp">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_margin="10dp"
            android:weightSum="10">

            <ImageView
                android:layout_width="0dp"
                android:layout_height="match_parent"
                android:layout_weight="2"
                android:src="@mipmap/neural" />

            <LinearLayout
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:layout_weight="8"
                android:orientation="vertical">
                <TextView
                    android:id="@+id/txtTenKH"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:gravity="center"
                    android:src="@mipmap/neural"
                    android:text="Lập trình Android"
                    android:textSize="20sp"
                    android:textStyle="bold" />
                <TextView
                    android:id="@+id/txtNoiDungKH"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:gravity="center"
                    android:src="@mipmap/neural"
                    android:text="Ca2 - 246 | Thi 22/02/2022" />
                <Button
                    android:id="@+id/btnKH"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_gravity="center"
                    android:backgroundTint="#1884D6"
                    android:text="Đăng ký khóa học" />
            </LinearLayout>
        </LinearLayout>
    </androidx.cardview.widget.CardView>
</LinearLayout>
```

## Tạo adapter GetAllCourseAdapter quản lý danh sách khóa học

```
public class GetAllCourseAdapter extends BaseAdapter {

    ArrayList<KhoaHoc> alKhoaHoc = new ArrayList<>();
    Context context;
    QuanLyLichHocSQLite quanLyLichHoc;

    public GetAllCourseAdapter(ArrayList<KhoaHoc> alKhoaHoc, Context context, QuanLyLichHocSQLite quanLyLichHoc) {
        this.alKhoaHoc = alKhoaHoc;
        this.context = context;
        this.quanLyLichHoc = quanLyLichHoc;
    }

    @Override
    public int getCount() { return alKhoaHoc.size(); }

    @Override
    public Object getItem(int i) { return alKhoaHoc.get(i); }

    @Override
    public long getItemId(int i) { return alKhoaHoc.get(i).getId(); }

    @Override
    public View getView(int i, View view, ViewGroup viewGroup) {
        ViewOfItem viewOfItem;
        LayoutInflater inflater = ((Activity) context).getLayoutInflater();

        if (view == null) {
            view = inflater.inflate(R.layout.item_course, root: null);
            viewOfItem = new ViewOfItem();
            viewOfItem.txtTenKH = view.findViewById(R.id.txtTenKH);
            viewOfItem.txtNoiDungKH = view.findViewById(R.id.txtNoiDungKH);
            viewOfItem.btnKH = view.findViewById(R.id.btnKH);

            view.setTag(viewOfItem);
        } else {
            viewOfItem = (ViewOfItem) view.getTag();
        }

        viewOfItem.txtTenKH.setText(alKhoaHoc.get(i).getTenkh());
        String noiDung = alKhoaHoc.get(i).getLichhoc() + " | Thi ngày " + alKhoaHoc.get(i).getLichthi();
        viewOfItem.txtNoiDungKH.setText(noiDung);

        if(alKhoaHoc.get(i).isChecked()){
            viewOfItem.btnKH.setText("Hủy đăng ký khóa học");
            viewOfItem.btnKH.setBackgroundColor(Color.parseColor("colorString: "#D65418"));
        }else{
            viewOfItem.btnKH.setText("Đăng ký khóa học");
            viewOfItem.btnKH.setBackgroundColor(Color.parseColor("colorString: "#1884D6"));
        }
    }
}
```

Trong hàm **getView** viết sự kiện click vào nút **Đăng ký/Hủy đăng ký khóa học**

```
//đăng ký/hủy khóa học
viewOfItem.btnKH.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent(context, unregisterAndRegisterCourseServices.class);
        //thay đổi mssv phù hợp với chức năng đăng nhập/đăng ký
        intent.putExtra(name: "masv", value: "ps00709");
        intent.putExtra(name: "makh", alKhoaHoc.get(i).getId());

        //đăng ký khóa học - true | hủy đăng ký khóa học - false
        intent.putExtra(name: "isRegister", !alKhoaHoc.get(i).isChecked());

        intent.setAction("unRegisterAndRegisterCourseServices");
        context.startService(intent);
    }
});
```

**Lưu ý:** Data mã số sinh viên các bạn lấy từ chức năng đăng nhập/đăng ký mà ASM yêu cầu



### c. Chức năng **HIỂN THỊ THÔNG TIN KHÓA HỌC ĐÃ ĐĂNG KÝ (THÔNG TIN KHÓA HỌC, LỊCH HỌC, LỊCH THI)**

Tương tự chức năng đăng ký/hủy đăng ký khóa học ở trên, trong **MyCourseActivity**, tạo **BroadcastReceiver** để nhận dữ liệu gửi về từ service khi load toàn bộ danh sách khóa học đã đăng ký và thực hiện hủy khóa học

```
private final BroadcastReceiver getAllCourseRegisterReceiver = (context, intent) → {
    int resultCode = intent.getIntExtra( name: "resultCode", RESULT_CANCELED);
    if (resultCode == RESULT_OK) {
        String action = intent.getStringExtra( name: "action");
        switch (action) {
            case "GetAllCourseServices":
            case "unRegisterAndRegisterCourseServices":
                alKhoaHoc.clear();
                ArrayList<KhoaHoc> alCourseRegister = (ArrayList<KhoaHoc>) intent.getSerializableExtra( name: "allCourseRegister");

                for (KhoaHoc khdk : alCourseRegister) {
                    khdk.setCheck(true);
                    alKhoaHoc.add(khdk);
                }

                GetAllCourseAdapter adapter = new GetAllCourseAdapter(alKhoaHoc, context: MyCourseActivity.this, quanLyLichHocSQLite);
                listViewAllCourseRegister.setAdapter(adapter);
                adapter.notifyDataSetChanged();
                break;
            default:
                break;
        }
    }
};

@Override
protected void onPause() {
    super.onPause();
    LocalBroadcastManager.getInstance(this).unregisterReceiver(getAllCourseRegisterReceiver);
}
```

Trong **onCreate** thực hiện đăng ký 2 services vừa tạo ở trên thông qua **LocalBroadcastManager**

```
public class MyCourseActivity extends AppCompatActivity {
    ArrayList<KhoaHoc> alKhoaHoc = new ArrayList<>();
    QuanLyLichHocSQLite quanLyLichHocSQLite;
    ListView listViewAllCourseRegister;
    Button btnCapture;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_my_course);

        listViewAllCourseRegister = findViewById(R.id.listViewAllCourseRegister);
        btnCapture = findViewById(R.id.btnCapture);

        quanLyLichHocSQLite = new QuanLyLichHocSQLite( context: this);

        IntentFilter filterGetAllCourse = new IntentFilter( action: "GetAllCourseServices");
        LocalBroadcastManager.getInstance(this).registerReceiver(getAllCourseRegisterReceiver, filterGetAllCourse);

        IntentFilter filterRegisterCourse = new IntentFilter( action: "unRegisterAndRegisterCourseServices");
        LocalBroadcastManager.getInstance(this).registerReceiver(getAllCourseRegisterReceiver, filterRegisterCourse);

        Intent intent = new Intent( packageContext: this, GetAllCourseServices.class);
        //thay đổi mssv phù hợp với chức năng đăng nhập/dăng ký
        intent.putExtra( name: "mssv", value: "ps00709");
        intent.putExtra( name: "isMine", value: true);
        intent.setAction("GetAllCourseServices");
        startService(intent);
    }
}
```

**Lưu ý:** Data mã số sinh viên các bạn lấy từ chức năng đăng nhập/dăng ký mà ASM yêu cầu

Tạo thêm hàm **ShareDetailCourse** để thực hiện chức năng chia sẻ thông tin khóa học đã đăng ký

```
private void ShareDetailCourse() {
    String content = "ĐinhNT đã chia sẻ những khóa học đã đăng ký\n\n";
    for (KhoaHoc khdk : aKhoaHoc) {
        content = content.concat(khdk.getTenkh())
            .concat("\nLịch học: ")
            .concat(khdk.getLichhoc())
            .concat("\nLịch thi: ")
            .concat(khdk.getLichthi())
            .concat("\n-----\n");
    }

    Intent intent = new Intent(Intent.ACTION_SEND);
    intent.putExtra(Intent.EXTRA_TEXT, content);
    intent.putExtra(Intent.EXTRA_SUBJECT, value: "ĐinhNT đã chia sẻ");
    intent.setType("text/plain");

    startActivity(Intent.createChooser(intent, title: "Chia sẻ nội dung thông qua"));
}
```

Trong onCreate bổ sung sự kiện click vào nút chia sẻ

```
btnCapture.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        ShareDetailCourse();
    }
});
```

## 2. Chức năng HIỂN THỊ BẢN ĐỒ

Tham khảo phần hướng dẫn trong file **Hướng dẫn tích hợp Google\_map.pdf**

Trong Activity bản đồ (**MapsActivity**) implements **OnMapReadyCallback** và thực hiện viết code trong hàm **onMapReady** như bên dưới

```
public class MapsActivity extends AppCompatActivity implements OnMapReadyCallback {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_maps);

        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager().findFragmentById(R.id.fragMap);
        mapFragment.getMapAsync(this);
    }

    @Override
    public void onMapReady(@NonNull GoogleMap googleMap) {
        LatLng FPT = new LatLng(10.8529391, 106.6295448);
        googleMap.addMarker(new MarkerOptions().position(FPT).title("FPT Polytechnic"));
        CameraUpdate cameraUpdate = CameraUpdateFactory.newLatLngZoom(FPT, 18);
        googleMap.animateCamera(cameraUpdate, 3000, null);
    }
}
```

### 3. Chức năng XEM TIN TỨC THÔNG QUA RSS

Code giao diện hiển thị danh sách tin tức

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

    <TextView
        android:id="@+id/title"
        android:layout_width="match_parent"
        android:layout_height="64dp"
        android:background="#28BF18"
        android:gravity="center"
        android:text="TIN TỨC GIAO DỤC"
        android:textColor="@color/white"
        android:textSize="20sp"
        android:textStyle="bold" />

    <ListView
        android:id="@+id/listViewNews"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="10dp" />

</LinearLayout>
```

Tạo file XMLDOMParser hỗ trợ việc parse XML

```
public class XMLDOMParser {
    public Document getDocument(String xml) {
        Document document = null;
        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
        try {
            DocumentBuilder db = factory.newDocumentBuilder();
            InputSource is = new InputSource();
            is.setCharacterStream(new StringReader(xml));
            is.setEncoding("UTF-8");
            document = db.parse(is);
        } catch (ParserConfigurationException e) {
            Log.e( tag: "Error: ", e.getMessage(), e);
            return null;
        } catch (SAXException e) {
            Log.e( tag: "Error: ", e.getMessage(), e);
            return null;
        } catch (IOException e) {
            Log.e( tag: "Error: ", e.getMessage(), e);
            return null;
        }
        return document;
    }

    public String getValue(Element item, String name) {
        NodeList nodes = item.getElementsByTagName(name);
        return this.getTextNodeValue(nodes.item( 0 ));
    }

    private final String getTextNodeValue(Node elem) {
        Node child;
        if (elem != null) {
            if (elem.hasChildNodes()) {
                for (child = elem.getFirstChild(); child != null; child = child.getNextSibling()) {
                    if (child.getNodeType() == Node.TEXT_NODE) {
                        return child.getNodeValue();
                    }
                }
            }
        }
        return "";
    }
}
```



Trong **NewsActivity** tạo class **ReadRSS extends AsyncTask<String, Void, String>** để lấy toàn bộ data từ link RSS

```
private class ReadRSS extends AsyncTask<String, Void, String> {

    @Override
    protected String doInBackground(String... strings) {
        StringBuilder content = new StringBuilder();
        try {
            URL url = new URL(strings[0]);
            InputStreamReader inputStreamReader = new InputStreamReader(url.openConnection().getInputStream());
            BufferedReader bufferedReader = new BufferedReader(inputStreamReader);
            String line = "";
            while ((line = bufferedReader.readLine()) != null) {
                content.append(line);
            }
            bufferedReader.close();

        } catch (IOException e) {
            e.printStackTrace();
        }
        return content.toString();
    }

    @Override
    protected void onPostExecute(String s) {
        super.onPostExecute(s);

        XMLDOMParser parser = new XMLDOMParser();

        Document document = parser.getDocument(s);

        NodeList nodeList = document.getElementsByTagName("item");

        String title = "";
        for (int i = 0; i < nodeList.getLength(); i++) {
            Element element = (Element) nodeList.item(i);
            title = parser.getValue(element, "title");
            arrTitle.add(title);
            arrLink.add(parser.getValue(element, "link"));
        }
        adapter.notifyDataSetChanged();
    }
}
```

Trong **onCreate** thực hiện gọi **ReadRSS** ở trên và add toàn bộ danh sách tin tức đọc được từ RSS vào listview

```
public class NewsActivity extends AppCompatActivity {

    ListView lstTieuDe;
    ArrayList<String> arrTitle, arrLink;
    ArrayAdapter adapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_news);

        lstTieuDe = findViewById(R.id.listViewNews);
        arrTitle = new ArrayList<>();
        arrLink = new ArrayList<>();

        adapter = new ArrayAdapter<String>(context: this, android.R.layout.simple_list_item_1, arrTitle);
        lstTieuDe.setAdapter(adapter);

        new ReadRSS().execute("https://vnexpress.net/rss/giao-duc.rss");

        lstTieuDe.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {
                Intent intent = new Intent(packageContext: NewsActivity.this, ReadNewsActivity.class);
                intent.putExtra(name: "linkNews", arrLink.get(i));
                startActivity(intent);
            }
        });
    }
}
```

Tạo thêm activity **ReadNewsActivity** để đọc chi tiết từng tin tức

Layout:

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    tools:context=".ReadNewsActivity">

    <WebView
        android:id="@+id/webView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>

</LinearLayout>
```

Trong onCreate

```
public class ReadNewsActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_read_news);

        WebView webView = findViewById(R.id.webView);

        Intent intent = getIntent();

        String link = intent.getStringExtra("linkNews");

        webView.loadUrl(link);

        webView.setWebViewClient(new WebViewClient());
    }
}
```

#### 4. Chức năng CHIA SẺ NỘI DUNG

Tạo giao diện cho chức năng

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

    <TextView
        android:id="@+id/title"
        android:layout_width="match_parent"
        android:layout_height="64dp"
        android:background="#3385FF"
        android:gravity="center"
        android:text="CHIA SẺ NỘI DUNG ĐẾN VỚI MỌI NGƯỜI"
        android:textColor="@color/white"
        android:textSize="20sp"
        android:textStyle="bold" />

    <EditText
        android:id="@+id/edtContent"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@android:color/transparent"
        android:gravity="top"
        android:hint="Nhập nội dung cần chia sẻ"
        android:inputType="text|textMultiLine"
        android:lines="8"
        android:maxLines="10"
        android:minLines="6"
        android:padding="15dp" />
```

```

<LinearLayout
    android:id="@+id/linearShare"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="10dp">

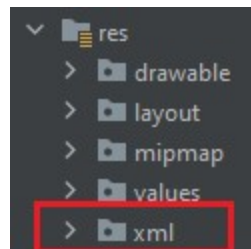
    <ImageView
        android:id="@+id/imgShare"
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:src="@mipmap/picture" />

    <TextView
        android:id="@+id/txtImage"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginStart="10dp"
        android:gravity="start|center"
        android:text="Thêm hình ảnh" />
</LinearLayout>

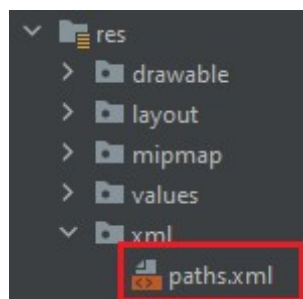
<Button
    android:id="@+id/btnShare"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#1655C1"
    android:layout_margin="10dp"
    android:text="Chia sẻ" />
</LinearLayout>

```

Trong **res**, tạo thêm thư mục **xml** để quản lý paths khi lấy hình ảnh để chia sẻ



Trong thư mục **xml**, tạo file **paths** để quản lý đường dẫn khi lấy hình ảnh từ điện thoại để chia sẻ



```

paths.xml x
1  <?xml version="1.0" encoding="utf-8"?>
2  <paths>
3      <cache-path
4          name="shared_images"
5          path="images/" />
6  </paths>

```



Trong **AndroidManifest**, trong thẻ **application**, khai báo **provider** lấy hình ảnh từ điện thoại

```
<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="ASM_AndroidNC"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.ASM_AndroidNC">
    <activity...>
    <activity...>
    <activity...>
    <activity...>
    <activity...>
    <activity...>
    <activity...>
    <activity...>

    <meta-data...>

    <provider
        android:name="androidx.core.content.FileProvider"
        android:authorities="com.dinhnt.asm_androidnc.fileprovider"
        android:exported="false"
        android:grantUriPermissions="true">
        <meta-data
            android:name="android.support.FILE_PROVIDER_PATHS"
            android:resource="@xml/paths" />
        </provider>

    <service...>
    <service...>
</application>
```

Cấp quyền cho chức năng

```
<uses-permission android:name="android.permission.INTERNET"
    tools:ignore="ManifestOrder" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"
    tools:ignore="ScopedStorage" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
```

Code chức năng click vào chọn hình ảnh, lấy hình ảnh từ điện thoại và hiển thị lên imageView

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_social);

    EditText edtContent = findViewById(R.id.edtContent);
    Button btnShare = findViewById(R.id.btnShare);
    LinearLayout linearShare = findViewById(R.id.linearShare);
    txtImage = findViewById(R.id.txtImage);
    imgShare = findViewById(R.id.imgShare);

    linearShare.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent i = new Intent();
            i.setType("image/*");
            i.setAction(Intent.ACTION_GET_CONTENT);

            chooseImage.launch(i);
        }
    });

    btnShare.setOnClickListener(new View.OnClickListener() {...});
}
```

```
ActivityResultLauncher<Intent> chooseImage = registerForActivityResult(
    new ActivityResultContracts.StartActivityForResult(),
    new ActivityResultCallback<ActivityResult>() {
        @Override
        public void onActivityResult(ActivityResult result) {
            if (result.getResultCode() == Activity.RESULT_OK) {
                Intent data = result.getData();
                Uri selectedImageUri = data.getData();
                if (null != selectedImageUri) {
                    imgShare.setImageURI(selectedImageUri);
                    txtImage.setText("Lựa chọn lại hình ảnh");
                    BitmapDrawable bitmapDrawable = (BitmapDrawable) imgShare.getDrawable();
                    bitmap = bitmapDrawable.getBitmap();
                }
            }
        }
    });
```

Tiếp tục thực hiện chức năng share nội dung cho các ứng dụng khác (zalo, facebook, messenger,...)

Viết hàm **getImageToShare** để lấy URI của hình ảnh chia sẻ

```
private Uri getImageToShare(Bitmap bitmap) {
    File imagefolder = new File(getCacheDir(), "images");
    Uri uri = null;
    try {
        imagefolder.mkdirs();
        File file = new File(imagefolder, "shared_image.png");
        FileOutputStream outputStream = new FileOutputStream(file);
        bitmap.compress(Bitmap.CompressFormat.PNG, 90, outputStream);
        outputStream.flush();
        outputStream.close();
        uri = FileProvider.getUriForFile(context, this, authority: "com.dinhnt.asm_androidnc.fileprovider", file);
    } catch (Exception e) {
        Toast.makeText(context, this, text: "" + e.getMessage(), Toast.LENGTH_LONG).show();
    }
    return uri;
}
```

Viết sự kiện click và nút share để chia sẻ nội dung

```
btnShare.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent(Intent.ACTION_SEND);
        intent.putExtra(Intent.EXTRA_TEXT, edtContent.getText().toString());
        intent.putExtra(Intent.EXTRA_SUBJECT, value: "DinhNT chia sẻ");

        if(bitmap != null){
            Uri uri = getImageToShare(bitmap);
            intent.putExtra(Intent.EXTRA_STREAM, uri);
            intent.setType("image/png");
        }else{
            intent.setType("text/plain");
        }

        startActivity(Intent.createChooser(intent, title: "Chia sẻ nội dung thông qua"));
    }
});
```

Chúc các bạn thành công

**DinhNT <3**