**TRƯỜNG CAO ĐẲNG LÝ TỰ TRỌNG**

**-= ♛ =-**



**Học phần: Lập Trình Thiết Bị Di Động**

**Đề Tài : APP NGHE NHẠC OFFLINE**

**Nhóm : …..**

**- Nguyễn Tô Hiệu**

**- Đỗ Thị Lan Anh**

**- Đỗ Thị Tuyết Mai**

**Lớp : 17C1 - LTM2**

**GV Hướng dẫn:** **Mai Chiêm Tuấn**

TP.Hồ Chí Minh, Ngày ….. Tháng ….. Năm 2019

**LỜI CẢM ƠN**

**TRƯỜNG CAO ĐẲNG LÝ TỰ TRỌNG**

Em xin chân thành cảm ơn Trường Cao Đẳng Lý Tự Trọng và Khoa Công Nghệ Thông Tin đã tạo điều kiện cho em học tập cũng như thực tập trong học kì này. Trong đồ án này chúng em đã tích lũy được nhiều kinh nghiệm và những vài học quý báo cho bản thân. Những kinh nghiệm và bài học trong đồ án sẽ giúp em hoàn thiện hơn trong học tập cũng như trong công việc. Em xin chân thành cảm ơn thầy ***Mai Chiếm Tuấn*** đã luôn luôn tận tình chỉ dẫn, giảng dạy và trang bị cho em những kiến thức cần thiết nhất và giúp đỡ em rất nhiều để hoàn thành tốt đồ án này. Trong quá trình thực tập em đã cố gắng để hoàn thành công việc với tất cả những nổ lực của bản thân, nhưng cũng không thể tránh khỏi những thiếu sót trong quá trình thực tập rất mong nhận được sự thông cảm của thầy. Một lần nữa em xin chân thành cảm ơn thầy ***Mai Chiêm Tuấn .***

TP.Hồ Chí Minh, Ngày …..Tháng ….. Năm 2019

Kí Tên(Nhóm trưởng)

Nguyễn Tô Hiệu

**Nhận xét của giáo viên hướng dẫn** ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... .................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... .................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... .......................................................................................................................

....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... .................................................................................................................... ....................................................................................................................... ....................................................................................................................... .......................................................................................................................

**MỤC LỤC**

[**CHƯƠNG 1 : GIỚI THIỆU VỀ ANDROID** 5](#_Toc12537475)

[**1) Giới thiệu về hệ điều hành Android.** 5](#_Toc12537476)

[**2) Lịch Sử Phát Triển Của Android :** 5](#_Toc12537477)

[**3) Các thành phần cơ bản của một project Android**. 5](#_Toc12537478)

[**Chương II. Phân tích nhu cầu thị trường** 6](#_Toc12537479)

[**1) Tại sao phải Nghe nhạc offline ?** 6](#_Toc12537480)

[**2) Ưu điểm việc dùng phần mềm so với không dùng phần mềm** 7](#_Toc12537481)

[**3) Việc sử dụng trên điệm thoại di động có phù hợp không ?** 7](#_Toc12537482)

[**Chương III. Xây dựng phần mềm** 7](#_Toc12537483)

[**1) Cài đặt Android và môi trường phát triển android như esclipse, SDK** 7](#_Toc12537484)

[**2) Phân tích các chức năng** 10](#_Toc12537485)

[**3) Xây dựng giao diện** 11](#_Toc12537486)

[**4) Thực hiện các Class** 14](#_Toc12537487)

[**Chương IV : Kế Hoạch Thực Hiện Đồ Án** 26](#_Toc12537488)

[**Chương V. Tổng kết** 29](#_Toc12537489)

# **CHƯƠNG 1 : GIỚI THIỆU VỀ ANDROID**

## **1) Giới thiệu về hệ điều hành Android.**

Android là một hệ điều hành điện thoại di động mã nguồn mở, hiện đại, miễn phí do Google phát   
triển dựa trên nền tảng của Linux. Android là nền tảng cho thiết bị di động bao gồm một hệ điều   
hành, midware và một số ứng dụng chủ đạo. Bộ công cụ Android SDK cung cấp các công cụ và bộ   
thư viện các hàm API cần thiết để phát triển ứng dụng cho nền tảng Android sử dụng ngôn ngữ lập   
trình Java.   
Mô hình kiến trúc nền tảng Android

## **2) Lịch Sử Phát Triển Của Android :**

Google mua lại Tổng công ty Android vào ngày 17 tháng 8 năm 2005, biến nó thành một bộ phận trực thuộc Google. Những nhân viên của chủ chốt của Tổng công ty Android, gồm Rubin, Miner và White, vẫn tiếp tục ở lại công ty làm việc sau thương vụ này. Vào thời điểm đó không có nhiều thông tin về Tổng công ty, nhưng nhiều người đồn đoán rằng Google dự tính tham gia thị trường điện thoại di động sau bước đi này. Tại Google, nhóm do Rubin đứng đầu đã phát triển một nền tảng thiết bị di động phát triển trên nền nhân Linux. Google quảng bá nền tảng này cho các nhà sản xuất điện thoại và các nhà mạng với lời hứa sẽ cung cấp một hệ thống uyển chuyển và có khả năng nâng cấp. Google đã liên hệ với hàng loạt hãng phần cứng cũng như đối tác phần mềm, bắn tin cho các nhà mạng rằng họ sẵn sàng hợp tác với các cấp độ khác nhau

## **3) Các thành phần cơ bản của một project Android**.

**+) Activity:** Là một trong những thành phần chính của bất kỳ một ứng dụng Android nào. Activity   
được dùng để hiển thị một màn hình, chỉ một việc mà người dùng có thể thực hiện trong một ứng   
dụng Android, nó tương tác với người sử dụng. 

**+) B.Intent:** Là một cấu trúc dữ liệu mô tả cách thức, đối tượng thực hiện của một Activity, là cầu nối   
giữa các Activity, giúp ta triệu gọi cũng như truyền các dữ liệu cần thiết để thực hiện một Activity từ   
một Activity khác. Ứng dụng Android thường bao gồm nhiều Activity, mỗi Activity hoạt động độc   
lập với nhau và thực hiện những công việc khác nhau.

**+) Bundle:** Ta thường dùng đối tượng Bundle để đóng gói dữ liệu cho việc truyền tải qua các   
Activity khác nhau. Bundle có 2 loại phương thức: putXXX để cho bên Bundle gửi key đi, ứng với   
kiểu dữ liệu nào thì put đúng kiểu đó. Ở bên nhận Bundle thì dùng các phương thức getXXX tương   
ứng để lấy dữ liệu theo key bên gửi.

- XML là ngôn ngữ đánh dấu nhằm đơn giản hóa việc chia sẻ dữ liệu giữa các hệ thống khác nhau, đặc   
biệt là các hệ thống được kết nối Internet. XML cung cấp 1 phương tiện dùng văn bản để mô tả thông   
tin và áp dụng 1 cấu trúc kiểu cây cho thông tin đó.

# **Chương II. Phân tích nhu cầu thị trường**

## **1) Tại sao phải Nghe nhạc offline ?**

- Mạng điện thoại di động xuất hiện tại Việt Nam từ đầu những năm 1990 và theo thời gian số lượng các thuê bao cũng như các nhà cung cấp dịch vụ đi động tại Việt Nam ngày càng tăng. Do nhu cầu trao đôi thông tin ngày càng tăng và nhu cầu sử dụng sản phẩm công nghệ cao nhiều tính năng, cầu hình cao, chất lượng tốt, kiêu đáng mẫu mà đẹp, phong phú nên nhà cung cấp phải luôn luôn cải thiện, nâng cao những sản phẩm của mình. Do đó việc xây dựng các ứng dụng cho điện thoại di động đang là một ngành công nghiệp mới đây tiềm năng và hứa hẹn nhiều sự phát triển vượt bậc của ngành khoa học kĩ thuật.

- Cùng với sự phát triển của thị trường điện thoại di động là sự phát triển mạnh mẽ của xu hướng lập trình phần mềm ứng dụng cho các thiết bị đi động. Phần mềm, ứng dụng cho điện thoại di động hiện nay rất đa dạng và phong phú trên các hệ điều hành di động cũng phát triên mạnh mẽ và đang thay đối từ ngày. Các hệ điều hành J2ME, Android, IOS, Hybrid, Web based Mobile Application đã có rất phát triển trên thị trường truyền thông di động.

- Trong vài năm trở lại đây, hệ điều hành Android ra đời với sự kế thừa những ưu việt của các hệ điều hành ra đời trước và sự kết hợp của nhiều công nghệ tiên tiến nhất hiện nay, đã được nhà phát triển công nghệ rất nỗi tiếng hiện nay là Google.

- Android đã nhanh chóng là đối thủ cạnh tranh mạnh mẽ với các hệ điều hành trước đó và đang là hệ điều hành di động của tương lai và được nhiều người ưa chuộng nhất. Ngày nay, với sự phát triển nhanh chóng của xã hội, nhu cầu sử dụng các dịch vụ mà không có internet là rất cần thiết, vì vậy em đã chọn đề tài **“App nghe nhạc ofline“.**

## **2) Ưu điểm việc dùng phần mềm so với không dùng phần mềm**

**-** Bạncó thể nghe nhạc ở máy mà không cần có wifi hay 3G có thể nghe nhạc không ?

- Máy bạn có thể thể động tìm ra bài hát mà bạn đã chép và hiển thị ở trong thiết bị không ?

=> Đương nhiên là không thể vậy nên ứng dụng của chúng em ra đời nhằm giải quyết những vấn đề trên.

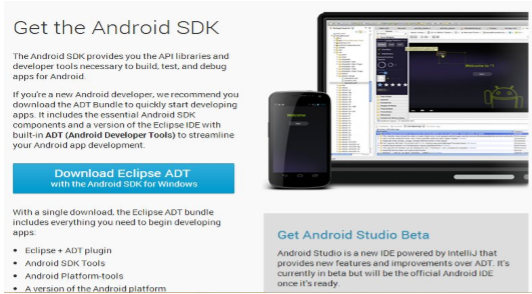
## **3) Việc sử dụng trên điệm thoại di động có phù hợp không ?**

- Ngày nay, tính di động, tiện lợi đang được yêu cầu rất cao khi phát triển 1 phần mềm, phần mềm không những cần được cài đặt trên máy tính, mà còn cả trên thiết bị di động như Iphone, Ipad, Samsung, nokia...Đây là 1 xu thế tất yếu nên việc phát triển trên điện thoại di động là rất

# **Chương III. Xây dựng phần mềm**

## **1) Cài đặt Android và môi trường phát triển android như esclipse, SDK**

Bước 1: Tải bộ cài đặt Android ADT Bundle Đầu tiên chúng ta cần tải công cụ & bộ thư viện hỗ trợ phát triển ứng dụng Android (Android SDK) ở trang <http://developer.android.com/sdk/index.html>



Chú ý:

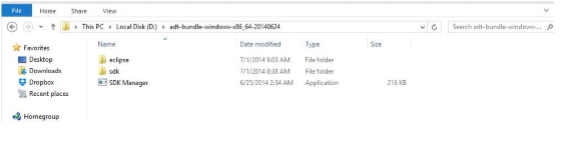
♣ Chúng ta cần chọn phiên bản Android SDK 32-bit hay Android SDK 64-bit phù hợp với hệ thống máy tính dự kiến sử dụng để lập trình Android.

♣ Android mặc định đóng gói môi trường phát triển ứng dụng Eclipse có tích hợp sẵn các công cụ để phát triển ứng dụng Adnroid (Android Developer Tools - ADT plugin), Android SDK Tools, cùng Android Platform-tools vào trong 1 file ADT Bundle duy nhất để người dùng có thể tải về các thành phần cần thiết chỉ 1 lần duy nhất và triển khai môi trường phát triển ứng dụng Android một cách nhanh nhất có thể.

Bước 2: Cài đặt Java để lập trình ứng dụng Android: Máy tính dùng để lập trình Android cần phải được cài đặt Java trước. Nếu không thấy biểu tượng Java trong Control Panel của Windows, chúng ta cần tải Java về và cài đặt từ địa chỉ : http://www.oracle.com/technetwork/java/javase/downloads/index.html

Bước 3: Cài đặt Eclipse & ADT Plugin Giải nén file Android ADT Bundle đã download được ở bước trên vào một thư mục bất kì nào đó trên máy tính. Android đã đính kèm môi trường phát triển Eclipse tích hợp ATD Plugin đã được đóng gói sẵn trong đó.

Mẹo: bạn nên để Android SDK ở ổ đĩa khác ổ đã cài hệ điều hành Windows, vì như thế khi có nhu cầu cài lại máy, thì bản cài đặt Android SDK này vẩn còn, chúng ta không cần phải download lại nữa.



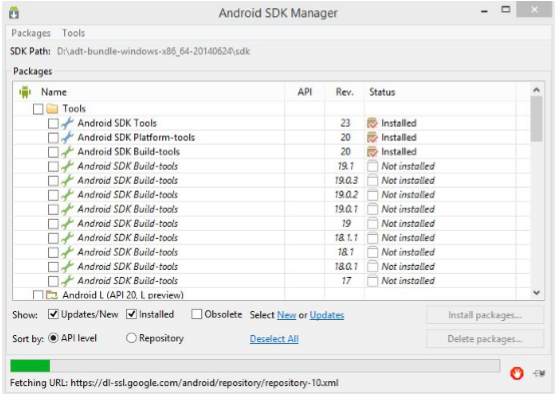
Trong trường hợp chúng ta đã cài đặt Eclipse trên máy tính trước đây rồi, thì chỉ cần cài thêm ADT Plugin (Android Development Tools - ADT Plugin) cho bản Eclipse đó bằng cách:

♣ Khởi động phần mềm Eclipse

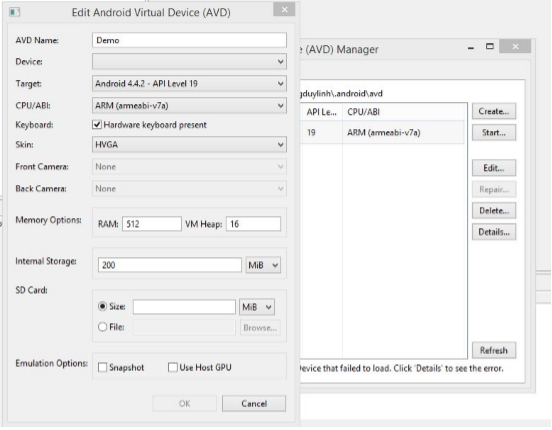
♣ Vào menu Help chọn mục Intall New Software…

♣ Nhập vào đường link sau đây trong mục “Work with” : https://dl-ssl.google.com/android/eclipse để tiến hành cài đặt ADT Plugin cho Eclipse

Bước 4: Cài đặt Android SDK Manager: Bật Eclipse lên, chọn Workpace. Chọn theo các bước hướng dẫn trên hình để cài Android SDK Manager



Bước 5: Tạo máy ảo Android để chạy ứng dụng Android: Máy ảo Android là chương trình giả lập thiết bị Android (Android Virtual Device – AVD) để giúp lập trình viên kiểm tra ứng dụng nhanh ngay trên máy tính trước khi chính thức triển khai trên thiết bị Android thực tế.



Bước 6: Tạo và chạy ứng dụng Android đầu tiên Tạo project Android đầu tiên bằng cách vào menu File => New => Android Application Project

## **2) Phân tích các chức năng**

-Để Xây dựng 1 phần mềm có thể đáp ứng nhu cầu người sử dụng muốn , đảm bảo là hệ thống này cần cho người dùng, tránh làm ra 1 phần mềm mà thị trường không cần thiết.

-Phân tích chức năng giúp xác định các chức năng cần thiết một cách rõ ràng đầy đủ,hạn chế chi phí phân tích lại trong quá trình phát triển phần mềm.

-Phần mềm nghe nhạc offline : là một phần mềm để người dung có thể nghe nhạc mọi lúc trong cả khi không có kết nối internet.

+ Chức năng xem danh sách album : 1 phần mềm nghe nhạc offline không chỉ xem được 1 album mà còn xem được nhiều album có trong thiết bị của bạn để lựa chọn loại nhạc và thể loại mà mình muốn nghe ,nên việc xem danh sách là cần thiết.

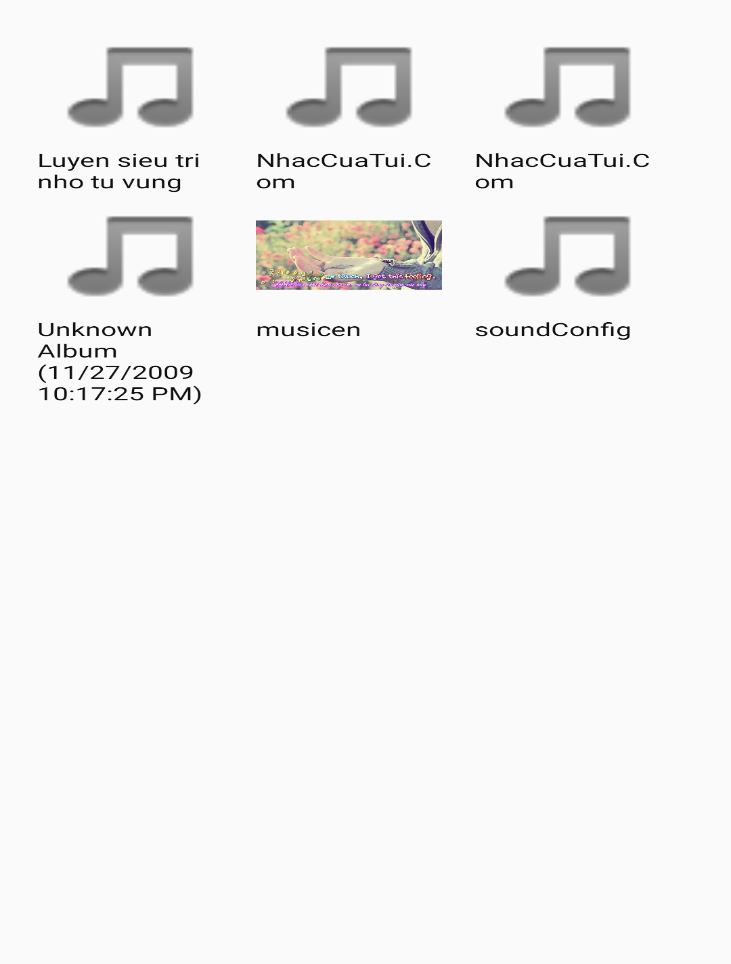
+ Chức năng xem danh sách bài hát trong 1 album : người dùng có thể lựa chọn các bài hát mà mình muốn nghe nên việc xem được danh sách nhạc giúp người dùng có những lựa chọn mà mình thích .

+ Chức năng thông tin của 1 bài hát : người dùng có thể nghe nhạc biết được thông tin của bài hát như tên, hình ảnh ,thời lượng và người dùng có thể tua nhanh hoặc lựa chọn phút nghe - chuyển bài và dừng trong quá trình nghe, tạo nên tiện ích cho người dùng một cách tốt nhất.

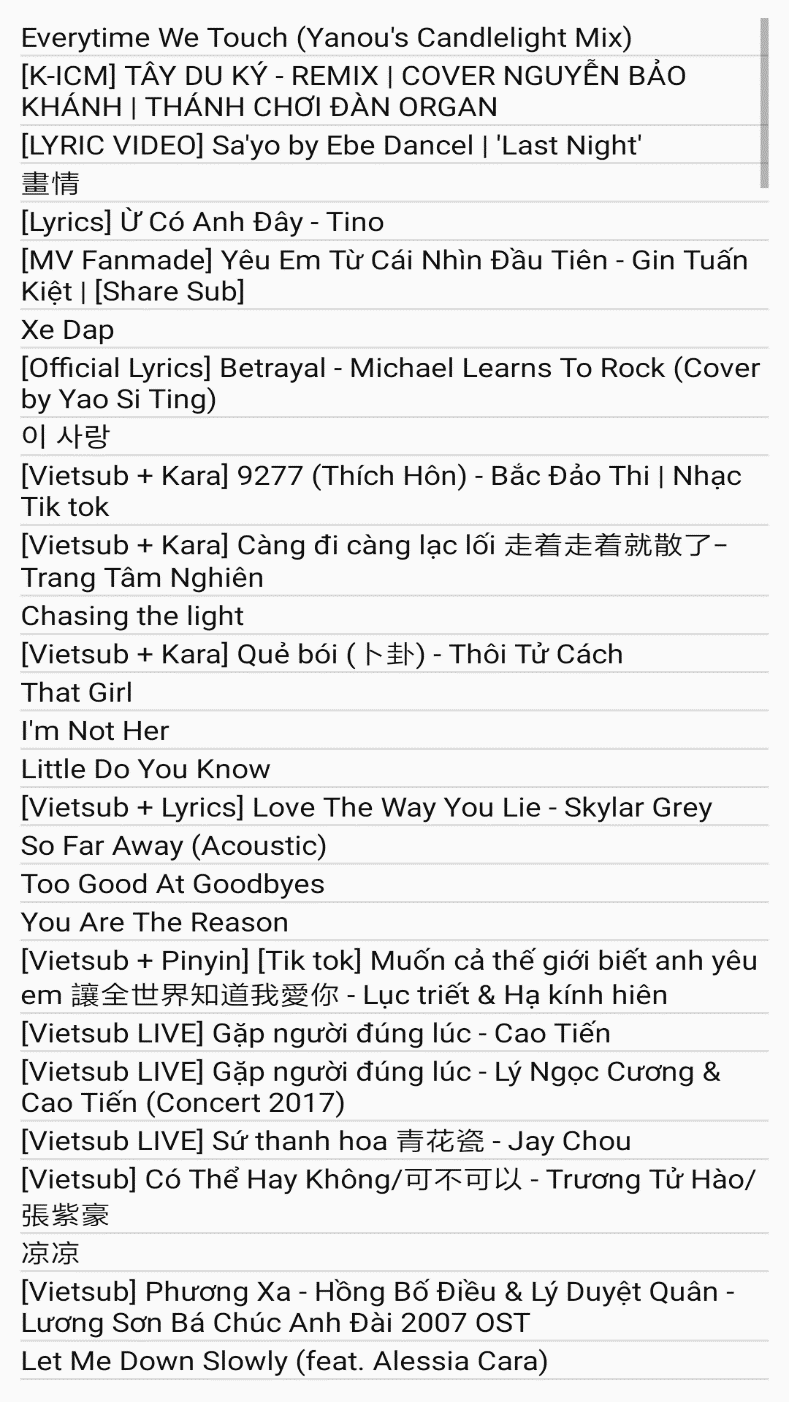
## **3) Xây dựng giao diện**

- Sau khi phân tích các chức năng cần có, tiếp theo là đi xây dựng giao diện, việc xây dựng giao diện giúp định hình phần mềm 1 cách rõ ràng hơn.

- Đầu tiên là giao diện khi người sử dụng mở phần mềm: Gồm các danh sách album – tên album



* Khi người dung nhấn vào Album : Hiện ra danh sách các bài hát trong alnum đó . Thông tin bài hát gồm tên bài hát – ca sĩ thực hiện.

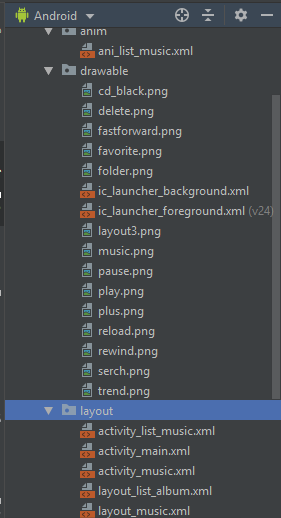
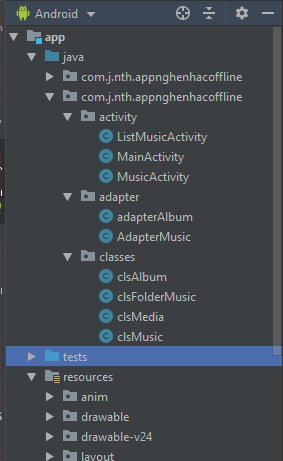


- Sau khi click vào tên bài hát : Gồm tên bài hát , hình , các nút chức năng chuyển bài , dừng bài và lặp lại bài hát ngoài ra còn có thông tin thời gain bài hát , thời lượng của bài hát trong quá trình chạy



## **4) Thực hiện các Class**

**-**Cấu trúc thư mục

****

**-** Trong cấu trúc trên có 3 pakage :

+) com.j.nth. appnghenhacoffline.activity

\***MainActivity**

package com.j.nth.appnghenhacoffline.activity;  
  
import android.Manifest;  
import android.content.ContentResolver;  
import android.content.Intent;  
import android.content.pm.PackageManager;  
import android.database.Cursor;  
import android.media.MediaPlayer;  
import android.net.Uri;  
import android.os.Parcelable;  
import android.provider.MediaStore;  
import android.support.annotation.NonNull;  
import android.support.v4.app.ActivityCompat;  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.View;  
import android.widget.AdapterView;  
import android.widget.ArrayAdapter;  
import android.widget.GridView;  
import android.widget.ListView;  
import android.widget.Toast;  
  
import com.j.nth.appnghenhacoffline.R;  
import com.j.nth.appnghenhacoffline.adapter.AdapterMusic;  
import com.j.nth.appnghenhacoffline.adapter.adapterAlbum;  
import com.j.nth.appnghenhacoffline.classes.clsAlbum;  
import com.j.nth.appnghenhacoffline.classes.clsMedia;  
import com.j.nth.appnghenhacoffline.classes.clsMusic;  
  
import java.io.File;  
import java.io.IOException;  
import java.util.ArrayList;  
  
public class MainActivity extends AppCompatActivity {  
  
 GridView gridviewMusic;  
 ArrayList<clsAlbum> listAlbum;  
 adapterAlbum adapterAlbum;  
 int READ\_EXTERNAL\_STORAGE=1;  
 clsMedia clsmedia;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 connectView();  
 permisstion();  
 listAlbum = new ArrayList();  
 clsmedia = new clsMedia();  
 listAlbum = clsmedia.getAlbum(MainActivity.this);  
 adapterAlbum = new adapterAlbum(this,R.layout.layout\_list\_album,listAlbum);  
 gridviewMusic.setAdapter(adapterAlbum);  
 gridviewMusic.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
 @Override  
 public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
 clsAlbum album =listAlbum.get(position);  
 Intent intent = new Intent(MainActivity.this,ListMusicActivity.class);  
 intent.putExtra("nameAlbum", String.valueOf(album.getNameAlbum()));  
 startActivity(intent);  
 }  
 });  
   
 }  
  
 void permisstion()  
 {  
 ActivityCompat.requestPermissions(MainActivity.this,new String[]{Manifest.permission.READ\_EXTERNAL\_STORAGE},READ\_EXTERNAL\_STORAGE);  
 }  
  
 @Override  
 public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {  
 if(requestCode==READ\_EXTERNAL\_STORAGE && permissions.length > 0 && grantResults[0] == PackageManager.PERMISSION\_GRANTED)  
 {  
  
 }  
 else  
 {  
 Toast.makeText(MainActivity.this,"Bạn chưa cấp quyền",Toast.LENGTH\_SHORT).show();  
 }  
 super.onRequestPermissionsResult(requestCode, permissions, grantResults);  
 }  
  
 private void connectView() {  
 gridviewMusic = findViewById(R.id.listMusic);  
 }  
}

\* ListMusicActivity

package com.j.nth.appnghenhacoffline.activity;  
  
import android.content.Intent;  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.AdapterView;  
import android.widget.ListView;  
  
import com.j.nth.appnghenhacoffline.R;  
import com.j.nth.appnghenhacoffline.adapter.AdapterMusic;  
import com.j.nth.appnghenhacoffline.classes.clsMedia;  
import com.j.nth.appnghenhacoffline.classes.clsMusic;  
  
import java.util.ArrayList;  
  
public class ListMusicActivity extends AppCompatActivity {  
 ListView listviewmusic;  
 AdapterMusic arrayAdapter;  
 public static ArrayList<clsMusic> *clsMusicsArr*;  
 clsMedia clsmedia;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_list\_music*);  
 connectView();  
 clsmedia = new clsMedia();  
 *clsMusicsArr* = new ArrayList<>();  
 final Intent intent = getIntent();  
 String nameAlbum = intent.getStringExtra("nameAlbum");  
 *clsMusicsArr* = clsmedia.getMusic(this,nameAlbum);  
 arrayAdapter = new AdapterMusic(this,R.layout.*layout\_music*,*clsMusicsArr*);  
 listviewmusic.setAdapter(arrayAdapter);  
 listviewmusic.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
 @Override  
 public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
 Intent intent1 = new Intent(ListMusicActivity.this,MusicActivity.class);  
 intent.putExtra("position",position);  
 startActivity(intent1);  
 }  
 });  
 }  
 void connectView()  
 {  
 listviewmusic = findViewById(R.id.*listMusic*);  
 }  
}

\* MusicActivity

package com.j.nth.appnghenhacoffline.activity;  
  
import android.content.Intent;  
import android.media.MediaPlayer;  
import android.net.Uri;  
import android.os.Handler;  
import android.support.constraint.ConstraintLayout;  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.ImageButton;  
import android.widget.ImageView;  
import android.widget.LinearLayout;  
import android.widget.SeekBar;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import com.j.nth.appnghenhacoffline.R;  
import com.j.nth.appnghenhacoffline.classes.clsMusic;  
  
import java.text.SimpleDateFormat;  
  
public class MusicActivity extends AppCompatActivity {  
 LinearLayout linearLayout;  
 MediaPlayer mediaPlayer;  
 TextView txtTenBH,txtTimeSong,txtTimeTotal;  
 SeekBar skB\_Song;  
 ImageButton btnBack,btnPlay,btnNext,btnReload;  
 ImageView songImage;  
 int position = 0;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_music*);  
 connectView();  
 //background  
  
 linearLayout.setBackgroundResource(R.drawable.*layout3*);  
 Intent intent = getIntent();  
 position = intent.getIntExtra("position",0);  
 clsMusic music = ListMusicActivity.*clsMusicsArr*.get(position);  
 if(music.getImage(music.getLocation(),position)!=null)  
 songImage.setImageBitmap(music.getImage(music.getLocation(),position));  
 else  
 songImage.setImageResource(R.drawable.*cd\_black*);  
  
 mediaPlayer = MediaPlayer.*create*(getApplicationContext(),Uri.*parse*(music.getLocation()));  
 mediaPlayer.start();  
 SetTimeToTal();  
 UpdateTimeSong();  
 txtTenBH.setText(music.getTitle());  
  
 btnNext.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 position++;  
 if(position > ListMusicActivity.*clsMusicsArr*.size() -1)  
 {  
 position = 0;  
 }  
 if(mediaPlayer.isPlaying())  
 {  
 mediaPlayer.stop();  
 }  
 clsMusic music = ListMusicActivity.*clsMusicsArr*.get(position);  
 mediaPlayer = MediaPlayer.*create*(getApplicationContext(), Uri.*parse*(music.getLocation()));  
  
 txtTenBH.setText(music.getTitle());  
 mediaPlayer.start();  
 btnPlay.setImageResource(R.drawable.*pause*);  
 SetTimeToTal();  
 UpdateTimeSong();  
  
 }  
 });  
  
 btnBack.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 position--;  
 if(position < 0)  
 {  
 position = ListMusicActivity.*clsMusicsArr*.size() -1;  
 }  
 if(mediaPlayer.isPlaying())  
 {  
 mediaPlayer.stop();  
 }  
 clsMusic music = ListMusicActivity.*clsMusicsArr*.get(position);  
 mediaPlayer = MediaPlayer.*create*(getApplicationContext(), Uri.*parse*(music.getLocation()));  
  
 txtTenBH.setText(music.getTitle());  
 mediaPlayer.start();  
 btnPlay.setImageResource(R.drawable.*pause*);  
 SetTimeToTal();  
 UpdateTimeSong();  
 }  
 });  
  
 btnPlay.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view)  
 {  
  
 if (mediaPlayer.isPlaying())  
 {  
 // nếu đang chạy nhạc -> pause -> đổi hình play  
 mediaPlayer.pause();  
 btnPlay.setImageResource(R.drawable.*play*);  
  
 }  
 else  
 // đang ngừng -> phát -> đổi hình pause  
 {  
 mediaPlayer.start();  
 btnPlay.setImageResource(R.drawable.*pause*);  
 }  
 SetTimeToTal();  
 UpdateTimeSong();  
 }  
 });  
  
 skB\_Song.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {  
 @Override  
 public void onProgressChanged(SeekBar seekBar, int i, boolean b) {  
  
 }  
  
 @Override  
 public void onStartTrackingTouch(SeekBar seekBar) {  
  
 }  
  
 @Override  
 public void onStopTrackingTouch(SeekBar seekBar) {  
 mediaPlayer.seekTo(skB\_Song.getProgress());  
 }  
 });  
 }  
  
 private void SetTimeToTal()  
 {  
 SimpleDateFormat DinhDangGio = new SimpleDateFormat("mm:ss");  
 txtTimeTotal.setText(DinhDangGio.format(mediaPlayer.getDuration()));  
 // gán max của skB\_Song = mediaPlayer.getDuration()  
 skB\_Song.setMax(mediaPlayer.getDuration());  
 }  
  
 private void UpdateTimeSong()  
 {  
 final Handler handler = new Handler();  
 handler.postDelayed(new Runnable() {  
 @Override  
 public void run() {  
 SimpleDateFormat DinhDangTime = new SimpleDateFormat("mm:ss");  
 txtTimeSong.setText(DinhDangTime.format(mediaPlayer.getCurrentPosition()));  
 //update process skB\_Song  
 skB\_Song.setProgress(mediaPlayer.getCurrentPosition());  
  
 // kiểm tra thời gian bài hát nếu kết thúc -> tự động next bài  
 mediaPlayer.setOnCompletionListener(new MediaPlayer.OnCompletionListener() {  
 @Override  
 public void onCompletion(MediaPlayer mediaPlayer2) {  
 position++;  
 if(position > ListMusicActivity.*clsMusicsArr*.size() -1)  
 {  
 position = 0;  
 }  
 if(mediaPlayer.isPlaying())  
 {  
 mediaPlayer.stop();  
 }  
 clsMusic music = ListMusicActivity.*clsMusicsArr*.get(position);  
 mediaPlayer = MediaPlayer.*create*(getApplicationContext(), Uri.*parse*(music.getLocation()));  
  
 txtTenBH.setText(music.getTitle());  
 mediaPlayer.start();  
 btnPlay.setImageResource(R.drawable.*pause*);  
 SetTimeToTal();  
 UpdateTimeSong();  
 }  
 });  
  
 handler.postDelayed(this,500);  
 }  
 },100);  
 }  
  
 void notifyicon()  
 {  
  
 }  
  
 @Override  
 protected void onDestroy() {  
 if(mediaPlayer!=null)  
 if(mediaPlayer.isPlaying()) mediaPlayer.stop();  
 super.onDestroy();  
 }  
  
 void connectView()  
 {  
 linearLayout = findViewById(R.id.*screen*);  
 txtTenBH = findViewById(R.id.*txt\_TenBH*);  
 txtTimeSong = findViewById(R.id.*txt\_TimeSong*);  
 txtTimeTotal = findViewById(R.id.*txt\_TimeTotal*);  
 skB\_Song = findViewById(R.id.*seekbar*);  
 btnBack = findViewById(R.id.*imgB\_Trai*);  
 btnPlay = findViewById(R.id.*img\_Play*);  
 btnNext = findViewById(R.id.*imgB\_Phai*);  
 btnReload = findViewById(R.id.*imgB\_reload*);  
 songImage = findViewById(R.id.*songImage*);  
 }  
}

+) com.j.nth. appnghenhacoffline.adapter

\* adapterAlbum

package com.j.nth.appnghenhacoffline.adapter;  
  
import android.content.Context;  
import android.graphics.BitmapFactory;  
import android.graphics.drawable.Drawable;  
import android.net.Uri;  
import android.util.Log;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.BaseAdapter;  
import android.widget.ImageView;  
import android.widget.TextView;  
  
import com.j.nth.appnghenhacoffline.R;  
import com.j.nth.appnghenhacoffline.classes.clsAlbum;  
import com.j.nth.appnghenhacoffline.classes.clsMusic;  
  
import java.util.ArrayList;  
  
public class adapterAlbum extends BaseAdapter {  
  
 Context context;  
 ArrayList listAlbum;  
 int layout;  
  
 public adapterAlbum(Context context, int layout, ArrayList listAlbum) {  
 this.context = context;  
 this.listAlbum = listAlbum;  
 this.layout = layout;  
 }  
  
 @Override  
 public int getCount() {  
 return listAlbum.size();  
 }  
  
 @Override  
 public Object getItem(int position) {  
 return listAlbum.get(position);  
 }  
  
 @Override  
 public long getItemId(int position) {  
 return 0;  
 }  
  
 class viewHolder  
 {  
 TextView txtNameAlbum;  
 ImageView imageView;  
 }  
 @Override  
 public View getView(int position, View convertView, ViewGroup parent) {  
 viewHolder holder;  
 if(convertView == null) {  
 LayoutInflater inflater = (LayoutInflater) context.getSystemService(Context.*LAYOUT\_INFLATER\_SERVICE*);  
 convertView = inflater.inflate(layout, null);  
 holder = new viewHolder();  
 holder.txtNameAlbum = convertView.findViewById(R.id.*txtNameAlbum*);  
 holder.imageView = convertView.findViewById(R.id.*imageView*);  
 convertView.setTag(holder);  
 }  
 else  
 {  
 holder = (viewHolder) convertView.getTag();  
 }  
 clsAlbum album = (clsAlbum) listAlbum.get(position);  
 holder.txtNameAlbum.setText(album.getNameAlbum());  
 if(album.getImageAlbum()!= null)//Log.d("AAA",album.getImageAlbum());  
 {  
 holder.imageView.setImageURI(Uri.*parse*(album.getImageAlbum()));  
 }  
 return convertView;  
 }  
}

\* AdapterMusic

package com.j.nth.appnghenhacoffline.adapter;  
  
import android.content.Context;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.view.animation.Animation;  
import android.view.animation.AnimationUtils;  
import android.widget.BaseAdapter;  
import android.widget.TextView;  
  
import com.j.nth.appnghenhacoffline.R;  
import com.j.nth.appnghenhacoffline.classes.clsMusic;  
  
import org.w3c.dom.Text;  
  
import java.util.ArrayList;  
  
public class AdapterMusic extends BaseAdapter {  
  
 Context context;  
 ArrayList listMusic;  
 int layout;  
  
 public AdapterMusic(Context context, int layout ,ArrayList listMusic) {  
 this.context = context;  
 this.listMusic = listMusic;  
 this.layout = layout;  
 }  
  
 @Override  
 public int getCount() {  
 return listMusic.size();  
 }  
  
 @Override  
 public Object getItem(int position) {  
 return listMusic.get(position);  
 }  
  
 @Override  
 public long getItemId(int position) {  
 return 0;  
 }  
  
 class viewHolder  
 {  
 TextView txtTitleMusic;  
 }  
  
 @Override  
 public View getView(int position, View convertView, ViewGroup parent) {  
 viewHolder holder;  
 if(convertView == null) {  
 LayoutInflater inflater = (LayoutInflater) context.getSystemService(Context.*LAYOUT\_INFLATER\_SERVICE*);  
 convertView = inflater.inflate(layout, null);  
 holder = new viewHolder();  
 holder.txtTitleMusic = convertView.findViewById(R.id.*txtTitleMusic*);  
 convertView.setTag(holder);  
 }  
 else  
 {  
 holder = (viewHolder) convertView.getTag();  
 }  
 clsMusic music = (clsMusic) listMusic.get(position);  
 holder.txtTitleMusic.setText(music.getTitle());  
 //gan animation  
 Animation animation = AnimationUtils.*loadAnimation*(context,R.anim.*ani\_list\_music*);  
 convertView.setAnimation(animation);  
 return convertView;  
 }  
}

+) com.j.nth. appnghenhacoffline.classes

\* clsAlbum

package com.j.nth.appnghenhacoffline.classes;  
  
public class clsAlbum {  
 String imageAlbum;  
 String nameAlbum;  
  
 public clsAlbum(String imageAlbum, String nameAlbum) {  
 this.imageAlbum = imageAlbum;  
 this.nameAlbum = nameAlbum;  
 }  
  
 public String getImageAlbum() {  
 return imageAlbum;  
 }  
  
 public void setImageAlbum(String imageAlbum) {  
 this.imageAlbum = imageAlbum;  
 }  
  
 public String getNameAlbum() {  
 return nameAlbum;  
 }  
  
 public void setNameAlbum(String nameAlbum) {  
 this.nameAlbum = nameAlbum;  
 }  
  
}

\* clsFolderMusic

package com.j.nth.appnghenhacoffline.classes;  
  
public class clsFolderMusic {  
 String path;  
 String parentFolder;  
  
 public clsFolderMusic(String path, String parentFolder) {  
 this.path = path;  
 this.parentFolder = parentFolder;  
 }  
  
 public String getPath() {  
 return path;  
 }  
  
 public void setPath(String path) {  
 this.path = path;  
 }  
  
 public String getParentFolder() {  
 return parentFolder;  
 }  
  
 public void setParentFolder(String parentFolder) {  
 this.parentFolder = parentFolder;  
 }  
}

\* clsMedia

package com.j.nth.appnghenhacoffline.classes;  
  
import android.content.ContentResolver;  
import android.content.Context;  
import android.database.Cursor;  
import android.graphics.Bitmap;  
import android.graphics.BitmapFactory;  
import android.media.MediaMetadataRetriever;  
import android.provider.MediaStore;  
import android.widget.Toast;  
  
import java.util.ArrayList;  
  
public class clsMedia {  
  
 public ArrayList getAlbum(Context context)  
 {  
 ArrayList<clsAlbum> arrayListAlbum;  
 arrayListAlbum = new ArrayList();  
 String[] projection = new String[] { MediaStore.Audio.Albums.*\_ID*, MediaStore.Audio.Albums.*ALBUM*, MediaStore.Audio.Albums.*ARTIST*, MediaStore.Audio.Albums.*ALBUM\_ART*, MediaStore.Audio.Albums.*NUMBER\_OF\_SONGS* };  
 String selection = null;  
 String[] selectionArgs = null;  
 String sortOrder = MediaStore.Audio.Media.*ALBUM* + " ASC";  
 Cursor cursor = context.getContentResolver().query(MediaStore.Audio.Albums.*EXTERNAL\_CONTENT\_URI*, projection, selection, selectionArgs, sortOrder);  
 if (cursor != null && cursor.moveToFirst())  
 {  
 int name = cursor.getColumnIndex(MediaStore.Audio.Albums.*ALBUM*);  
 int image = cursor.getColumnIndex(MediaStore.Audio.Albums.*ALBUM\_ART*);  
 do{  
 clsAlbum album = new clsAlbum(cursor.getString(image),cursor.getString(name));  
 arrayListAlbum.add(album);  
 }while (cursor.moveToNext());  
 }  
 cursor.close();  
 return arrayListAlbum;  
 }  
  
 public ArrayList<clsMusic> getMusic(Context context,String nameAlbum)  
 {  
 ArrayList<clsMusic> clsMusicArrayList = new ArrayList<>();  
 String[] column = { MediaStore.Audio.Media.*DATA*,  
 MediaStore.Audio.Media.*ALBUM\_ID*, MediaStore.Audio.Media.*TITLE*,  
 MediaStore.Audio.Media.*DISPLAY\_NAME*,  
 MediaStore.Audio.Media.*MIME\_TYPE*, };  
 String orderBy = android.provider.MediaStore.Audio.Media.*TITLE*;  
  
 String where = android.provider.MediaStore.Audio.Media.*ALBUM* + "=?";  
 String whereVal[] = {nameAlbum};  
 Cursor cursor =context.getContentResolver().query(MediaStore.Audio.Media.*EXTERNAL\_CONTENT\_URI*,  
 column,where,whereVal,null);  
 int songTitle = cursor.getColumnIndex(MediaStore.Audio.Media.*TITLE*);  
 int songArtist = cursor.getColumnIndex(MediaStore.Audio.Media.*ARTIST*);  
 int location = cursor.getColumnIndex(MediaStore.Audio.Media.*DATA*);  
 if(cursor != null)  
 {  
 MediaMetadataRetriever metaRetriver = new MediaMetadataRetriever();  
  
 while (cursor.moveToNext())  
 {  
 clsMusic music = new clsMusic(cursor.getString(songTitle),  
 cursor.getString(location));  
  
 clsMusicArrayList.add(music);  
 }  
 }  
 return clsMusicArrayList;  
 }  
}

\* clsMusic

package com.j.nth.appnghenhacoffline.classes;  
  
import android.graphics.Bitmap;  
import android.graphics.BitmapFactory;  
import android.media.MediaMetadataRetriever;  
import android.util.Log;  
  
import com.j.nth.appnghenhacoffline.R;  
  
import java.io.ByteArrayInputStream;  
import java.io.InputStream;  
  
public class clsMusic {  
 String title;  
 String location;  
  
 public String getTitle() {  
 return title;  
 }  
  
 public void setTitle(String title) {  
 this.title = title;  
 }  
  
 public String getLocation() {  
 return location;  
 }  
  
 public void setLocation(String location) {  
 this.location = location;  
 }  
  
 public clsMusic(String title, String location) {  
 this.title = title;  
 this.location = location;  
 }  
  
 public Bitmap getImage(String mPath,int positon)  
 {  
 try {  
 MediaMetadataRetriever metaRetriver = new MediaMetadataRetriever();;  
 metaRetriver.setDataSource(mPath);  
 byte[] art;  
 art = metaRetriver.getEmbeddedPicture();  
 Bitmap songImage = BitmapFactory.*decodeByteArray*(art, positon, art.length);  
 Log.*d*("AAA",art.length+"");  
 return songImage;  
 }catch (Exception e)  
 {  
 return null;  
 }  
  
 }  
}

# **Chương IV : Kế Hoạch Thực Hiện Đồ Án**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Ngày thực hiện** | **Thực hiện** | **Nội dung** |
| 1 | 17/02/19 - 22/02/19 | Đỗ Thị Lan Anh | Khảo sát nhu cầu thực tiễn về ứng dụng trên điện thoại di động |
| Đỗ Thị Tuyết Mai | Khảo sát nhu cầu thực tiễn về ứng dụng trên điện thoại di động |
| Nguyễn Tô Hiệu | Khảo sát nhu cầu thực tiễn về ứng dụng trên điện thoại di động |
| 2 | 24/02/19 - 01/03/19 | Nguyễn Tô Hiệu | Phân tích, lựa chọn công cụ, giải pháp phù hợp với điều kiện và tính khả thi để thực hiện đề tài |
| Đỗ Thị Tuyết Mai | Phân tích, lựa chọn công cụ, giải pháp phù hợp với điều kiện và tính khả thi để thực hiện đề tài |
| Đỗ Thị Lan Anh | Phân tích, lựa chọn công cụ và thư viên để bắt đầu quá trình viết code |
| 3 | 03/03/19 - 08/03/19 | Đỗ Thị Tuyết Mai | Nghiên cứu và viết bản thảo quá trình hình thành và phát triển của hệ điều hành Android, ưu và nhược điểm của Android |
| Đỗ Thị Lan Anh | Nghiên cứu và viết bản thảo các phiên bản của hệ điều hành Android, các thiết bị cài đặt Android |
| Nguyễn Tô Hiệu | Nghiên cứu và viết bản thảo kiến trúc hệ điều hành Android. |
| 4 | 11/03/19 - 16/03/19 | Đỗ Thị Tuyết Mai | Nghiên cứu và viết bản thảo cài đặt Android Studio |
| Đỗ Thị Lan Anh | Nghiên cứu và viết bản thảo cài đặt Java |
| Nguyễn Tô Hiệu | Nghiên cứu và viết bản thảo cài đặt máy ảo Android Emulator |
| 5 | 17/03/19 - 22/03/19 | Nguyễn Tô Hiệu | Nghiên cứu và viết bản thảo giao diện CardView(widget),DrawerLayout(widget),RecyclerView(widget) |
| Đỗ Thị Tuyết Mai | Nghiên cứu và viết bản thảo giao diện Constraint Layout , Linear Layout |
| Đỗ Thị Lan Anh | Nghiên cứu và viết bản thảo giao diện GridLayout,Relative Layout |
| 6 | 24/03/19 - 06/04/19 | Đỗ Thị Tuyết Mai | chỉnh sửa word định kỳ |
| Đỗ Thị Lan Anh | chỉnh sửa word định kỳ |
| Nguyễn Tô Hiệu | chỉnh sửa word định kỳ |
| 7 | 08/04/19 - 12/04/19 | Nguyễn Tô Hiệu | Nghiên cứu và thống nhất các Activity trong app |
| Đỗ Thị Lan Anh | Nghiên cứu về ImageButton & ImageList |
| Đỗ Thị Tuyết Mai | Nghiên cứu và viết bản thảo xử lý các sự kiện trên SeekBar |
| 8 | 14/04/19 - 19/04/19 | Đỗ Thị Tuyết Mai | Thiết kế giao diện Activity\_Main và activity\_music |
| Đỗ Thị Lan Anh | Thiết kế giao diện activity\_list\_music và layout\_music |
| Nguyễn Tô Hiệu | Thiết kế giao diện layout\_list\_album và ani\_list\_music |
| 9 | 20/04/19 - 27/04/19 | Nguyễn Tô Hiệu | Tạo class : clsMedia |
| Đỗ Thị Lan Anh | Tạo class : clsMusic |
| Đỗ Thị Tuyết Mai | Tạo class : clsAlbum & clsFolderMusic |
| 10 | 28/04/19 - 03/05/19 | Nguyễn Tô Hiệu | Viết code xử lý nghiệp vụ “ListMusicActivity & adapterAlbum ” |
| Đỗ Thị Tuyết Mai | Viết code xử lý nghiệp vụ “MusicActivity ” |
| Đỗ Thị Lan Anh | Viết code xử lý nghiệp vụ giao diện ứng dụng “MainAcvity & AdapterMusic ” |
| 11 | 13/05/19 - 19/05/19 | Nguyễn Tô Hiệu | Hoàn thiện báo cáo đồ án :  Chương IV : Kế hoạch thực hiện đồ án |
| Đỗ Thị Lan Anh | Hoàn thiện báo cáo đồ án  Chương II : Phân tích nhu cầu của thị trường  Chương IV: Tổng Kết |
| Đỗ Thị Tuyết Mai | Hoàn thiện báo cáo đồ án :  Chương III: Xây dựng phần mềm  Chương I : Giới thiệu về Android |
| 12 | 20/05/19 - 27/05/19 | Đỗ Thị Lan Anh | Chỉnh sửa báo cáo đồ án  Chương II : Phân tích nhu cầu của thị trường  Chương V: Tổng Kết |
| Nguyễn Tô Hiệu | Chỉnh sửa và Fix 1 số lỗi liên quan đến App  Chương IV : Kế hoạch thực hiện đồ án |
| Đỗ Thị Tuyết Mai | Chỉnh sửa báo cáo đồ án  Chương I : Giới thiệu về Android  Chương III: Xây dựng phần mềm |
| 13 | …../……/…… | Đỗ Thị Lan Anh | Nộp báo cáo + Đĩa CD |
| Đỗ Thị Tuyết Mai | Nộp báo cáo + Đĩa CD |
| Nguyễn Tô Hiệu | Nộp báo cáo + Đĩa CD |

# **Chương V. Tổng kết**

-Trải qua quá trình làm project nghe nhạc offline trên android, giúp em rút ra một số kinh nghiệm

+ Quá trình phân tích rất quan trọng, càng phân tích sâu thì việc hiên thực càng dễ dàng.Trong quá trình phân tích project này chúng em nhận thấy mình còn chưa hiểu rõ được vấn đề yêu cầu thực sự của người dùng, phần mềm xây như thế nào là hợp lí nhất dẫn đến phần mềm chưa hoàn thiện về chức năng,chưa hỗ trợ đắc lực cho người sử dụng.

+ Việc tìm hiểu ngôn ngữ mới cần nhiều thời gian đễ có thể vận dụng nó một cách hiệuquả.

+ Qua Project này, em cũng gặt hái được một số kinh nghiệm, kiến thức phát triển một phần mềm trên android, hiểu về cách thức sử dụng phần mền trên các phần mền khác, cũng như làm sao đễ đáp ứng nhu cầu của người một cách hiệu quả.