|  |
| --- |
| **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **KHOA: CÔNG NGHỆ THÔNG TIN**  **---------------------------------------**  Nhận diện thương hiệu Logo HaUI |
| BÁO CÁO HỌC PHẦN: PHÁT TRIỂN ỨNG DỤNG TRÊN THIẾT BỊ DI ĐỘNG |
|  |
| **QUẢN LÝ GỌI MÓN CỬA HÀNG PHỞ ABC** |
|  |
| GVHD: ThS. Vũ Thị Dương  Nhóm: 2 |
| Sinh viên: Phạm Văn Thư  Vũ Đức Vượng |
| Lớp:20201IT8030002 Khóa: 12 |
| **Hà Nội – Năm 2020** |

# Lời cảm ơn

Trước tiên với tình cảm sâu sắc và chân thành nhất, cho phép nhóm chúng em xin được bày tỏ lòng biết ơn đến cô Vũ Thị Dương đã tạo điều kiện hỗ trợ, giúp đỡ chúng em trong suốt quá trình học tập và nghiên cứu đề tài này.

Trong quá trình làm bài tập lớn, nhóm chúng em còn nhiều hạn chế và bỡ ngỡ nên không tránh khỏi những thiếu sót, chúng em rất mong nhận được những ý kiến đóng góp quý báu của quý Thầy Cô để kiến thức của em trong lĩnh vực này được hoàn thiện hơn đồng thời có điều kiện bổ sung, nâng cao ý thức của mình.

Chúng em xin chân thành cảm cô!

Nhóm 2

# Mục lục

[Lời cảm ơn 2](#_Toc60710175)

[Mục lục 3](#_Toc60710176)

[Chương 1. PHẦN MỞ ĐẦU 4](#_Toc60710177)

[1.1 Ý tưởng và chủ đề nghiên cứu 5](#_Toc60710178)

[1.2 Tổng quan về lập trình trên thiết bị di động 5](#_Toc60710179)

[1.2.1 Giới thiệu về lập trình Android 5](#_Toc60710180)

[1.2.2 Tổng quan về lập trình Android 6](#_Toc60710181)

[1.2.3 Cấu trúc của hệ điều hành Android 7](#_Toc60710182)

[1.2.4 Giới thiệu phần mềm lập trình Android-Android Studio 9](#_Toc60710183)

[1.3 Nội dung thực hiện 9](#_Toc60710184)

[1.3.1 Phạm Văn Thư 9](#_Toc60710185)

[1.3.2 Vũ Đức Vượng 10](#_Toc60710186)

[Chương 2. KẾT QUẢ NGHIÊN CỨU 11](#_Toc60710187)

[2.1 Tìm hiểu yêu cầu bài toán 11](#_Toc60710188)

[2.2 Phân tích thiết kế bài toán 11](#_Toc60710189)

[2.2.1 Phân tích thiết kế bài toán 11](#_Toc60710190)

[2.2.2 Thiết kế hệ thống 13](#_Toc60710191)

[2.3 Thực hiên bài toán 15](#_Toc60710192)

[2.3.1 Phạm Văn Thư-Đăng nhập, đặt món ăn, hiển thị danh sách gọi món 15](#_Toc60710193)

[2.3.2 Vũ Đức Vượng – Sửa, xóa đặt món ăn và giới thiệu về cửa hàng 25](#_Toc60710194)

[2.3.2.1. Sửa đặt món. 25](#_Toc60710195)

[Chương 3. Kiến thức lĩnh hội và bài học kinh nghiệm 30](#_Toc60710196)

[3.1 Kiến thức lĩnh hội 30](#_Toc60710197)

[3.2 Bài học kinh nghiệm và hướng phát triển 30](#_Toc60710198)

[Tài liệu tham khảo 31](#_Toc60710199)

# PHẦN MỞ ĐẦU

## Ý tưởng và chủ đề nghiên cứu

Hiện nay, có rất nhiều các quán ăn đã được mở ra, việc quản lý các món được gọi là một việc khá khó khăn do giờ cao điểm khách sẽ vào rất đông và gọi rất nhiều món, vì thế nhóm chúng em quyết định lựa chọn đề tài “QUẢN LÝ GỌI MÓN CỬA HÀNG PHỞ ABC” để tạo ra một ứng dụng di động giúp cho người quản lý có thể quản lý các món đã được gọi trong cửa hàng.

## Tổng quan về lập trình trên thiết bị di động

### Giới thiệu về lập trình Android

Android là hệ điều hành trên nền tảng di động được phát triển dựa trên hệ điều hành Linux bởi Google. Kể từ khi ra mắt, Android đã ngày càng trở nên phổ biến với lượng người dùng ngày một tăng.

Các ứng dụng trên nền tảng này được phát triển sử dụng các **ngôn ngữ lập trình**như Java hoặc Kotlin. Việc trở thành nền tảng hệ điều hành di động phổ biến nhất giúp đảm bảo nhu cầu đối với các nhà phát triển ứng dụng trên Android. Bên cạnh đó, với đặc điểm là mã nguồn mở, Android sở hữu một hệ sinh thái đa dạng các nhà phát triển ứng dụng.

Tính đến đầu năm 2016, có hơn một tỷ người tích cực sử dụng Android, một hệ điều hành di động giống Unix chạy trên điện thoại, máy tính bảng, TV thông minh, đồng hồ thông minh và hàng loạt thiết bị khác. Mặc dù lập trình Android được Google phát triển và bảo trì, nhưng đây là phần mềm miễn phí và nguồn mở có thể được sử dụng trong lập trình di động và tùy chỉnh bởi bất kỳ ai. Do đó, các thiết bị Android được sản xuất bởi một số nhà sản xuất thiết bị lớn, như Samsung, LG, Sony và HTC.



Một trong những yếu tố chính thúc đẩy thành công của lập trình Android là Google Play, một nền tảng phân phối ứng dụng Android mà người dùng có thể sử dụng để tải xuống và cài đặt ứng dụng trên thiết bị của họ. Với hơn một triệu ứng dụng và hàng tỷ lượt tải xuống, Google Play lớn hơn và có lưu lượng truy cập nhiều hơn so với các nền tảng tương tự cho iOS và Windows 10.

### Tổng quan về lập trình Android

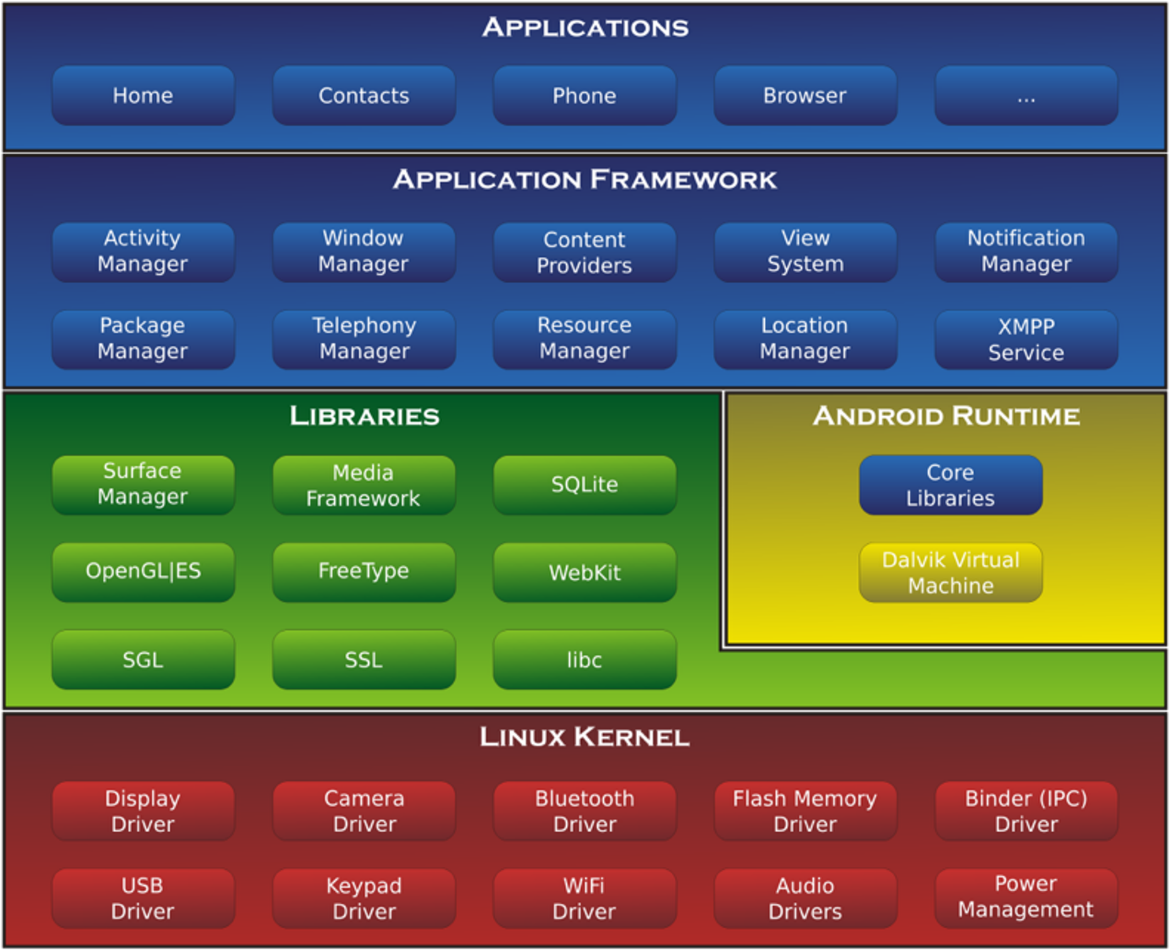
Android có đặc điểm là một nền tảng mã nguồn mở, điều này cho phép các nhà mạng di động, lập trình viên có thể điều chỉnh và thiết kế các ứng dụng một cách linh hoạt và dễ dàng.

Nền tảng Android gồm 5 bộ phận chính, được chứa trong 4 lớp:

* Nhân Linux
* Thư viện
* Android Runtime
* Android Framework
* Application

Đối với việc lập trình Android, lập trình viên sẽ phải làm việc với tầng Android Runtime và Application. Trên Runtime chứa tệp các thư viện cốt lõi cho phép lập trình viên viết các ứng dụng sử dụng ngôn ngữ lập trình như Java. Mọi ứng dụng được viết sẽ được lưu trữ tại tầng Application.

### Cấu trúc của hệ điều hành Android



Hình 1: Cấu trúc của hệ điều hành android

**Linux Kernel:**

Linux Kernel là lớp thấp nhất. Nó cung cấp các chức năng cơ bản như quản lý tiến trình, quản lý bộ nhớ, quản lý thiết bị như: Camera, bàn phím, màn hình,… Ngoài ra, nó còn quản lý mạng, driver của các thiết bị, điều này gỡ bỏ sự khó khăn về giao tiếp với các thiết bị ngoại vi.

**Libraries:**

Phía trên Linux Kernel là tập hợp các bộ thư viện mã nguồn mở WebKit, bộ thư viện nổi tiến libc, cơ sở dữ liệu SQLite hữu ích cho việc lưu trữ và chia sẻ dữ liệu, bộ thư viện thể phát, ghi âm về âm thanh, hoặc video. Thư viện SSL chịu trách nhiệm cho bảo mật Internet.

**Android Runtime:**

Đây là thành phần thứ 3 trong cấu trúc, thuộc về lớp 2 tính từ dưới lên. Phần này cung cấp một thành phần quan trọng gọi là Dalvik Virtual Machine là một máy ảo Java đặt biệt, được thiết kế tối ưu cho Android.

Máy ảo Dalvik sử dụng các tính năng cốt lõi của Linux như quản lý bộ nhớ, đa luồng, mà thực chất là bên trong ngôn ngữ Java. Máy ảo Dalvik cho phép tất cả các ứng dụng Android chạy trong tiến trình riêng của nó.

Android Runtime cũng cung cấp bộ thư viện cốt lõi, cho phép các lập trình viên Android sử dụng để viết các ứng dụng Android.

**Application Framework:**

Lớp Application Framework cung cấp nhiều dịch vụ cấp cao hơn cho các ứng dụng trong các class Java. Các lập trình viên cũng được phép sử dụng các dịch vụ này trong các ứng dụng của họ.

**Applications:**

Bạn sẽ thấy tất cả các ứng dụng Android ở lớp trên cùng. Ứng dụng bạn viết sẽ được cài đặt vào lớp này

Trong quá trình làm việc, chúng ta sẽ gần như chỉ làm việc với tầng xanh lam (Applications và Application Framework) và xanh lá (Libraries). Chương trình Android được viết bằng ngôn ngữ Java và được máy ảo DVM / ART trong mỗi thiết bị Android biên dịch ra mã máy.

### Giới thiệu phần mềm lập trình Android-Android Studio



Tháng 5 năm 2013, Google công bố Android Studio, một môi trường phát triển ứng dụng tích hợp (IDE) dành riêng cho Android, mã nguồn mở, dựa trên IDE Java IntelliJ của hãng JetBrains (đối thủ với Eclipse và Netbeans, vốn khá quen thuộc với dân lập trình Java).

Android Studio chạy trên Windows, Mac và Linux, nhằm thay thế cho Eclipse Android Development Tool (ADT) vốn được sử dụng làm IDE chính trong các năm trước đó.

## Nội dung thực hiện

### Phạm Văn Thư

* Chức năng order món ăn: Khi khách hàng muốn đặt món ăn, người quản lý sẽ ấn vào nút Đặt món trên màn hình, màn hình đặt món sẽ hiển thị các thông tin về món ăn, bàn như số bàn, tên món ăn, số lượng bát muốn ăn, sở thích của khách hàng, có khuyến mãi hay không.
* Chức năng hiển thị các danh sách món ăn lên màn hình: Sau khi đăng nhập thành công, hệ thống sẽ hiển thị danh sách các món ăn đã được đặt lên danh sách, thông tin về món ăn bao gồm: số bàn, tên món ăn, số lượng, sở thích, có khuyến mãi hay không.
* Chức năng đăng nhập: khi bắt đầu khởi động chương trình, người sử dụng sẽ phải đăng nhập vào hệ thống bằng tài khoản và mật khẩu để có thể sử dụng tiếp chương trình.

### Vũ Đức Vượng

* Chức năng sửa thông tin các món ăn đã order: Khi người sử dụng muốn thay đổi thông tin của món ăn, người sử dụng cần ấn và giữ một lúc trên thanh thông tin món ăn, một menu sẽ hiện ra bao gồm 2 nút là sửa, người sử dụng ấn vào nút sửa, sẽ hiển thị ra màn hình thông tin các món ăn cần sửa, sau khi sửa thành công và ấn vào nút cập nhật, sẽ hiển thị lại danh sách các món ăn
* Chức năng xóa các món ăn sau khi đã được phục vụ: Nếu khách hàng không muốn ăn nữa hoặc món ăn đã được phục vụ, người sử dụng có thể ấn và giữ lâu trong thanh thông tin món ăn, một menu sẽ hiện ra, khi người dùng ấn vào nút xóa sẽ hiển thị thông báo có muốn xóa hay không, nếu có thì sẽ xóa thanh thông tin về món ăn, nếu không thì sẽ trở về trạng thái cũ.
* Chức năng xem thông tin cửa hàng: Người dùng có thể ấn vào nút thông tin cửa hàng trên menu, người sử dụng có thể xem được thông tin về cửa hàng.

# KẾT QUẢ NGHIÊN CỨU

**Đề tài:** Quản lý gọi món cửa hàng phở ABC

Với đề tài này, nhóm chúng em đã vận dụng các kiến thức trên lớp và các kiến thức tự tìm hiểu để có thể thực hiện tốt nhất. Chúng em thực hiện một số modul sau:

* Chức năng đăng nhập.
* Chức năng hiển thị các món đã gọi.
* Chức năng gọi món.
* Chức năng sửa các món đã gọi.
* Chức năng xóa món ăn sau khi được phục vụ
* Chức năng xem thông tin cửa hàng.

## Tìm hiểu yêu cầu bài toán

## Phân tích thiết kế bài toán

### Phân tích thiết kế bài toán

#### Biểu đồ use case

****

Hình 2 Biểu đồ use case

#### Phân tích yêu cầu người dùng

* **Mô tả yêu cầu người dùng**

Ứng dụng di động “Quản lý gọi món quán phở ABC” với các chức năng đăng nhập; chức năng xem danh sách gọi món, gọi món, sửa và xóa thông tin món, xem thông tin cửa hàng.

* **Yêu cầu chức năng cần có trong hệ thống:**

**Đăng nhập:**

Chức năng đăng nhập: khi người dùng khởi động chương trình, hệ thống sẽ hiển thị màn hình đăng nhập, người dùng cần điền tài khoản và mật khẩu đã được đăng kí để đăng nhập.

**Chức năng hiển thị danh sách món ăn**

Sau khi người dùng đăng nhập, hệ thống sẽ hiển thị tất cả các món ăn đã được đặt trên danh sách.

**Chức năng bảo trì món ăn:**

* **Thêm món ăn**: người dùng ấn vào nút đặt món, hệ thống sẽ hiển thị các thông tin món ăn như: số bàn, tên món ăn, số lượng, sở thích, có khuyến mại hay không, nếu người dùng ấn đồng ý thì hệ thống sẽ thêm món ăn vào trong cơ sở dữ liệu và hiển thị màn hình danh sách món ăn.
* **Sửa món ăn**: Nếu người sử dụng muốn sửa thông tin món ăn trên danh sách đặt món, người dùng cần ấn và giữ một lúc, menu sẽ hiện ra gồm 2 nút sửa và xóa, người dùng ấn vào nút sửa, hệ thống sẽ hiển thị màn hình thông tin của món ăn cho phép sửa, sau khi sửa thông tin, người dùng ấn vào nút cập nhật để cập nhật thông tin món ăn và hiển thị lại danh sách món ăn.
* **Xóa món ăn**: Nếu người sử dụng muốn xóa thông tin món ăn trên danh sách đặt món, người dùng cần ấn và giữ một lúc, menu sẽ hiện ra gồm 2 nút sửa và xóa, người dùng ấn vào nút xóa, hệ thống sẽ hiển thị thông báo xóa món ăn.

**Chức năng hiển thị thông tin cửa hàng**

* **Yêu cầu phi chức năng cần có trong hệ thống:**

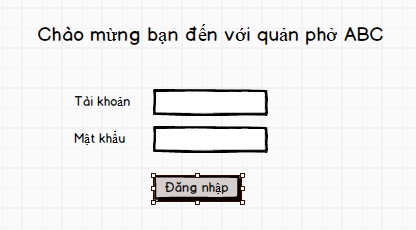
Người dùng cần phải đăng nhập đúng tài khoản và mật khẩu nếu muốn truy cập vào ứng dụng này.

Người dùng phải thêm đầy đủ dữ liệu cho các trường nếu muốn đặt món ăn mới.

### Thiết kế hệ thống

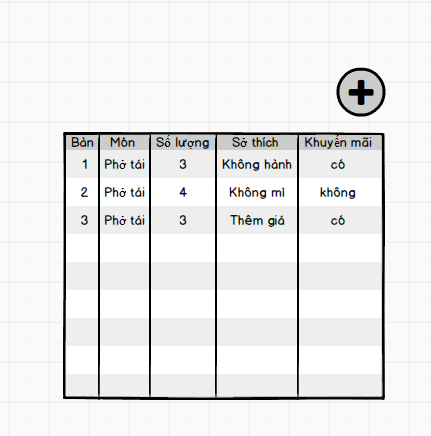
**Mô tả các màn hình**

* **Màn hình đăng nhập**



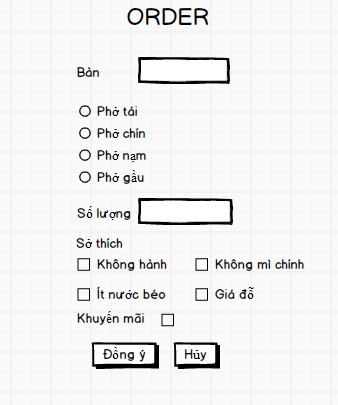
Hình 3 Màn hình đăng nhập

* + - **Màn hình hiển thị danh sách gọi món**



Hình 4 Màn hình hiển thị danh sách sản phẩm

* + - **Màn hình đặt món**



Hình 5 Màn hình đặt món

## Thực hiên bài toán

### Phạm Văn Thư-Đăng nhập, đặt món ăn, hiển thị danh sách gọi món

#### Đăng nhập

Tạo một lớp User

|  |
| --- |
| import androidx.room.ColumnInfo; import androidx.room.Entity; import androidx.room.PrimaryKey;  @Entity public class User {  @PrimaryKey(autoGenerate = true)  private int id;  @ColumnInfo  private String userName;  @ColumnInfo  private String passWord;   public User(String userName, String passWord) {  this.userName = userName;  this.passWord = passWord;  }  public User(){}   public int getId() {  return id;  }   public void setId(int id) {  this.id = id;  }   public String getUserName() {  return userName;  }   public void setUserName(String userName) {  this.userName = userName;  }   public String getPassWord() {  return passWord;  }   public void setPassWord(String passWord) {  this.passWord = passWord;  } } |

|  |
| --- |
| import androidx.room.Dao; import androidx.room.Insert; import androidx.room.Query;  import com.phamvanthu.nhom2\_thu2.Pho; import com.phamvanthu.nhom2\_thu2.User;  import java.util.List;  @Dao public interface UserDao {  @Query("SELECT \* FROM User WHERE userName=:us AND passWord=:ps")  List<User> getAllUs(String us,String ps);   @Query("SELECT \* FROM User")  List<User> getAll();   @Insert  void insert(User... users);  } |

|  |
| --- |
| import android.content.Context;  import androidx.room.Database; import androidx.room.Room; import androidx.room.RoomDatabase;  import com.phamvanthu.nhom2\_thu2.Pho; import com.phamvanthu.nhom2\_thu2.User;  @Database(entities = {Pho.class, User.class}, version = 2) public abstract class PhoDatabase extends RoomDatabase {  private static final String *DATABASE\_NAME*="quanpho.db";  private static PhoDatabase *instance*;   public static synchronized PhoDatabase getInstance(Context context){  if(*instance*==null){  *instance*= Room.*databaseBuilder*(context.getApplicationContext(), PhoDatabase.class, *DATABASE\_NAME*)  .allowMainThreadQueries()  .build();  }  return *instance*;  }  public abstract PhoDao phoDao();  public abstract UserDao userDao(); } |

|  |
| --- |
| import androidx.appcompat.app.AppCompatActivity; import androidx.databinding.DataBindingUtil;  import android.content.Intent; import android.os.Bundle; import android.view.View; import android.widget.Toast;  import com.phamvanthu.nhom2\_thu2.database.PhoDatabase; import com.phamvanthu.nhom2\_thu2.databinding.ActivityLoginBinding;  import java.util.ArrayList; import java.util.List;  public class LoginActivity extends AppCompatActivity {   private String userName="";  private String passWord="";  List<User> userList=new ArrayList<>();  private ActivityLoginBinding binding;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  binding = DataBindingUtil.*setContentView*(this, R.layout.*activity\_login*);  addUser();  binding.btnlogin.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  userName=binding.edtUsername.getText().toString();  passWord=binding.edtPassword.getText().toString();  for (User user: userList){  if(user.getUserName().equals(userName)&&user.getPassWord().equals(passWord)){  Intent intent=new Intent(LoginActivity.this, MainActivity.class);  startActivity(intent);  return;  }  }  Toast.*makeText*(LoginActivity.this, "Đăng nhập không thành công! ", Toast.*LENGTH\_LONG*).show();  }  });   }  private void addUser(){  userList.addAll(PhoDatabase.*getInstance*(this).userDao().getAll());  List<User> orderList = new ArrayList<>(PhoDatabase.*getInstance*(this).userDao().getAll());  if (orderList.size() == 0) {  initdataUser();  }  }   private void initdataUser() {  User user1=new User("Admin1","12345");  User user2=new User("Admin2","54321");  User user3=new User("Admin3","54321");  PhoDatabase.*getInstance*(this).userDao().insert(user1);  PhoDatabase.*getInstance*(this).userDao().insert(user2);  PhoDatabase.*getInstance*(this).userDao().insert(user3);  }  } |

#### Đặt món ăn

|  |
| --- |
| import androidx.room.ColumnInfo; import androidx.room.Entity; import androidx.room.PrimaryKey;  @Entity public class Pho {   @PrimaryKey(autoGenerate = true)  private int id;   @ColumnInfo  private int ban;   @ColumnInfo  private String mon;   @ColumnInfo  private int soluong;   @ColumnInfo  private String sothich;   @ColumnInfo  private String km;   public Pho() {  }   public Pho(int ban, String mon, int soluong, String sothich, String km) {  this.ban = ban;  this.mon = mon;  this.soluong = soluong;  this.sothich = sothich;  this.km = km;  }   public int getId() {  return id;  }   public void setId(int id) {  this.id = id;  }   public int getBan() {  return ban;  }   public void setBan(int ban) {  this.ban = ban;  }   public String getMon() {  return mon;  }   public void setMon(String mon) {  this.mon = mon;  }   public int getSoluong() {  return soluong;  }   public void setSoluong(int soluong) {  this.soluong = soluong;  }   public String getSothich() {  return sothich;  }   public void setSothich(String sothich) {  this.sothich = sothich;  }   public String getKm() {  return km;  }   public void setKm(String km) {  this.km = km;  }   } |

|  |
| --- |
| import androidx.room.Database; import androidx.room.Room; import androidx.room.RoomDatabase;  import com.phamvanthu.nhom2\_thu2.Pho; import com.phamvanthu.nhom2\_thu2.User;  @Database(entities = {Pho.class, User.class}, version = 2) public abstract class PhoDatabase extends RoomDatabase {  private static final String *DATABASE\_NAME*="quanpho.db";  private static PhoDatabase *instance*;   public static synchronized PhoDatabase getInstance(Context context){  if(*instance*==null){  *instance*= Room.*databaseBuilder*(context.getApplicationContext(), PhoDatabase.class, *DATABASE\_NAME*)  .allowMainThreadQueries()  .build();  }  return *instance*;  }  public abstract PhoDao phoDao();  public abstract UserDao userDao(); } |

|  |
| --- |
| import androidx.room.Dao; import androidx.room.Delete; import androidx.room.Insert; import androidx.room.Query; import androidx.room.Update;  import com.phamvanthu.nhom2\_thu2.Pho;  import java.util.List;  @Dao public interface PhoDao {  @Query("SELECT \* FROM Pho")  List<Pho> getAll();   @Query("DELETE FROM Pho")  void deleteAll();   @Insert  void insert(Pho... phos);   @Delete  void delete(Pho... phos);   @Update  void update(Pho phos); } |

|  |
| --- |
| import android.app.Activity; import android.app.AlertDialog; import android.app.Dialog; import android.content.DialogInterface; import android.content.Intent; import android.os.Bundle;  import android.view.View;  import androidx.appcompat.app.AppCompatActivity; import androidx.databinding.DataBindingUtil;  import com.phamvanthu.nhom2\_thu2.databinding.AddActivityBinding;   public class AddOder extends AppCompatActivity implements View.OnClickListener {   private AddActivityBinding binding;   private int banAn =0;  private String monAn = "";  private int soLuong =0 ;  private String soThich = "";  private String kM = "Không";    @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  binding = DataBindingUtil.*setContentView*(this, R.layout.*add\_activity*);  binding.btnOk.setOnClickListener(this);  binding.rbTai.setChecked(true);   binding.btnCancel.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Dialog dialog = new AlertDialog.Builder(AddOder.this)  .setTitle("Thông báo")  .setMessage("Bạn có muốn đóng không?")  .setNegativeButton("Hủy", new DialogInterface.OnClickListener() {  @Override  public void onClick(DialogInterface dialog, int which) {   }  }).setPositiveButton("Đồng ý", new DialogInterface.OnClickListener() {  @Override  public void onClick(DialogInterface dialog, int which) {  finish();  }  }).show();  }  });  }    @Override  public void onClick(View v) {   banAn = Integer.*parseInt*(binding.edtBan.getText().toString());  soLuong = Integer.*parseInt*(binding.edtSoluong.getText().toString());   if (binding.rbTai.isChecked()) {  monAn = "Phở tái";  }  if (binding.rbChin.isChecked()) {  monAn = "Phở chín";  }  if (binding.rbNam.isChecked()) {  monAn = "Phở nậm";  }  if (binding.rbGau.isChecked()) {  monAn = "Phở gầu";  }   if (binding.chkKhonghanh.isChecked()) {  soThich = "Không hành";  } else if (binding.chkKhongmychinh.isChecked()) {  soThich += "Không mì";  } else if (binding.chkItnuoc.isChecked()) {  soThich += "ít nước";  } else if (binding.chkGiado.isChecked()) {  soThich += "Thêm giá";  }  if (binding.ckKm.isChecked()) {  kM = "Có";  }  if (banAn != 0 && monAn != "" && soLuong != 0 && soThich != "") {  Intent output = getIntent();  output.putExtra("banAn", banAn);  output.putExtra("monAn", monAn);  output.putExtra("soLuong", soLuong);  output.putExtra("soThich", soThich);  output.putExtra("kM", kM);  setResult(Activity.*RESULT\_OK*, output);  finish();  }    } } |

|  |
| --- |
| import android.content.Intent; import android.os.Bundle; import android.view.View; import android.widget.Toast;  import androidx.annotation.Nullable; import androidx.appcompat.app.AppCompatActivity; import androidx.databinding.DataBindingUtil;  import com.phamvanthu.nhom2\_thu2.database.PhoDatabase; import com.phamvanthu.nhom2\_thu2.databinding.ActivityMainBinding;  import java.util.ArrayList; import java.util.List;  public class MainActivity extends AppCompatActivity implements PhoAdapter.SpinnerListener {   private static final int *SECOND\_ACTIVITY\_REQUEST\_CODE* = 0;  private ActivityMainBinding binding;  private PhoAdapter adapter;  private List<Pho> oderList = new ArrayList<>();   private int banAn;  private String monAn = "";  private int soLuong;  private String soThich = "";  private String kM = "";      @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  binding = DataBindingUtil.*setContentView*(this, R.layout.*activity\_main*);  binding.btnadd.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Intent intent=new Intent(MainActivity.this, AddOder.class);  startActivityForResult(intent, *SECOND\_ACTIVITY\_REQUEST\_CODE*);  }  });   initDataIntoDatabase();   initData();  }   private void initData() {  adapter = new PhoAdapter(getLayoutInflater());  adapter.setData(oderList);  binding.rcOder.setAdapter(adapter);  adapter.setListener(this);  }    private void initDataIntoDatabase() {  oderList.addAll(PhoDatabase.*getInstance*(this).phoDao().getAll());  List<Pho> orderList = new ArrayList<>(PhoDatabase.*getInstance*(this).phoDao().getAll());  if (orderList.size() == 0) {  addDataIntoDatabase();  }  }   private void addDataIntoDatabase() {  Pho pho1=new Pho(1, "Phở tái" ,4, "Không hành", "Có");  Pho pho2=new Pho(2, "Phở chín",3, "Giá đỗ", "Có");  Pho pho3=new Pho(3, "Phở nạm",4, "Ít nước béo", "Không");   *// chèn dữ liệu vào database* PhoDatabase.*getInstance*(this).phoDao().insert(pho1);  PhoDatabase.*getInstance*(this).phoDao().insert(pho2);  PhoDatabase.*getInstance*(this).phoDao().insert(pho3);   oderList.addAll(PhoDatabase.*getInstance*(this).phoDao().getAll());  }    @Override  public void onSpinnerClicked(int position) {   }  @Override  public void onSpinnerLongClicked(int position) {   }   @Override  protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {  super.onActivityResult(requestCode, resultCode, data);   if(requestCode==*SECOND\_ACTIVITY\_REQUEST\_CODE* && resultCode==*RESULT\_OK*){  banAn=data.getIntExtra("banAn", 0);  monAn=data.getStringExtra("monAn");  soLuong=data.getIntExtra("soLuong", 0);  soThich=data.getStringExtra("soThich");  kM=data.getStringExtra("kM");   Pho pho=new Pho(banAn, monAn, soLuong, soThich, kM);  PhoDatabase.*getInstance*(this).phoDao().insert(pho);   oderList.clear();  oderList.addAll( PhoDatabase.*getInstance*(this).phoDao().getAll());  adapter.notifyDataSetChanged();   Toast.*makeText*(this, "Order thành công",Toast.*LENGTH\_SHORT*).show();  }  } } |

### Vũ Đức Vượng – Sửa, xóa đặt món ăn và giới thiệu về cửa hàng

# Sửa đặt món.

Hàm xử lý khi ấn giữ một item trong RecyclerView

|  |
| --- |
| @Override **public void** onSpinnerLongClicked(**final int** position) {  **a**=position;  androidx.appcompat.widget.PopupMenu popupMenu = **new** androidx.appcompat.widget.PopupMenu(MainActivity.**this**, **binding**.**rcOder**.getChildAt(position));  popupMenu.getMenuInflater().inflate(R.menu.***item\_menu***, popupMenu.getMenu());  popupMenu.setOnMenuItemClickListener(**new** PopupMenu.OnMenuItemClickListener() {  @SuppressLint(**"NonConstantResourceId"**)  @Override  **public boolean** onMenuItemClick(MenuItem menuItem) {  **switch** (menuItem.getItemId()) {  **case** R.id.***menu\_sua***:  doEdit();  **break**;  **case** R.id.***menu\_xoa***:  doDelete();  **break**;  }  **return false**;  }  });  popupMenu.show(); } |

Hàm xử lý khi kích vào Sửa

|  |
| --- |
| **private void** doEdit() {  Intent intent = **new** Intent(MainActivity.**this**, Edit\_Order.**class**);  intent.putExtra(**"Ban"**,**oderList**.get(position).getBan());  intent.putExtra(**"Mon"**,**oderList**.get(position).getMon());  intent.putExtra(**"SoLuong"**,**oderList**.get(position).getSoluong());  intent.putExtra(**"SoThich"**,**oderList**.get(position).getSothich());  intent.putExtra(**"KhuyenMai"**,**oderList**.get(position).getKm());  startActivityForResult(intent, 2); } |

Hàm xử lý Sửa

|  |
| --- |
| **package** com.example.nhom2\_android\_project;  **import** androidx.appcompat.app.AppCompatActivity; **import** android.app.Activity; **import** android.content.Intent; **import** android.os.Bundle; **import** android.view.View; **import** android.widget.Toast; **import** androidx.databinding.DataBindingUtil; **import** java.util.\*; **import** com.example.nhom2\_android\_project.databinding.EditOrderBinding;  **public class** Edit\_Order **extends** AppCompatActivity {  **private int banAn** = 0;  **private** String **monAn** = **""**;  **private int soLuong** = 0;  **private** String **soThich** = **""**;  **private** String **kM** = **"Không"**;   **private** PhoAdapter **adapter**;  **private** List<Pho> **oderList** = **new** ArrayList<>();   **private** EditOrderBinding **binding**;   @Override  **protected void** onCreate(Bundle savedInstanceState) {   String mon = getIntent().getStringExtra(**"Mon"**);  String sothich = getIntent().getStringExtra(**"SoThich"**);  String km = getIntent().getStringExtra(**"KhuyenMai"**);  **super**.onCreate(savedInstanceState);  **binding** = DataBindingUtil.*setContentView*(**this**, R.layout.***edit\_\_order***);  **binding**.**edtBan**.setText(getIntent().getIntExtra(**"Ban"**, 0)+**""**);  **if**(mon.equals(**"Phở tái"**)==**true**){  **binding**.**rbTai**.setChecked(**true**);  }  **if**(mon.equals(**"Phở chín"**)==**true**){  **binding**.**rbChin**.setChecked(**true**);  }  **if**(mon.equals(**"Phở nạm"**)==**true**){  **binding**.**rbNam**.setChecked(**true**);  }  **if**(mon.equals(**"Phở gầu"**)==**true**){  **binding**.**rbGau**.setChecked(**true**);  } *// binding.edtMon.setText(getIntent().getStringExtra("Mon"));* **binding**.**edtSoluong**.setText(getIntent().getIntExtra(**"SoLuong"**, 0)+**""**); *// binding.edtSothich.setText(getIntent().getStringExtra("SoThich"));* **if**(sothich.equals(**"Không hành"**)==**true**){  **binding**.**chkKhonghanh**.setChecked(**true**);  }  **if**(sothich.equals(**"Không mì"**)==**true**){  **binding**.**chkKhongmychinh**.setChecked(**true**);  }  **if**(sothich.equals(**"Thêm giá"**)==**true**){  **binding**.**chkGiado**.setChecked(**true**);  }  **if**(sothich.equals(**"Ít nước"**)==**true**){  **binding**.**chkItnuoc**.setChecked(**true**);  } *// binding.edtKm.setText(getIntent().getStringExtra("KhuyenMai"));* **if**(km.equals(**"Có"**)==**true**){  **binding**.**ckKm**.setChecked(**true**);  } **else** {  **binding**.**ckKm**.setChecked(**false**);  }  **binding**.**btnOk**.setOnClickListener(**new** View.OnClickListener() {  @Override  **public void** onClick(View view) {  **banAn**=Integer.*parseInt*(**binding**.**edtBan**.getText().toString());  **soLuong** = Integer.*parseInt*(**binding**.**edtSoluong**.getText().toString());  **if**(**binding**.**rbTai**.isChecked()){  **monAn**=**"Phở tái"**;  }**if**(**binding**.**rbChin**.isChecked()){  **monAn**=**"Phở chín"**;  }**if**(**binding**.**rbNam**.isChecked()){  **monAn**=**"Phở nậm"**;  }**if**(**binding**.**rbGau**.isChecked()){  **monAn**=**"Phở gầu"**;  }   **if**(**binding**.**chkKhonghanh**.isChecked()){  **soThich**=**"Không hành"**;  }**else if**(**binding**.**chkKhongmychinh**.isChecked()){  **soThich**+=**"Không mì"**;  }**else if**(**binding**.**chkItnuoc**.isChecked()){  **soThich**+=**"ít nước"**;  }**else if**(**binding**.**chkGiado**.isChecked()){  **soThich**+=**"Thêm giá"**;  }  **if** (**binding**.**ckKm**.isChecked()){  **kM**=**"Có"**;  }  **if** (**banAn** != 0 && **monAn** != **null** && **soLuong** != 0 && **soThich** != **null** && **kM** != **null**) {  Intent output = **new** Intent();  output.putExtra(**"newBan"**, **banAn**);  output.putExtra(**"newMon"**, **monAn**);  output.putExtra(**"newSl"**, **soLuong**);  output.putExtra(**"newSt"**, **soThich**);  output.putExtra(**"newKm"**, **kM**);  setResult(Activity.***RESULT\_OK***, output);  finish();  }**else** {  Toast.*makeText*(Edit\_Order.**this**, **"Mời bạn nhập đủ dữ liệu"**, Toast.***LENGTH\_LONG***).show();  }  }  });  } } |

Hàm trả về kết quả Sửa

|  |
| --- |
| @Override **protected void** onActivityResult(**int** requestCode, **int** resultCode, @Nullable Intent data) {  **super**.onActivityResult(requestCode, resultCode, data);  **if** (requestCode == 2 && resultCode == ***RESULT\_OK*** && data != **null**) {  **banAn**=data.getIntExtra(**"newBan"**, 0);  **monAn**=data.getStringExtra(**"newMon"**);  **soLuong**=data.getIntExtra(**"newSl"**, 0);  **soThich**=data.getStringExtra(**"newSt"**);  **kM**=data.getStringExtra(**"newKm"**);   **oderList**.get(**a**).setBan(**banAn**);  **oderList**.get(**a**).setMon(**monAn**);  **oderList**.get(**a**).setSoluong(**soLuong**);  **oderList**.get(**a**).setSothich(**soThich**);  **oderList**.get(**a**).setKm(**kM**);  PhoDatabase.*getInstance*(**this**).phoDao().update(**oderList**.get(**a**));  **oderList**.clear();  **oderList**.addAll(PhoDatabase.*getInstance*(getApplicationContext()).phoDao().getAll());  **adapter**.notifyDataSetChanged();  Toast.*makeText*(**this**, **"Sửa thành công"**, Toast.***LENGTH\_SHORT***).show();  } }s |

* + - 1. **Xoá đặt món.**

Hàm xử lý khi kích vào Xoá

|  |
| --- |
| **private void** doDelete() {  Dialog dialog = **new** AlertDialog.Builder(MainActivity.**this**)  .setTitle(**"Thông báo"**)  .setMessage(**"Bạn có muốn xoá không?"**)  .setNegativeButton(**"Đồng ý"**, **new** DialogInterface.OnClickListener() {  @Override  **public void** onClick(DialogInterface dialog, **int** which) {  **if** (**a** != 0) {  PhoDatabase.*getInstance*(getApplicationContext()).phoDao().delete(**oderList**.get(**a**));  **oderList**.clear();  **oderList**.addAll(PhoDatabase.*getInstance*(getApplicationContext()).phoDao().getAll());  **adapter**.notifyDataSetChanged();  } **else** {  Toast.*makeText*(MainActivity.**this**, **"Mời bạn chọn hoá đơn cần xoá"**, Toast.***LENGTH\_LONG***).show();  }  }  }).setPositiveButton(**"Huỷ"**, **new** DialogInterface.OnClickListener() {  @Override  **public void** onClick(DialogInterface dialog, **int** which) {  dialog.cancel();  }  }).show();  } |

* + - 1. **Giới thiệu về cửa hàng**

|  |
| --- |
| 1. **package** com.example.nhom2\_android\_project;  **import** androidx.appcompat.app.AppCompatActivity; **import** android.os.Bundle; **import** android.widget.TextView;  **public class** App\_Infor **extends** AppCompatActivity {   TextView **txt\_appinfor**, **txt\_footer**;   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***app\_\_infor***);  **txt\_appinfor** = findViewById(R.id.***txt\_appinfor***);  **txt\_footer** = findViewById(R.id.***txt\_footer***);   **txt\_appinfor**.setText(**"Phần mềm quản lý gọi món của hàng phở Thu Vương"** +  **"\nPhiên bản: 1.0"** +  **"\nPhần mềm quản lý gọi món là ứng dụng của nhóm 2 giúp cho việc đặt phở của nhà hàng Thu Vương dễ dàng quản lý hơn."** +  **"Ứng dụng đưa ra các thống kê đặt hàng rõ ràng cũng như cung cấp việc thêm, sửa, xoá đặt phở với đầy đủ các loại phở mà nhà hàng cung cấp."** );  **txt\_footer**.setText(**"Copyright © Nhom 12 KTPM2"** +  **"\nPham Van Thu - Vu Duc Vuong"**);  } } |

# Kiến thức lĩnh hội và bài học kinh nghiệm

## Kiến thức lĩnh hội

* Qua quá trình tìm hiểu đề tài, Nhóm chúng em đã tiếp thu được rất nhiều kiến thức, cụ thể:
* Cách xây dựng ứng dụng một ứng dụng quản lý trên hệ điều hành android (Cụ thể là quản lý gọi món của cửa hàng phở)
* Các kỹ năng teamwork, phân phối thời gian, kỹ năng tìm kiếm tài liệu
* Trau dồi thêm các kỹ năng lập trình cũng như tư duy lôgic giải quyết vấn đề

## Bài học kinh nghiệm và hướng phát triển

* Quá trình tìm hiểu và xây dựng đề tài giúp nhóm em rút ra được những bài học kinh nghiệm. Nhận thấy bản thân mình còn yếu ở đâu để tiếp tục trau dồi hơn nữa
* Nhóm có dự định phát triển sản phẩm thêm nữa:
  + - * + Cải tiến giao diện phù hợp hơn
        + Xây dựng thêm nhiều chức năng quản lý hơn
        + Độ bảo mật tốt hơn

# Tài liệu tham khảo

<https://developer.android.com/reference/android/arch/persistence/room/Database>