

TRƯỜNG CAO ĐẲNG CÔNG NGHỆ THỦ ĐỨC
KHOA CÔNG NGHỆ THÔNG TIN



PHÂN TÍCH THIẾT KẾ HỆ THỐNG

Lập Trình Di Động 2

APP QUẢN LÝ BÁN HÀNG

Giảng Viên: Trương Bá Thái

Sinh Viên Thực Hiện:

1. Lê Khương Tường Vy
2. Mai Thị Quanh

Tp.Hồ Chí Minh , ngày 14 tháng 07 năm 2020

Nhật Ký Hoạt Động Nhóm

STT	Họ Tên Sinh Viên	Công Việc Phân Công	Tự Đánh Giá	Nhóm Đánh Giá	Ký Tên
1	Mai Thị Quanh	<ul style="list-style-type: none">-Thiết kế màn hình trang chủ-Thiết kế navigation-Thiết kế dialog đăng nhập , đăng ký.-Thiết kế database-Viết SRS-Việc chức năng thêm ,xóa ,sửa , hiển thị sản phẩm- Thiết kế màn hình và chức năng giỏ hàng			
2	Lê Khương Tường Vy	<ul style="list-style-type: none">-Thiết kế giao diện danh sách Điện Thoại và LapTop- Thiết kế màn hình Thông Tin Cửa Hàng- Thiết kế màn hình Thông Tin Khách Hàng- Thiế kế màn hình Thống Kê-Viết SRS			

Mục Lục

Chương I. Thiết kế giao diện nâng cao trong android	5
Chương II. Điều khiển màn hình tương tác và cảm biến	7
Chương III. Đồ họa và hoạt cảnh	9
Chương IV. Đa phương tiện và truyền thông thoại	11
Chương V. Bản đồ trực tuyến, map và location	13
Chương VI. Tổng quan ứng dụng quản lý bán hàng	14
1. Giới thiệu	14
1.1 Tổng quan.....	14
1.2 Phạm vi	14
2. Yêu cầu chức năng	14
2.1 Màn hình trang chủ	14
2.2 Màn hình menu	17
2.3 Màn hình đăng ký	18
2.4 Màn hình đăng nhập	21
2.5 Màn hình thông tin sản phẩm	22
2.6 Màn hình danh sách đơn hàng	25
2.7 Màn hình thanh toán	26
2.8 Màn hình thông tin khách hàng	28
2.9 Màn hình tình trạng đơn hàng	30
2.10 Màn hình hiển thị danh sách sản phẩm.....	32
2.11 Màn hình liên hệ nhân viên	33
Chương VII. Tổng quan ứng dụng và cài đặt.....	35
1. Màn hình đăng nhập.....	35
2. Màn hình Đăng ký	36
3. Màn hình Trang chủ	38
4. Màn hình menu	39
5. Màn hình Danh sách điện thoại.....	40
6. Màn hình hiển thị danh sách Laptop	41
7. Màn hình thanh toán	42
8. Màn hình Chi tiết sản phẩm	43
9. Màn hình giỏ hàng	45
10. Màn hình thông tin khách hàng	47
11. Màn hình liên hệ.....	48
12. Màn hình Maps	50

CHƯƠNG VIII. KẾT QUẢ ĐẠT ĐƯỢC.....	51
1.Kết quả đạt được	51
2.Kết luận	51

Chương I. Thiết kế giao diện nâng cao trong android

1. Lý thuyết

- Style

Kiểu là tập hợp các tùy chỉnh thuộc tính chế độ xem

kích thước văn bản hoặc màu nền, nên được áp dụng cho nhiều chế độ xem trong toàn bộ ứng dụng

- Theme

Nhóm nhiều kiểu lại với nhau trong một phần tử chung

Áp dụng cho toàn bộ bối cảnh (Hoạt động hoặc ứng dụng)

Xác định kiểu nên áp dụng cho tất cả các chế độ xem trong ngữ cảnh được xác định

Mỗi hoạt động có một chủ đề được áp dụng cho nó

Kiểu và Chủ đề được định nghĩa trong các giá trị / style.xml

- Widget

Không phải là tiện ích ứng dụng (ví dụ: ứng dụng thời tiết)

Thường đề cập đến các Chế độ xem chuyên biệt được đặt trong các tệp bố cục, chẳng hạn như Nút, TextView, CheckBox, v.v. GridView

- Recycler

Có thể cuộn theo chiều ngang (Khó với ListView)

Tích hợp với những người có chế độ xem

Hoạt ảnh khi Thêm, Cập nhật và Xóa các mục

Bộ chuyển đổi thông báo

thông báoDataSetChanged (), thông báoItemInserted (), thông báoItemRemoved () và thông báoItemChanged ()

Phải tạo bộ điều hợp để điền dữ liệu vào đó

Lớp con của lớp RecyclerView.Adapter

Tạo chế độ xem cho từng mục danh sách

Thay thế nội dung của chúng khi từng mục trong tập dữ liệu không còn hiển thị

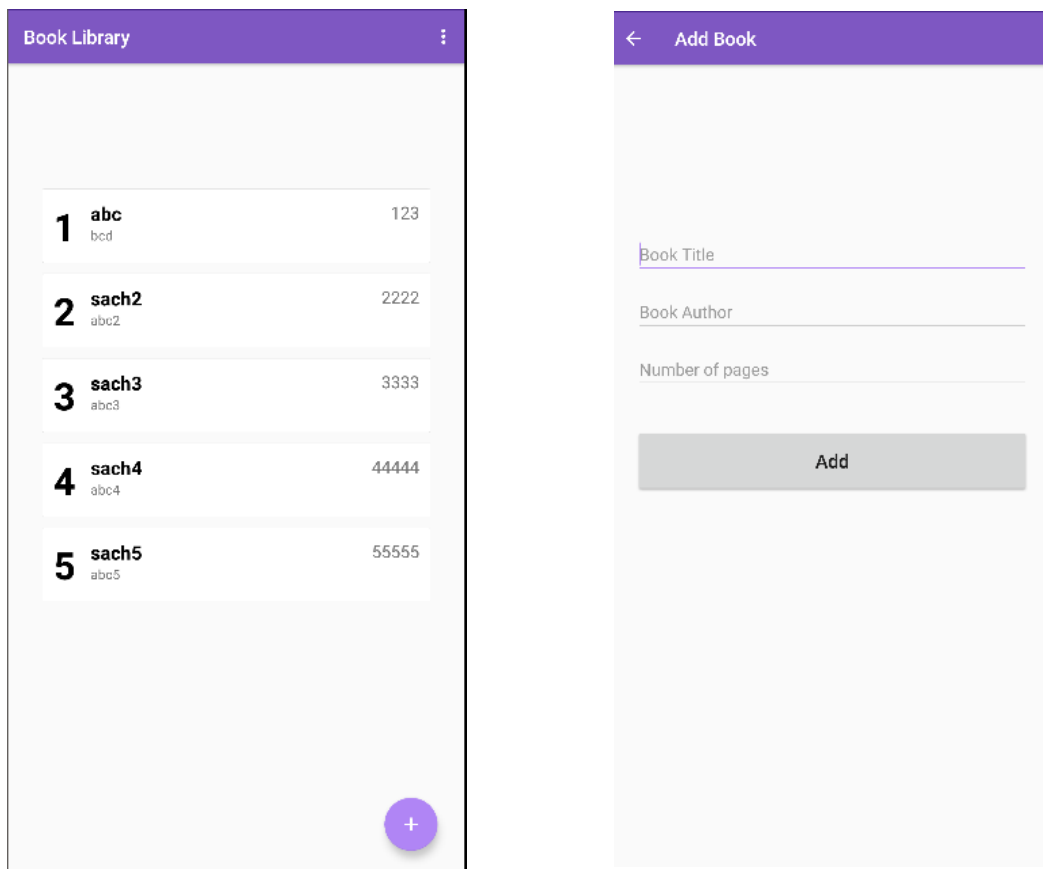
Mỗi bộ chuyển đổi

Chỉ định loại dữ liệu mà bộ điều hợp sẽ hoạt động

Tạo chế độ xem

Liên kết dữ liệu với các chế độ xem

2. Ví dụ thực hành



- Ứng dụng Book Library sử dụng hiển thị danh sách RecyclerView, button thêm sách sử dụng FloatingActionButton

Chương II. Điều khiển màn hình tương tác và cảm biến

1. Lý Thuyết

- Touch Event

Thu thập dữ liệu chuyển động: onTouchEvent ()

Xác định cử chỉ tương ứng là gì khi sử dụng lớp GestureDetector trong Android

onTouchEvent ()

onDown()

onFling ()

onLongPress ()

onScroll ()

onShowPress ()

onDoubleTap ()

- CustomView

- Sensor

Sử dụng Khung cảm biến Android, bao gồm:

Bộ cảm biến

cảm biến

Bộ cảm biến

Cảm biến

Hai bước với cảm biến

Chỉ định cảm biến trong Bản kê khai Android

Sử dụng <used-Feature>

Ví dụ: Làm việc với một la bàn

Cảm biến chuyển động

Gia tốc kế, con quay hồi chuyển

Trọng lực, gia tốc tuyến tính, vector quay

Cảm biến môi trường

Độ ẩm

Ánh sáng

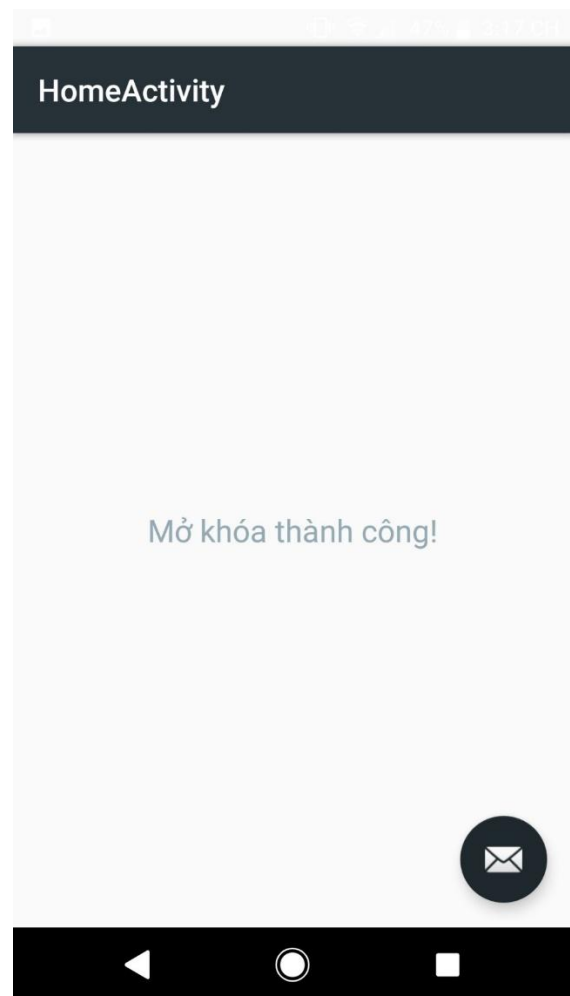
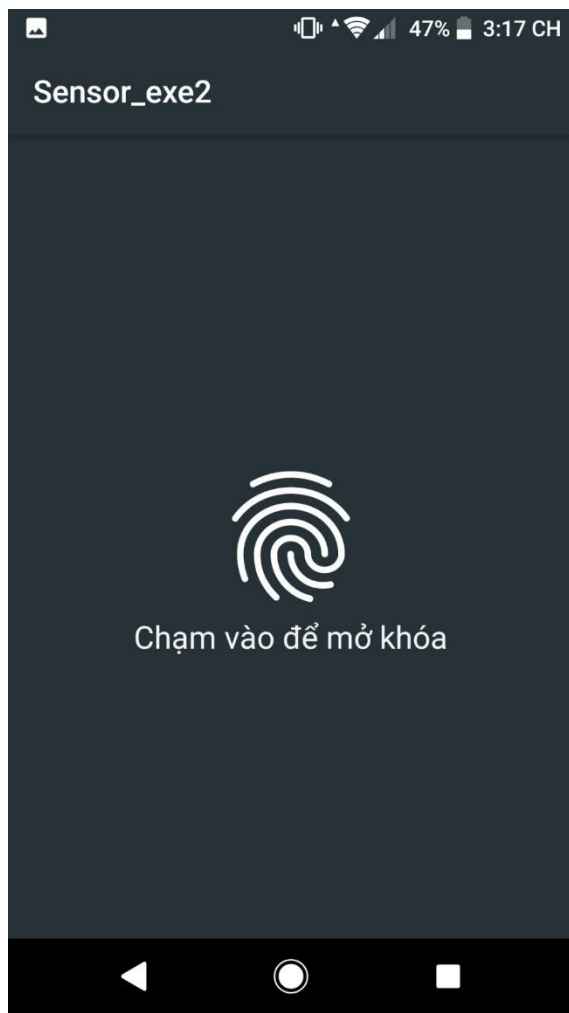
Sức ép

Nhiệt độ

Cảm biến vị trí

Trường địa từ

2. Ví dụ thực hành



- Ứng dụng kiểm tra dấu vân tay sử dụng cảm biến sensor có sẵn

Chương III. Đồ họa và hoạt cảnh

1. Lý thuyết

- Splashscreen
- Hoạt cảnh chuyển động sử dụng hình trong drawable

Mối quan hệ giữa bố cục của bạn, cảnh, chuyển tiếp và hoạt hình cuối cùng

Hoạt ảnh chuyển tiếp

Tạo bố cục bắt đầu (Cảnh bắt đầu)

Tạo bố cục kết thúc (Kết thúc cảnh)

Tạo tệp chuyển tiếp: Loại thay đổi

AutoTransition (chuyển đổi mặc định): Làm mờ dần, di chuyển và thay đổi kích thước, sau đó mờ dần (theo thứ tự đó)

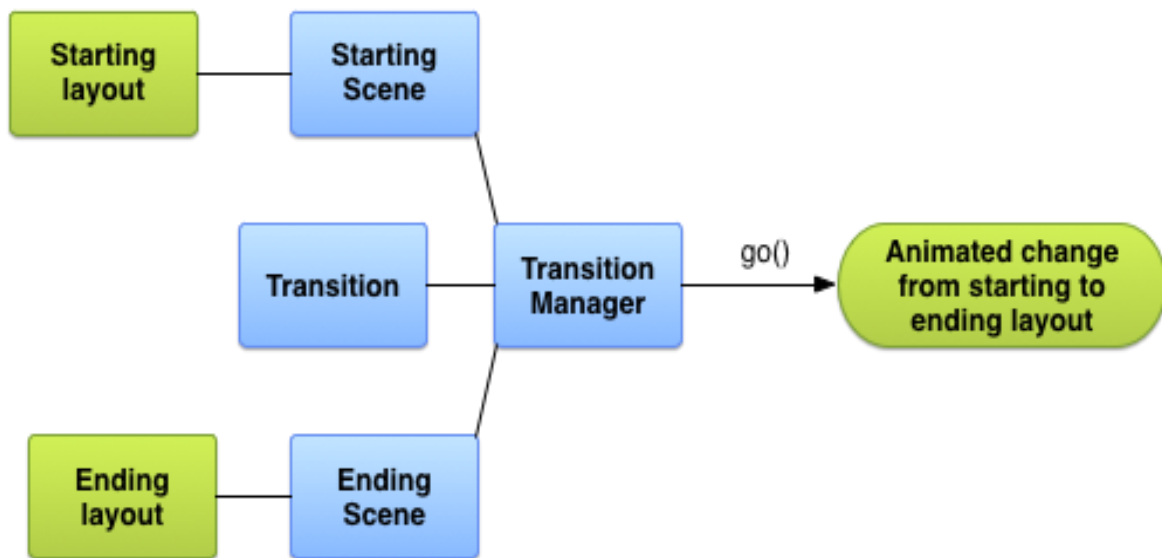
Làm mờ: Làm mờ dần, mờ dần (mặc định) hoặc cả hai (chỉ định thứ tự)

ChangeBounds: Di chuyển và thay đổi kích thước

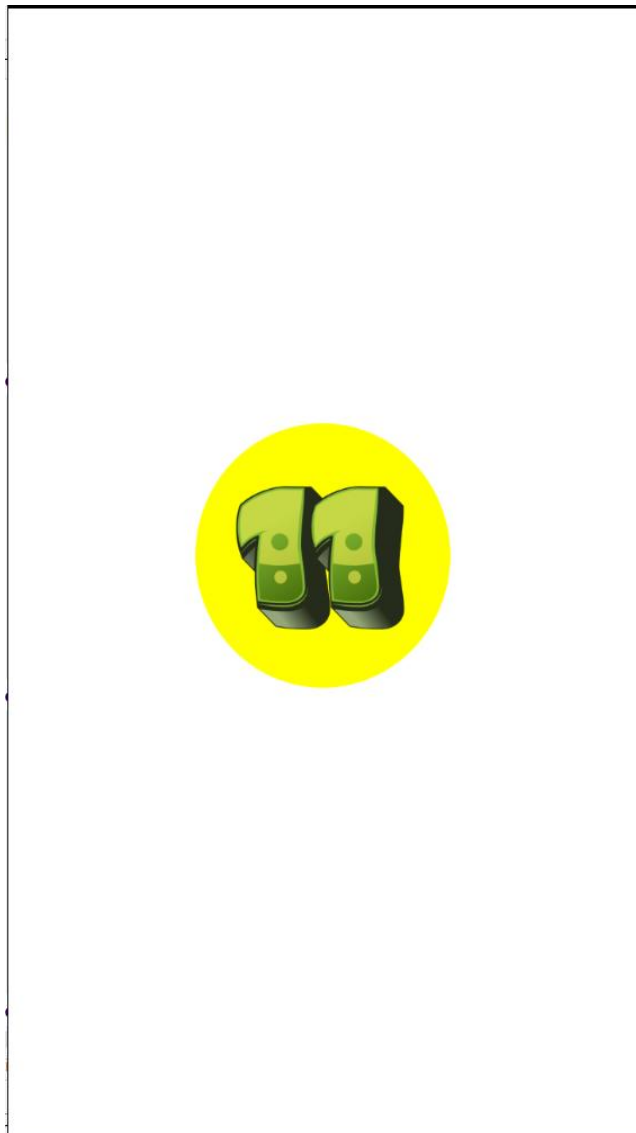
Trượt, nổ, tô màu lại...

Áp dụng hoạt ảnh bằng TransitionManager

Android Transition Framework: Tự động tạo các khung hình cần thiết để tạo hoạt ảnh từ các cảnh đầu đến cuối



2. Ví dụ thực hành



ĐĂNG NHẬP



Nhập tên người dùng _____

Nhập mật khẩu _____

ĐĂNG KÝ **ĐĂNG NHẬP**

THOÁT

Chương IV. Đa phương tiện và truyền thông thoại

1. Lý Thuyết

- Playing sound effects với SoundPool

Đối với clip âm thanh ngắn

Để có thể tạo hiệu ứng đặc biệt với âm thanh của chúng tôi

Thay đổi tỷ lệ chơi

Nhiều âm thanh để phát cùng lúc

Được hỗ trợ: .3gp, .flac, .mp3, .mid, .xmf, .mxmf, .ogg, .wav

Hoạt động tốt trên API 21 trở lên

- Playing audio và video với MediaPlayer

Hỗ trợ nhiều loại tệp hơn SoundPool

.mkv, .webm, .mp4, .m4a ...

Hỗ trợ tập tin nhạc hoặc video lớn

Cách làm

Tái sử dụng PlayMusicControl

Tạo dự án mới

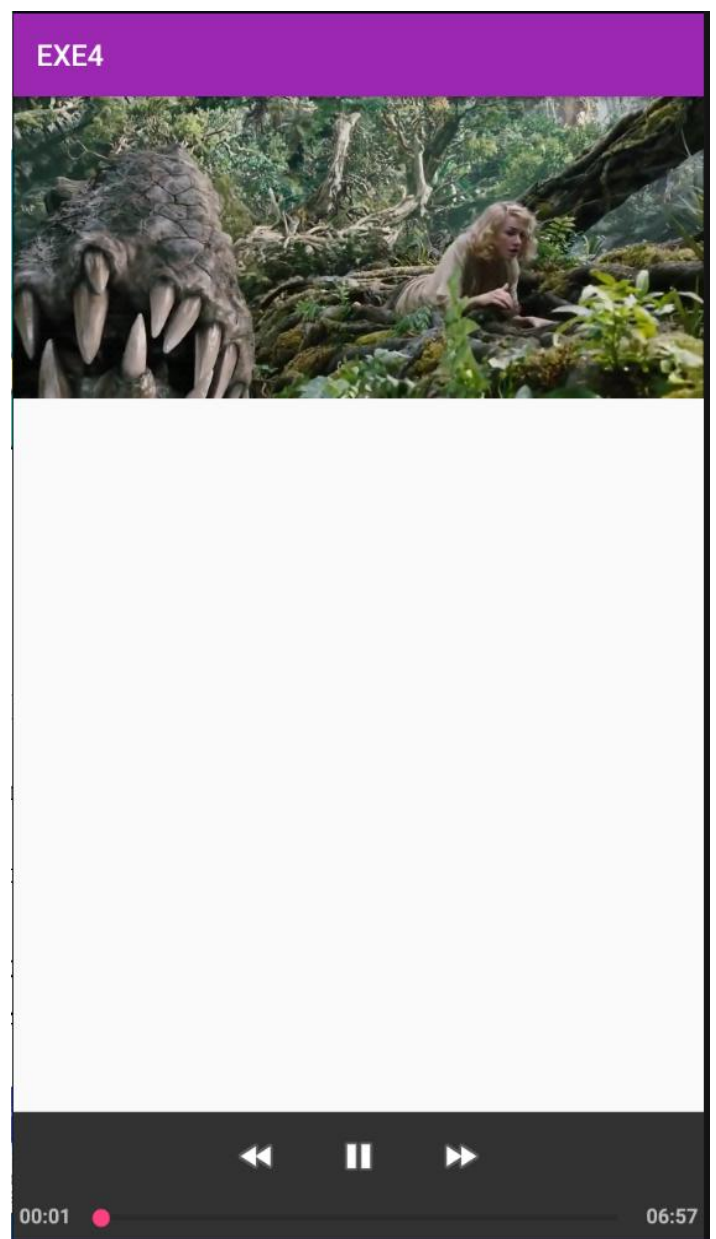
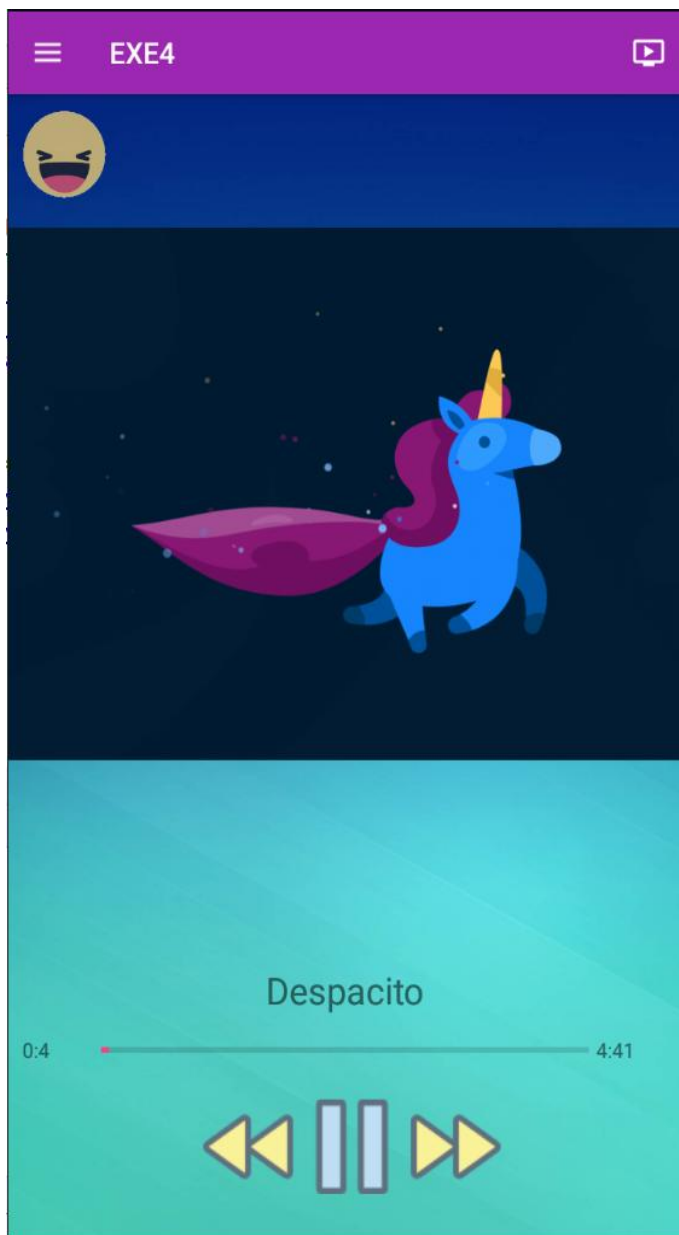
Dự án trống: Đa phương tiện

Sao chép nội dung bố cục của dự án CustomViews sang bố cục chính

Sao chép PlayMusicViewControl.java, MusicPlayerFunction

- Sử dụng camera trong android
- Sử dụng gọi điện thoại
- Network

2. Ví dụ thực hành



Chương V. Bản đồ trực tuyến, map và location

1. Lý thuyết

- ConnectionCallbacks trong android
- Setup for GoogleAPIClient
- Sử dụng Google Maps Activity

2. Ví dụ thực hành



- App Google maps, hiển thị bản đồ từ vệ tinh, định vị vị trí từ thiết bị hoặc tọa độ setup sẵn

Chương VI. Tổng quan ứng dụng quản lý bán hàng

1. Giới thiệu

1.1 Tổng quan

- Bạn đã, đang hay chuẩn bị kinh doanh? Vậy hãy làm quen ngay với người bạn mang tên phần mềm quản lý bán hàng – một công cụ đắc lực cho các nhà bán hàng. Các ứng dụng quản lý bán hàng trên điện thoại ngày càng cải tiến chỉ cần có kết nối internet, chủ shop hoàn toàn chủ động quản lý cửa hàng của mình ngay cả khi không có mặt tại shop.
- **Đăng nhập dễ dàng:** Chỉ cần đăng ký tài khoản hoặc đăng nhập nhanh với tài khoản là đã có thể truy cập phần mềm và sử dụng ngay.
- **Dễ sử dụng:** Giao diện phần mềm được thiết kế đơn giản, bố cục rõ ràng, hợp lý, giúp người mới sử dụng cũng có thể dễ dàng nắm bắt.
- **Bán hàng nhanh chóng:** Phần mềm được tối ưu thân thiện cho người dùng, thuận tiện cho các thao tác chọn sản phẩm, tính tiền,...
- **Xử lý nhanh:** Hoàn tất đơn nhanh chóng, đẩy thông tin tự động sang các đối tác vận chuyển. 1 phần mềm quản lý bán hàng miễn phí tốt cần có tính năng gợi ý phí vận chuyển giúp bạn lựa chọn được đơn vị vận chuyển phù hợp nhất.
- **Quản lý và bán hàng từ trực tuyến đến cửa hàng:** Phần mềm quản lý bán hàng miễn phí tốt nhất không chỉ giúp bạn bán hàng và quản lý các đơn bán tại cửa hàng mà còn có khả năng xử lý và **quản lý bán hàng online** hiệu quả.
- **Quán xuyên cửa hàng mọi lúc mọi nơi:** Hệ thống báo cáo từ tổng quan đến chi tiết sẽ giúp bạn theo dõi hiệu quả kinh doanh của cửa hàng chỉ với 1 chiếc smartphone.
- **Sử dụng không giới hạn:** Mặc dù là phần mềm quản lý bán hàng miễn phí nhưng vẫn phải chất. Các tính năng trên phần mềm có thể sử dụng hoàn toàn không bị giới hạn.

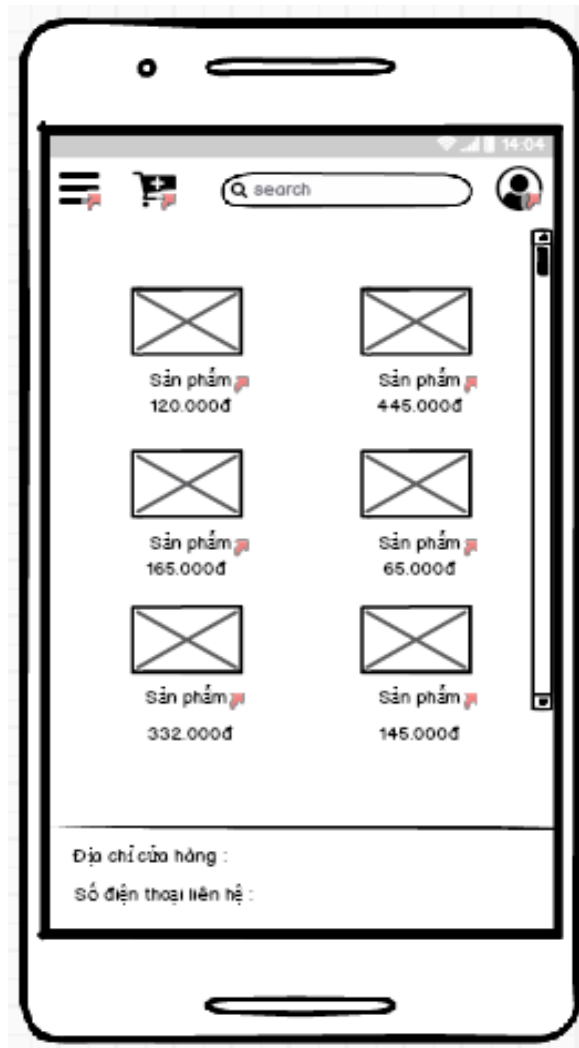
1.2 Phạm vi

- Phiên bản dùng thử
- Sử dụng giả lập
- Kết nối database

2. Yêu cầu chức năng

2.1 Màn hình trang chủ

2.1.1 Giao diện màn hình chính



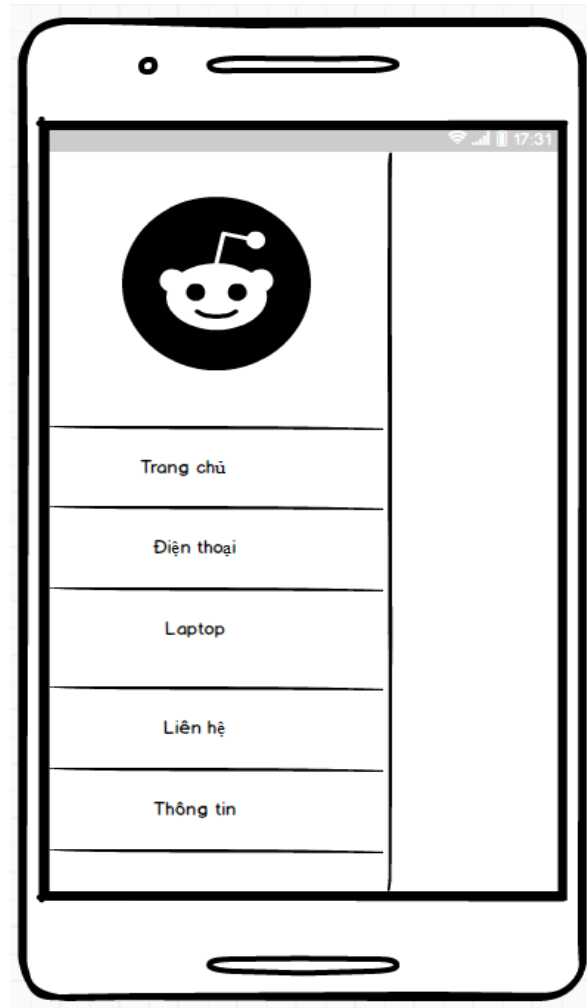
2.1.2 Yêu cầu chức năng

Mục	Miêu tả	Hoạt động	Phản ứng
Màn hình chính	Màn hình chính gồm có: <ul style="list-style-type: none"> - Slide Menu - Icon Button giỏ hàng - Textbox tìm kiếm - ListView Item - Image Item 	N/A	N/A
Slide Menu	Khi người dùng muốn tìm sản phẩm theo phân loại	Chạm vào Slide Menu trên màn hình chính	Sử dụng biến Intent để thiết lập điều hướng khi khách hàng chạm vào slider menu, màn hình sẽ chuyển sang màn hình có Slide Menu với các chức năng

Icon Button giỏ hàng	Khi người dùng muốn xem giỏ hàng	Chạm vào button giỏ hàng trên màn hình chính	Sử dụng biến Intent để thiết lập điều hướng khi khách hàng chạm vào icon giỏ hàng, màn hình sẽ chuyển sang màn hình đơn hàng(dữ liệu từ database DSDH)
Textbox tìm kiếm	Cho phép người dùng gõ vào món hàng cần tìm	Chạm vào textbox tìm kiếm và gõ kí tự	So sánh dữ liệu nhập của khách hàng với database, nếu trùng khớp sẽ hiển thị sản phẩm khách hàng cần tìm, nếu sai sẽ báo lỗi "không có sản phẩm phù hợp"
Icon button Tìm kiếm	Khi người dùng muốn tìm món hàng	Chạm vào Icon Button Tìm kiếm	Lấy dữ liệu từ databse(DS sản phẩm) và hiển thị bằng ListView cùng với từ khóa mà khách hàng đã nhập trước đó
ListView Item	Hiển thị danh sách các món hàng	N/A	N/A
Image Item	Hiển thị hình ảnh sản phẩm	N/A	N/A

2.2 Màn hình menu

2.2.1 Giao diện



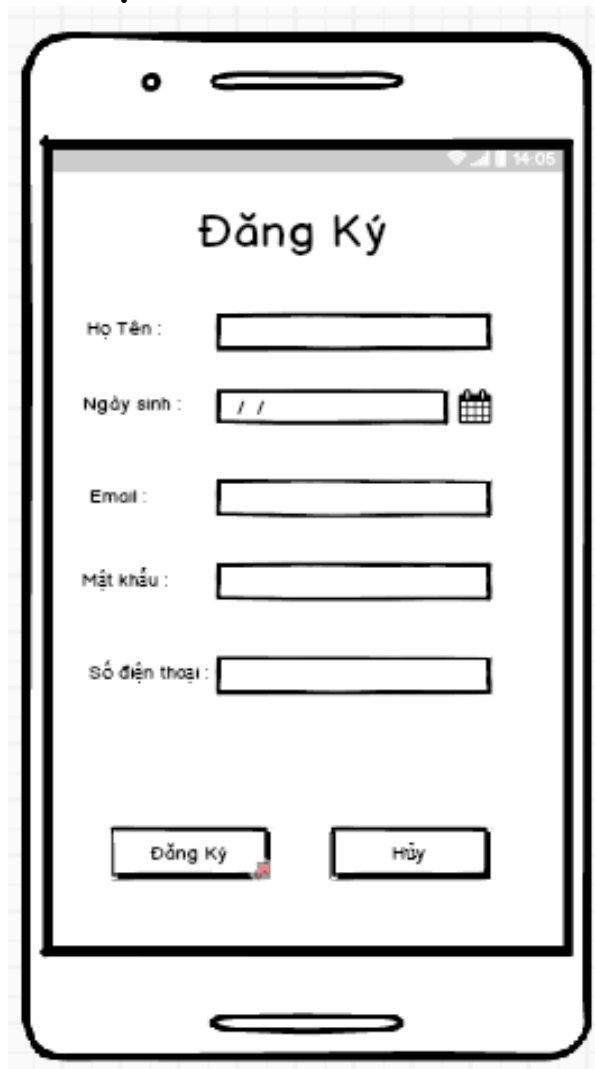
2.2.2 Mô tả chức năng

Mục	Miêu tả	Hoạt động	Phản ứng
Màn hình Slide Menu	Màn hình Slide Menu gồm có: <ul style="list-style-type: none">- Image Avatar- Link Đăng nhập- Link Đăng ký- Danh sách sản phẩm- Thoát	N/A	N/A
Image Avatar	Hiển thị người dùng	N/A	N/A
Link Đăng nhập	Khi người dùng muốn đăng nhập tài khoản	Chạm vào link đăng nhập	Sử dụng biến Intent để thiết lập điều hướng khi khách hàng chạm vào link đăng nhập, màn hình sẽ

			chuyển sang màn hình đăng nhập
Link Đăng ký	Khi người dùng muốn đăng ký tài khoản	Chạm vào link đăng ký	Sử dụng biến Intent để thiết lập điều hướng khi khách hàng chạm vào link đăng kí, màn hình sẽ chuyển sang màn hình đăng kí
Hiển thị danh sách sản phẩm	Hiển thị các danh mục sản phẩm	Chạm vào hiển thị danh sách sản phẩm	Hiển thị ra các danh mục sản phẩm

2.3 Màn hình đăng ký

2.3.1 Giao diện



2.3.2 Mô tả

Mục	Miêu tả	Hoạt động	Phản ứng
-----	---------	-----------	----------

Màn hình đăng kí	Màn hình đăng kí gồm có: <ul style="list-style-type: none"> - Label Đăng ký - Label họ và tên - Textbox họ và tên - Label ngày sinh - Date Picker ngày sinh - Label Email - Textbox Email - Label mật khẩu - Textbox mật khẩu - Label số điện thoại - Textbox số điện thoại - Button đăng kí - Button Hủy 	N/A	N/A
Label Đăng ký	Hiện thị trên màn hình cho người dùng biết đó là trang đăng ký	N/A	N/A
Label Họ và tên	Hiện thị cho người dùng xem vị trí đó cần gõ họ và tên	N/A	N/A
Textbox Họ và tên	Cho phép người dùng nhập Họ và tên vào	Chạm vào ô textbox và gõ họ và tên người dùng	Kiểm tra nếu khách hàng nhập đúng sẽ xuất hiện dấu ✓, nếu nhập sai sẽ thông báo “Họ và tên không thể chứa kí tự đặc biệt hoặc số”
Label Ngày sinh	Hiện thị cho người dùng xem vị trí đó cần gõ Ngày sinh	N/A	N/A
DatePicker Ngày sinh	Cho phép người dùng chọn ngày tháng năm sinh	Chạm vào DatePicker Ngày sinh	Hiện thị lịch có ngày tháng năm
Label Email	Hiện thị cho người dùng xem vị trí đó cần gõ Email	N/A	N/A
Textbox Email	Cho phép người dùng nhập thông tin Email	Chạm vào Textbox Email và gõ kí tự	Kiểm tra nếu khách hàng nhập đúng sẽ xuất hiện dấu ✓, nếu nhập sai sẽ thông báo “Email không thể chứa kí tự đặc biệt”
Label mật khẩu	Hiện thị cho người dùng xem vị trí đó cần gõ mật khẩu	N/A	N/A

Textbox Mật khẩu	Cho phép người dùng nhập Mật khẩu	Chạm vào textbox Mật khẩu và gõ kí tự	Kiểm tra nếu khách hàng nhập đúng sẽ xuất hiện dấu ✓, nếu mật khẩu nhập không đủ yêu cầu sẽ xuất câu “Mật khẩu phải có ít nhất 1 kí tự”.
Textbox Số điện thoại	Cho phép người dùng nhập thông tin Số điện thoại	Chạm vào textbox Số điện thoại và gõ kí tự	Kiểm tra nếu khách hàng nhập đúng sẽ xuất hiện dấu ✓, nếu nhập sai sẽ thông báo “Số điện thoại không thể chứa kí tự đặc biệt hoặc chữ”
Label Nhập lại mật khẩu	Hiện thị cho người dùng xem vị trí đó cần gõ lại mật khẩu	N/A	N/A
Textbox Nhập lại mật khẩu	Cho phép người dùng Nhập Lại Mật Khẩu	Chạm vào textbox Nhập Lại Mật khẩu và gõ kí tự	Kiểm tra nếu khách hàng nhập đúng sẽ xuất hiện dấu ✓, nếu mật khẩu nhập không khớp sẽ xuất câu “Mật khẩu không khớp”
Button Đăng ký	Khi người dùng muốn đăng kí tài khoản	Chạm vào button Đăng kí	Dữ liệu của khách hàng khi nhập xong tất cả thông tin sẽ được lưu vào Database(thông tin khách hàng) và điều hướng sang màn hình chính
Button Hủy	Khi người dùng muốn hủy, không đăng ký tài khoản	Chạm vào button Hủy	Sử dụng biến Intent để thiết lập điều hướng khi khách hàng chạm vào button hủy, màn hình sẽ quay về màn hình chính

2.4 Màn hình đăng nhập

2.4.1 Giao diện



2.4.2 Mô tả

Mục	Sự miêu tả	Hoạt động	Phản ứng
Màn hình đăng nhập	Màn hình đăng nhập gồm có: <ul style="list-style-type: none">- Label đăng nhập- Image Logo- Label Tên đăng nhập- Label Mật khẩu	N/A	N/A

	<ul style="list-style-type: none"> - Textbox Mật khẩu - Link Quên mật khẩu - Button Đăng nhập - Button Hủy 		
Label Đăng nhập	Hiển thị cho người dùng biết đó là trang Đăng nhập	N/A	N/A
Image Logo	Hiển thị logo của shop		
Label Tên Tài Khoản	Hiển thị cho người dùng xem vị trí đó cần gõ Tên tài khoản	N/A	N/A
Textbox Tên Tài Khoản	Cho phép người dùng gõ Tên tài khoản đã đăng ký vào ô textbox	Chạm vào textbox Tên tài khoản và gõ kí tự	Hiển thị các kí tự vừa gõ
Label Mật khẩu	Hiển thị cho người dùng xem vị trí đó cần gõ mật khẩu	N/A	N/A
Textbox Mật khẩu	Cho phép người dùng gõ Mật khẩu đã đăng ký vào ô textbox	Chạm vào ô textbox và gõ kí tự	Hiển thị dấu * thay cho kí tự vừa gõ
Button Đăng nhập	Khi người dùng muốn đăng nhập vào tài khoản của mình	Chạm vào button Đăng nhập	Username và Password của khách hàng sẽ được so sánh trong Database(danh sách tài khoản), nếu trùng khớp sẽ cho phép đăng nhập, còn nếu sai sẽ thông báo “Tài khoản hoặc mật khẩu không đúng”
Button Hủy	Khi người dùng không muốn đăng nhập vào tài khoản cá nhân của mình	Chạm vào button Hủy	Sử dụng biến Intent để thiết lập điều hướng khi khách hàng chạm vào button hủy, màn hình sẽ quay về màn hình chính

2.5 Màn hình thông tin sản phẩm

2.5.1 giao diện



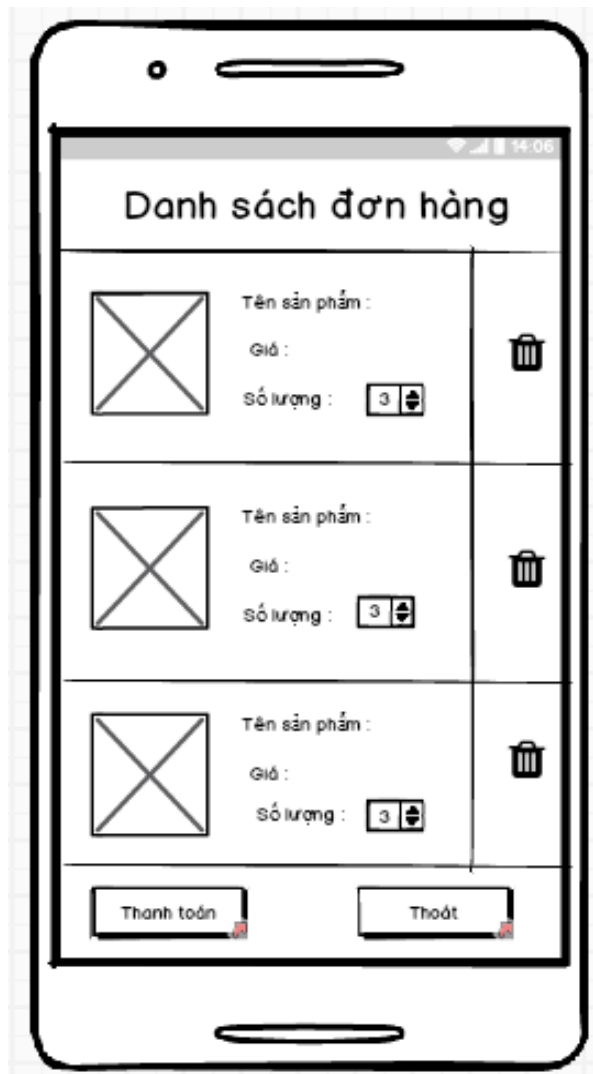
2.5.2 Mô tả

Mục	Miêu tả	Hoạt động	Phản ứng
Màn hình thông tin sản phẩm	<p>Màn hình thông tin sản phẩm gồm có:</p> <ul style="list-style-type: none"> - Label Thông tin sản phẩm - Label Tên sản phẩm - Image sản phẩm - Label Số lượng - Combo box Số lượng - Label Giá tiền - Label Chi tiết sản phẩm - Label Comment - Button Mua - Button Trở về 	N/A	N/A
Label Thông tin sản phẩm	<p>Hiển thị cho người dùng biết đó là trang Thông tin sản phẩm</p>	N/A	N/A

Label Tên sản phẩm	Hiển thị cho người dùng biết Tên sản phẩm đó là gì	N/A	N/A
Image sản phẩm	Hiển thị hình ảnh của sản phẩm	N/A	N/A
Label Số lượng	Hiển thị cho người dùng biết vị trí đó là chọn Số lượng mà người dùng cần mua	N/A	N/A
Combo box Số lượng	Cho phép người dùng chọn số Số lượng cần mua	Chạm vào mũi tên trên combo box rồi chạm chọn số lượng	Hiển thị số lượng vừa chọn trên Combo box
Label Giá tiền	Hiển thị giá tiền cho người dùng xem	N/A	N/A
Label Chi tiết sản phẩm	Hiển thị chi tiết của sản phẩm gồm chất liệu, kiểu dáng, màu sắc, xuất xứ,...	N/A	N/A
Label Comment	Hiển thị các comment của những khách hàng khác cho người dùng xem	N/A	N/A
Button Mua	Khi người dùng muốn mua sản phẩm	Chạm vào button Mua	Sử dụng biến Intent để thiết lập điều hướng khi khách hàng chạm vào button mua, màn hình sẽ chuyển sang màn hình danh sách đơn hàng
Button Trở về	Khi người dùng muốn trở về trang trước đó(màn hình chính)	Chạm vào button Trở về	Sử dụng biến Intent để thiết lập điều hướng khi khách hàng chạm vào button trở về, màn hình sẽ quay về màn hình chính


2.6 Màn hình danh sách đơn hàng

2.6.1 Giao diện



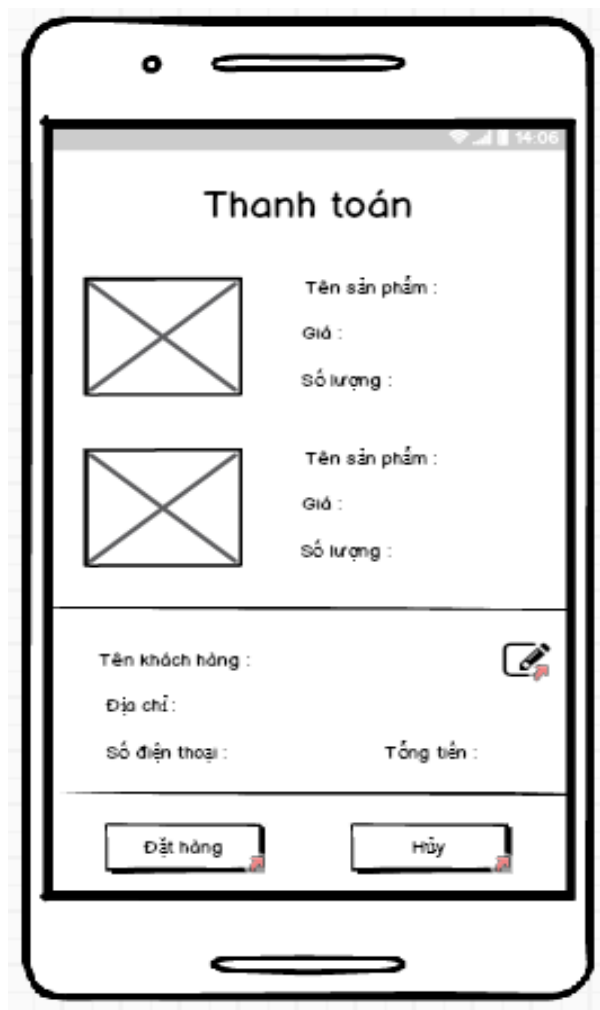
2.6.2 Mô tả

Mục	Miêu tả	Hoạt động	Phản ứng
-----	---------	-----------	----------

Màn hình danh sách đơn hàng(khách hàng)	Cho phép khách hàng xem danh sách các đơn hàng đã đặt mua Màn hình gồm có: <ul style="list-style-type: none"> - 5 textview - 1 numberstepper - 2 button 	Không có	Không có
Textview	Hiển thị thông tin các sản phẩm	Không có	Không có
Numberstepper Số lượng	Khách hàng có thể thay đổi số lượng sản phẩm đã đặt	Chạm vào nút 	Số lượng sẽ thay đổi theo ý định của khách hàng
Button thanh toán	Cho phép khách hàng thanh toán các sản phẩm đã đặt mua	Chạm vào nút thanh toán	Sử dụng biến Intent để thiết lập điều hướng khi khách hàng chạm vào button thanh toán, màn hình sẽ chuyển sang màn hình thanh toán
Button thoát	Thoát khỏi màn hình danh sách sản phẩm đặt mua	Chạm vào nút thoát	Sử dụng biến Intent để thiết lập điều hướng khi khách hàng chạm vào button thoát, màn hình sẽ trở về màn hình thông tin sản phẩm

2.7 Màn hình thanh toán

2.7.1 Giao diện



2.7.2 Mô tả

Mục	Miêu tả	Hoạt động	Phản ứng
Màn hình thanh toán	Cho phép khách hàng thanh toán các sản phẩm đã đặt mua Màn hình gồm: <ul style="list-style-type: none"> - 1 label thanh toán - 7 textview - 3 button 	Không có	Không có
3TextView(first)	Hiển thị thông tin chi tiết sản phẩm	Không có	Hệ thống lấy dữ liệu từ data của Sản phẩm để đẩy lên và cho hiện lên thông tin của sản phẩm mà khách hàng đã mua.


4 textView(second)	Hiển thị thông tin của khách hàng đặt mua, địa chỉ giao, phí vận chuyển và tổng tiền	Không có	Hệ thống sẽ lấy dữ liệu từ bảng thông tin khách hàng để hiển thị lên màn hình
Button đặt hàng	Cho phép khách hàng đặt mua những sản phẩm đã chọn	Chạm vào nút đặt hàng	Sau khi chạm vào buuton “Đặt hàng” thì tất cả dữ liệu về thông tin đặt hàng sẽ được lưu trữ vào database và Chuyển sang màn hình tình trạng đơn hàng
Button hủy	Hủy thanh toán các đơn hàng	Chạm vào nút hủy	Trở về màn hình danh sách các đơn hàng
Button chỉnh sửa thông tin khách hàng	Cho phép chỉnh sửa thông tin khách hàng	Chạm vào biểu tượng bút sửa	Chuyển sang màn hình thông tin khách hàng

2.8 Màn hình thông tin khách hàng

2.8.1 Giao diện



2.8.2 Mô tả


Mục	Miêu tả	Hoạt động	Phản ứng
Màn hình thông tin khách hàng	Hiển thị thông tin khách hàng đã login Màn hình gồm: <ul style="list-style-type: none"> - 5 textview - 5 edit text - 2 button 	Không có	Không có
5 TextView	Hiển thị thông tin của khách hàng	Không có	Hệ thống sẽ lấy dữ liệu từ bảng dữ liệu của khách hàng để cập nhật và hiển thị lên màn hình
5 Edit Text	Cho phép khách hàng thay đổi thông tin cá nhân	Chạm vào 	Thông tin khách hàng thay đổi khi khách hàng nhập xong và thông tin này sẽ

			được cập nhật lại dữ liệu trên bảng dữ liệu của khách hàng
Button cập nhật	Cho phép khách hàng cập nhật lại thông tin đã nhập trên màn hình	Chạm vào nút cập nhật	Tất cả thông tin sẽ được lưu lại và cập nhật lại vào bảng dữ liệu tài khoản
Button hủy	Thoát màn hình	Chạm vào hủy	Trở về màn hình login


2.9 Màn hình tình trạng đơn hàng

2.9.1 Giao diện

Thông tin khách hàng



Họ Tên :

Ngày sinh : 

Email :

Mật khẩu :

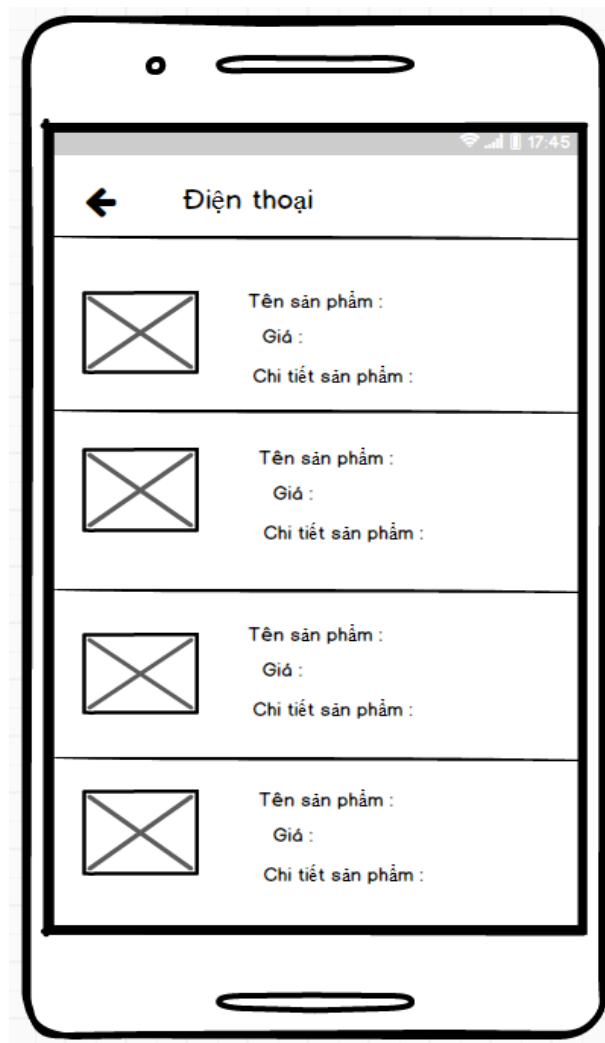
Số điện thoại :

2.9.2 Mô tả

Mục	Miêu tả	Hoạt động	Phản ứng
Màn hình tình trạng các đơn hàng	Khách hàng có thể xem tình trạng các đơn hàng đã đặt mua Màn hình gồm: <ul style="list-style-type: none">- 5 textview- 2 button	Không có	Không có
5 textview(1)	Hiển thị thông tin của sản phẩm, ngày thông tin giao hàng, tình trạng và tổng tiền	Không có	Hệ thống sẽ lấy dữ liệu từ bảng đơn hàng đã được lưu trữ lúc mua hàng để cập nhật lên màn hình
Button thoát	Thoát khỏi màn hình tình trạng đơn hàng	Chạm vào nút thoát	Trở về màn hình chính
Button trở về	Thoát khỏi màn hình tình trạng đơn hàng	Chạm vào nút trở về	Trở về màn hình thanh toán

2.10 Màn hình hiển thị danh sách sản phẩm

2.10.1 Giao diện

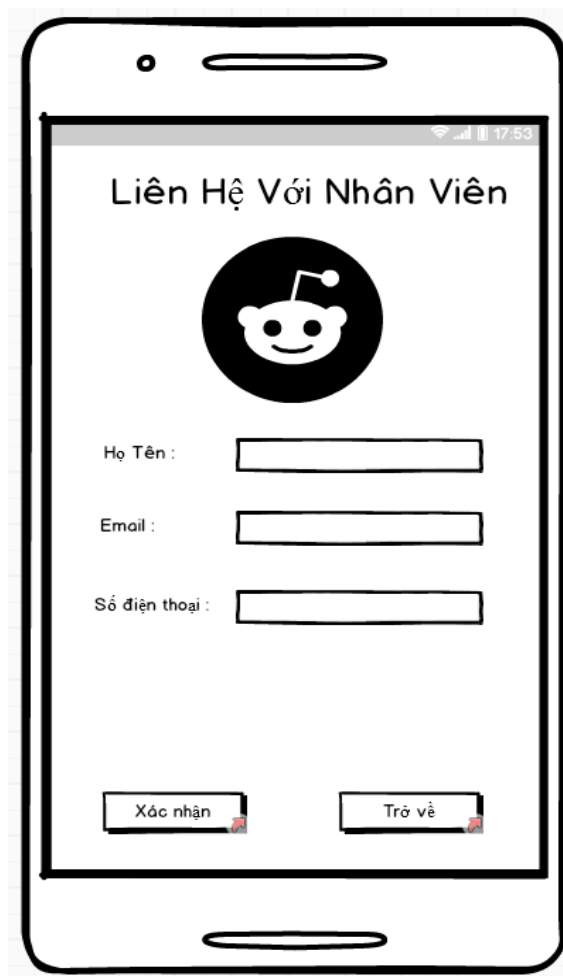


2.10.2 Mô tả

Mục	Miêu tả	Hoạt động	Phản ứng
Màn hình hiển thị danh mục sản phẩm	Khách hàng có thể xem các loại mặt hàng đang được bày bán theo danh mục đã chọn trước đó Màn hình gồm: <ul style="list-style-type: none"> - 3 textview - ImageView - ListView 	Không có	Không có
3 textview(1)	Hiển thị thông tin của sản phẩm	Không có	Thông tin của sản phẩm đã được nhập vào trước đó
ImageView	Hiển thị hình ảnh sản phẩm	Không có	Không có
ListView hiển thị sản phẩm	Hiển thị danh sách sản phẩm	Chạm vào bất kì đâu trên 1 list view	Chuyển qua màn hình hiển thị thông tin chi tiết của 1 sản phẩm đã click trước đó

2.11 Màn hình liên hệ nhân viên

2.11.1 Màn hình



2.11.2 Mô tả

Mục	Miêu tả	Hoạt động	Phản ứng
Màn hình hiển thị thông tin liên hệ với nhân viên	Khách hàng nhập thông tin để nhân viên liên hệ Màn hình gồm: - 3 editText - 2 button	Không có	Không có
3 editText	Khách hàng nhập thông tin để liên hệ với nhân viên	Nhập trường thông tin	Hiện thị thông tin đã nhập
Button Xác nhận	Lưu thông tin đã nhập	click	Lưu thông tin đã nhập
Button Trở về	Thoát khỏi màn hình nhập thông tin	click	Chuyển qua màn hình trước đó

Chương VII. Tổng quan ứng dụng và cài đặt

1. Màn hình đăng nhập

13.1 Giao diện màn hình

ĐĂNG NHẬP



ĐĂNG KÝ

ĐĂNG NHẬP

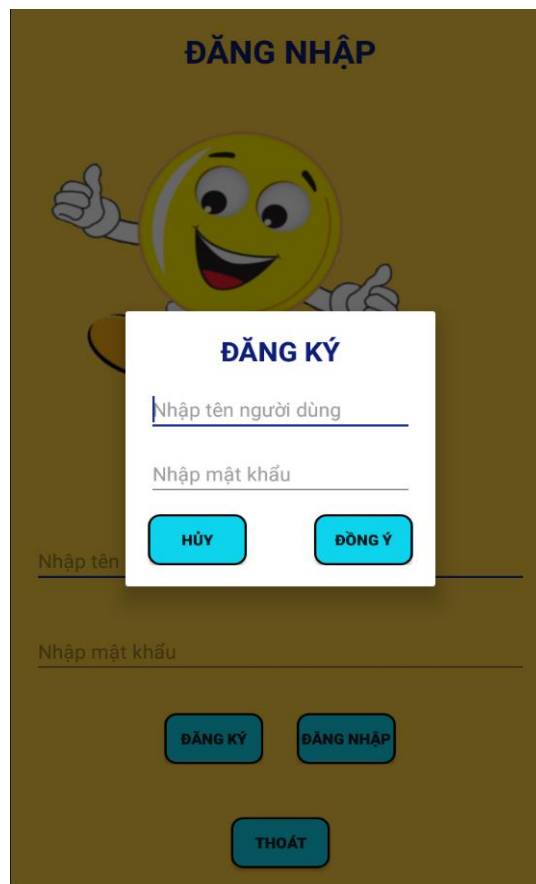
THOÁT

13.2 Code

```
btndangnhap.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        if (edtuser.getText().length() != 0 && edtpassword.getText().length() != 0){  
            if (edtuser.getText().toString().equals(ten) && edtpassword.getText().toString().equals(mk)){  
                Toast.makeText(context: DangNhapActivity.this, text: "Bạn đã đăng nhập thành công", Toast.LENGTH_SHORT).show();  
                Intent intent = new Intent( packageContext: DangNhapActivity.this, TrangChuActivity.class);  
                startActivity(intent);  
            }else if(edtuser.getText().toString().equals("Quanh") && edtpassword.getText().toString().equals("123456")){  
                Toast.makeText(context: DangNhapActivity.this, text: "Bạn đã đăng nhập thành công", Toast.LENGTH_SHORT).show();  
                Intent intent = new Intent( packageContext: DangNhapActivity.this, TrangChuActivity.class);  
                startActivity(intent);  
            }else {  
                Toast.makeText(context: DangNhapActivity.this, text: "Bạn đã đăng nhập thất bại", Toast.LENGTH_SHORT).show();  
            }  
        }else {  
            Toast.makeText(context: DangNhapActivity.this, text: "Mời bạn nhập đủ thông tin", Toast.LENGTH_SHORT).show();  
        }  
    }  
});
```

2. Màn hình Đăng ký

2.1 Giao diện màn hình



2.2 Code

```
btndangky.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final Dialog dialog = new Dialog( context: DangNhapActivity.this);
        dialog.setTitle("Hộp thoại xử lý");
        dialog.setCancelable(false);
        dialog.setContentview(R.layout.customdialog);
        final EditText edttk = (EditText)dialog.findViewById(R.id.edttk);
        final EditText edtmk = (EditText)dialog.findViewById(R.id.edtmk);
        Button btnhuy = (Button)dialog.findViewById(R.id.buttonhuy);
        Button btndongy = (Button)dialog.findViewById(R.id.buttondongy);
        btndongy.setOnClickListener((v) -> {
            ten = edttk.getText().toString().trim();
            mk = edtmk.getText().toString().trim();

            edtuser.setText(ten);
            edtpassword.setText(mk);

            dialog.cancel();
        });
        btnhuy.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                dialog.cancel();
            }
        });
        dialog.show();
    }
});
```

3. Màn hình Trang chủ

3.1 Giao diện màn hình



3.2 Code

```
private void ActionViewFlipper() {
    ArrayList<String> mangquangcao = new ArrayList<>();
    mangquangcao.add("https://tikicdn.com/media/catalog/product/j/7/j7pro-preorder-kv-v1.u3059.d20170613.t112936.249930.jpg");
    mangquangcao.add("https://tintuc.hoanghamobile.com/tin-tuc/wp-content/uploads/2019/08/cach-tat-quang-cao-tren-dien-thoai-xiaomi-1.jpg");
    mangquangcao.add("https://phongvu.vn/hoi-dap/wp-content/uploads/2018/03/MacBook_Pro_13_inch-e1522292522356.jpg");
    mangquangcao.add("https://tenovi.net/public/upload/images/2018/05/07/microsoft-khang-dinh-surface-pro-la-laptop-quotthuong-dangquot-cho-sinh-vien");
    for (int i = 0; i < mangquangcao.size(); i++)
    {
        ImageView imageView = new ImageView(getApplicationContext());
        Picasso.with(getApplicationContext()).load(mangquangcao.get(i)).into(imageView);
        imageView.setScaleType(ImageView.ScaleType.FIT_XY);
        viewFlipper.addView(imageView);
    }
    viewFlipper.setFlipInterval(5000);
    viewFlipper.setAutoStart(true);
    //Gan animation
    Animation animation_slide_in = AnimationUtils.loadAnimation(getApplicationContext(), R.anim.slide_in_right);
    Animation animation_slide_out = AnimationUtils.loadAnimation(getApplicationContext(), R.anim.slide_out_right);
    viewFlipper.setInAnimation(animation_slide_in);
    viewFlipper.setOutAnimation(animation_slide_out);
}
```

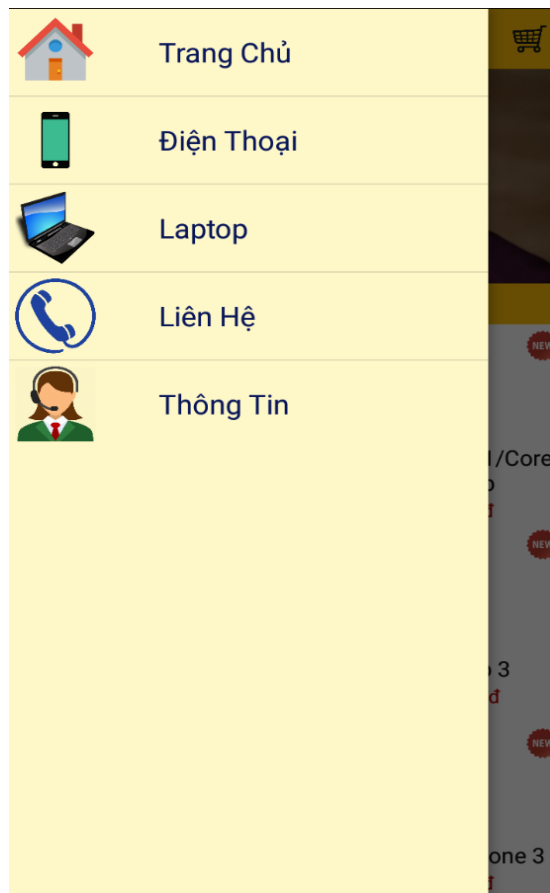
```

private void GetDuLieuSPMoiNhat() {
    RequestQueue requestQueue = Volley.newRequestQueue(getApplicationContext());
    JSONArrayRequest jsonArrayRequest = new JSONArrayRequest(Server.Duongdansanphammoinhat, (response) → {
        if (response != null){
            int ID = 0;
            String Tensanpham = "";
            Integer Giasanpham = 0;
            String Hinhahnsanpham = "";
            String Motasanpham = "";
            int IDsanpham = 0;
            for (int i = 0; i < response.length(); i++){
                try {
                    JSONObject jsonObject = response.getJSONObject(i);
                    ID = jsonObject.getInt( name: "id");
                    Tensanpham = jsonObject.getString( name: "tensp");
                    Giasanpham = jsonObject.getInt( name: "giasp");
                    Hinhahnsanpham = jsonObject.getString( name: "hinhanhsp");
                    Motasanpham = jsonObject.getString( name: "motasp");
                    IDsanpham = jsonObject.getInt( name: "idsanpham");
                    mangsanpham.add(new Sanpham(ID,Tensanpham,Giasanpham,Hinhahnsanpham,Motasanpham,IDsanpham));
                    sanphamAdapter.notifyDataSetChanged();
                } catch (JSONException e) {
                    e.printStackTrace();
                }
            }
        }
    }
}

```

4. Màn hình menu

4.1 Giao diện màn hình



4.2 Code

```
private void GetDulieuLoaisp() {
    RequestQueue requestQueue = Volley.newRequestQueue(getApplicationContext());
    JSONArrayRequest jsonArrayRequest = new JSONArrayRequest(Server.DuongdanLoaisp, (response) -> {
        if (response != null){
            for (int i = 0; i < response.length(); i++){
                try {
                    JSONObject jsonObject = response.getJSONObject(i);
                    id = jsonObject.getInt( name: "id");
                    tenloaisp = jsonObject.getString( name: "tenloaisp");
                    hinhanhloaisp = jsonObject.getString( name: "hinhanhloaisp");
                    mangloaisp.add(new Loaisp(id,tenloaisp,hinhanhloaisp));
                    loaispAdapter.notifyDataSetChanged();
                } catch (JSONException e) {
                    e.printStackTrace();
                }
            }
            mangloaisp.add( index: 3,new Loaisp( id: 0, tenloaisp: "Liên Hệ", hinhanhloaisp: "https://taiappdidong.com/upload/images/icon-lien-he_4047575377260834244.png"));
            mangloaisp.add( index: 4,new Loaisp( id: 0, tenloaisp: "Thông Tin", hinhanhloaisp: "https://cdn2.iconfinder.com/data/icons/rcons-users-color/32/support_woman-512.png"));
            mangloaisp.add( index: 5,new Loaisp( id: 0, tenloaisp: "Thông Kê", hinhanhloaisp: "https://image.flaticon.com/icons/png/512/1321/1321907.png"));
        }
    }, new Response.ErrorListener() {
        @Override
        public void onErrorResponse(VolleyError error) {
            CheckConnection.ShowToast_Short(getApplicationContext(),error.toString());
        }
    });
    requestQueue.add(jsonArrayRequest);
}
```

5. Màn hình Danh sách điện thoại

5.1 Giao diện màn hình

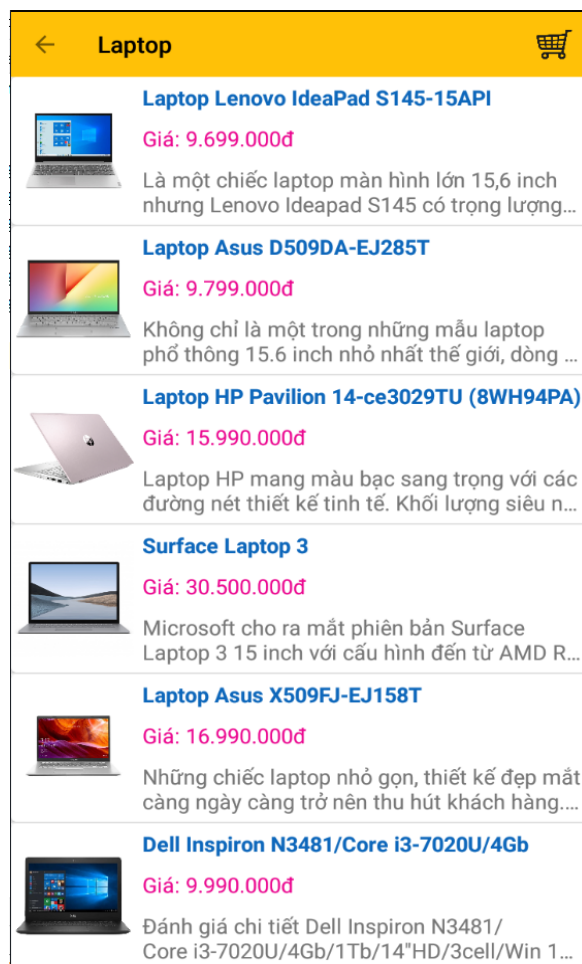


5.2 Code

```
private void LoadMoreData() {  
    lvdt.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
        @Override  
        public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
            Intent intent = new Intent(getApplicationContext(), ChiTietSanPham.class);  
            intent.putExtra( name: "thongtinsanpham", mangdt.get(position));  
            startActivity(intent);  
        }  
    });  
    lvdt.setOnScrollListener(new AbsListView.OnScrollListener() {  
        @Override  
        public void onScrollStateChanged(AbsListView view, int scrollState) {  
        }  
  
        @Override  
        public void onScroll(AbsListView view, int firstVisibleItem, int visibleItemCount, int totalItemCount) {  
            if (firstVisibleItem + visibleItemCount == totalItemCount && totalItemCount != 0 && isLoading == false && limitadata == false){  
                isLoading = true;  
                ThreadData threadData = new ThreadData();  
                threadData.start();  
            }  
        }  
    });  
}
```

6. Màn hình hiển thị danh sách Laptop

6.1 Giao diện màn hình



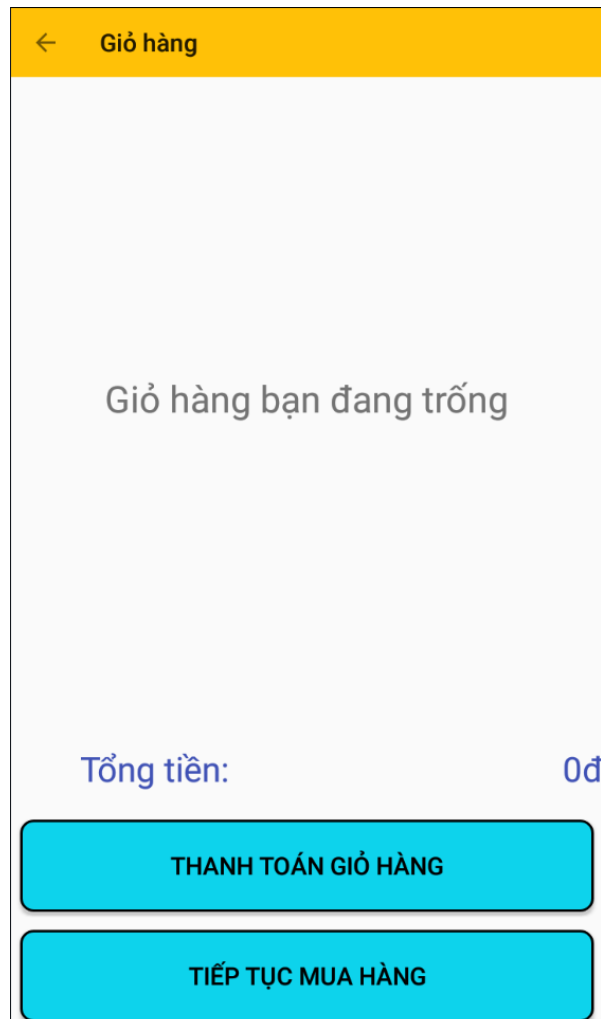
6.2 Code

```
private void LoadMoreData() {  
    lvlaptop.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
        @Override  
        public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
            Intent intent = new Intent(getApplicationContext(), ChiTietSanPham.class);  
            intent.putExtra( name: "thongtinsanpham", manglaptop.get(position));  
            startActivity(intent);  
        }  
    });  
    lvlaptop.setOnScrollListener(new AbsListView.OnScrollListener() {  
        @Override  
        public void onScrollStateChanged(AbsListView view, int scrollState) {  
        }  
        @Override  
        public void onScroll(AbsListView view, int firstVisibleItem, int visibleItemCount, int totalItemCount) {  
            if (firstVisibleItem + visibleItemCount == totalItemCount && totalItemCount != 0 && isLoading == false && limitadata == false){  
                isLoading = true;  
                ThreadData threadData = new ThreadData();  
                threadData.start();  
            }  
        }  
    });  
}
```

Plugin Update
Android Grad

7. Màn hình thanh toán

13.1 Giao diện màn hình



13.2 Code

```
private void EventButton() {  
    btntieptucmua.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            Intent intent = new Intent(getApplicationContext(),MainActivity.class);  
            startActivity(intent);  
        }  
    });  
    btntthanhtoan.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            if (MainActivity.manggiohang.size() > 0){  
                Intent intent = new Intent(getApplicationContext(),Thongtinkhachhang.class);  
                startActivity(intent);  
            }else {  
                CheckConnection.ShowToast_Short(getApplicationContext(),thongbao: "Giỏ hàng của bạn chưa có sản phẩm để thanh toán");  
            }  
        }  
    });  
}
```

8. Màn hình Chi tiết sản phẩm

8.1 Giao diện màn hình



8.2 Code

```
private void CatchEventSpinner() {

    ArrayAdapter<String> adapter = new ArrayAdapter<>(
        context: this,
        R.layout.custom_spinner,
        getResources().getStringArray(R.array.list)
    );
    adapter.setDropDownViewResource(R.layout.custom_spinner_dropdown);
    spinner.setAdapter(adapter);

    spinner.setOnItemClickListener(new AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {
            String selectedItem = spinner.getItemAtPosition(i).toString();
            Toast.makeText(getApplicationContext(),selectedItem,Toast.LENGTH_LONG).show();
        }
        @Override
        public void onNothingSelected(AdapterView<?> adapterView) {

        }
    });
}
```

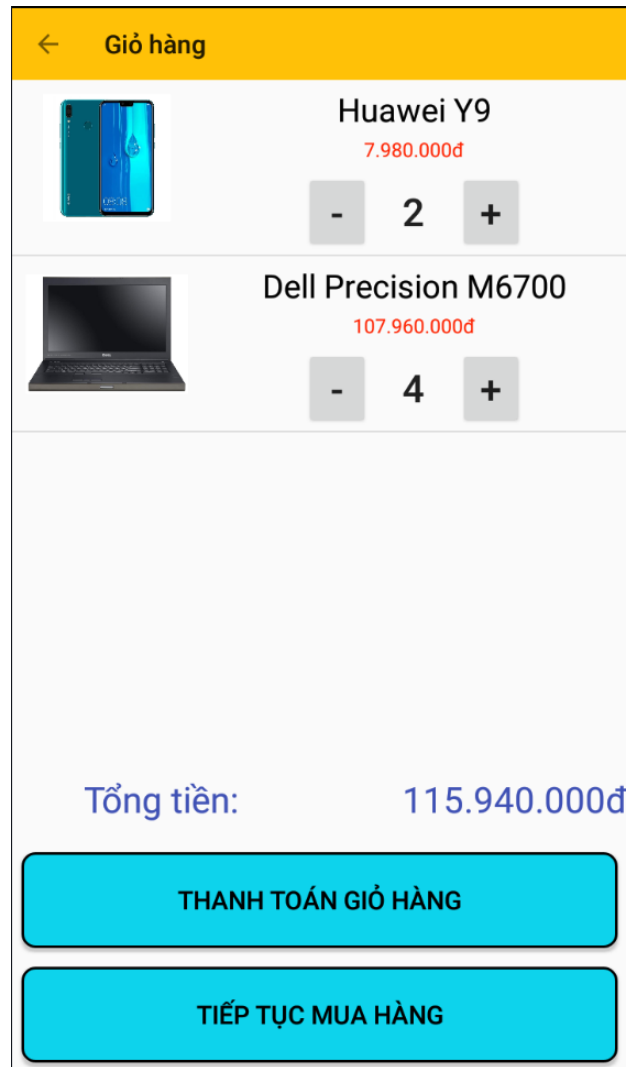
```
private void GetInformation() {

    Sanpham sanpham = (Sanpham) getIntent().getSerializableExtra( name: "thongtinsanpham");
    id = sanpham.getID();
    TenChitiet = sanpham.getTensanpham();
    GiaChitiet = sanpham.getGiasanpham();
    HinhanhChitiet = sanpham.getHinhahnsanpham();
    Motachitiet = sanpham.getMotasampham();
    Idsanpham = sanpham.getIDSanpham();
    txtten.setText(TenChitiet);
    DecimalFormat decimalFormat = new DecimalFormat( pattern: "###,###,###");
    txtgia.setText("Giá : " + decimalFormat.format(GiaChitiet) + "đ");
    txtmota.setText(MotaChitiet);
    Picasso.with(getApplicationContext()).load(HinhanhChitiet)
        .placeholder(R.drawable.noimage)
        .error(R.drawable.er)
        .into(imgChitiet);
}
```

```
private void ActionToolbar() {
    setSupportActionBar(toolbarChitiet);
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);
    toolbarChitiet.setNavigationOnClickListener((v) -> { finish(); });
}
```

9. Màn hình giỏ hàng

13.1 Giao diện màn hình

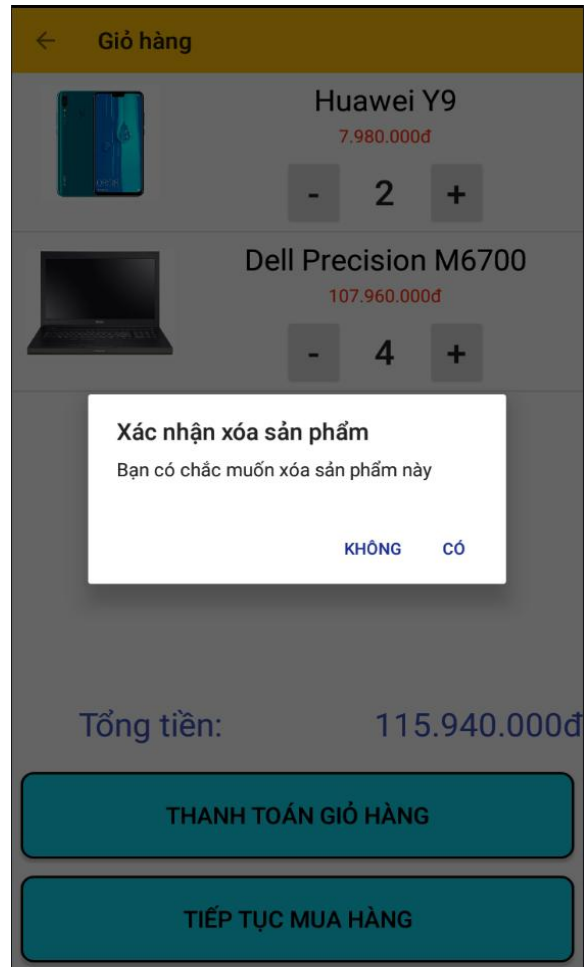


13.2 Code

```
private void EventButton() {  
    btntieptucmua.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            Intent intent = new Intent(getApplicationContext(), MainActivity.class);  
            startActivity(intent);  
        }  
    });  
    btnthanhtoan.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            if (MainActivity.manggiohang.size() > 0){  
                Intent intent = new Intent(getApplicationContext(), Thongtinhachhang.class);  
                startActivity(intent);  
            }else {  
                CheckConnection.ShowToast_Short(getApplicationContext(), thôngbao: "Giỏ hàng của bạn chưa có sản phẩm để thanh toán");  
            }  
        }  
    });  
}
```

Màn hình danh sách đơn hàng

13.3 Giao diện màn hình



13.4 Code

```
private void CatchOnItemListView() {
    lvgiohang.setOnItemLongClickListener(new AdapterView.OnItemClickListener() {
        @Override
        public boolean onItemLongClick(AdapterView<?> parent, View view, final int position, long id) {
            AlertDialog.Builder builder = new AlertDialog.Builder(context: GiohangActivity.this);
            builder.setTitle("Xác nhận xóa sản phẩm");
            builder.setMessage("Bạn có chắc muốn xóa sản phẩm này");
            builder.setPositiveButton(text: "CÓ", new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int which) {
                    if (MainActivity.manggiohang.size() <= 0){
                        txtthongbao.setVisibility(View.VISIBLE);
                    }else {
                        MainActivity.manggiohang.remove(position);
                        giohangAdapter.notifyDataSetChanged();
                        EvenUtil();
                        if (MainActivity.manggiohang.size() <= 0){
                            txtthongbao.setVisibility(View.VISIBLE);
                        }else {
                            txtthongbao.setVisibility(View.INVISIBLE);
                            giohangAdapter.notifyDataSetChanged();
                            EvenUtil();
                        }
                    }
                }
            });
            builder.setNegativeButton(text: "Không", new DialogInterface.OnClickListener() {
```

10. Màn hình thông tin khách hàng

13.1 Giao diện màn hình

13.2 Code

```
private void EventButton() {  
    btnxacnhan.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            final String ten = edtttenkhachhang.getText().toString().trim();  
            final String sdt = edtsdt.getText().toString().trim();  
            final String email = edtemail.getText().toString().trim();  
            if (ten.length() > 0 && sdt.length() > 0 && email.length() > 0){  
                RequestQueue requestQueue = Volley.newRequestQueue(getApplicationContext());  
                StringRequest stringRequest = new StringRequest(Request.Method.POST, Server.Duongdandonhang, new Response.Listener<String>() {  
                    @Override  
                    public void onResponse(final String madonhang) {  
                        Log.d( tag: "madonhang",madonhang);  
                        if (Integer.parseInt(madonhang) > 0){  
                            RequestQueue queue = Volley.newRequestQueue(getApplicationContext());  
                            StringRequest request = new StringRequest(Request.Method.POST, Server.Duongdanchitietdonhang, new Response.Listener<String>() {  
                                @Override  
                                public void onResponse(String response) {  
                                    if (response.equals("1")){  
                                        MainActivity.manggiohang.clear();  
                                        CheckConnection.ShowToast_Short(getApplicationContext(), thongbao: "Bạn đã thêm dữ liệu giỏ hàng thành công");  
                                        Intent intent = new Intent(getApplicationContext(),MainActivity.class);  
                                        startActivity(intent);  
                                        CheckConnection.ShowToast_Short(getApplicationContext(), thongbao: "Mời bạn tiếp tục mua hàng");  
                                    }else {  
                                        CheckConnection.ShowToast_Short(getApplicationContext(), thongbao: "Dữ liệu giỏ hàng của bạn đã bị lỗi");  
                                    }  
                                }  
                            }  
                        }  
                    }  
                }  
            }  
        }  
    }  
}
```

```

    }
}, new Response.ErrorListener() {
    @Override
    public void onErrorResponse(VolleyError error) {

    }
}){
    @Override
    protected Map<String, String> getParams() throws AuthFailureError {
        JSONArray jsonArray = new JSONArray();
        for (int i = 0 ; i < MainActivity.manggiohang.size() ; i++){
            JSONObject jsonObject = new JSONObject();
            try {
                jsonObject.put( name: "madonhang", madonhang);
                jsonObject.put( name: "masanpham", MainActivity.manggiohang.get(i).getIdsp());
                jsonObject.put( name: "tensanpham", MainActivity.manggiohang.get(i).getTensp());
                jsonObject.put( name: "giasanpham", MainActivity.manggiohang.get(i).getGiasp());
                jsonObject.put( name: "soluongsanpham", MainActivity.manggiohang.get(i).getSoluongsp());
            } catch (JSONException e) {
                e.printStackTrace();
            }
            jsonArray.put(jsonObject);
        }
        HashMap<String,String> hashMap = new HashMap<String, String>();
        hashMap.put("json", jsonArray.toString());
        return hashMap;
    }
};
queue.add(request);
}

```

11. Màn hình liên hệ

13.1 Giao diện màn hình



13.2 Code

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_Lien_he);
    Anhxa();
    ActionToolbar();
}

private void ActionToolbar() {
    setSupportActionBar(toolbarlienhe);
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);
    toolbarlienhe.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            finish();
        }
    });
}

private void Anhxa() {
    toolbarlienhe = (Toolbar) findViewById(R.id.toolbarlienhe);
}
```

12. Màn hình Maps

13.1 Giao diện màn hình



13.2 Code

```
@Override
public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;
    if (ActivityCompat.checkSelfPermission( context: this, Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission
        // TODO: Consider calling
        //   ActivityCompat#requestPermissions
        // here to request the missing permissions, and then overriding
        //   public void onRequestPermissionsResult(int requestCode, String[] permissions,
        //                                           int[] grantResults)
        // to handle the case where the user grants the permission. See the documentation
        // for ActivityCompat#requestPermissions for more details.
        return;
    }
    mMap.setMyLocationEnabled(true);

    // Add a marker in Sydney and move the camera
    LatLng TDC = new LatLng( v: 10.854906, v1: 106.765242);
    mMap.addMarker(new MarkerOptions().position(TDC).title("Thủ Đức").snippet("53 Võ Văn Ngân - Phường Linh Chiểu").icon(BitmapDescriptorFactory.defaultMarker()));
    mMap.setMapType(GoogleMap.MAP_TYPE_NORMAL);
    CameraPosition cameraPosition = new CameraPosition.Builder().target(TDC).zoom(90).build();
    mMap.animateCamera(CameraUpdateFactory.newCameraPosition(cameraPosition));
}
```

CHƯƠNG VIII. KẾT QUẢ ĐẠT ĐƯỢC

1.Kết quả đạt được

- Hiển thị được danh sách các sản phẩm từ cơ sở dữ liệu
- Ứng dụng navigation trong app
- Hiển thị được các sản phẩm theo danh mục
- Sử dụng animation trong app
- Thêm , xóa , sửa, hiển thị sản phẩm
- Sử dụng chọn ảnh từ thư mục và chụp ảnh từ camera
- Sử dụng CustomSpinner
- Hiển thị biểu đồ thống kê

2.Kết luận

- Những mặt đã làm được :
 - +Kết nối database vào app
 - +Sử dụng được những cardview, recyclerview, bản đồ... trong app
- Những mặt chưa làm được :
 - +Chưa tìm kiếm được sản phẩm theo tên