

**BỘ NÔNG NGHIỆP VÀ PHÁT TRIỂN NÔNG THÔN
TRƯỜNG ĐẠI HỌC THỦY LỢI**



BÀI TẬP LỚN

**HỌC PHẦN: PHÁT TRIỂN ỨNG DỤNG CHO CÁC THIẾT BỊ
DI ĐỘNG**

**ĐỀ TÀI: Xây dựng ứng dụng Ôn thi lái xe máy
(Java Android App)**

Mã Sinh Viên	Họ và Tên	Ngày Sinh	Điểm	
			Bảng Số	Bảng Chữ
2051063778	Phạm Công Khanh	03/01/2002		
2051063799	Đinh Quang Thao	02/11/2002		
2151062761	Lê Văn Hà	16/02/2003		
2151060279	Nguyễn Tiến Lực	06/09/2003		

CÁN BỘ CHẤM THI 1

CÁN BỘ CHẤM THI 2

Hà Nội, năm 2024

LỜI NÓI ĐẦU

Trong bối cảnh cách mạng công nghiệp 4.0 đang phát triển mạnh mẽ, công nghệ thông tin và truyền thông đã trở thành những yếu tố then chốt trong việc nâng cao chất lượng cuộc sống. Đặc biệt, sự bùng nổ của các thiết bị di động thông minh cùng với sự phổ biến của internet đã mở ra nhiều cơ hội mới trong lĩnh vực giáo dục và đào tạo. Những ứng dụng học tập trên điện thoại di động không chỉ giúp tiết kiệm thời gian mà còn mang lại hiệu quả học tập vượt trội nhờ tính tiện lợi và khả năng tùy biến cao.

Ứng dụng "Ôn thi giấy phép lái xe máy" ra đời từ nhu cầu thực tế của người dùng trong việc ôn tập và chuẩn bị cho kỳ thi lấy giấy phép lái xe máy. Đây là một yêu cầu bắt buộc đối với bất kỳ ai muốn tham gia giao thông an toàn và hợp pháp. Tuy nhiên, không phải ai cũng có thời gian và điều kiện để tham gia các lớp học truyền thống hoặc ôn tập theo cách truyền thống. Chính vì vậy, việc phát triển một ứng dụng ôn thi trên nền tảng di động không chỉ là một giải pháp tối ưu mà còn là một xu hướng tất yếu của thời đại.

Ứng dụng "Ôn thi giấy phép lái xe máy" được xây dựng với mục tiêu cung cấp cho người dùng một công cụ ôn tập hiệu quả, chính xác và tiện lợi nhất. Ứng dụng tích hợp các bộ đề thi mới nhất, cập nhật liên tục theo quy định của Bộ Giao thông Vận tải, giúp người dùng làm quen với cấu trúc đề thi và các dạng câu hỏi thường gặp. Bên cạnh đó, ứng dụng còn cung cấp các tính năng hỗ trợ như chế độ thi thử, ôn tập lý thuyết, cung cấp thông tin các loại biển báo, và tra cứu luật để người dùng có thể tham gia giao thông an toàn và đúng luật.

Không chỉ dừng lại ở việc cung cấp nội dung ôn tập, ứng dụng còn chú trọng đến trải nghiệm người dùng. Giao diện của ứng dụng được thiết kế đơn giản, thân thiện, dễ sử dụng, phù hợp với mọi đối tượng từ học sinh, sinh viên đến người đi làm. Người dùng có thể ôn tập bất cứ lúc nào, bất cứ nơi đâu, chỉ cần có một chiếc điện thoại thông minh kết nối internet.

Chúng em hy vọng rằng, với sự hỗ trợ của ứng dụng "Ôn thi giấy phép lái xe máy", việc ôn tập và thi lấy giấy phép lái xe máy sẽ trở nên dễ dàng và hiệu quả hơn bao giờ hết. Qua đó, góp phần nâng cao nhận thức và kỹ năng tham gia giao thông an toàn cho cộng đồng, giảm thiểu tai nạn giao thông và xây dựng một môi trường giao thông an toàn, văn minh.

PHÂN CÔNG CÔNG VIỆC

Thành viên	Công việc
Phạm Công Khanh (Nhóm trưởng)	<p>Thực hiện thiết kế giao diện và code các chức năng:</p> <ul style="list-style-type: none"> • Màn hình chính • Thanh điều hướng • Thi thử sát hạch • Biển báo (Biển báo cấm) • Tra cứu luật (Màn hình chính của Tra cứu luật; Nồng độ cồn, chất kích thích; Hiệu lệnh, chỉ dẫn) • Code php dữ liệu API, thiết kế các file cơ sở dữ liệu law.json, learning.json, sign.json <p>Soạn báo cáo Word, vẽ Figma, đẩy dự án lên Github</p>
Nguyễn Tiến Lực	<p>Thực hiện thiết kế giao diện và code chức năng:</p> <ul style="list-style-type: none"> • Học lý thuyết <p>Vẽ Figma</p>
Lê Văn Hà	<p>Thực hiện thiết kế giao diện và code chức năng:</p> <ul style="list-style-type: none"> • Biển báo (Biển báo nguy hiểm; Biển báo hiệu lệnh) <p>Vẽ Figma</p>
Đinh Quang Thao	<p>Thực hiện thiết kế giao diện và code chức năng:</p> <ul style="list-style-type: none"> • Tra cứu luật (Dừng xe đỗ xe; Tốc độ, khoảng cách an toàn; Đường cấm, đường một chiều) <p>Vẽ Figma</p>

MỤC LỤC

LỜI NÓI ĐẦU	2
PHÂN CÔNG CÔNG VIỆC	3
CHƯƠNG 1. MÔ TẢ BÀI TOÁN	7
1.1. Giới thiệu	7
1.2. Chức năng chính	7
1.3. Yêu cầu phi chức năng	7
CHƯƠNG 2. PHÂN TÍCH YÊU CẦU VÀ THIẾT KẾ HỆ THỐNG	8
2.1. Phân tích yêu cầu:	8
a) Xác định người dùng:.....	8
b) Thu thập yêu cầu:.....	8
c) Phân tích yêu cầu:	8
2.2. Thiết kế hệ thống:	8
a) Xác định các lớp:	8
b) Môi quan hệ giữa các lớp:	10
c) Biểu đồ lớp cho mô hình miền:.....	10
d) Thiết kế giao diện:	12
2.3. Triển khai:	12
2.4. Vận hành và bảo trì:.....	13
CHƯƠNG 3. KẾT QUẢ THỰC HIỆN	14
3.1. Công nghệ đã sử dụng	14
3.2. Tiến độ thực hiện	14
Bước 1. Tạo dự án mới:	14
Bước 2. Tạo các package:	14
Bước 3. Tạo các lớp:	15
Bước 4. Cài đặt các thư viện:	16
Bước 5. Viết code:.....	16
Bước 6. Thiết lập cơ sở dữ liệu:	16
Bước 7. Phát triển giao diện người dùng:	17
Bước 8. Chạy và kiểm thử:	17
Bước 9. Triển khai:.....	17

3.3. Hình ảnh sản phẩm	17
a. Phạm Công Khanh:	17
b. Nguyễn Tiến Lực:	29
c. Lê Văn Hà:	30
d. Đinh Quang Thao	32
KẾT LUẬN	35
DANH MỤC TÀI LIỆU THAM KHẢO	36
PHỤ LỤC	37

MỤC LỤC HÌNH ẢNH

<u>Hình 1: Giao diện màn hình chính</u>	18
<u>Hình 2: Giao diện thanh điều hướng (NavigationDrawer)</u>	19
<u>Hình 3: Giao diện Thi thử sát hạch</u>	20
<u>Hình 4: Giao diện hiển thị lên dialog khi nhấn vào nút “Làm bài”</u>	21
<u>Hình 5: Giao diện làm bài thi thử</u>	22
<u>Hình 6: Giao diện hiển thị dialog khi nộp bài với kết quả TRƯỢT</u>	23
<u>Hình 7: Giao diện hiển thị dialog khi nộp bài với kết quả ĐẠT</u>	24
<u>Hình 8: Giao diện biển báo cấm</u>	25
<u>Hình 9: Giao diện tra cứu luật</u>	26
<u>Hình 10: Giao diện Nồng độ cồn, chất kích thích</u>	27
<u>Hình 11: Giao diện Hiệu lệnh, chỉ dẫn</u>	28
<u>Hình 12: Giao diện Học lý thuyết</u>	29
<u>Hình 13: Giao diện Biển báo nguy hiểm</u>	30
<u>Hình 14: Giao diện biển báo hiệu lệnh</u>	31
<u>Hình 15: Giao diện luật Dừng xe, đỗ xe</u>	32
<u>Hình 16: Giao diện luật Tốc độ, khoảng cách an toàn</u>	33
<u>Hình 17: Đường cấm, đường một chiều</u>	34

CHƯƠNG 1. MÔ TẢ BÀI TOÁN

1.1. Giới thiệu

Ứng dụng "Ôn thi giấy phép lái xe máy" là một công cụ hỗ trợ người dùng ôn tập và thi thử để chuẩn bị cho kỳ thi sát hạch lái xe máy. Ứng dụng này được xây dựng bằng Java và phát triển trên nền tảng Android, nhằm cung cấp cho người dùng một phương tiện học tập tiện lợi, hiệu quả và chính xác.

1.2. Chức năng chính

Ứng dụng "Ôn thi giấy phép lái xe máy" có các chức năng chính sau:

- **Thi thử sát hạch:** Cung cấp 5 bộ đề thi, mỗi đề gồm 25 câu hỏi và thời gian làm bài là 19 phút. Người dùng cần trả lời đúng ít nhất 21/25 câu để đạt yêu cầu, nếu không sẽ bị trượt.
- **Ôn tập lý thuyết:** Hiện thị toàn bộ các câu hỏi và đáp án chính xác, giúp người dùng nắm vững kiến thức.
- **Biển báo:** Cung cấp thông tin về các loại biển báo, bao gồm biển báo cấm, biển báo nguy hiểm và biển báo hiệu lệnh, kèm theo ý nghĩa của từng loại biển báo.
- **Tra cứu luật:** Cho phép người dùng tra cứu thông tin từ 5 bộ luật, gồm: Nồng độ cồn, chất kích thích; Hiệu lệnh, chỉ dẫn; Dừng xe, đỗ xe; Tốc độ, khoảng cách an toàn; Đường cấm, đường một chiều.

1.3. Yêu cầu phi chức năng

- **Dễ sử dụng:** Giao diện của ứng dụng cần được thiết kế thân thiện, dễ hiểu và dễ sử dụng cho mọi đối tượng người dùng.
- **Hiệu năng:** Ứng dụng cần hoạt động nhanh chóng, mượt mà và hiệu quả, ngay cả khi có nhiều công việc.
- **Độ tin cậy:** Dữ liệu cần được lưu trữ an toàn và bảo mật, đảm bảo không bị mất mát hoặc xâm nhập trái phép.

CHƯƠNG 2. PHÂN TÍCH YÊU CẦU VÀ THIẾT KẾ HỆ THỐNG

2.1. Phân tích yêu cầu:

a) Xác định người dùng:

- Người dùng cuối là bất kỳ ai muốn ôn thi giấy phép lái xe máy. Họ có thể là học sinh, sinh viên, nhân viên văn phòng, hoặc bất kỳ ai cần chuẩn bị cho kỳ thi sát hạch lái xe máy.
- Người dùng có thể có các mức độ am hiểu về công nghệ khác nhau, do đó ứng dụng cần dễ sử dụng và thân thiện với người dùng.

b) Thu thập yêu cầu:

- Dựa trên mô tả bài toán, ta đã xác định được các chức năng chính (thi thử sát hạch, ôn tập lý thuyết, biển báo, tra cứu luật) và yêu cầu phi chức năng (dễ sử dụng, hiệu năng, độ tin cậy) của ứng dụng.

c) Phân tích yêu cầu:

- **Thi thử sát hạch:** Cung cấp 5 bộ đề thi, mỗi đề gồm 25 câu hỏi với thời gian làm bài là 19 phút. Người dùng cần trả lời đúng ít nhất 21/25 câu để đạt yêu cầu. Chức năng này giúp người học làm quen với cấu trúc đề thi và đánh giá khả năng của mình.
- **Câu hỏi:** Cung cấp, cập nhật tài liệu câu hỏi ôn thi bao gồm: id, nội dung câu hỏi, các phương án của câu hỏi và đáp án chính xác. Thực hiện chức năng hiển thị câu hỏi.
- **Ôn tập lý thuyết:** Hiện thị toàn bộ các câu hỏi và đáp án chính xác, giúp người dùng nắm vững kiến thức lý thuyết. Chức năng này hỗ trợ người dùng trong việc học và ôn tập một cách có hệ thống.
- **Biển báo:** Cung cấp thông tin về các loại biển báo, bao gồm biển báo cấm, biển báo nguy hiểm và biển báo hiệu lệnh, kèm theo ý nghĩa của từng loại biển báo. Chức năng này giúp người dùng hiểu rõ hơn về các quy định giao thông.
- **Tra cứu luật:** Cho phép người dùng tra cứu thông tin từ 5 bộ luật, gồm: nồng độ cồn, chất kích thích; hiệu lệnh, chỉ dẫn; dừng xe, đỗ xe; tốc độ, khoảng cách an toàn; đường cấm, đường một chiều. Chức năng này giúp người dùng nắm rõ các quy định pháp luật liên quan đến giao thông.

2.2. Thiết kế hệ thống:

a) Xác định các lớp:

- **Lớp Exam (Thi thử sát hạch):**
 - Thuộc tính:
 - String namedethi: Tên đề thi
 - int imagedethi: Ảnh biểu tượng cho đề thi
 - int soluongCauhoi: Số lượng câu hỏi

- int thoigian: Thời gian làm bài
- Phương thức:
 - getNamedethi, setNamedethi
 - getImagedethi, setImagedethi
 - getSoluongCauhoi, setSoluongCauhoi
 - getThoigian, setThoigian
- **Lớp Question (Câu hỏi):**
 - Thuộc tính:
 - int id: Mã câu hỏi
 - String question: Câu hỏi
 - String ideaA: Câu hỏi A
 - String ideaB: Câu hỏi B
 - String ideaC: Câu hỏi C
 - String ideaD: Câu hỏi D
 - int answer: Đáp án của câu hỏi
 - int userAnswer: Đáp án người dùng chọn
 - Phương thức:
 - getUserAnswer, setUserAnswer
 - isCorrect
 - getId, setId
 - getQuestion, setQuestion
 - getIdeaA, setIdeaA
 - getIdeaB, setIdeaB
 - getIdeaC, setIdeaC
 - getIdeaD, setIdeaD
 - getAnswer, setAnswer
 - toString
- **Lớp Learning (Ôn tập lý thuyết):**
 - Thuộc tính:
 - String maCauHoi: Mã câu hỏi

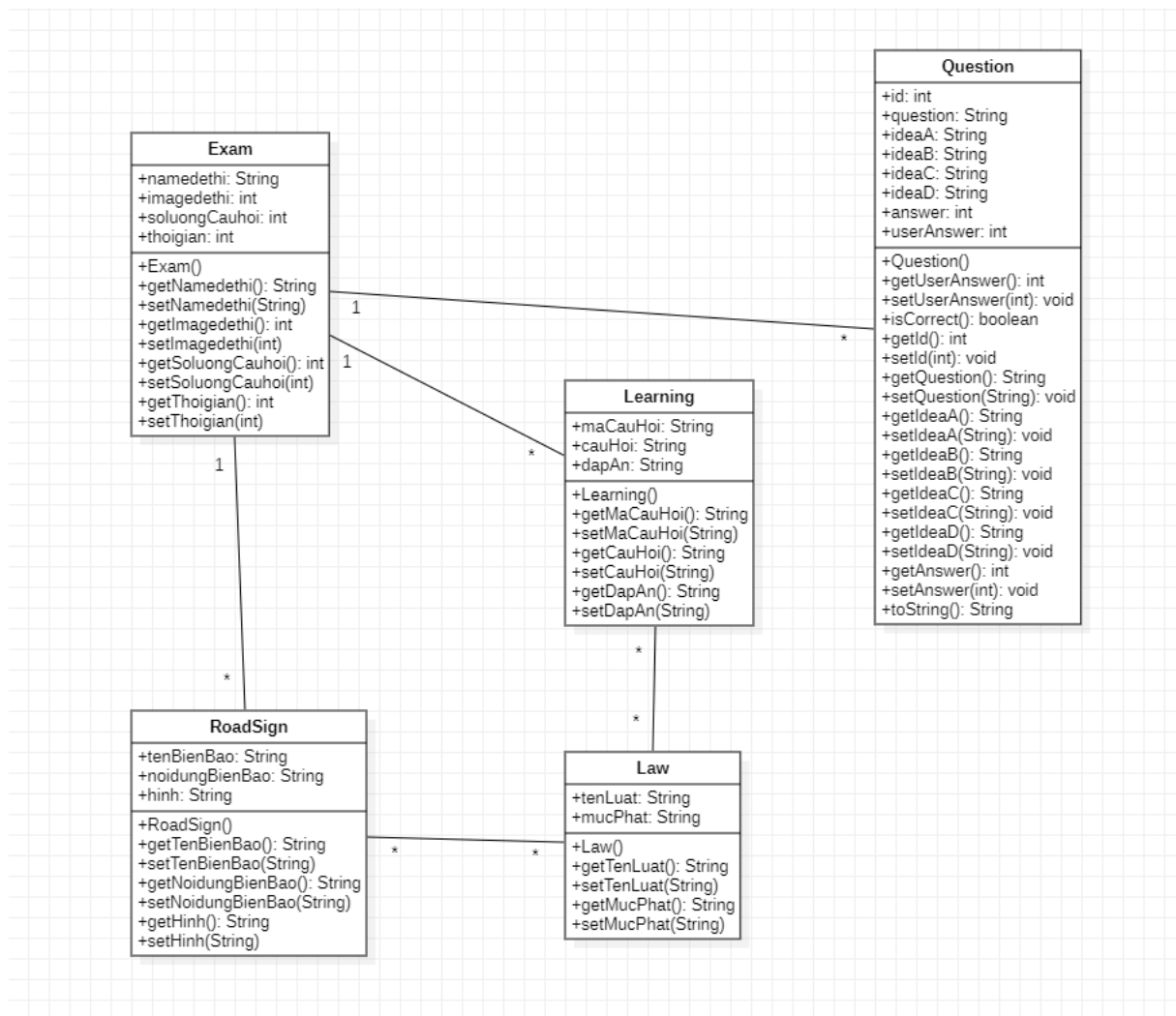
- String cauHoi: Câu hỏi
- String dapAn: Đáp án
- Phương thức:
 - getMaCauHoi, setMaCauHoi
 - getCauHoi, setCauHoi
 - getDapAn, setDapAn
- **Lớp RoadSign (Biển báo):**
 - Thuộc tính:
 - String tenBienBao: Tên biển báo
 - String noidungBienBao: Nội dung biển báo
 - String hinh: Hình biển báo
 - Phương thức:
 - getTenBienBao, setTenBienBao
 - getNoidungBienBao, setNoidungBienBao
 - getHinh, setHinh
- **Lớp Law (Tra cứu luật):**
 - Thuộc tính:
 - String tenLuot: Tên luật
 - String mucPhat: Mức phạt nếu vi phạm luật
 - Phương thức:
 - getTenLuot, setTenLuot
 - getMucPhat, setMucPhat

b) Mối quan hệ giữa các lớp:

- **Exam - Learning:** Mỗi đề thi (**Exam**) bao gồm nhiều câu hỏi (**Learning**). Đây là mối quan hệ **Một-Nhiều**.
- **Exam – Question** có liên kết theo dạng **Một-Nhiều**. Một **Exam** có thể chứa nhiều **Question**.
- **Learning - RoadSign - Law:** Mỗi câu hỏi (**Learning**) có thể liên quan đến một biển báo (**RoadSign**) hoặc một luật giao thông (**Law**). Các mối quan hệ này có thể là **Nhiều-Nhiều** nếu một câu hỏi liên quan đến nhiều luật hoặc biển báo.

c) Biểu đồ lớp cho mô hình miền:

- Tập trung vào các thực thể nghiệp vụ và mối quan hệ giữa chúng.
- Thường không chứa các lớp liên quan đến giao diện người dùng, cơ sở dữ liệu, hay các chi tiết kỹ thuật khác.
- Trong ứng dụng "Ôn thi giấy phép lái xe máy", biểu đồ lớp cho mô hình miền chứa các thực thể như:
 - **Exam**: các thuộc tính bao gồm namedethi, imagedethi, soluongCauhoi, thoigian. Tạo ra 5 đề, mỗi đề có một bộ câu hỏi. Bao gồm tên đề, số lượng câu hỏi, thời gian làm bài, điểm số. Thực hiện chức năng làm bài thi.
 - **Question**: các thuộc tính bao gồm id, question, ideaA, ideaB, ideaC, ideaD, answer, userAnswer. Thực hiện chức năng hiển thị lên câu hỏi cho đề thi.
 - **Learning**: các thuộc tính bao gồm maCauHoi, cauHoi, dapAn. Thực hiện chức năng hiển thị các câu hỏi để ôn tập và cung cấp đáp án cho từng câu hỏi.
 - **RoadSign**: các thuộc tính bao gồm tenBienBao, noidungBienBao, hinh. Thực hiện chức năng hiển thị thông tin và ý nghĩa của từng loại biển báo.
 - **Law**: các thuộc tính bao gồm tenLuot, mucPhat. Thực hiện chức năng tra cứu luật và thông tin các hình phạt nếu vi phạm.



d) Thiết kế giao diện:

- Hiện thị menu chính với các lựa chọn:
 - Thi thử sát hạch
 - Ôn tập lý thuyết
 - Biển báo
 - Tra cứu luật
 - Cột điều hướng (NavigationDrawer)
- Sử dụng các thông báo rõ ràng để hướng dẫn người dùng lựa chọn.

2.3. Triển khai:

- **Viết code:** Sử dụng Java để cài đặt các class trong mô hình MVC, đọc/ghi file, xử lý dữ liệu và hiển thị giao diện.
 - Cài đặt các class cho các thực thể như DeThiSatHach, Question, BienBao,

và BoLuat.

- Xử lý dữ liệu thi thử sát hạch, ôn tập lý thuyết, biển báo và tra cứu luật.
- Phát triển các tính năng như thi thử, ôn tập lý thuyết, và hiển thị thông tin biển báo, luật giao thông.
- **Kiểm thử:** Thực hiện các bài kiểm thử đơn vị (unit test) và kiểm thử tích hợp (integration test) để đảm bảo tính chính xác và tin cậy của ứng dụng.

2.4. Vận hành và bảo trì:

- Cài đặt và triển khai:
 - Hướng dẫn người dùng cách chạy ứng dụng trên Android Studio.
 - Giải thích cách thiết lập môi trường phát triển và cài đặt các thư viện cần thiết.
- Bảo trì: Sửa lỗi phát sinh, cập nhật chức năng mới (nếu có) và cải thiện hiệu năng của ứng dụng.

CHƯƠNG 3. KẾT QUẢ THỰC HIỆN

3.1. Công nghệ đã sử dụng

- Ngôn ngữ lập trình: Java
- Công cụ: Android Studio
- Thư viện và công cụ hỗ trợ:
 - Retrofit: để thực hiện các kết nối API cho việc truy xuất dữ liệu.
 - Gson: chuyển đổi giữa các đối tượng Java và các chuỗi JSON.
 - Các thư viện hỗ trợ khác: thư viện đọc/ghi file, xử lý dữ liệu JSON.

3.2. Tiến độ thực hiện

Link github tới dự án: https://github.com/khanhtlu2020/CSE441_PROJECT

Hướng dẫn các bước đã thực hiện:

Bước 1. Tạo dự án mới:

- Mở Android Studio.
- Chọn "File/New/New Project...".
- Chọn "Empty Views Activity" (hoặc loại activity mà bạn muốn).
- Nhấn "Next".
- Nhập thông tin dự án:
 - Name: Nhập tên dự án (ví dụ: "DrivingTestApp").
 - Package name: Nhập tên gói (ví dụ: "com.example.drivingtestapp").
 - Save location: Chọn vị trí lưu trữ dự án.
 - Language: Chọn ngôn ngữ lập trình Java.
 - Minimum SDK: Chọn SDK tối thiểu (ví dụ: API 24 hoặc mới hơn).
 - Build configuration language: Chọn Kotlin DSL (khuyến nghị).
- Nhấn "Finish" để tạo dự án.

Bước 2. Tạo các package:

- Trong cửa sổ "Project", click chuột phải vào thư mục "src".
- Chọn "New" -> "Package".
- Tạo các package sau:

- Model: chứa các lớp mô hình như Exam, Question, Learning, RoadSign, Law.
- View: chứa các Activity hoặc Fragment.
- Nhấn "OK" để tạo package.

Bước 3. Tạo các lớp:

- Trong mỗi package, click chuột phải và chọn "New" -> "Java Class" để tạo các lớp tương ứng:
 - Question
 - Exam
 - ExamActivity
 - ExamActivity1
 - ExamActivity2
 - ExamActivity3
 - ExamActivity4
 - ExamListActivity
 - Law
 - LawActivity
 - LawActivity1
 - LawActivity2
 - LawActivity3
 - LawActivity4
 - LawAdapter
 - LawListActivity
 - Learning
 - LearningActivity
 - LearningAdapter

- MainActivity
 - MyArrayAdapter
 - QuestionFragment
 - RoadSign
 - RoadSignActivity
 - RoadSignAdapter
 - ViewPagerAdapter
- Cài đặt các thuộc tính và phương thức cho từng lớp dựa trên thiết kế đã phân tích.
 - Khai báo các activity trong AndroidManifest.xml.

Bước 4. Cài đặt các thư viện:

- Mở file build.gradle.kts của Module và thêm các dependency cho Retrofit, Gson, Espresso.
- Đồng bộ lại dự án để tải về và cài đặt các thư viện.

Bước 5. Viết code:

- Bắt đầu viết code cho từng lớp, thực hiện các chức năng của ứng dụng:
 - **Exam, Question, Learning, RoadSign, Law:** Cài đặt các thuộc tính và phương thức getter, setter cơ bản.
 - **MainActivity, ExamListActivity, ExamActivity, ExamActivity1, ExamActivity2, ExamActivity3, ExamActivity4, LearningActivity, RoadSignActivity, LawActivity, LawActivity1, LawActivity2, LawActivity3, LawActivity4:** cài đặt logic xử lý yêu cầu của người dùng, điều phối các lớp khác để hoàn thành các chức năng.
 - **MyArrayAdapter, LearningAdapter, RoadSignAdapter, LawAdapter, QuestionFragment, ViewPagerAdapter:** Cài đặt các phương thức để kết nối, đọc, ghi dữ liệu từ file JSON/ApiService.

Bước 6. Thiết lập cơ sở dữ liệu:

- Tạo lớp java **ApiService** để cung cấp các phương thức tương tác với API Web trong ứng dụng Android: Nhấp chuột phải vào package muốn tạo, "New" -

> "Java Class" -> Interface và đặt tên là ApiService. Sau đó tiếp tục tạo ApiService1, ApiService2, ApiService3, ApiService4 để tạo ra mỗi đề một bộ câu hỏi.

- Tạo các cơ sở dữ liệu dưới dạng JSON (law.json, learning.json, sign.json): Nhấp chuột phải vào package muốn tạo, "New" -> "Folder" -> "Assets Folder" và chọn Finish. Tiếp tục thêm vào các file json.
- Tạo file xml network_security_config.xml để cấu hình bảo mật mạng cho ứng dụng Android. Sau đó khai báo trong AndroidManifest.xml.

Bước 7. Phát triển giao diện người dùng:

- Sử dụng XML để thiết kế giao diện cho các màn hình thi thử sát hạch, ôn tập lý thuyết, biển báo và tra cứu luật.
- Sử dụng XML để thiết kế NavigationDrawer (thanh điều hướng).

Bước 8. Chạy và kiểm thử:

- Chạy ứng dụng từ Android Studio để kiểm tra các chức năng.
- Kiểm thử các tính năng chính:
 - Thi thử sát hạch
 - Ôn tập lý thuyết
 - Biển báo
 - Tra cứu luật
 - Thanh điều hướng kết nối (intent) giữa các activity với nhau.
- Kiểm tra các chức năng của ứng dụng, sửa lỗi và hoàn thiện code.

Bước 9. Triển khai:

- Nếu muốn chia sẻ ứng dụng, sử dụng chức năng "Build Bundle(s) / APK(s)" để tạo file .apk.
- Chia sẻ file .apk cho người dùng để cài đặt và sử dụng trên thiết bị Android.

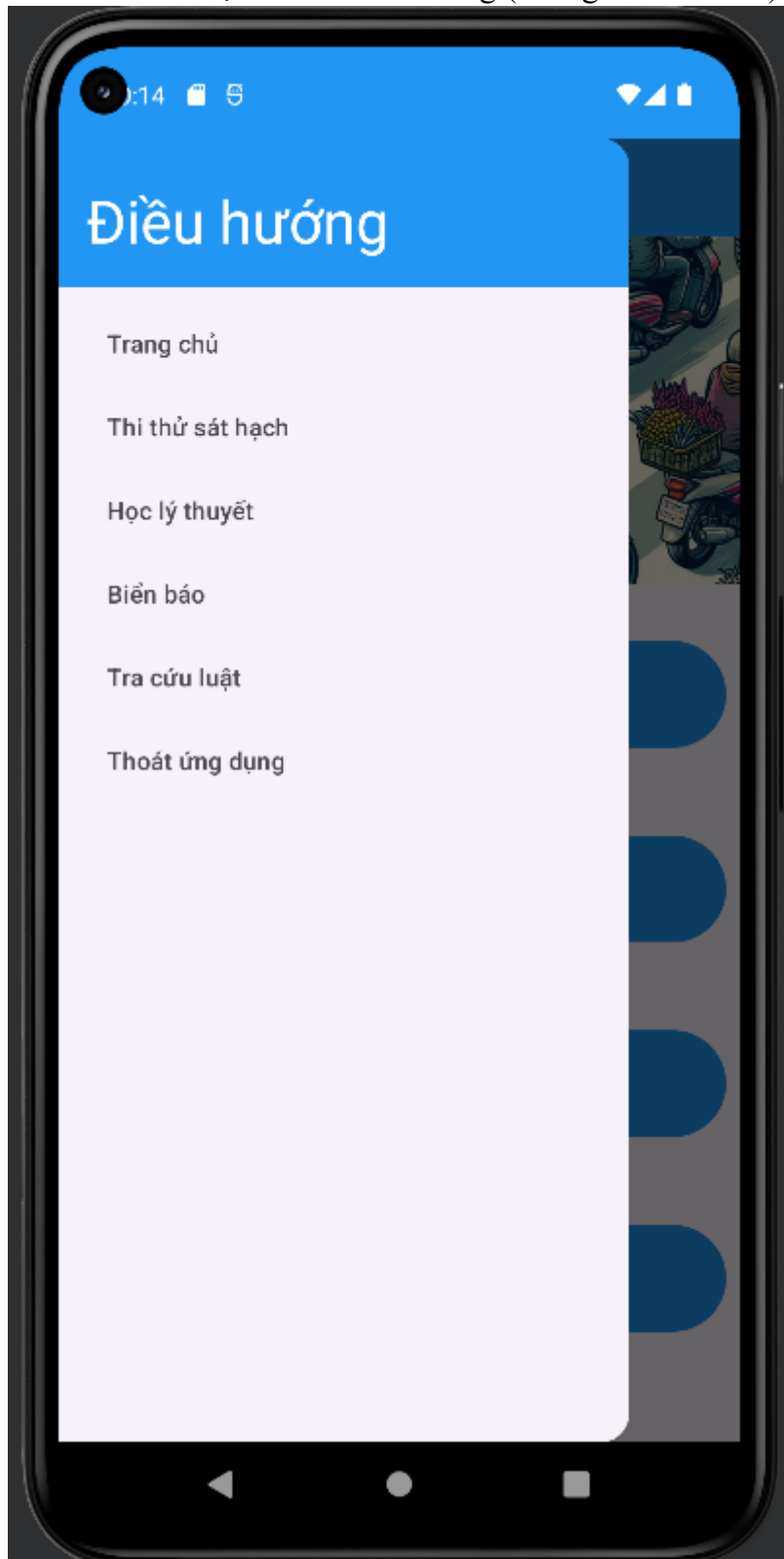
3.3. Hình ảnh sản phẩm

a. Phạm Công Khanh:

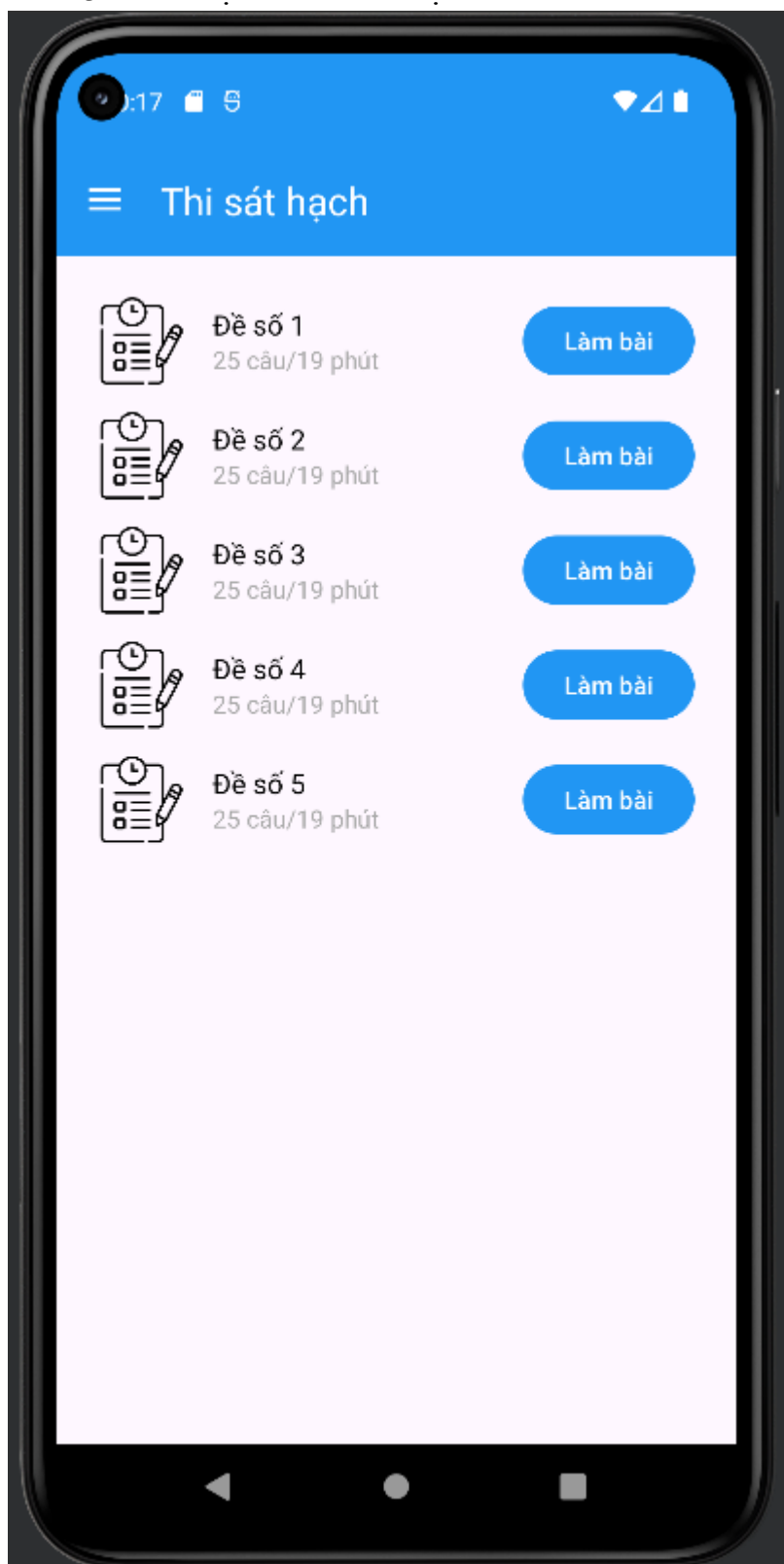
- Hình 1: Giao diện màn hình chính



- Hình 2: Giao diện thanh điều hướng (NavigationDrawer)



- Hình 3: Giao diện Thi thử sát hạch



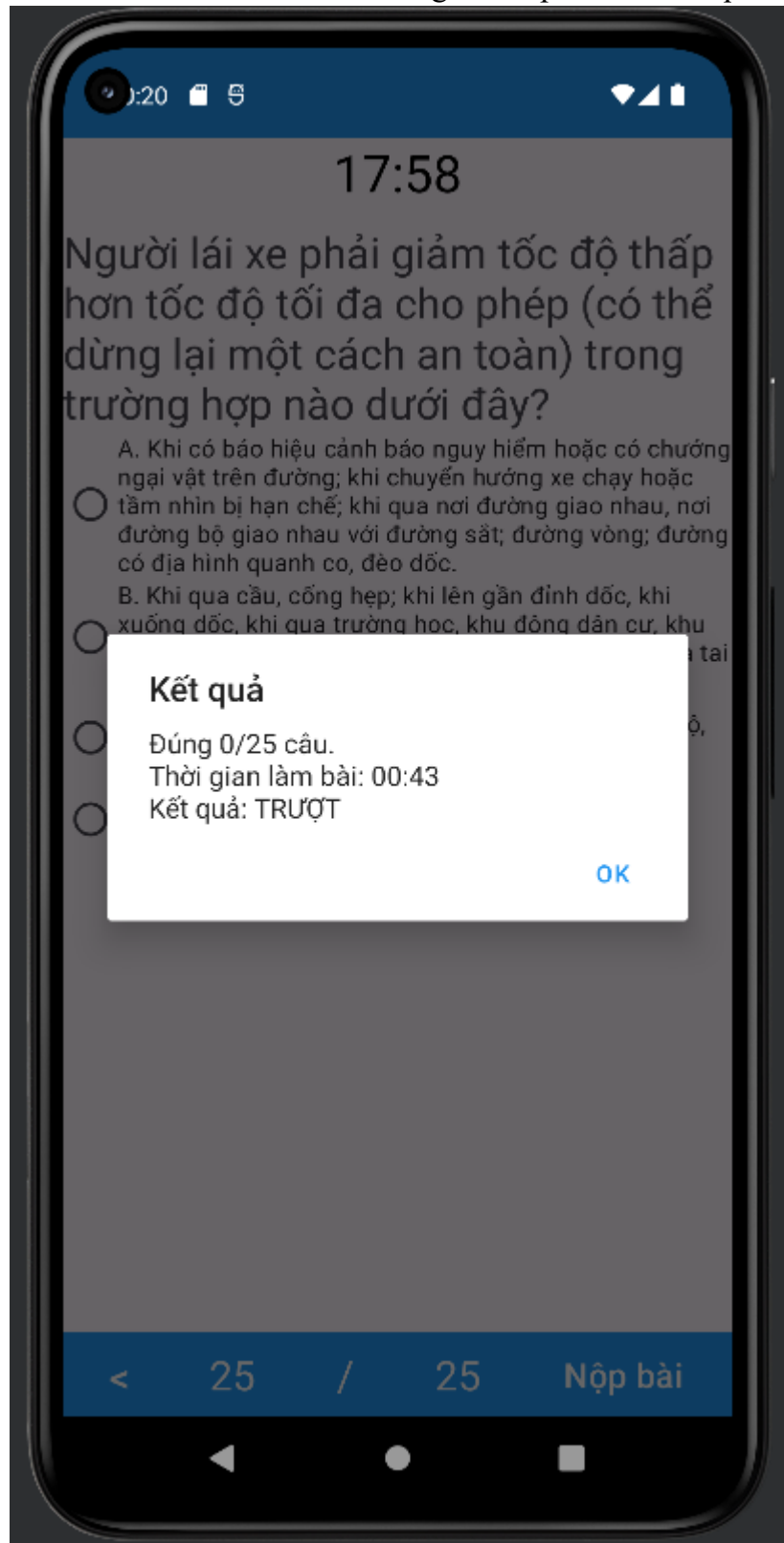
- Hình 4: Giao diện hiển thị lên dialog khi nhấn vào nút “Làm bài”



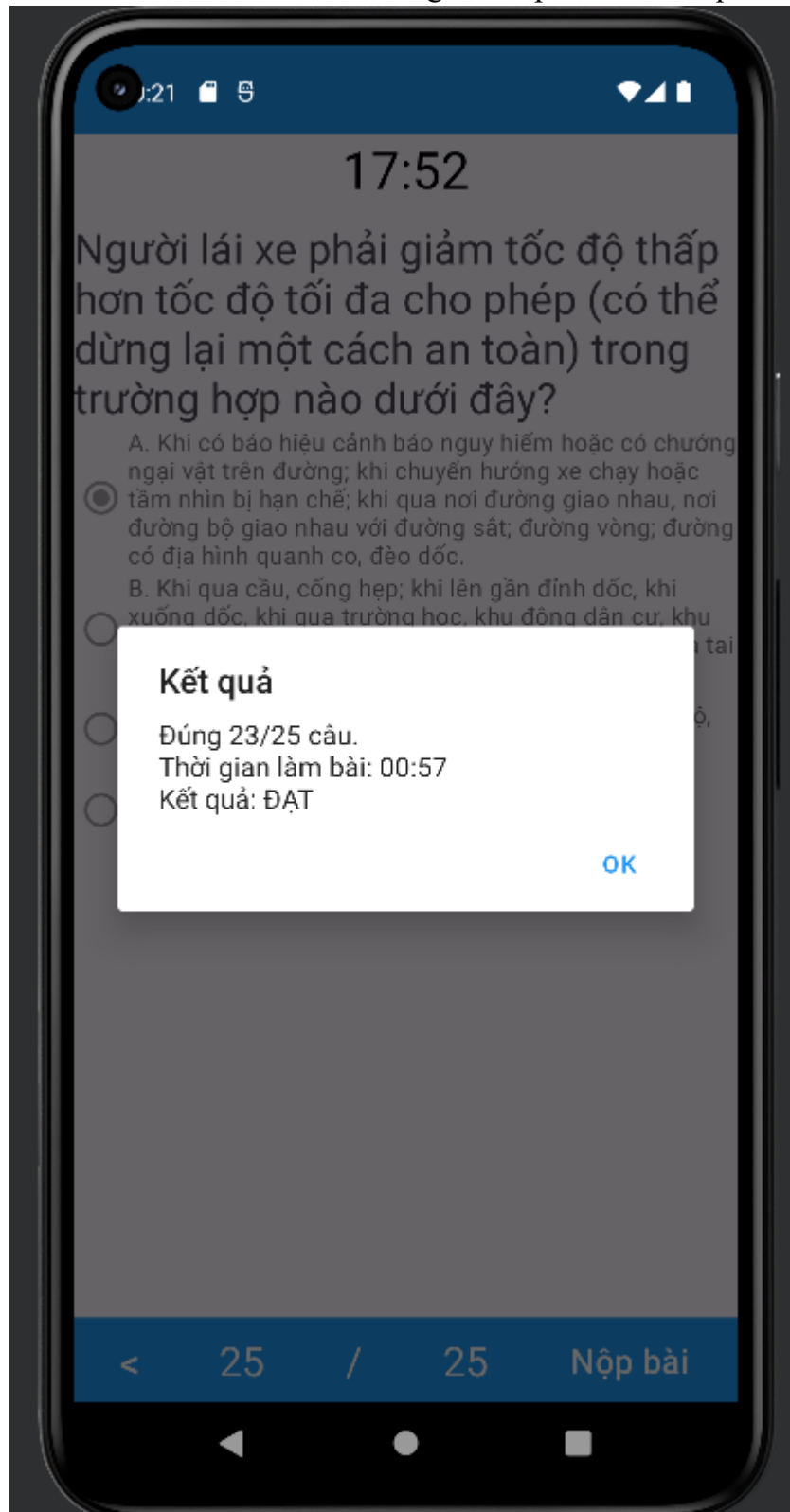
- Hình 5: Giao diện làm bài thi thử



- Hình 6: Giao diện hiển thị dialog khi nộp bài với kết quả TRƯỢT



- Hình 7: Giao diện hiển thị dialog khi nộp bài với kết quả ĐẠT



- Hình 8: Giao diện biên báo cấm



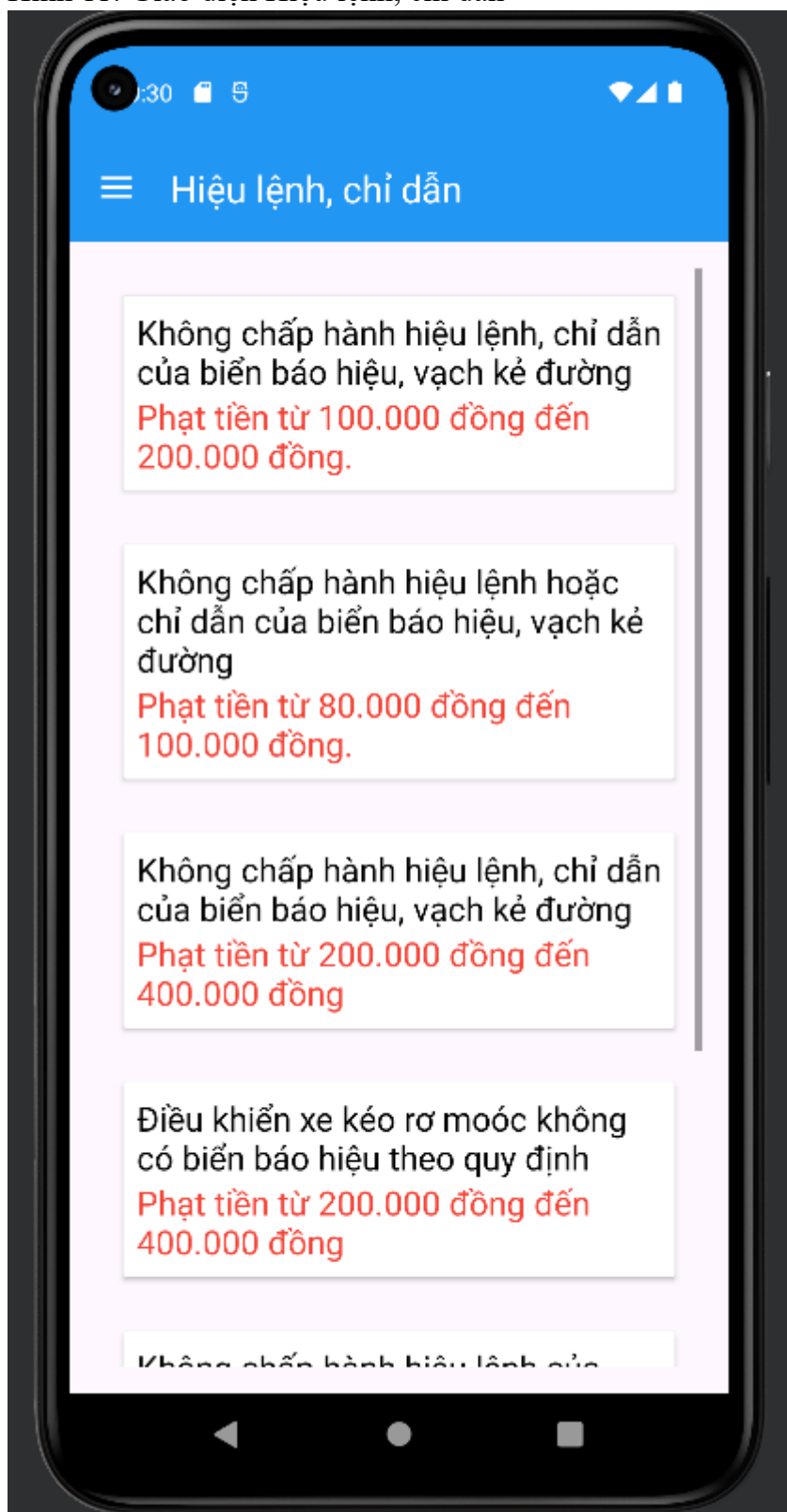
- Hình 9: Giao diện tra cứu luật



- Hình 10: Giao diện Nồng độ cồn, chất kích thích



- Hình 11: Giao diện Hiệu lệnh, chỉ dẫn



b. Nguyễn Tiên Lực:

- Hình 12: Giao diện Học lý thuyết



c. Lê Văn Hà:

- Hình 13: Giao diện Biển báo nguy hiểm



- Hình 14: Giao diện biển báo hiệu lệnh



d. Đinh Quang Thao

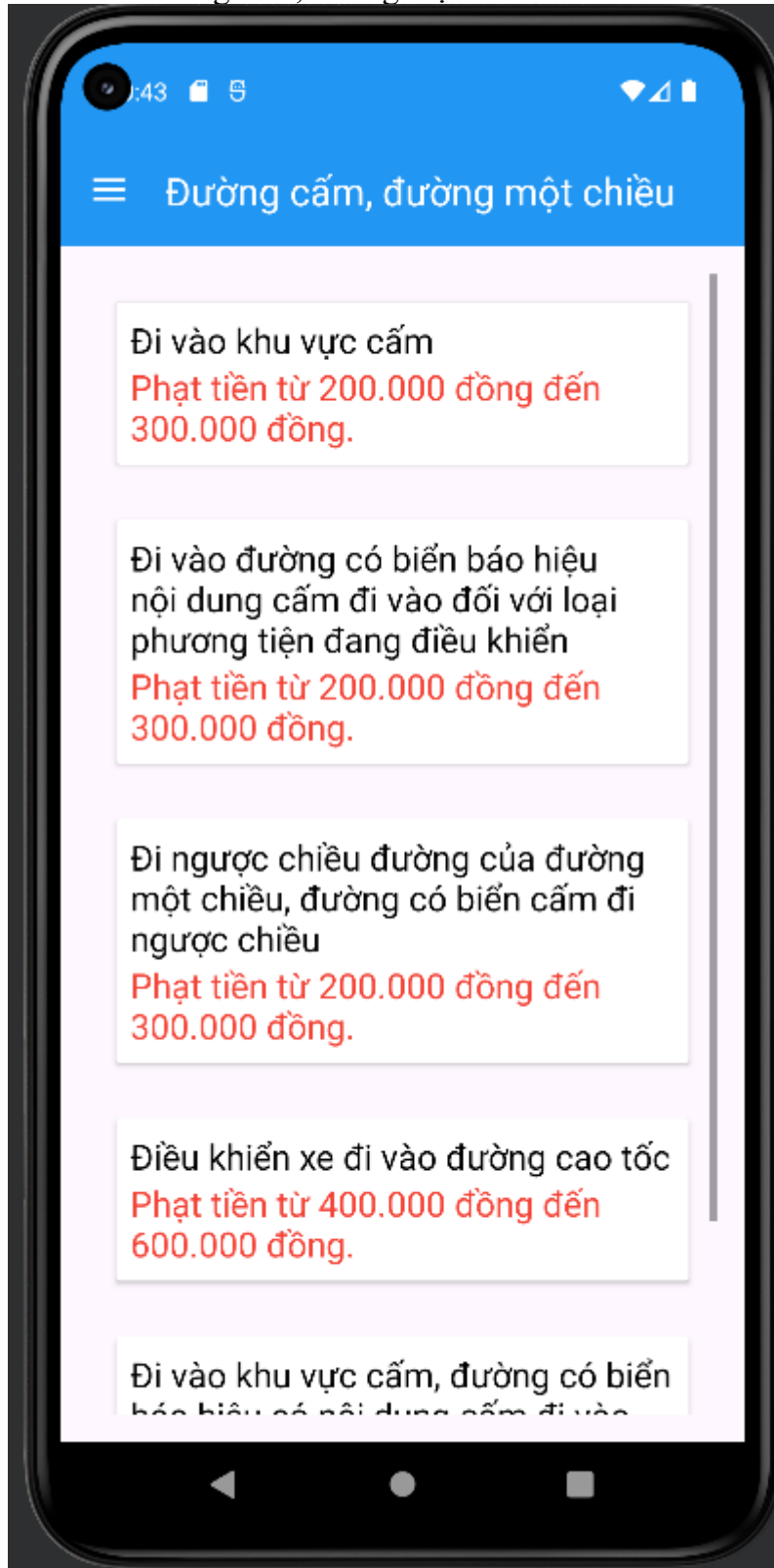
- Hình 15: Giao diện luật Dừng xe, đỗ xe



- Hình 16: Giao diện luật Tốc độ, khoảng cách an toàn



- Hình 17: Đường cấm, đường một chiều



KẾT LUẬN

Ưu điểm của ứng dụng "Ôn thi giấy phép lái xe máy" bao gồm tính tiện lợi và dễ sử dụng. Người dùng có thể ôn thi bất kỳ lúc nào và ở đâu, chỉ cần có một thiết bị di động và kết nối internet. Ứng dụng cung cấp các câu hỏi ôn tập và bài thi thử được tổ chức theo cấu trúc giống như bài thi thực tế, giúp người dùng làm quen với dạng câu hỏi và cách thức thi. Ngoài ra, tính năng ghi nhớ kết quả thi và phân tích các câu trả lời sai giúp người dùng nhận biết điểm yếu của mình và cải thiện kiến thức một cách hiệu quả.

Nhược điểm của ứng dụng có thể là sự phụ thuộc vào kết nối internet, gây khó khăn cho người dùng ở các khu vực có tín hiệu yếu hoặc không có kết nối internet. Đôi khi, các câu hỏi trong ứng dụng có thể không cập nhật kịp thời theo thay đổi của luật giao thông hoặc quy định thi, dẫn đến việc cung cấp thông tin không chính xác. Thêm vào đó, giao diện người dùng có thể chưa thực sự thân thiện và hấp dẫn, gây khó khăn cho một số người dùng trong việc tương tác với ứng dụng.

Hướng phát triển của ứng dụng có thể bao gồm việc tích hợp chức năng học offline để người dùng có thể ôn thi mà không cần kết nối internet. Ngoài ra, việc cập nhật thường xuyên và nhanh chóng các câu hỏi và kiến thức mới từ các nguồn đáng tin cậy sẽ giúp đảm bảo tính chính xác và cập nhật của ứng dụng. Để cải thiện trải nghiệm người dùng, có thể đầu tư vào việc tối ưu giao diện và bổ sung các tính năng tương tác như video hướng dẫn hoặc trò chuyện trực tiếp với chuyên gia tư vấn. Chúng em sẽ cố gắng, tiếp tục tìm tòi nghiên cứu để phát triển ứng dụng tốt hơn trong tương lai.

DANH MỤC TÀI LIỆU THAM KHẢO

- [1]. Margaret Kozak Polk (2024), *Coding Android Apps*, Chapman & Hall
- [2]. <https://123docz.net/document/10599431-bao-cao-bai-tap-lon-mon-phat-trien-ung-dung-cho-cac-thiet-bi-di-dong.htm>
- [3]. <https://www.thegioididong.com/game-app/top-5-phan-mem-hoc-lai-xe-b1-b2-mien-phi-tren-dien-1308854>

PHỤ LỤC

- Code minh họa đầy đủ đã thực hiện

- MainActivity.java:

```
package com.example.drivingtestapp;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.appcompat.widget.Toolbar;
import android.view.MenuItem;
import androidx.annotation.NonNull;

import com.google.android.material.navigation.NavigationView;

public class MainActivity extends AppCompatActivity implements
NavigationView.OnNavigationItemSelectedListener {

    Button btnExam, btnLearn, btnSign, btnLaw;
    private DrawerLayout drawerLayout;

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

        btnExam = findViewById(R.id.btnExam);
        btnLearn = findViewById(R.id.btnLearn);
        btnSign = findViewById(R.id.btnSign);
        btnLaw = findViewById(R.id.btnLawDetail);

        btnExam.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(MainActivity.this,
ExamListActivity.class);
                startActivity(intent);
            }
        });

        btnLearn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(MainActivity.this,
LearningActivity.class);
                startActivity(intent);
            }
        });

        btnSign.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(MainActivity.this,
RoadSignActivity.class);
```

```

        startActivity(intent);
    }
});

btnLaw.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(MainActivity.this,
LawListActivity.class);
        startActivity(intent);
    }
});

Toolbar toolbar = findViewById(R.id.toolbar_main);
setSupportActionBar(toolbar);

drawerLayout = findViewById(R.id.drawer_layout_main);
NavigationView navigationView = findViewById(R.id.nav_view);

ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(this,
drawerLayout, toolbar, R.string.navigation_drawer_open,
R.string.navigation_drawer_close);
drawerLayout.addDrawerListener(toggle);
toggle.syncState();
navigationView.setNavigationItemSelectedListener(this);
}

@Override
public boolean onNavigationItemSelected(@NonNull MenuItem item) {
    int id = item.getItemId();
    if (id == R.id.nav_main) {
        drawerLayout.closeDrawer(GravityCompat.START);
    } else if (id == R.id.nav_thithusathach) {
        startActivity(new Intent(this, ExamListActivity.class));
    } else if (id == R.id.nav_hoclythuyet) {
        startActivity(new Intent(this, LearningActivity.class));
    } else if (id == R.id.nav_bienbao) {
        startActivity(new Intent(this, RoadSignActivity.class));
    } else if (id == R.id.nav_tracuuluat) {
        startActivity(new Intent(this, LawListActivity.class));
    } else if (id == R.id.nav_bienbao) {
        startActivity(new Intent(this, RoadSignActivity.class));
    } else if (id == R.id.nav_thoat) {
        exit();
    }
    drawerLayout.closeDrawer(GravityCompat.START);
    return true;
}

private void exit() {
    finishAffinity();
    System.exit(0);
}
}

```

- **Exam.java:**

```
package com.example.drivingtestapp;

import java.util.ArrayList;

public class Exam {
    private String namedethi;
    private int imagedethi;
    private int soluongCauhoi;
    private int thoigian;

    public Exam(String namedethi, int imagedethi, int soluongCauhoi,
int thoigian) {
        this.namedethi = namedethi;
        this.imagedethi = imagedethi;
        this.soluongCauhoi = soluongCauhoi;
        this.thoigian = thoigian;
    }

    public String getNamedethi() {
        return namedethi;
    }

    public void setNamedethi(String namedethi) {
        this.namedethi = namedethi;
    }

    public int getImagedethi() {
        return imagedethi;
    }

    public void setImagedethi(int imagedethi) {
        this.imagedethi = imagedethi;
    }

    public int getSoluongCauhoi() {
        return soluongCauhoi;
    }

    public void setSoluongCauhoi(int soluongCauhoi) {
        this.soluongCauhoi = soluongCauhoi;
    }

    public int getThoigian() {
        return thoigian;
    }

    public void setThoigian(int thoigian) {
        this.thoigian = thoigian;
    }

    private static int lastExamId = 0;

    public static ArrayList<Exam> createExamsList(int numExams) {
        ArrayList<Exam> exams = new ArrayList<>();
        lastExamId = 0;

        for (int i = 1; i <= numExams; i++) {
            int soluongCauhoi = 25;
            int thoigian = 19;

            int imagedethi =
```

```
android.R.drawable.checkbox_off_background;  
  
        exams.add(new Exam("Đề số " + ++lastExamId, imagedethi,  
        soluongCauhoi, thoigian));  
    }  
  
    return exams;  
}  
}
```


- ExamListActivity.java:

```
package com.example.drivingtestapp;

import android.content.Intent;
import android.os.Bundle;
import android.view.MenuItem;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.google.android.material.navigation.NavigationView;

import java.util.ArrayList;

public class ExamListActivity extends AppCompatActivity implements
NavigationView.OnNavigationItemSelectedListener {

    ArrayList<Exam> exams;
    private DrawerLayout drawerLayout;

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

        RecyclerView recyclerView_dethi = (RecyclerView)
        findViewById(R.id.recyclerView_dethi);

        exams = Exam.createExamsList(5);
        MyArrayAdapter adapter = new MyArrayAdapter(exams);
        recyclerView_dethi.setAdapter(adapter);
        recyclerView_dethi.setLayoutManager(new
        LinearLayoutManager(this));

        Toolbar toolbar = findViewById(R.id.toolbar_thi);
        setSupportActionBar(toolbar);

        drawerLayout = findViewById(R.id.drawer_layout_thi);
        NavigationView navigationView =
        findViewById(R.id.nav_view_thi);
        ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(this,
        drawerLayout, toolbar, R.string.navigation_drawer_open,
        R.string.navigation_drawer_close);
        drawerLayout.addDrawerListener(toggle);
        toggle.syncState();

        navigationView.setNavigationItemSelectedListener(this);
    }

    @Override
    public boolean onNavigationItemSelectedListener(@NonNull MenuItem item) {
        int id = item.getItemId();

        if (id == R.id.nav_main) {
            startActivity(new Intent(this, MainActivity.class));
        } else if (id == R.id.nav_thithusathach) {
```

```

        drawerLayout.closeDrawer(GravityCompat.START);
    } else if (id == R.id.nav_hoclythuyet) {
        startActivity(new Intent(this, LearningActivity.class));
    } else if (id == R.id.nav_bienbao) {
        startActivity(new Intent(this, RoadSignActivity.class));
    } else if (id == R.id.nav_tracuuluat) {
        startActivity(new Intent(this, LawListActivity.class));
    } else if (id == R.id.nav_bienbao) {
        startActivity(new Intent(this, RoadSignActivity.class));
    } else if (id == R.id.nav_thoat) {
        exit();
    }

    drawerLayout.closeDrawer(GravityCompat.START);
    return true;
}

private void exit() {
    finishAffinity();
    System.exit(0);
}
}

```

- **ExamActivity.java (trương tự ExamActivity1.java, ExamActivity2.java, ExamActivity3.java, ExamActivity4.java):**

```
package com.example.drivingtestapp;

import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.graphics.Color;
import android.os.Bundle;
import android.os.CountDownTimer;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;
import androidx.viewpager2.widget.ViewPager2;

import com.example.drivingtestapp.models.Question;
import com.example.drivingtestapp.services.ApiService;

import java.util.ArrayList;
import java.util.List;

import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class ExamActivity extends AppCompatActivity {
    public static int score = 20;
    private ArrayList<Question> questions = new ArrayList<>();
    ViewPager2 viewPager;
    Button btnPrev, btnNext, btnSubmit;
    TextView cur, total, cDown;
    CountDownTimer countDownTimerTotal;
    int timeLeftInMillisTotal;

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

        viewPager = findViewById(R.id.viewpager);
        btnPrev = findViewById(R.id.btnPrev);
        btnNext = findViewById(R.id.btnNext);
        btnSubmit = findViewById(R.id.btnSubmit);
        cur = findViewById(R.id.current);
        total = findViewById(R.id.total);
        cDown = findViewById(R.id.countDown);

        timeLeftInMillisTotal = 1140000;
        countDownTimerTotal = new CountDownTimer(timeLeftInMillisTotal,
1000) {
            @Override
            public void onTick(long millisUntilFinished) {
                timeLeftInMillisTotal = (int) millisUntilFinished;
                updateCountDownText();
            }
        }
    }
}
```

```

    }

    @Override
    public void onFinish() {
        Intent intent = new Intent();
        intent.putExtra("result", score);
        setResult(RESULT_OK, intent);
        ExamActivity.this.finish();
    }
}.start();

getQuestions();

btnPrev.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        int previousItem = viewPager.getCurrentItem() - 1;
        if (previousItem >= 0) {
            viewPager.setCurrentItem(previousItem);
        }
    }
});

btnNext.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        int nextItem = viewPager.getCurrentItem() + 1;
        if (nextItem < questions.size()) {
            viewPager.setCurrentItem(nextItem);
        }
    }
});

btnSubmit.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        submitExam();
    }
});
}

private void getQuestions() {
    ApiService.apiService.questionCall().enqueue(new
    Callback<List<Question>>() {
        @Override
        public void onResponse(Call<List<Question>> call,
        Response<List<Question>> response) {
            questions = new ArrayList<>(response.body());
            for (Question question : questions) {
                Log.i("QuestionsLog", question.toString());
            }
            viewPager.setUserInputEnabled(false);
            ViewPagerAdapter adapter = new
            ViewPagerAdapter(ExamActivity.this, questions);
            viewPager.setAdapter(adapter);
            cur.setText("1");
            total.setText(String.valueOf(questions.size()));
            viewPager.registerOnPageChangeCallback(new
            ViewPager2.OnPageChangeCallback() {
                @Override
                public void onPageSelected(int position) {
                    cur.setText(String.valueOf(position + 1));
                    updateButtonVisibility(position);
                }
            });
        }
    });
}

```

```

        }
    });
    updateButtonVisibility(0);
}

@Override
public void onFailure(Call<List<Question>> call, Throwable
t) {
    Toast.makeText(ExamActivity.this, "Error: " +
t.getMessage(), Toast.LENGTH_SHORT).show();
    Log.e("ApiError", t.getMessage(), t);
}
});
}

private void updateButtonVisibility(int position) {
    if (position == 0) {
        btnPrev.setVisibility(View.GONE);
        btnNext.setVisibility(View.VISIBLE);
        btnSubmit.setVisibility(View.GONE);
    } else if (position == questions.size() - 1) {
        btnNext.setVisibility(View.GONE);
        btnPrev.setVisibility(View.VISIBLE);
        btnSubmit.setVisibility(View.VISIBLE);
    } else {
        btnPrev.setVisibility(View.VISIBLE);
        btnNext.setVisibility(View.VISIBLE);
        btnSubmit.setVisibility(View.GONE);
    }
}

private void updateCountDownText() {
    int minutes = (int) ((timeLeftInMillisTotal / 1000) / 60);
    int seconds = (int) ((timeLeftInMillisTotal / 1000) % 60);
    String timeLeftFormatted = String.format("%02d:%02d", minutes,
seconds);
    cDown.setText(timeLeftFormatted);
    if (timeLeftInMillisTotal < 10000) {
        cDown.setTextColor(Color.RED);
    } else {
        cDown.setTextColor(Color.BLACK);
    }
}

private void submitExam() {
    int correctAnswers = calculateCorrectAnswers();
    int totalQuestions = questions.size();
    String timeTaken = getTimeTaken();

    String passStatus = correctAnswers >= 21 ? "ĐẠT" : "TRƯỢT";

    String resultMessage = String.format("Đúng %d/%d câu.\nThời
gian làm bài: %s\nKết quả: %s",
        correctAnswers, totalQuestions, timeTaken, passStatus);

    new AlertDialog.Builder(this)
        .setTitle("Kết quả")
        .setMessage(resultMessage)
        .setPositiveButton(android.R.string.ok, new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int
which) {

```

```

        Intent intent = new Intent(ExamActivity.this,
ExamListActivity.class);
        intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP
| Intent.FLAG_ACTIVITY_NEW_TASK);
        startActivity(intent);
        finish();
    }
    })
    .show();
}

private int calculateCorrectAnswers() {
    int correctAnswers = 0;
    for (Question question : questions) {
        if (question.isCorrect()) {
            correctAnswers++;
        }
    }
    return correctAnswers;
}

private String getTimeTaken() {
    int minutes = (int) ((1140000 - timeLeftInMillisTotal) / 1000)
/ 60;
    int seconds = (int) ((1140000 - timeLeftInMillisTotal) / 1000)
% 60;
    return String.format("%02d:%02d", minutes, seconds);
}
}

```

- **Question.java:**

- **package** com.example.drivingtestapp.models;

```
import java.io.Serializable;

public class Question implements Serializable {
    private int id;
    private String question;
    private String ideaA;
    private String ideaB;
    private String ideaC;
    private String ideaD;
    private int answer;
    private int userAnswer = -1;

    public Question(int id, String question, String ideaA, String
ideaB, String ideaC, String ideaD, int answer) {
        this.id = id;
        this.question = question;
        this.ideaA = ideaA;
        this.ideaB = ideaB;
        this.ideaC = ideaC;
        this.ideaD = ideaD;
        this.answer = answer;
    }

    public int getUserAnswer() {
        return userAnswer;
    }

    public void setUserAnswer(int userAnswer) {
        this.userAnswer = userAnswer;
    }

    public boolean isCorrect() {
        return userAnswer == answer;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getQuestion() {
        return question;
    }

    public void setQuestion(String question) {
        this.question = question;
    }

    public String getIdeaA() {
        return ideaA;
    }

    public void setIdeaA(String ideaA) {
        this.ideaA = ideaA;
    }
}
```

```

    public String getIdeaB() {
        return ideaB;
    }

    public void setIdeaB(String ideaB) {
        this.ideaB = ideaB;
    }

    public String getIdeaC() {
        return ideaC;
    }

    public void setIdeaC(String ideaC) {
        this.ideaC = ideaC;
    }

    public String getIdeaD() {
        return ideaD;
    }

    public void setIdeaD(String ideaD) {
        this.ideaD = ideaD;
    }

    public int getAnswer() {
        return answer;
    }

    public void setAnswer(int answer) {
        this.answer = answer;
    }

    @Override
    public String toString() {
        return "Question{" +
            "id=" + id +
            ", question='" + question + '\'' +
            ", ideaA='" + ideaA + '\'' +
            ", ideaB='" + ideaB + '\'' +
            ", ideaC='" + ideaC + '\'' +
            ", ideaD='" + ideaD + '\'' +
            ", answer=" + answer +
            ", userAnswer=" + userAnswer +
            '}';
    }
}

```


- **MyArrayAdapter.java:**

```
package com.example.drivingtestapp;

import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import java.util.List;

public class MyArrayAdapter extends
RecyclerView.Adapter<MyArrayAdapter.ViewHolder> {
    private List<Exam> mExams;

    private static final Class[] activities = new Class[]{
        ExamActivity.class,
        ExamActivity1.class,
        ExamActivity2.class,
        ExamActivity3.class,
        ExamActivity4.class,
    };

    public MyArrayAdapter(List<Exam> exams) {
        mExams = exams;
    }

    public static class ViewHolder extends RecyclerView.ViewHolder {
        public ImageView deImageView;
        public TextView deTextView;
        public TextView soluongCauhoiTextView;
        public Button lamBaiButton;

        public ViewHolder(View itemView) {
            super(itemView);
            deImageView = itemView.findViewById(R.id.imgDe);
            deTextView = itemView.findViewById(R.id.txtTenLuot);
            soluongCauhoiTextView =
itemView.findViewById(R.id.txtMucPhat);
            lamBaiButton = itemView.findViewById(R.id.btnLamBai);
        }
    }

    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType) {
        Context context = parent.getContext();
        LayoutInflater inflater = LayoutInflater.from(context);
        View examView = inflater.inflate(R.layout.exam_list, parent,
false);
        return new ViewHolder(examView);
    }
}
```

```

@Override
public void onBindViewHolder(@NonNull ViewHolder holder, int
position) {
    Exam exam = mExams.get(position);

    TextView textViewDe = holder.deTextView;
    textViewDe.setText(exam.getNamedethi());

    TextView textViewSoluongCauhoi = holder.soluongCauhoiTextView;
    String cauhoiVaThoigian = exam.getSoluongCauhoi() + " câu/" +
exam.getThoigian() + " phút";
    textViewSoluongCauhoi.setText(cauhoiVaThoigian);

    ImageView imageView = holder.deImageView;
    imageView.setImageResource(R.drawable.exam);

    Button buttonLamBai = holder.lamBaiButton;
    buttonLamBai.setText("Làm bài");
    buttonLamBai.setEnabled(true);

    buttonLamBai.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Context context = v.getContext();

            new AlertDialog.Builder(context)
                .setTitle("BẮT ĐẦU LÀM BÀI")
                .setMessage("\nĐề thi gồm 25 câu hỏi làm bài
trong thời gian 19 phút.\n" +
"\nĐể vượt qua bài thi, bạn cần trả lời
đúng 21/25 câu hỏi.")
                .setPositiveButton("Bắt đầu", new
DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog,
int which) {
                        Class activityToStart =
activities[position % activities.length];
                        Intent intent = new Intent(context,
activityToStart);
                        intent.putExtra("EXAM_ID",
exam.getNamedethi());
                        context.startActivity(intent);
                    }
                })
                .setNegativeButton("Hủy", null)
                .show();
        }
    });
}

@Override
public int getItemCount() {
    return mExams.size();
}
}

```

- **ViewPagerAdapter.java:**

```
package com.example.drivingtestapp;

import android.os.Bundle;

import androidx.annotation.NonNull;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentActivity;
import androidx.viewpager2.adapter.FragmentStateAdapter;

import com.example.drivingtestapp.models.Question;

import java.util.List;

public class ViewPagerAdapter extends FragmentStateAdapter{

    private List<Question> questions;

    public ViewPagerAdapter(@NonNull FragmentActivity fragmentActivity,
List<Question> questions) {
        super(fragmentActivity);
        this.questions = questions;
    }

    @NonNull
    @Override
    public Fragment createFragment(int position) {
        if(questions == null){
            return null;
        }
        Question question = questions.get(position);
        QuestionFragment questionFragment = new QuestionFragment();
        Bundle bundle = new Bundle();
        bundle.putSerializable("question",question);
        questionFragment.setArguments(bundle);
        // questionFragment.countDown();
        return questionFragment;
    }

    @Override
    public int getItemCount() {
        return questions.size();
    }
}
```

○ QuestionFragment.java:

```
package com.example.drivingtestapp;

import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;
import android.widget.Toast;

import androidx.fragment.app.Fragment;

import com.example.drivingtestapp.models.Question;

public class QuestionFragment extends Fragment {

    private View view;
    private Question ques;
    private boolean isTimeUp = false;

    TextView question;
    RadioGroup rdoG;
    RadioButton idA, idB, idC, idD;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        view = inflater.inflate(R.layout.fragment_question, container,
false);

        question = view.findViewById(R.id.ques);
        rdoG = view.findViewById(R.id.rdoGR);
        idA = view.findViewById(R.id.idea1);
        idB = view.findViewById(R.id.idea2);
        idC = view.findViewById(R.id.idea3);
        idD = view.findViewById(R.id.idea4);

        Bundle bundle = getArguments();

        if (bundle != null) {
            ques = (Question) bundle.getSerializable("question");
            if (ques != null) {
                question.setText(ques.getQuestion());
                idA.setText(ques.getIdeaA());
                idB.setText(ques.getIdeaB());
                idC.setText(ques.getIdeaC());
                idD.setText(ques.getIdeaD());
            }
        }

        rdoG.setOnCheckedChangeListener(new
RadioGroup.OnCheckedChangeListener() {
            @Override
            public void onCheckedChanged(RadioGroup group, int
checkedId) {
                int selectedAnswer = -1; // -1 nếu không có lựa chọn
```

```

        if (checkedId == idA.getId()) {
            selectedAnswer = 1;
        } else if (checkedId == idB.getId()) {
            selectedAnswer = 2;
        } else if (checkedId == idC.getId()) {
            selectedAnswer = 3;
        } else if (checkedId == idD.getId()) {
            selectedAnswer = 4;
        }

        ques.setUserAnswer(selectedAnswer);

        if (ques.isCorrect()) {
            Toast.makeText(getApplicationContext(), "Đúng",
Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(getApplicationContext(), "Sai",
Toast.LENGTH_SHORT).show();
            ExamActivity.score -= 1;
        }

        rdoG.setEnabled(false);
        idA.setEnabled(false);
        idB.setEnabled(false);
        idC.setEnabled(false);
        idD.setEnabled(false);
    }
});
return view;
}
}

```

- **ApiService.java (tương tự ApiService.java, ApiService.java, ApiService.java, ApiService.java):**

```
package com.example.drivingtestapp.services;

import com.example.drivingtestapp.models.Question;
import com.google.gson.Gson;
import com.google.gson.GsonBuilder;

import java.util.List;

import retrofit2.Call;
import retrofit2.Retrofit;
import retrofit2.converter.gson.GsonConverterFactory;
import retrofit2.http.GET;

public interface ApiService {
    Gson gson = new GsonBuilder().setDateFormat("yyyy-MM-dd HH:mm:ss").create();

    // http://10.0.84.176/AndroidWebService/views/QuestionView.php

    ApiService apiService = new Retrofit.Builder()
        .baseUrl("http://192.168.1.35/")
        //192.168.1.35
        //172.20.10.2
        .addConverterFactory(GsonConverterFactory.create())
        .build()
        .create(ApiService.class);

    @GET("/AndroidWebService/views/QuestionView.php")
    public Call<List<Question>> questionCall();
}
```

- **Learning.java:**

```
package com.example.drivingtestapp;

public class Learning {
    private String maCauHoi;
    private String cauHoi;
    private String dapAn;

    public Learning(String maCauHoi, String cauHoi, String dapAn) {
        this.maCauHoi = maCauHoi;
        this.cauHoi = cauHoi;
        this.dapAn = dapAn;
    }

    public String getMaCauHoi() {
        return maCauHoi;
    }

    public void setMaCauHoi(String maCauHoi) {
        this.maCauHoi = maCauHoi;
    }

    public String getCauHoi() {
        return cauHoi;
    }

    public void setCauHoi(String cauHoi) {
        this.cauHoi = cauHoi;
    }

    public String getDapAn() {
        return dapAn;
    }

    public void setDapAn(String dapAn) {
        this.dapAn = dapAn;
    }
}
```

- **LearningAdapter.java:**

```
package com.example.drivingtestapp;

import android.app.Activity;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import java.util.ArrayList;

public class LearningAdapter extends
RecyclerView.Adapter<LearningAdapter.ViewHolder> {

    private Activity context;
    private ArrayList<Learning> myArray;
    private int layoutId;

    public LearningAdapter(Activity context, int layoutId,
ArrayList<Learning> arr) {
        this.context = context;
        this.layoutId = layoutId;
        this.myArray = arr;
    }

    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType) {
        LayoutInflater inflater = context.getLayoutInflater();
        View itemView = inflater.inflate(layoutId, parent, false);
        return new ViewHolder(itemView);
    }

    @Override
    public void onBindViewHolder(@NonNull ViewHolder holder, int
position) {
        Learning myItem = myArray.get(position);

        holder.maCauHoi.setText(myItem.getMaCauHoi());
        holder.cauHoi.setText(myItem.getCauHoi());
        holder.dapAn.setText(myItem.getDapAn());
    }

    @Override
    public int getItemCount() {
        return myArray.size();
    }

    public static class ViewHolder extends RecyclerView.ViewHolder {
        TextView maCauHoi, cauHoi, dapAn;

        public ViewHolder(@NonNull View itemView) {
            super(itemView);
            maCauHoi = itemView.findViewById(R.id.txtCauHoi);
            cauHoi = itemView.findViewById(R.id.txtLuaChon);
            dapAn = itemView.findViewById(R.id.txtDapAn);
        }
    }
}
```


} }

- **LearningActivity.java:**

```
package com.example.drivingtestapp;

import android.content.Intent;
import android.content.res.AssetManager;
import android.os.Bundle;
import android.view.MenuItem;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.google.android.material.navigation.NavigationView;

import org.json.JSONArray;
import org.json.JSONObject;

import java.io.InputStream;
import java.util.ArrayList;
import java.util.Scanner;

public class LearningActivity extends AppCompatActivity implements
NavigationView.OnNavigationItemSelectedListener {

    RecyclerView recyclerViewLearning;
    ArrayList<Learning> itemList;
    LearningAdapter adapter;
    private DrawerLayout drawerLayout;

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

        Toolbar toolbar = findViewById(R.id.toolbar_learning);
        setSupportActionBar(toolbar);

        drawerLayout = findViewById(R.id.drawer_layout_learning);
        NavigationView navigationView =
        findViewById(R.id.nav_view_thi);
        ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(this,
drawerLayout, toolbar, R.string.navigation_drawer_open,
R.string.navigation_drawer_close);
drawerLayout.addDrawerListener(toggle);
toggle.syncState();

        navigationView.setNavigationItemSelectedListener(this);

        recyclerViewLearning =
        findViewById(R.id.recyclerView_learning);
        itemList = new ArrayList<>();
        loadItemsFromJson();
        recyclerViewLearning.setLayoutManager(new
        LinearLayoutManager(this));
        adapter = new LearningAdapter(this, R.layout.learning_list,
itemList);
        recyclerViewLearning.setAdapter(adapter);
```

```

    }

    private void loadItemsFromJson() {
        try {
            AssetManager assetManager = getAssets();
            InputStream inputStream =
assetManager.open("learning.json");
            String jsonStr = convertStreamToString(inputStream);
            JSONObject jsonObj = new JSONObject(jsonStr);
            JSONArray jsonArray = jsonObj.getJSONArray("hoclythuyet");

            for (int i = 0; i < jsonArray.length(); i++) {
                JSONObject itemObj = jsonArray.getJSONObject(i);
                String maCauHoi = itemObj.getString("MaCauHoi");
                String cauHoi = itemObj.getString("CauHoi");
                String dapAn = itemObj.getString("DapAn");

                itemList.add(new Learning(maCauHoi, cauHoi, dapAn));
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    private String convertStreamToString(InputStream inputStream) {
        Scanner scanner = new Scanner(inputStream).useDelimiter("\\A");
        return scanner.hasNext() ? scanner.next() : "";
    }

    @Override
    public boolean onNavigationItemSelected(@NonNull MenuItem item) {
        int id = item.getItemId();

        if (id == R.id.nav_main) {
            startActivity(new Intent(this, MainActivity.class));
        } else if (id == R.id.nav_thithusathach) {
            startActivity(new Intent(this, ExamListActivity.class));
        } else if (id == R.id.nav_hoclythuyet) {
            drawerLayout.closeDrawer(GravityCompat.START);
        } else if (id == R.id.nav_bienbao) {
            startActivity(new Intent(this, RoadSignActivity.class));
        } else if (id == R.id.nav_tracuuluat) {
            startActivity(new Intent(this, LawListActivity.class));
        } else if (id == R.id.nav_bienbao) {
            startActivity(new Intent(this, RoadSignActivity.class));
        } else if (id == R.id.nav_thoat) {
            exit();
        }

        drawerLayout.closeDrawer(GravityCompat.START);
        return true;
    }

    private void exit() {
        finishAffinity();
        System.exit(0);
    }
}

```

- **Law.java:**

```
package com.example.drivingtestapp;

public class Law {
    private String tenLuat;
    private String mucPhat;

    public Law(String tenLuat, String mucPhat) {
        this.tenLuat = tenLuat;
        this.mucPhat = mucPhat;
    }

    public String getTenLuat() {
        return tenLuat;
    }

    public void setTenLuat(String tenLuat) {
        this.tenLuat = tenLuat;
    }

    public String getMucPhat() {
        return mucPhat;
    }

    public void setMucPhat(String mucPhat) {
        this.mucPhat = mucPhat;
    }
}
```

- **LawListActivity.java:**

```
package com.example.drivingtestapp;

import android.content.Intent;
import android.os.Bundle;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;

import com.google.android.material.navigation.NavigationView;

public class LawListActivity extends AppCompatActivity implements
NavigationView.OnNavigationItemSelectedListener {

    Button btnNghiDinh, btnHieuLenh, btnDungXe, btnTocDo, btnDuongCam;
    private DrawerLayout drawerLayout;

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

        btnNghiDinh = findViewById(R.id.btnNongDoCon);
        btnHieuLenh = findViewById(R.id.btnHieuLenh);
        btnDungXe = findViewById(R.id.btnDungXe);
        btnTocDo = findViewById(R.id.btnTocDo);
        btnDuongCam = findViewById(R.id.btnDuongCam);

        btnNghiDinh.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(LawListActivity.this,
LawActivity.class);
                startActivity(intent);
            }
        });

        btnHieuLenh.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(LawListActivity.this,
LawActivity1.class);
                startActivity(intent);
            }
        });

        btnDungXe.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(LawListActivity.this,
LawActivity2.class);
                startActivity(intent);
            }
        });
    }
}
```

```

        btnTocDo.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(LawListActivity.this,
                LawActivity3.class);
                startActivity(intent);
            }
        });

        btnDuongCam.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(LawListActivity.this,
                LawActivity4.class);
                startActivity(intent);
            }
        });

        Toolbar toolbar = findViewById(R.id.toolbar_lawlist);
        setSupportActionBar(toolbar);

        drawerLayout = findViewById(R.id.drawer_layout_lawlist);
        NavigationView navigationView =
        findViewById(R.id.nav_view_lawlist);

        ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(this,
        drawerLayout, toolbar, R.string.navigation_drawer_open,
        R.string.navigation_drawer_close);
        drawerLayout.addDrawerListener(toggle);
        toggle.syncState();
        navigationView.setNavigationItemSelectedListener(this);
    }

    @Override
    public boolean onNavigationItemSelected(@NonNull MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.nav_main) {
            startActivity(new Intent(this, MainActivity.class));
        } else if (id == R.id.nav_thithusathach) {
            startActivity(new Intent(this, ExamListActivity.class));
        } else if (id == R.id.nav_hoclythuyet) {
            startActivity(new Intent(this, LearningActivity.class));
        } else if (id == R.id.nav_bienbao) {
            startActivity(new Intent(this, RoadSignActivity.class));
        } else if (id == R.id.nav_tracuuluat) {
            drawerLayout.closeDrawer(GravityCompat.START);
        } else if (id == R.id.nav_thoat) {
            exit();
        }
        drawerLayout.closeDrawer(GravityCompat.START);
        return true;
    }

    private void exit() {
        finishAffinity();
        System.exit(0);
    }
}

```

○ LawAdapter.java:

```
• package com.example.drivingtestapp;

import android.app.Activity;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import java.util.ArrayList;

public class LawAdapter extends
RecyclerView.Adapter<LawAdapter.ViewHolder> {

    private Activity context;
    private ArrayList<Law> myArray;
    private int layoutId;

    public LawAdapter(Activity context, int layoutId, ArrayList<Law>
arr) {
        this.context = context;
        this.layoutId = layoutId;
        this.myArray = arr;
    }

    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType) {
        LayoutInflater inflater = context.getLayoutInflater();
        View itemView = inflater.inflate(layoutId, parent, false);
        return new ViewHolder(itemView);
    }

    @Override
    public void onBindViewHolder(@NonNull ViewHolder holder, int
position) {
        Law myLaw = myArray.get(position);

        holder.tenLuat.setText(myLaw.getTenLuat());
        holder.mucPhat.setText(myLaw.getMucPhat());
    }

    @Override
    public int getItemCount() {
        return myArray.size();
    }

    public static class ViewHolder extends RecyclerView.ViewHolder {
        TextView tenLuat, mucPhat;

        public ViewHolder(@NonNull View itemView) {
            super(itemView);
            tenLuat = itemView.findViewById(R.id.txtTenLuat);
            mucPhat = itemView.findViewById(R.id.txtMucPhat);
        }
    }
}
```

- **LawActivity.java (tương tự LawActivity1.java, LawActivity2.java, LawActivity3.java, LawActivity4.java):**

```
package com.example.drivingtestapp;

import android.content.Intent;
import android.content.res.AssetManager;
import android.os.Bundle;
import android.view.MenuItem;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.google.android.material.navigation.NavigationView;

import org.json.JSONArray;
import org.json.JSONObject;

import java.io.InputStream;
import java.util.ArrayList;
import java.util.Scanner;

public class LawActivity extends AppCompatActivity implements
NavigationView.OnNavigationItemSelectedListener {

    RecyclerView recyclerViewLuat;
    ArrayList<Law> itemList;
    LawAdapter adapter;
    private DrawerLayout drawerLayout;

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

        Toolbar toolbar = findViewById(R.id.toolbar_nongdocon);
        setSupportActionBar(toolbar);

        drawerLayout = findViewById(R.id.drawer_layout_nongdocon);
        NavigationView navigationView =
        findViewById(R.id.nav_view_nongdocon);
        ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(this,
drawerLayout, toolbar, R.string.navigation_drawer_open,
R.string.navigation_drawer_close);
        drawerLayout.addDrawerListener(toggle);
        toggle.syncState();

        navigationView.setNavigationItemSelectedListener(this);

        recyclerViewLuat = findViewById(R.id.recyclerView_nongdocon);
        itemList = new ArrayList<>();
        loadItemsFromJson();
        recyclerViewLuat.setLayoutManager(new
LinearLayoutManager(this));
        adapter = new LawAdapter(this, R.layout.law_list, itemList);
        recyclerViewLuat.setAdapter(adapter);
```



```

    }

    private void loadItemsFromJson() {
        try {
            AssetManager assetManager = getAssets();
            InputStream inputStream = assetManager.open("law.json");
            String jsonStr = convertStreamToString(inputStream);
            JSONObject jsonObj = new JSONObject(jsonStr);
            JSONArray jsonArray = jsonObj.getJSONArray("nongdocon");

            for (int i = 0; i < jsonArray.length(); i++) {
                JSONObject itemObj = jsonArray.getJSONObject(i);
                String tenLuat = itemObj.getString("TenLuat");
                String mucPhat = itemObj.getString("MucPhat");

                itemList.add(new Law(tenLuat, mucPhat));
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    private String convertStreamToString(InputStream inputStream) {
        Scanner scanner = new Scanner(inputStream).useDelimiter("\\A");
        return scanner.hasNext() ? scanner.next() : "";
    }

    @Override
    public boolean onNavigationItemSelected(@NonNull MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.nav_main) {
            startActivity(new Intent(this, MainActivity.class));
        } else if (id == R.id.nav_thithusathach) {
            startActivity(new Intent(this, ExamListActivity.class));
        } else if (id == R.id.nav_hoclythuyet) {
            startActivity(new Intent(this, LearningActivity.class));
        } else if (id == R.id.nav_bienbao) {
            startActivity(new Intent(this, RoadSignActivity.class));
        } else if (id == R.id.nav_tracuuluat) {
            startActivity(new Intent(this, LawListActivity.class));
        }
        else if (id == R.id.nav_thoat) {
            exit();
        }
        drawerLayout.closeDrawer(GravityCompat.START);
        return true;
    }

    private void exit() {
        finishAffinity();
        System.exit(0);
    }
}

```

- **RoadSign.java:**

```
package com.example.drivingtestapp;

public class RoadSign {
    private String tenBienBao;
    private String noidungBienBao;
    private String hinh;

    public RoadSign(String tenBienBao, String noidungBienBao, String
hinh) {
        this.tenBienBao = tenBienBao;
        this.noidungBienBao = noidungBienBao;
        this.hinh = hinh;
    }

    public String getTenBienBao() {
        return tenBienBao;
    }

    public void setTenBienBao(String tenBienBao) {
        this.tenBienBao = tenBienBao;
    }

    public String getNoidungBienBao() {
        return noidungBienBao;
    }

    public void setNoidungBienBao(String noidungBienBao) {
        this.noidungBienBao = noidungBienBao;
    }

    public String getHinh() {
        return hinh;
    }

    public void setHinh(String hinh) {
        this.hinh = hinh;
    }
}
```

- RoadSignAdapter.java:

```
package com.example.drivingtestapp;

import android.app.Activity;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import java.util.ArrayList;

public class RoadSignAdapter extends
RecyclerView.Adapter<RoadSignAdapter.ViewHolder> {
    private Activity context;
    private ArrayList<RoadSign> roadSignList;
    private int layoutId;

    public RoadSignAdapter(Activity context, int layoutId,
ArrayList<RoadSign> roadSignList) {
        this.context = context;
        this.layoutId = layoutId;
        this.roadSignList = roadSignList;
    }

    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType) {
        LayoutInflater inflater = context.getLayoutInflater();
        View itemView = inflater.inflate(layoutId, parent, false);
        return new ViewHolder(itemView);
    }

    @Override
    public void onBindViewHolder(@NonNull ViewHolder holder, int
position) {
        RoadSign roadSign = roadSignList.get(position);
        holder.tenBienBao.setText(roadSign.getTenBienBao());
        holder.noiDungBienBao.setText(roadSign.getNoidungBienBao());
        // Load image resource by name
        int resId =
context.getResources().getIdentifier(roadSign.getHinh(), "drawable",
context.getPackageName());
        holder.hinh.setImageResource(resId);
    }

    @Override
    public int getItemCount() {
        return roadSignList.size();
    }

    public static class ViewHolder extends RecyclerView.ViewHolder {
        TextView tenBienBao, noiDungBienBao;
        ImageView hinh;

        public ViewHolder(@NonNull View itemView) {
            super(itemView);
            tenBienBao = itemView.findViewById(R.id.txtLaw);
        }
    }
}
```

```
        noiDungBienBao =  
itemView.findViewById(R.id.txtSignMeaning);  
        hinh = itemView.findViewById(R.id.imgSign);  
    }  
}  
}
```

- **RoadSignActivity.java:**

```
package com.example.drivingtestapp;

import android.content.Intent;
import android.content.res.AssetManager;
import android.os.Bundle;
import android.view.MenuItem;
import android.widget.TabHost;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.google.android.material.navigation.NavigationView;

import org.json.JSONArray;
import org.json.JSONObject;

import java.io.InputStream;
import java.util.ArrayList;
import java.util.Scanner;

public class RoadSignActivity extends AppCompatActivity implements
NavigationView.OnNavigationItemSelectedListener {
    // Declare UI elements
    private RecyclerView rv1, rv2, rv3;
    private TabHost tab;
    private DrawerLayout drawerLayout;

    // Declare data structures
    private ArrayList<RoadSign> list1, list2, list3;
    private RoadSignAdapter myarray1, myarray2, myarray3;

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

        // Initialize UI elements
        addControl();

        // Set tab change listener
        addEvent();

        Toolbar toolbar = findViewById(R.id.toolbar_sign);
        setSupportActionBar(toolbar);
        drawerLayout = findViewById(R.id.drawer_layout_sign);
        NavigationView navigationView =
        findViewById(R.id.nav_view_sign);
        ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(this,
        drawerLayout, toolbar, R.string.navigation_drawer_open,
        R.string.navigation_drawer_close);
        drawerLayout.addDrawerListener(toggle);
        toggle.syncState();
        navigationView.setNavigationItemSelectedListener(this);

        // Load initial data for all tabs
```

```

        loadItemsFromJson("bienbaocam"); // Load data for the first tab
        by default
    }

    private void loadItemsFromJson(String tabName) {
        try {
            AssetManager assetManager = getAssets();
            InputStream inputStream = assetManager.open("sign.json");
            String jsonStr = convertStreamToString(inputStream);
            JSONObject jsonObj = new JSONObject(jsonStr);
            JSONArray jsonArray = jsonObj.getJSONArray(tabName);

            ArrayList<RoadSign> list = new ArrayList<>();
            for (int i = 0; i < jsonArray.length(); i++) {
                JSONObject itemObj = jsonObj.getJSONObject(i);
                String tenBienBao = itemObj.getString("TenBienBao");
                String noiDungBienBao =
itemObj.getString("NoiDungBienBao");
                String hinh = itemObj.getString("Hinh");
                list.add(new RoadSign(tenBienBao, noiDungBienBao,
hinh));
            }
            updateList(tabName, list);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    private String convertStreamToString(InputStream inputStream) {
        Scanner scanner = new Scanner(inputStream).useDelimiter("\\A");
        return scanner.hasNext() ? scanner.next() : "";
    }

    private void updateList(String tabName, ArrayList<RoadSign> list) {
        if (tabName.equalsIgnoreCase("bienbaocam")) {
            list1.clear();
            list1.addAll(list);
            myarray1.notifyDataSetChanged();
        } else if (tabName.equalsIgnoreCase("bienbaonguyhiem")) {
            list2.clear();
            list2.addAll(list);
            myarray2.notifyDataSetChanged();
        } else if (tabName.equalsIgnoreCase("bienbaohieulenh")) {
            list3.clear();
            list3.addAll(list);
            myarray3.notifyDataSetChanged();
        }
    }

    private void addEvent() {
        tab.setOnTabChangeListener(new TabHost.OnTabChangeListener() {
            @Override
            public void onTabChanged(String tabId) {
                if (tabId.equalsIgnoreCase("t1")) {
                    loadItemsFromJson("bienbaocam");
                } else if (tabId.equalsIgnoreCase("t2")) {
                    loadItemsFromJson("bienbaonguyhiem");
                } else if (tabId.equalsIgnoreCase("t3")) {
                    loadItemsFromJson("bienbaohieulenh");
                }
            }
        });
    }
}

```

```

private void addControl() {
    // Find and initialize UI elements
    tab = (TabHost) findViewById(R.id.tabhost);
    tab.setup();

    // Create and configure tabs
    TabHost.TabSpec tab1 = tab.newTabSpec("t1");
    tab1.setContent(R.id.tab1);
    tab1.setIndicator("Biển báo cấm");
    tab.addTab(tab1);

    TabHost.TabSpec tab2 = tab.newTabSpec("t2");
    tab2.setContent(R.id.tab2);
    tab2.setIndicator("Biển báo nguy hiểm");
    tab.addTab(tab2);

    TabHost.TabSpec tab3 = tab.newTabSpec("t3");
    tab3.setContent(R.id.tab3);
    tab3.setIndicator("Biển báo hiệu lệnh");
    tab.addTab(tab3);

    rv1 = (RecyclerView) findViewById(R.id.rv1);
    rv1.setLayoutManager(new LinearLayoutManager(this));
    rv2 = (RecyclerView) findViewById(R.id.rv2);
    rv2.setLayoutManager(new LinearLayoutManager(this));
    rv3 = (RecyclerView) findViewById(R.id.rv3);
    rv3.setLayoutManager(new LinearLayoutManager(this));

    // Initialize data structures
    list1 = new ArrayList<>();
    list2 = new ArrayList<>();
    list3 = new ArrayList<>();

    // Initialize adapters
    myarray1 = new RoadSignAdapter(this, R.layout.roadsign_list,
list1);
    myarray2 = new RoadSignAdapter(this, R.layout.roadsign_list,
list2);
    myarray3 = new RoadSignAdapter(this, R.layout.roadsign_list,
list3);

    // Set adapters to RecyclerViews
    rv1.setAdapter(myarray1);
    rv2.setAdapter(myarray2);
    rv3.setAdapter(myarray3);
}

@Override
public boolean onNavigationItemSelected(@NonNull MenuItem item) {
    int id = item.getItemId();
    if (id == R.id.nav_main) {
        startActivity(new Intent(this, MainActivity.class));
    } else if (id == R.id.nav_thithusathach) {
        startActivity(new Intent(this, ExamListActivity.class));
    } else if (id == R.id.nav_hoclythuyet) {
        startActivity(new Intent(this, LearningActivity.class));
    } else if (id == R.id.nav_bienbao) {
        drawerLayout.closeDrawer(GravityCompat.START);
    } else if (id == R.id.nav_tracuuluat) {
        startActivity(new Intent(this, LawListActivity.class));
    } else if (id == R.id.nav_thoat) {
        exit();
    }
}

```

```
    }  
    drawerLayout.closeDrawer(GravityCompat.START);  
    return true;  
}  
  
private void exit() {  
    finishAffinity();  
    System.exit(0);  
}  
}
```