

**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 (Java Console App)

Mã Sinh Viên	Họ và Tên	Ngày Sinh	Lớp
2051063778	Phạm Công Khanh	03/01/2002	62TH3
2051063799	Đinh Quang Thao	02/11/2002	62TH3
2151062761	Lê Văn Hà	16/02/2003	63CNTT4
2151060279	Nguyễn Tiến Lực	06/09/2003	63CNTT4

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 project 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	6
1.1. Giới thiệu	6
1.2. Chức năng chính	6
1.3. Yêu cầu phi chức năng	6
CHƯƠNG 2. PHÂN TÍCH YÊU CẦU VÀ THIẾT KẾ HỆ THỐNG	7
2.1. Phân tích yêu cầu:	7
2.2. Thiết kế hệ thống:	7
2.3. Triển khai:	11
2.4. Vận hành và bảo trì:.....	12
CHƯƠNG 3. KẾT QUẢ THỰC HIỆN	13
3.1. Công nghệ đã sử dụng.....	13
3.2. Tiến độ thực hiện	13
3.3. Hình ảnh sản phẩm	16
KẾT LUẬN	34
DANH MỤC TÀI LIỆU THAM KHẢO.....	35

MỤC LỤC HÌNH ẢNH

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

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:

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.

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.

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:** Hiển 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:

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:

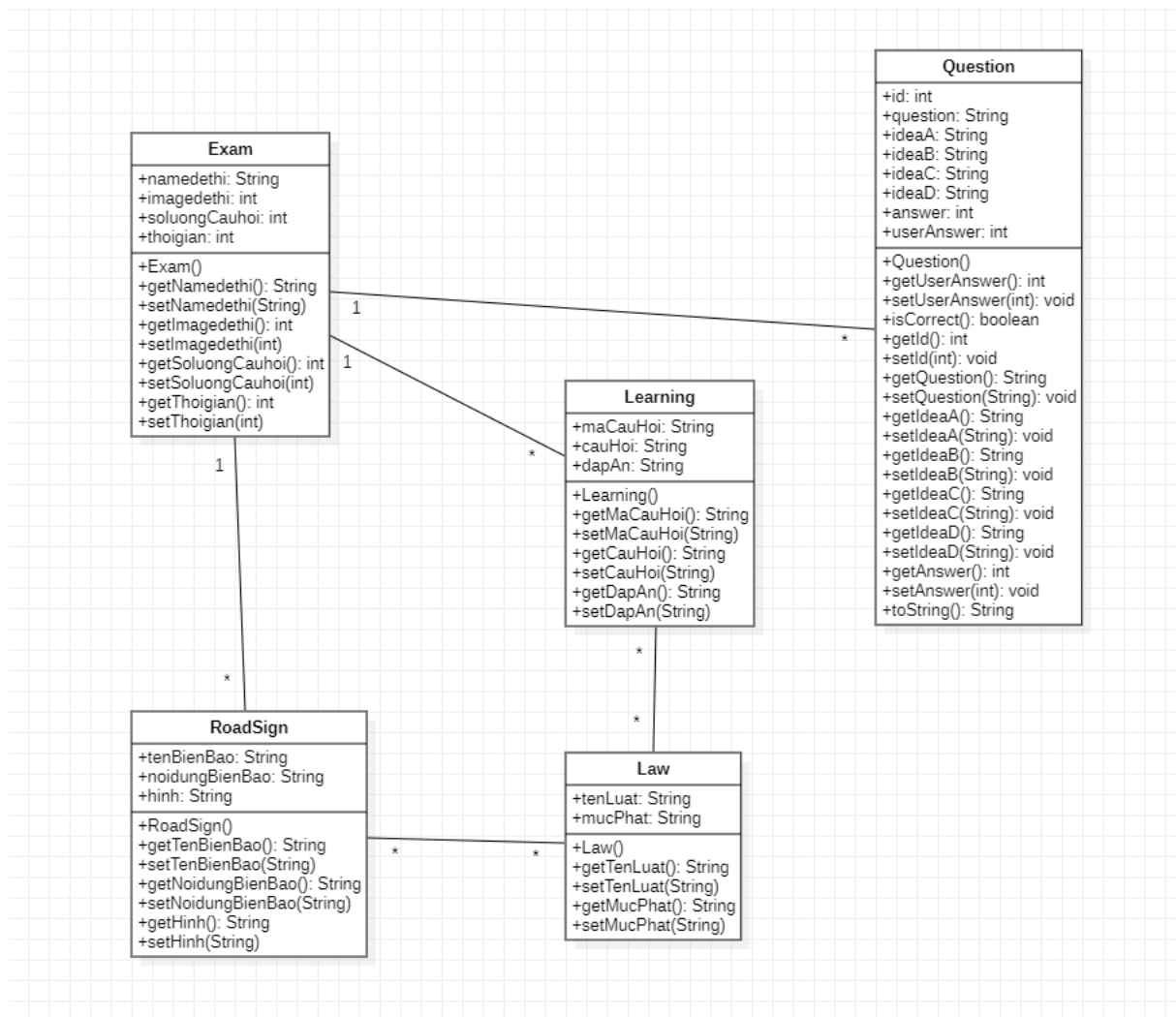
- String maCauHoi: Mã câu hỏi
- String cauHoi: Câu hỏi
- String dapAn: Đáp án
- Phương thức:
 - getMaCauHoi, setMaCauHoi
 - getCauHoi, setCauHoi
 - getDapAn, setDapAn
- **Lớp RoadSign (Biển báo):**
 - Thuộc tính:
 - String tenBienBao: Tên biển báo
 - String noidungBienBao: Nội dung biển báo
 - String 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

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.

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 `tenLuat`, `mucPhat`. Thực hiện chức năng tra cứu luật và thông tin các hình phạt nếu vi phạm.



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 BoLuot.
- Xử lý dữ liệu thi thử sát hạch, ôn tập lý thuyết, biên báo và tra cứu luật.
- Phát triển các tính năng như thi thử, ôn tập lý thuyết, và hiển thị thông tin biên 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 tối 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 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.

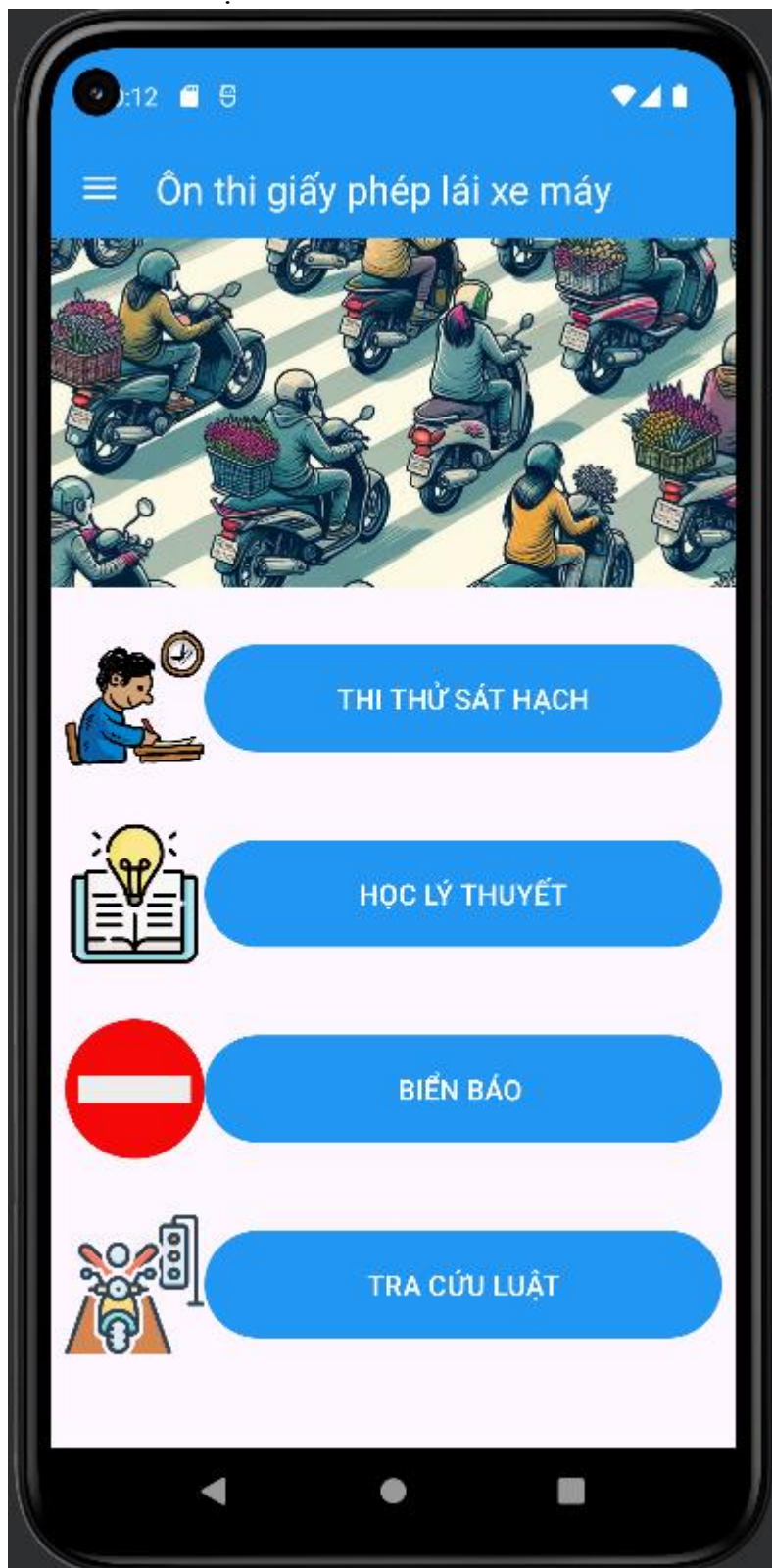
Bước 9. Triển khai:

- 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.

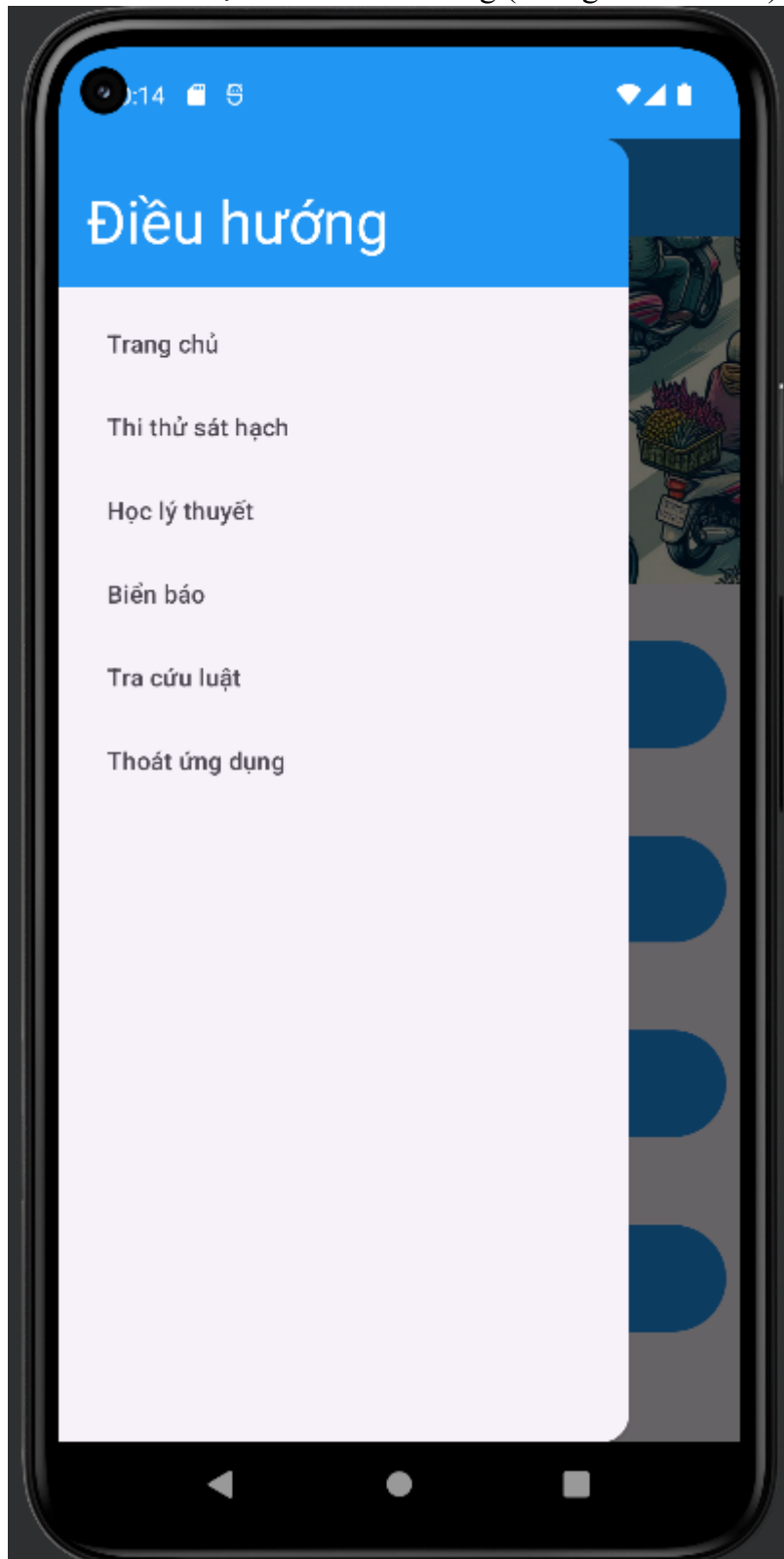
3.3. Hình ảnh sản phẩm

a. Phạm Công Khanh:

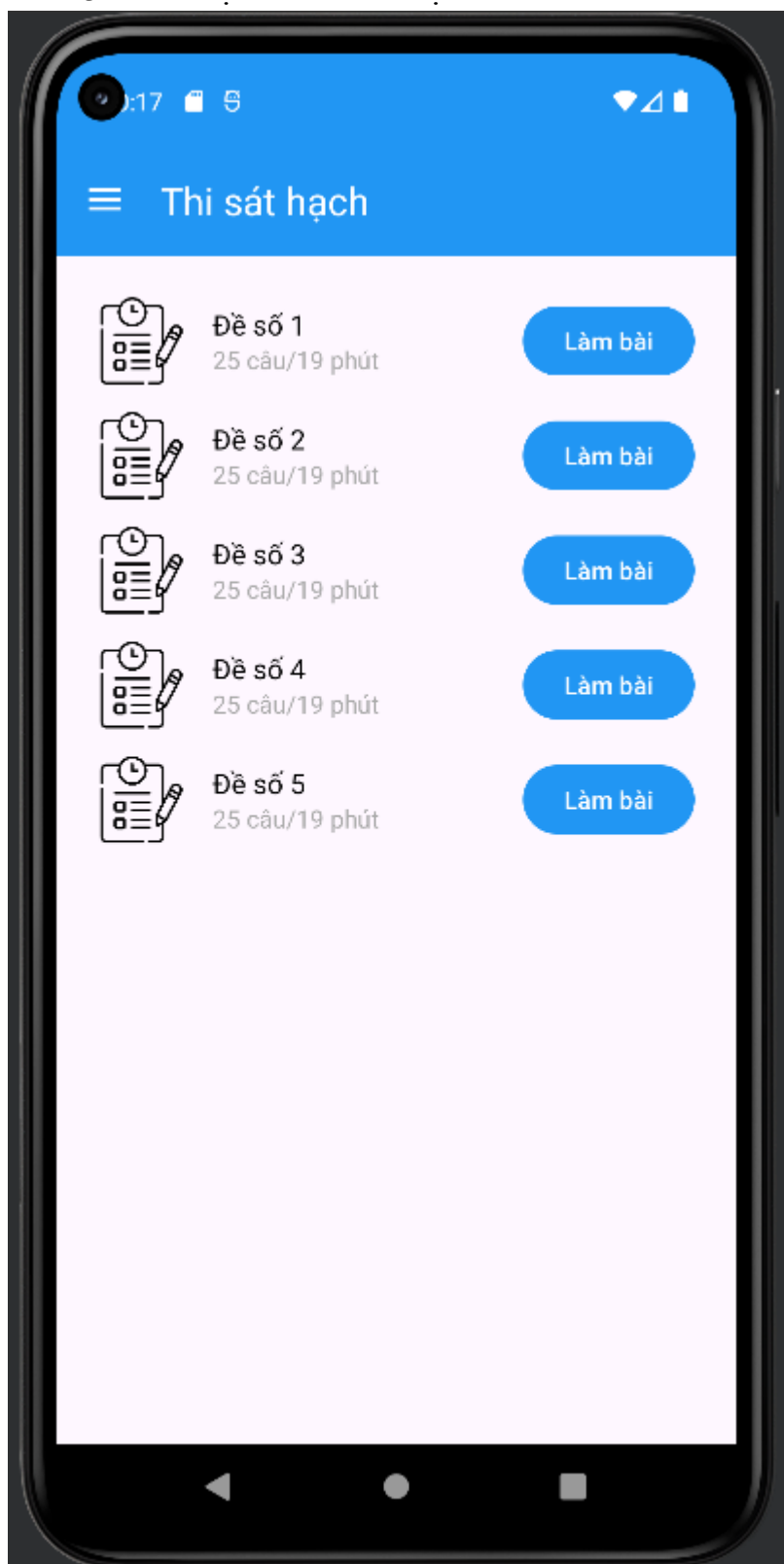
- Hình 1: Giao diện màn hình chính



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



- Hình 3: Giao diện Thi thử sát hạch



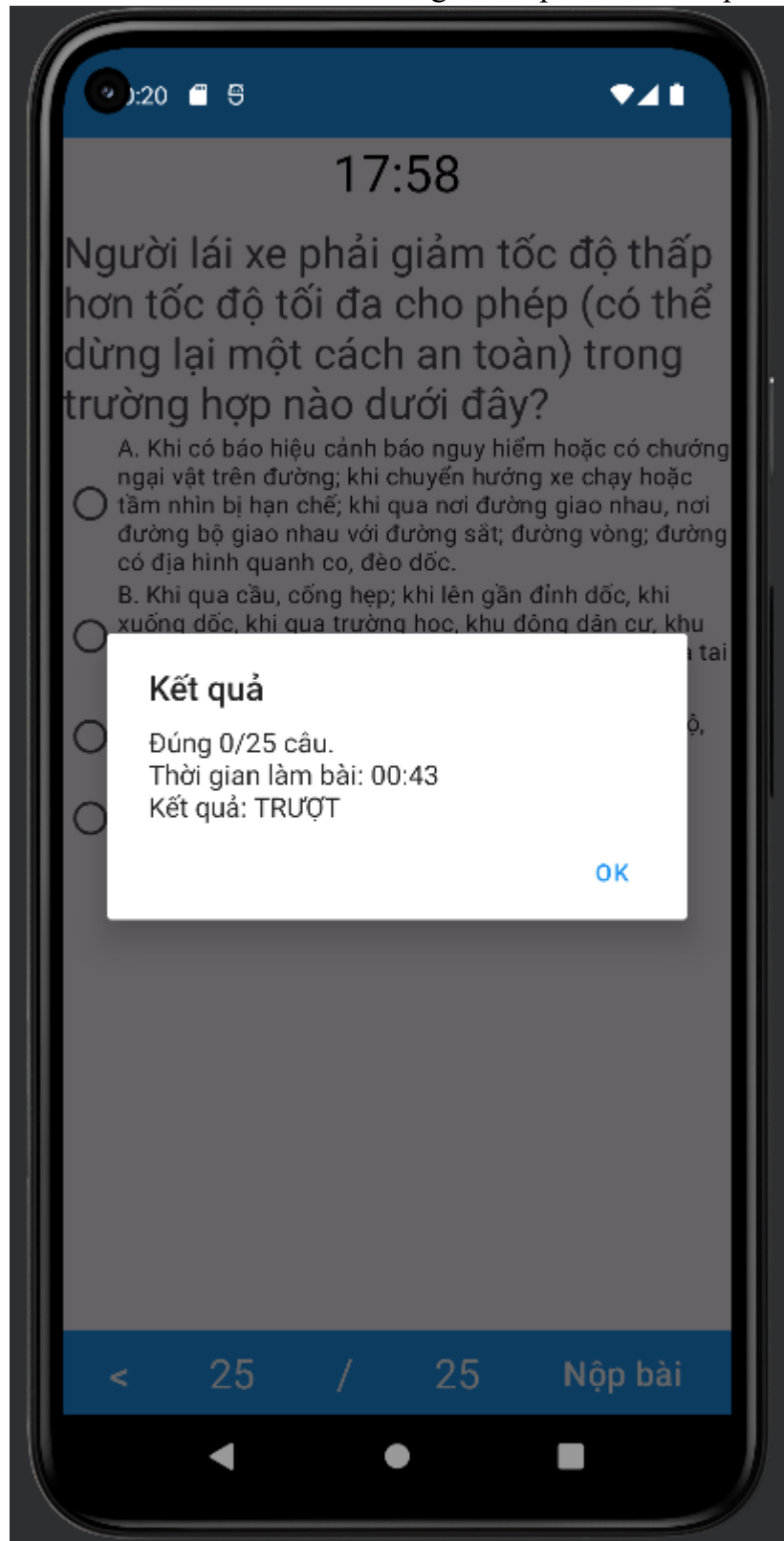
- Hình 4: Giao diện hiển thị lên dialog khi nhấn vào nút “Làm bài”



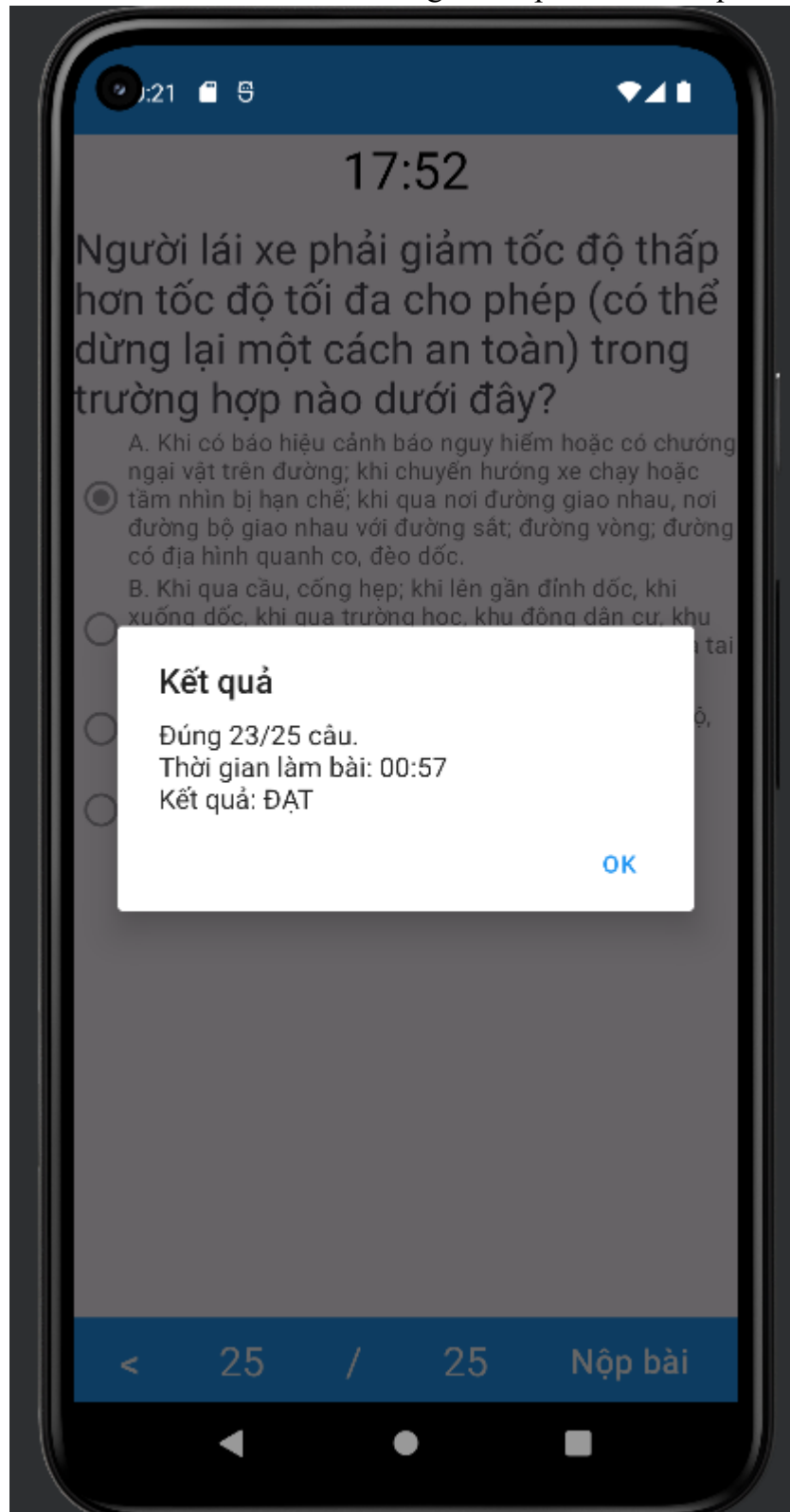
- Hình 5: Giao diện làm bài thi thử



- 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



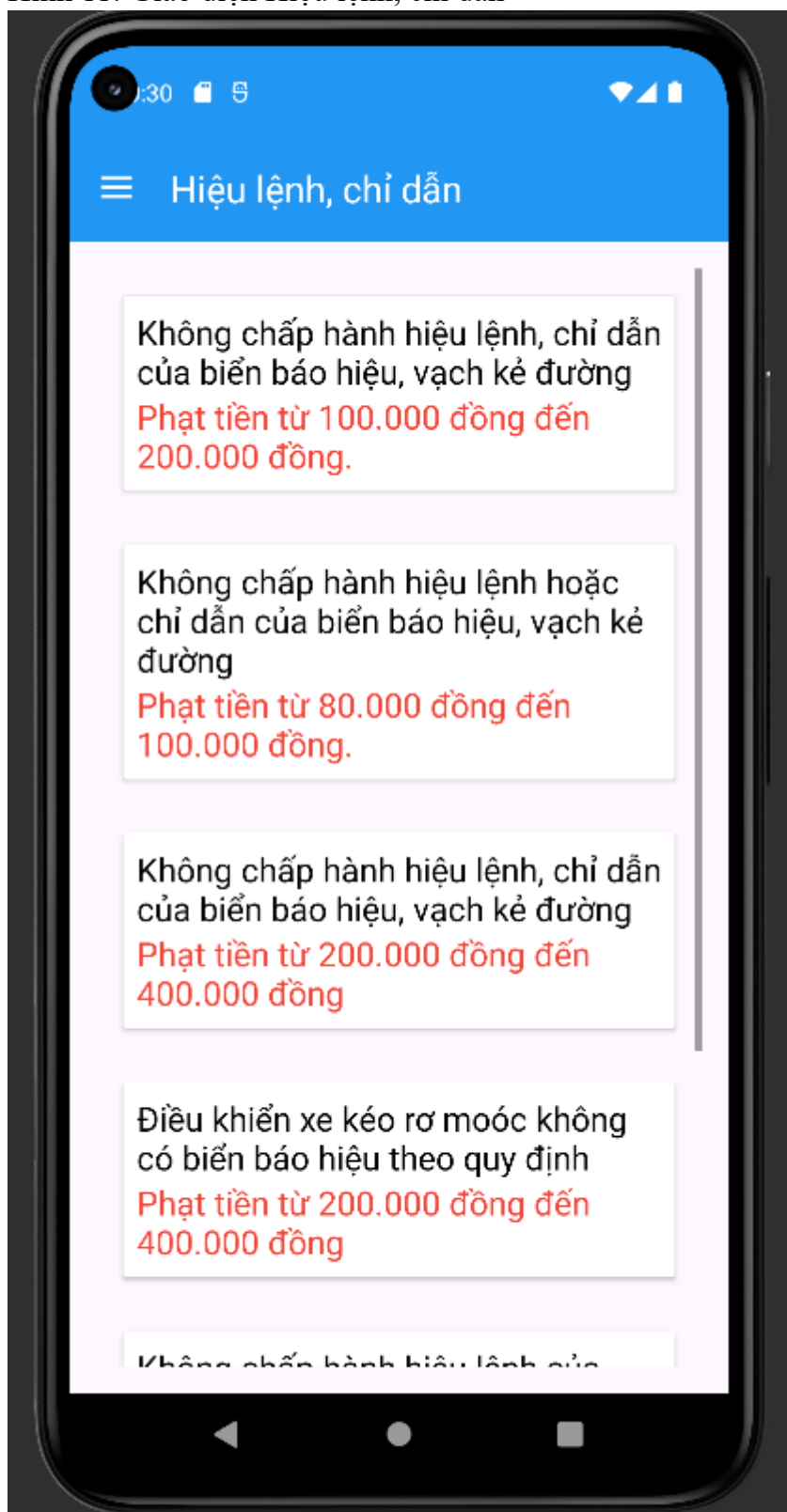
- Hình 9: Giao diện tra cứu luật



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

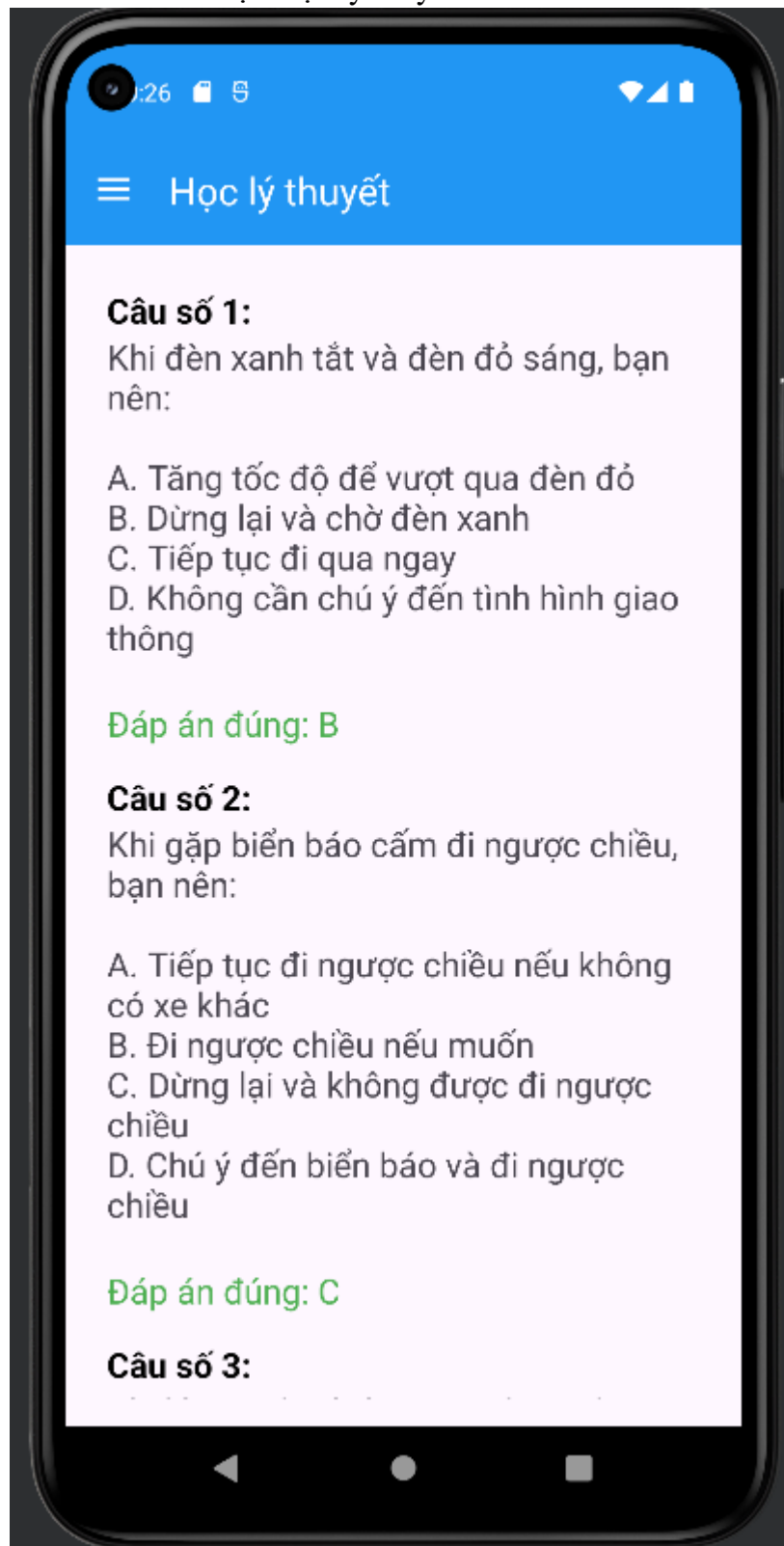


- Hình 11: Giao diện Hiệu lệnh, chỉ dẫn



b. Nguyễn Tiên Lược:

- Hình 12: Giao diện Học lý thuyết



c. Lê Văn Hà:

- Hình 13: Giao diện Biển báo nguy hiểm

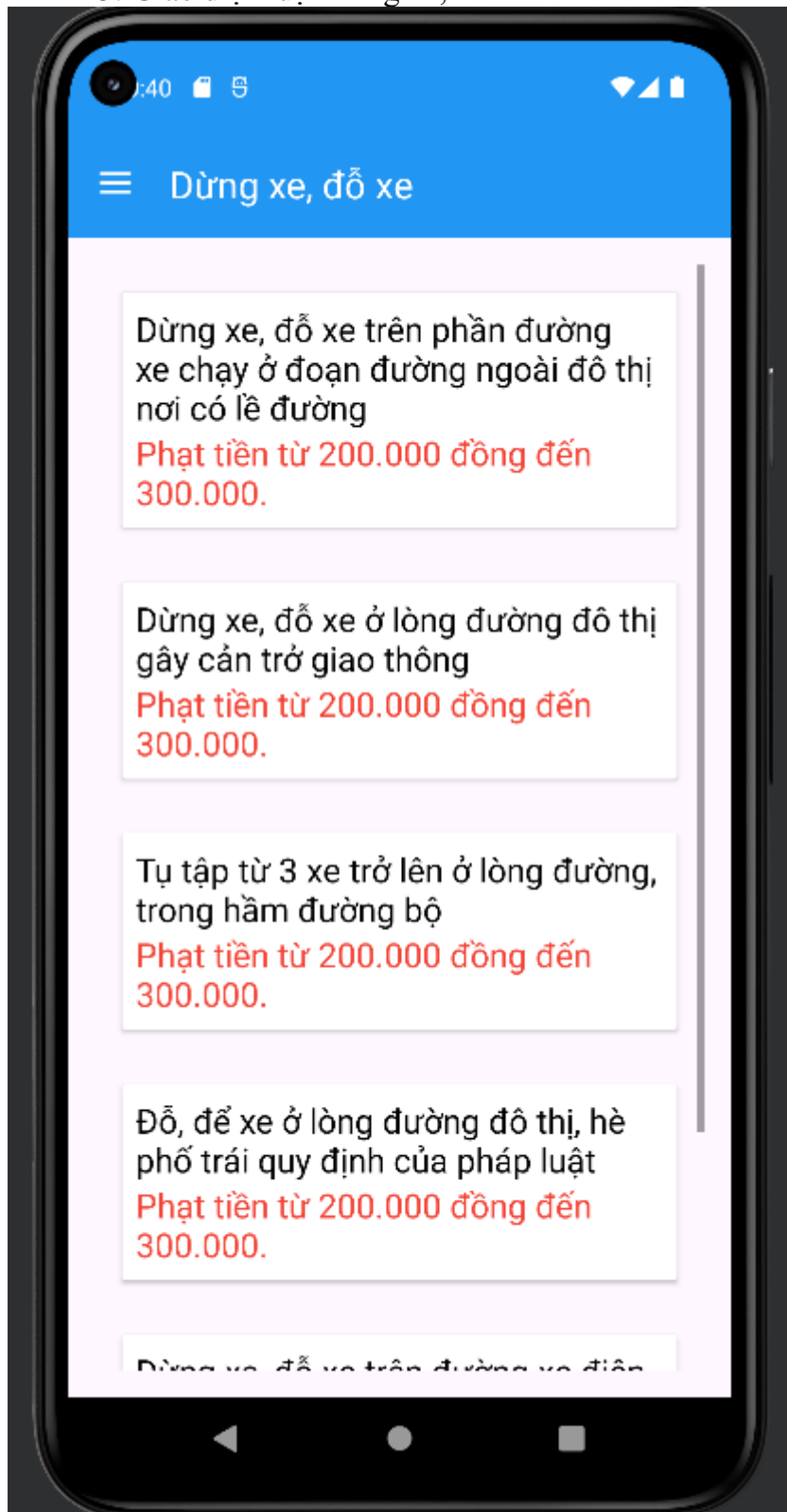


- Hình 14: Giao diện biển báo hiệu lệnh



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>