BỘ NÔNG NGHIỆP VÀ PHÁT TRIỂN NÔNG THÔN

**TRƯỜNG ĐẠI HỌC THỦY LỢI**

A blue and white logo

Description automatically generated

**BÀI TẬP LỚN**

**HỌC PHẦN: PHÁT TRIỂN DỰ ÁN DI ĐỘNG**

**ĐỀ TÀI: Xây dựng ứng dụng luyện thi bằng lái xe (Java DriverExam App)**

|  |  |  |
| --- | --- | --- |
| Mã Sinh Viên | Họ và Tên | Lớp |
| 2151170553 | Trần Trường Long | 63KTPM1 |
| 2151173798 | Nguyễn Đức Minh | 63KTPM1 |
| 2151173821 | Hoàng Đăng Tú Tài | 63KTPM1 |

### 

**Hà Nội, năm 2024**

BỘ NÔNG NGHIỆP VÀ PHÁT TRIỂN NÔNG THÔN

**TRƯỜNG ĐẠI HỌC THỦY LỢI**

A blue and white logo

Description automatically generated

**BÀI TẬP LỚN**

**HỌC PHẦN: LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG**

**ĐỀ TÀI: Xây dựng ứng dụng luyện thi bằng lái xe (Java DriverExam App)**

|  |  |  |  |
| --- | --- | --- | --- |
| Mã Sinh Viên | Họ và Tên | Điểm | |
| Bằng Số | Bằng Chữ |
| 2151170553 | Trần Trường Long |  |  |
| 2151173798 | Nguyễn Đức Minh |  |  |
| 2151173821 | Hoàng Đăng Tú Tài |  |  |

### 

### CÁN BỘ CHẤM THI 1 CÁN BỘ CHẤM THI 2

**Hà Nội, năm 2024**

**LỜI NÓI ĐẦU**

Trong bối cảnh giao thông ngày càng phát triển và trở nên phức tạp, việc sở hữu giấy phép lái xe không chỉ là một nhu cầu cần thiết mà còn là một trách nhiệm lớn lao đối với mỗi người lái. Nhằm hỗ trợ người dùng trong quá trình ôn tập và chuẩn bị cho kỳ thi cấp giấy phép lái xe hạng B1, B2 tại Việt Nam, chúng tôi đã phát triển ứng dụng "Thi Bằng Lái Xe Ô Tô".

Ứng dụng được thiết kế với giao diện thân thiện và dễ sử dụng, giúp người dùng dễ dàng tiếp cận và làm quen với các dạng câu hỏi trắc nghiệm. Bên cạnh việc cung cấp các bài thi thử, ứng dụng còn tích hợp nhiều tính năng hữu ích như lưu lịch sử thi, giải thích chi tiết cho mỗi câu hỏi, và phân tích các câu hỏi người dùng đã làm sai.

Chúng tôi hy vọng rằng ứng dụng này sẽ trở thành một công cụ hữu hiệu, hỗ trợ người dùng không chỉ trong việc ôn tập kiến thức lý thuyết giao thông mà còn giúp nâng cao ý thức và trách nhiệm khi tham gia giao thông. Với 600 câu hỏi trắc nghiệm chính thức và nhiều chế độ thi đa dạng, "Thi Bằng Lái Xe Ô Tô" sẽ mang đến cho người dùng những trải nghiệm học tập thú vị và hiệu quả.

**MỤC LỤC**

**MỤC LỤC HÌNH ẢNH**

**BẢNG PHÂN CHIA CÔNG VIỆC THEO CHỨC NĂNG CHÍNH:**

|  |  |  |
| --- | --- | --- |
| **STT** | **THÀNH VIÊN** | **CHỨC NĂNG ĐẢM NHẬN** |
| **1** | **Trần Trường Long** | * **Thi theo đề ngẫu nhiên** * **Thi theo bộ đề** |
| **2** | **Nguyễn Đức Minh** | * **Các câu bị sai** * **Các biển báo** * **60 câu điểm liệt** |
| **3** | **Hoàng Đăng Tú Tài** | * **Ôn tập theo câu hỏi** * **Mẹo ghi nhớ** * **Top 50 câu sai** |

**BẢNG CÁC TỪ VIẾT TẮT**

|  |  |  |
| --- | --- | --- |
| **STT** | **TỪ VIẾT TẮT** | **VIẾT ĐẦY ĐỦ** |
| 1 | CSDL | Cơ sở dữ liệu |

**CHƯƠNG 1. MÔ TẢ BÀI TOÁN**

**1.1. Giới thiệu**

Ứng dụng Thi Bằng Lái Xe Ô Tô là một phần mềm di động được phát triển nhằm hỗ trợ người dùng ôn tập và chuẩn bị cho kỳ thi cấp giấy phép lái xe hạng B1, B2 tại Việt Nam. Phần mềm này cung cấp các tính năng quan trọng như làm bài thi thử, xem danh sách các câu hỏi trắc nghiệm, lưu lịch sử thi và giải thích chi tiết cho mỗi câu hỏi.

Với giao diện thân thiện và dễ sử dụng, ứng dụng cho phép người dùng dễ dàng lựa chọn các bộ câu hỏi từ 600 câu trắc nghiệm chính thức. Ứng dụng cũng hỗ trợ sinh ngẫu nhiên câu hỏi và đáp án, giúp người dùng làm quen với nhiều dạng đề khác nhau, từ đó nắm vững kiến thức lý thuyết giao thông.

**1.2. Chức năng chính**

1. Thi đề ngẫu nhiên

Ứng dụng cung cấp danh sách đầy đủ 600 câu hỏi trắc nghiệm chính thức, được sử dụng trong kỳ thi cấp giấy phép lái xe tại Việt Nam. Chúng được chia ngẫu nhiên thành các bộ đề và tính năng này cho phép người dùng làm bài kiểm tra với 1 bộ đề bất kì. Khi làm sẽ có giới hạn thời gian tổng thể, người dùng có thể nộp bài bất cứ khi nào chứ không bắt buộc làm hết. Sau khi nộp bài sẽ trả về kết quả xem người dùng có hoàn thành bài thi hay không, với 1 số tiêu chí như không được làm sai câu điểm liệt, số câu đúng phải thỏa mãn với yêu cầu tối thiểu số câu đúng của bài thi.

2. Thi Theo bộ đề

Trái nghiệm với tính năng thi theo đề ngẫu nhiên, thì thi theo bộ đề lại cho phép người dùng có thể tự chọn 1 bộ đề và làm cho đến khi thông thạo nó. Nhìn chung thì hai tính năng này giống như hai dạng luyện tập để người dùng có thể tự do lựa chọn cho phù hợp với phương pháp học của họ. Khi làm sẽ có giới hạn thời gian tổng thể, người dùng có thể nộp bài bất cứ khi nào chứ không bắt buộc làm hết. Sau khi nộp bài sẽ trả về kết quả xem người dùng có hoàn thành bài thi hay không, với 1 số tiêu chí như không được làm sai câu điểm liệt, số câu đúng phải thỏa mãn với yêu cầu tối thiểu số câu đúng của bài thi.

3. Các câu bị sai

Hệ thống sẽ lưu lại các câu hỏi người dùng đã làm sai vào CSDL. Khi người dùng chọn làm mục này , họ sẽ tập trung học những kiến thức hổng, giúp cho việc luyện tập toàn diện hơn.

4. Ôn tập câu hỏi

Nếu như hai tính năng Thi theo bộ đề ngẫu nhiên hoặc cụ thể, sẽ giúp người dùng có trải nghiệm thi thử, với số câu và thời gian tương đồng với khi thi thật. Thì tính năng này lại phân chia câu hỏi theo mức độ khó, nội dung và bối cảnh để phân thành những dạng bài cụ thể cho người dùng.

5. Các biển báo

Tập hợp danh sách các biển báo giao thông, có phân nhóm thành biển cảnh báo, biển cấm,.. Để người dùng tập trung vào những dạng bài có biển báo, vì đây cũng là 1 dạng bài khó.

6. Các mẹo ghi nhớ

Tập hợp các mẹo làm bài, ghi nhớ giúp người dùng có thể tiết kiệm thời gian khi học, tuy nhiên người dùng chỉ nên sử dụng tính năng này sau khi đã nắm cơ bản được kiến thức, không nên lợi dụng để học qua loa, học vẹt.

7. 60 Câu điểm liệt.

Tính năng này tập trung phát triển nhằm hướng đến 1 yêu cầu vô cùng quan trọng khi thi. Đó là người dùng bắt buộc phải hoàn thành đúng tất cả các câu điểm liệt trong đề, thì mới xét đến các tiêu chí tiếp theo để xét người dùng có qua hay không. Người dùng chọn tính năng này để ôn tâp 60 câu điểm liệt.

8. Top 50 câu dễ sai.

Tính năng này tập trung vào 50 câu hỏi mà đa số người dùng thường xuyên làm sai, dễ làm sai khi làm bài thi. Người dùng chọn tính năng này để tập trung ôn 50 câu dễ sai.

9. Lưu trữ dữ liệu.

Ứng dụng Thi Bằng Lái Xe Ô Tô sử dụng **SQLite** làm giải pháp lưu trữ dữ liệu chính. SQLite là một cơ sở dữ liệu nhúng nhẹ, cung cấp cách thức lưu trữ và truy xuất dữ liệu hiệu quả mà không cần máy chủ cơ sở dữ liệu phức tạp. Dữ liệu trong ứng dụng này bao gồm các câu hỏi, thuộc tính phân loại và các thông tin liên quan đến kết quả thi của người dùng.

**Ưu điểm của việc sử dụng SQLite**

* **Hiệu năng cao:** SQLite hoạt động rất nhanh và hiệu quả ngay cả với lượng dữ liệu lớn như 600 câu hỏi trắc nghiệm và nhiều thông tin phân loại.
* **Dễ dàng tích hợp:** Là một cơ sở dữ liệu nhúng, SQLite không yêu cầu máy chủ riêng biệt, do đó dễ dàng tích hợp vào các ứng dụng di động và hoạt động tốt ngay cả khi ngoại tuyến.
* **Bảo mật và riêng tư:** Dữ liệu được lưu trữ cục bộ trên thiết bị người dùng, đảm bảo tính bảo mật và không yêu cầu kết nối mạng để truy cập dữ liệu.
* **Khả năng mở rộng:** SQLite dễ dàng mở rộng cho các bản cập nhật sau này khi cần thêm nhiều câu hỏi, thuộc tính hoặc thông tin mới.

**1.3. Yêu cầu phi chức năng**

* Dễ Sử Dụng

Ứng dụng được thiết kế với giao diện trực quan và thân thiện với người dùng. Tất cả các tính năng như xem câu hỏi, làm bài thi thử, và xem lại lịch sử thi đều dễ dàng truy cập và thao tác. Người dùng ở mọi cấp độ kỹ năng, kể cả những người không quen thuộc với công nghệ, có thể nhanh chóng làm quen và sử dụng ứng dụng một cách dễ dàng.

* Hiệu Năng

Ứng dụng đảm bảo hoạt động mượt mà và nhanh chóng, ngay cả khi xử lý một lượng lớn dữ liệu câu hỏi và bài thi. Nhờ việc sử dụng Room Database để lưu trữ dữ liệu cục bộ, người dùng có thể truy cập danh sách câu hỏi và kết quả thi một cách nhanh chóng mà không cần chờ tải dữ liệu từ server.

* Độ Tin Cậy

Dữ liệu của người dùng như kết quả bài thi và lịch sử thi được lưu trữ an toàn trong cơ sở dữ liệu cục bộ. Ứng dụng đảm bảo rằng thông tin này sẽ không bị mất mát ngay cả khi ứng dụng bị tắt đột ngột hoặc thiết bị gặp sự cố. Việc lưu trữ cục bộ cũng giúp ứng dụng hoạt động mà không cần kết nối internet, nâng cao tính khả dụng và độ tin cậy.

* Tính Tương Thích

Ứng dụng được phát triển để tương thích với nhiều loại thiết bị và kích cỡ màn hình khác nhau, từ điện thoại di động đến máy tính bảng. Điều này đảm bảo rằng người dùng có thể sử dụng ứng dụng trên nhiều nền tảng mà không gặp phải vấn đề về giao diện hay trải nghiệm.

* Khả Năng Mở Rộng

Ứng dụng được thiết kế linh hoạt và dễ dàng mở rộng thêm các tính năng mới hoặc cập nhật nội dung câu hỏi khi cần. Điều này đảm bảo rằng ứng dụng có thể phát triển và cải thiện theo nhu cầu của người dùng trong tương lai.

**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 những người đang chuẩn bị tham gia kỳ thi cấp giấy phép lái xe hạng B1, B2 tại Việt Nam. Họ có thể là học sinh, sinh viên, người đi làm, hoặc bất kỳ ai có nhu cầu thi và ôn tập các câu hỏi lý thuyết về luật giao thông đường bộ.
* Đặc điểm người dùng: Ứng dụng cần dễ sử dụng và thân thiện với người dùng, vì người dùng có thể có các mức độ hiểu biết khác nhau về công nghệ và điện thoại di động. Ứng dụng phải đảm bảo rằng người dùng có thể dễ dàng điều hướng và truy cập các tính năng như xem câu hỏi, thi thử, và xem lại lịch sử thi.

**Thu thập yêu cầu:**

* Chức năng chính: Dựa trên mô tả bài toán, các chức năng chính của ứng dụng bao gồm:
* Thi theo đề ngẫu nhiên
* Thi theo bộ đề
* Làm các câu đã bị sai
* Ôn tập câu hỏi
* Các biển báo
* Các mẹo ghi nhớ
* Các câu hỏi có hình ảnh(nếu đề bài yêu cầu).
* Giới hạn thời gian cho bài thi theo đề.
* Lưu trữ dữ liệu.
* Yêu cầu phi chức năng: Ứng dụng cần đảm bảo tính dễ sử dụng, hiệu suất cao, độ tin cậy, tính tương thích với nhiều loại thiết bị và khả năng mở rộng trong tương lai.

**Phân tích yêu cầu:**

* **Thi theo bộ đề ngẫu nhiên:** Người dùng có thể truy cập tính năng này để thi ngẫu nhiên 1 bộ đề. Với mỗi câu hỏi bao gồm nội dung câu hỏi, các đáp án lựa chọn (A, B, C, D) và phần giải thích. Các câu hỏi có thể bao gồm hình ảnh minh họa.
* **Thi theo bộ đề cụ thể:** Người dùng có thể truy cập tính năng này để chọn và luyện tập cụ thể, để thông thạo từng bộ đề 1 cách nhanh chóng. Với mỗi câu hỏi bao gồm nội dung câu hỏi, các đáp án lựa chọn (A, B, C, D) và phần giải thích. Các câu hỏi có thể bao gồm hình ảnh minh họa.
* **Làm các câu đã bị sai:** Người dùng có thể truy cập tính năng này để chọn và luyện tập tập trung vào các câu đã làm sai trước đây, phù hợp để khắc phục lỗ hổng kiến thức. Với mỗi câu hỏi bao gồm nội dung câu hỏi, các đáp án lựa chọn (A, B, C, D) và phần giải thích. Các câu hỏi có thể bao gồm hình ảnh minh họa.
* **Ôn tập câu hỏi theo dạng:** Người dùng có thể truy cập tính năng này để chọn và luyện tập cụ thể theo từng dạng câu hỏi. Các câu hỏi sẽ phân loại theo độ khó, hoặc dạng bài.Phù hợp với những người muốn luyện tập theo dạng bài cụ thể. Với mỗi câu hỏi bao gồm nội dung câu hỏi, các đáp án lựa chọn (A, B, C, D) và phần giải thích. Các câu hỏi có thể bao gồm hình ảnh minh họa.
* **Các biển báo:** Người dùng có thể truy cập tính năng này để học và ghi nhớ hệ thống biển báo giao thông, các biển báo có phân loại thành các loại như (biển báo cấm, biển báo nguy hiểm, biển báo hiệu lệnh, biển báo chỉ dẫn,…)
* **Các mẹo ghi nhớ:** Tổng hợp những mẹo để người dùng có thể học, hoặc làm bài thi 1 cách nhanh chóng hơn, tiết kiệm thời gian khi luyện tập và thực hành.
* **60 Câu điểm liệt:** Tính năng này tập trung phát triển nhằm hướng đến 1 yêu cầu vô cùng quan trọng khi thi. Đó là người dùng bắt buộc phải hoàn thành đúng tất cả các câu điểm liệt trong đề, thì mới xét đến các tiêu chí tiếp theo để xét người dùng có qua hay không. Người dùng chọn tính năng này để ôn tâp 60 câu điểm liệt. Với mỗi câu hỏi bao gồm nội dung câu hỏi, các đáp án lựa chọn (A, B, C, D) và phần giải thích. Các câu hỏi có thể bao gồm hình ảnh minh họa.
* **Top 50 câu dễ sai:** Tính năng này tập trung vào 50 câu hỏi mà đa số người dùng thường xuyên làm sai, dễ làm sai khi làm bài thi. Người dùng chọn tính năng này để tập trung ôn 50 câu dễ sai. Với mỗi câu hỏi bao gồm nội dung câu hỏi, các đáp án lựa chọn (A, B, C, D) và phần giải thích. Các câu hỏi có thể bao gồm hình ảnh minh họa.
* **Lưu trữ dữ liệu:** Ứng dụng lưu trữ các câu hỏi thi, thông tin phân loại, lịch sử thi và các câu trả lời sai vào cơ sở dữ liệu SQLite để dễ dàng truy xuất và xử lý trong các chức năng khác nhau.

Các bảng dữ liệu chính trong cơ sở dữ liệu SQLite của ứng dụng bao gồm:

**Bảng câu hỏi**: Lưu trữ 600 câu hỏi chính thức từ file JSON bao gồm nội dung câu hỏi, các đáp án, và đáp án đúng.

**Các cột chính:** ID câu hỏi, nội dung câu hỏi, danh sách đáp án, đáp án đúng, loại câu hỏi (ví dụ: lý thuyết, sa hình, biển báo).

**Bảng thuộc tính câu hỏi**: Lưu các thuộc tính phân loại của câu hỏi như nhóm biển báo, câu điểm liệt, mức độ khó, v.v.

**Các cột chính:** ID câu hỏi, loại câu hỏi, mức độ khó, thuộc tính đặc biệt (ví dụ: câu điểm liệt).

**Bảng kết quả thi**: Lưu trữ kết quả thi của người dùng, bao gồm các câu trả lời đúng và sai, số câu sai.

**Các cột chính:** ID thi, ID câu hỏi, trạng thái trả lời (đúng hoặc sai), thời gian thi, điểm số cuối cùng.

**Bảng câu hỏi sai**: Lưu lại danh sách các câu hỏi mà người dùng đã làm sai để họ có thể ôn tập sau.

**Các cột chính:** ID câu hỏi, số lần sai, lần thi gần nhất.

**2.2. Thiết kế hệ thống:**

**2.2.1. Biểu đồ lớp cho mô hình miền**

Biểu đồ lớp mô hình miền tập trung vào các **thực thể nghiệp vụ chính** của ứng dụng, bao gồm các lớp đại diện cho câu hỏi, loại bài thi, và dữ liệu người dùng. Dưới đây là mô tả cho các thực thể chính:

**Lớp chính trong mô hình miền:**

* **Question**: Đại diện cho một câu hỏi trắc nghiệm trong ứng dụng.
  + Thuộc tính: id, questionText, answers, options ,suggest, image.
* **DriverType**: Đại diện cho loại bằng lái xe (B1, B2,...).
  + Thuộc tính: id, title, quantity, description.
* **Exam**: Đại diện cho một bài thi, có thể bao gồm nhiều câu hỏi.
  + Thuộc tính: id, questions[], result, timeLimit
* **User**: Đại diện cho thông tin người dùng, có thể được mở rộng.
  + Thuộc tính: id, name, savedWrongQuestions[]
* **Test**: Đại diện cho một bộ đề thi hoặc bài kiểm tra ngẫu nhiên.
  + Thuộc tính: id, questions[], timeLimit, score

Các thực thể này sẽ có **mối quan hệ** với nhau. Ví dụ, một Exam có nhiều Question, một Test có thể liên kết với Exam và Question.

**2.2.2. Biểu đồ lớp cho MVC**

Biểu đồ lớp MVC mô tả kiến trúc tổng thể của ứng dụng, bao gồm các lớp thuộc về **Model (nghiệp vụ)**, **View (giao diện)**, và **Controller (điều khiển)**.

**Các lớp trong MVC:**

* **Model**:
  + **Question**: Lớp mô tả câu hỏi, bao gồm thông tin về câu hỏi, đáp án, hình ảnh liên quan.
  + **DriverType**: Lớp mô tả các loại bằng lái xe.
  + **Exam**: Lớp mô tả một bài thi, bao gồm danh sách câu hỏi và kết quả thi.
  + **User**: Lớp đại diện cho người dùng và lưu trữ câu hỏi sai.
* **View**:
  + **DriverTypeActivity**: Hiển thị danh sách loại bằng lái xe để người dùng chọn.
  + **ExamsActivity**: Hiển thị danh sách bài thi và cho phép người dùng bắt đầu thi.
  + **TestActivity**: Hiển thị câu hỏi, đáp án trong quá trình làm bài thi.
  + **ResultActivity**: Hiển thị kết quả thi sau khi hoàn thành.
  + **QuestionAdapter**: Hiển thị danh sách câu hỏi và các tuỳ chọn trả lời.
* **Controller**:
  + **MainActivity**: Điều hướng chính của ứng dụng, khởi tạo các hoạt động và chuyển đổi giữa các màn hình.
  + **JsonReader**: Đọc dữ liệu từ file JSON chứa danh sách câu hỏi và các thuộc tính liên quan.
  + **DatabaseHelper**: Xử lý việc lưu trữ và truy vấn dữ liệu trong SQLite.
  + **DriverTypeAdapter**: Xử lý việc hiển thị và tương tác với danh sách loại bằng lái.

**Mô tả chi tiết biểu đồ MVC:**

* **Model**: Gồm các lớp Question, DriverType, Exam, và User – chịu trách nhiệm lưu trữ và xử lý dữ liệu nghiệp vụ chính.
* **View**: Các lớp giao diện như DriverTypeActivity, ExamsActivity, TestActivity sẽ hiển thị dữ liệu từ model và nhận tương tác từ người dùng.
* **Controller**: Các lớp như MainActivity, JsonReader, và DatabaseHelper đảm nhiệm việc điều khiển luồng dữ liệu giữa Model và View, cũng như xử lý thao tác của người dùng.

**Dựa trên kiến trúc MVC đã chọn cho ứng dụng ôn thi GPLX, đây là các lớp có thể xác định:**

**Lớp Question:**

* **Thuộc tính**:
  + private String id: Mã định danh câu hỏi
  + private String questionText: Nội dung câu hỏi
  + private Options options: Các lựa chọn
  + private String answer: Đáp án
  + private String suggest: Gợi ý (nếu có)
  + private Image image: Hình ảnh đi kèm (nếu có)
* **Phương thức**:
  + Getters và setters cho các thuộc tính.
  + toString(): trả về chuỗi mô tả câu hỏi.

**Lớp DriverType:**

* **Thuộc tính**:
  + private int id: Mã định danh loại xe
  + private String title: Tiêu đề loại xe
  + private int quantity: Số lượng.
  + private String description: Mô tả loại xe
* **Phương thức**:
  + Getters và setters cho các thuộc tính.
  + toString(): trả về chuỗi mô tả loại xe.

**Lớp Exam:**

* **Thuộc tính**:
  + int id: ID của bài thi.
  + List<Question> questions: Danh sách các câu hỏi trong bài thi.
  + int timeLimit: Thời gian tối đa để hoàn thành bài thi.
  + int score: Điểm số đạt được.
* **Phương thức**:
  + addQuestion(Question question): Thêm câu hỏi vào bài thi.
  + removeQuestion(int questionId): Xóa câu hỏi theo ID.
  + calculateScore(): Tính điểm số dựa trên câu trả lời.

**Lớp User:**

* **Thuộc tính**:
  + int id: ID người dùng.
  + String name: Tên người dùng.
  + List<Question> savedWrongQuestions: Danh sách các câu hỏi người dùng đã làm sai.
* **Phương thức**:
  + addWrongQuestion(Question question): Thêm câu hỏi sai vào danh sách.
  + getSavedWrongQuestions(): Lấy danh sách các câu hỏi sai.

**Lớp QuestionAdapter:**

* **Mô tả**: Lớp QuestionAdapter là một lớp con của RecyclerView.Adapter, chịu trách nhiệm hiển thị danh sách các câu hỏi trong ứng dụng ôn thi GPLX. Nó quản lý cách các câu hỏi được hiển thị trong từng mục (item) của RecyclerView, bao gồm câu hỏi, hình ảnh minh họa và các tùy chọn trả lời.
* **Thuộc tính**:
  + questions: Một danh sách các đối tượng Question, mỗi đối tượng đại diện cho một câu hỏi trong bộ đề thi.
  + selectedAnswers: Một HashMap lưu trữ câu trả lời đã chọn cho mỗi câu hỏi. Khóa là ID câu hỏi, và giá trị là đáp án đã chọn (ví dụ: "a", "b",...).
  + criticals: Danh sách các câu hỏi điểm liệt (câu hỏi quan trọng) trong bài thi, giúp người dùng tập trung vào các câu hỏi này.
* **Phương thức**:
  + **Constructor**:
    - Nhận vào danh sách các câu hỏi và danh sách câu hỏi điểm liệt. Dữ liệu này được sử dụng để hiển thị câu hỏi và tùy chọn trả lời trong giao diện người dùng.
  + onCreateViewHolder(@NonNull ViewGroup parent, int viewType):
    - Tạo và trả về một đối tượng QuestionViewHolder mới, được liên kết với layout của một câu hỏi (XML question\_item).
  + onBindViewHolder(@NonNull QuestionViewHolder holder, int position):
    - Gán dữ liệu câu hỏi cho các thành phần trong view của RecyclerView, bao gồm câu hỏi, hình ảnh và các tùy chọn trả lời. Cũng thực hiện việc lắng nghe sự kiện thay đổi trên nhóm các RadioButton để lưu lại lựa chọn của người dùng.
    - Sử dụng thư viện Glide để tải ảnh từ URL vào ImageView, hoặc ẩn ImageView nếu không có ảnh.
    - Hiển thị và cập nhật các tùy chọn trả lời từ các thuộc tính của đối tượng Question (các tùy chọn A, B, C, D, E).
  + getSelectedKey(String selectedAnswerText, QuestionViewHolder holder):
    - Trả về khóa (key) của tùy chọn trả lời dựa trên văn bản của RadioButton được chọn. Ví dụ: nếu đáp án A được chọn, hàm này sẽ trả về "a".
  + getItemCount():
    - Trả về số lượng câu hỏi trong danh sách questions.
  + getSelectedAnswers():
    - Trả về HashMap chứa tất cả các câu trả lời mà người dùng đã chọn trong bài thi.

**Lớp DatabaseHelper:**

* **Mô tả**: Lớp DatabaseHelper quản lý cơ sở dữ liệu SQLite trong ứng dụng, bao gồm việc tạo, nâng cấp bảng, và thực hiện các thao tác lưu trữ, xóa, và truy xuất dữ liệu về các câu hỏi sai mà người dùng trả lời.
* **Thuộc tính**:
  + DATABASE\_NAME: Hằng số chứa tên của cơ sở dữ liệu, giá trị là "quiz.db".
  + TABLE\_NAME: Hằng số chứa tên bảng, giá trị là "incorrect\_answers".
  + COL\_1: Hằng số đại diện cho cột ID, được sử dụng làm khóa chính của bảng.
  + COL\_2: Hằng số đại diện cho cột QUESTION\_ID, lưu trữ ID của câu hỏi mà người dùng trả lời sai.
* **Phương thức**:
  + onCreate(SQLiteDatabase db):
    - Tạo bảng incorrect\_answers với hai cột:
      * ID: Kiểu INTEGER, là khóa chính, tự động tăng.
      * QUESTION\_ID: Kiểu INTEGER, lưu ID của câu hỏi.
    - Phương thức này được gọi khi cơ sở dữ liệu được tạo lần đầu tiên.
  + onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion):
    - Xóa bảng incorrect\_answers nếu đã tồn tại, sau đó gọi lại phương thức onCreate để tạo bảng mới. Được gọi khi có sự thay đổi phiên bản của cơ sở dữ liệu.
  + insertIncorrectAnswer(int questionId):
    - Chèn một dòng dữ liệu mới vào bảng incorrect\_answers, lưu trữ ID của câu hỏi sai vào cột QUESTION\_ID.
  + deleteCorrectAnswer(int questionId):
    - Xóa một dòng dữ liệu khỏi bảng incorrect\_answers dựa trên QUESTION\_ID khi người dùng trả lời đúng câu hỏi.
  + getIncorrectAnswers():
    - Truy xuất danh sách các ID câu hỏi mà người dùng trả lời sai từ bảng incorrect\_answers. Kết quả trả về dưới dạng danh sách ArrayList<Integer> chứa các ID câu hỏi.

**Lớp MainActivity:**

* **Mô tả**: Lớp MainActivity là lớp chính của ứng dụng, chịu trách nhiệm điều khiển giao diện màn hình chính (home screen) và xử lý các sự kiện khi người dùng tương tác với các nút chức năng. Các nút trên giao diện chính cho phép người dùng truy cập vào các chế độ ôn thi khác nhau trong ứng dụng, bao gồm thi ngẫu nhiên, thi theo bộ đề, câu điểm liệt, và các câu dễ sai.
* **Phương thức**:
  + **onCreate(Bundle savedInstanceState)**

**Mối quan hệ giữa các lớp:**

* MainActivity sử dụng Exam và User để quản lý thông tin bài thi và người dùng.
* Exam chứa danh sách các đối tượng Question.
* DatabaseHelper được sử dụng để lưu trữ và truy xuất câu hỏi từ cơ sở dữ liệu.
* QuestionAdapter hiển thị thông tin câu hỏi cho người dùng.
* Các lớp giao diện:
  + DriverTypeActivity: Hiển thị danh sách loại bằng lái xe để người dùng chọn.
  + ExamsActivity: Hiển thị danh sách bài thi và cho phép người dùng bắt đầu thi.
  + TestActivity: Hiển thị câu hỏi, đáp án trong quá trình làm bài thi.
  + ResultActivity: Hiển thị kết quả thi sau khi hoàn thành.