|  |
| --- |
| **BỘ GIÁO DỤC VÀ ĐÀO TẠO**  ĐẶNG TUẤN ANH XÂY DỰNG ỨNG DỤNG TRẮC NGHIỆM ÔN TOÁN LỚP 1 2021  **TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**  **ĐẶNG TUẤN ANH**  **XÂY DỰNG ỨNG DỤNG TRẮC NGHIỆM  ÔN TOÁN LỚP 1**  **ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC**  **HƯNG YÊN - 2021** |

|  |
| --- |
| **BỘ GIÁO DỤC VÀ ĐÀO TẠO**  **TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**  **ĐẶNG TUẤN ANH**  **XÂY DỰNG ỨNG DỤNG TRẮC NGHIỆM  ÔN TOÁN LỚP 1**  NGÀNH: CÔNG NGHỆ THÔNG TIN  CHUYÊN NGÀNH: KỸ THUẬT PHẦN MỀM  **ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC**  **NGƯỜI HƯỚNG DẪN**  **TS. CHU THỊ MINH HUỆ**  **HƯNG YÊN - 2021** |

**NHẬN XÉT**

**Nhận xét của giảng viên hướng dẫn:**

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

**GIẢNG VIÊN HƯỚNG DẪN**

*(Ký và ghi rõ họ tên)*

**LỜI CAM ĐOAN**

Em xin cam đoan đồ án tốt nghiệp “Xây dựng ứng dụng trắc nghiệm ôn toán lớp 1” là công trình nghiên cứu của bản thân. Những phần sử dụng tài liệu tham khảo trong đồ án đã được nêu rõ trong phần tài liệu tham khảo. Các số liệu, kết quả trình bày trong đồ án là hoàn toàn trung thực, nếu sai em xin chịu hoàn toàn trách nhiệm và chịu mọi kỷ luật của bộ môn và nhà trường đề ra.

*Hưng Yên, ngày … tháng … năm…..*

Sinh viên

Đặng Tuấn Anh

**LỜI CẢM ƠN**

Lời đầu tiên, em xin chân thành cảm ơn bộ môn Công nghệ phần mềm, trường  Đại học Sư phạm Kỹ thuật Hưng yên đã tạo điều kiện thuận lợi cho em thực hiện đồ án tốt nghiệp. Đặc biệt em xin chân thành cảm ơn cô Chu Thị Minh Huệ đã rất tận tình  hướng dẫn, chỉ bảo em trong suốt thời gian thực hiện Đồ án tốt nghiệp vừa qua.

Em cũng xin chân thành cảm ơn tất cả các Thầy, các Cô trong Trường đã tận tình giảng dạy, trang bị cho em những kiến thức cần thiết, quý báu để giúp em thực hiện  được đồ án của mình.

Mặc dù em đã có cố gắng, nhưng với trình độ còn hạn chế, trong quá trình thực hiện đề tài không tránh khỏi những thiếu sót. Em hi vọng sẽ nhận được những ý kiến nhận xét, góp ý của các Thầy giáo, Cô giáo về những vấn đề triển khai trong đồ án của mình.

MỤC LỤC

[DANH SÁCH CÁC THUẬT NGỮ 9](#_Toc75781721)

[DANH SÁCH BẢNG BIỂU 10](#_Toc75781722)

[DANH SÁCH HÌNH VẼ 11](#_Toc75781723)

[CHƯƠNG 1: MỞ ĐẦU 13](#_Toc75781724)

[1.1 Lý do chọn đồ án 13](#_Toc75781725)

[1.2 Mục tiêu của đồ án 13](#_Toc75781726)

[1.2.1 Mục tiêu tổng quát 13](#_Toc75781727)

[1.2.2 Mục tiêu cụ thể 13](#_Toc75781728)

[1.3 Giới hạn và phạm vi của đồ án 15](#_Toc75781729)

[1.3.1 Đối tượng nghiên cứu 15](#_Toc75781730)

[1.3.2 Phạm vi nghiên cứu 15](#_Toc75781731)

[1.4 Nội dung thực hiện 15](#_Toc75781732)

[1.5 Phương pháp tiếp cận 16](#_Toc75781733)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VỀ PHÁT TRIỂN ỨNG DỤNG 17](#_Toc75781734)

[2.1 Phương pháp phát triển phần mềm hướng đối tượng 17](#_Toc75781735)

[2.1.1 Phân tích và thiết kế hướng đối tượng (OOAD) 17](#_Toc75781736)

[2.1.2 Phân tích hướng đối tượng và thiết kế hướng đối tượng 19](#_Toc75781737)

[2.1.3 Lập trình hướng đối tượng OOP 19](#_Toc75781738)

[2.2 Công nghệ áp dụng 20](#_Toc75781739)

[2.2.1 Lập trình trên Android Studio 20](#_Toc75781740)

[2.2.2 Activity trong Android Studio 21](#_Toc75781741)

[2.2.3 Thư viện Android (Android libraries) và Android Runtime 22](#_Toc75781742)

[2.2.4 Ngôn ngữ lập trình Java 23](#_Toc75781743)

[2.2.5 Hệ quản trị SQLite 24](#_Toc75781744)

[CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 25](#_Toc75781745)

[3.1 Phân tích yêu cầu bài toán 25](#_Toc75781746)

[3.1.1 Khảo sát bài toán 25](#_Toc75781747)

[3.1.2 Phân tích yêu cầu bài toán 26](#_Toc75781748)

[3.2 Đặc tả yêu cầu phần mềm 28](#_Toc75781749)

[3.2.1 Các yêu cầu chức năng 29](#_Toc75781750)

[3.2.2 Biểu đồ lớp thực thể 39](#_Toc75781751)

[3.2.3 Các yêu cầu phi chức năng 41](#_Toc75781752)

[3.3 Thiết kế hệ thống 41](#_Toc75781753)

[3.3.1 Thiết kê cơ sở dữ liệu 41](#_Toc75781754)

[3.3.2 Thiết kế lớp đối tượng 42](#_Toc75781755)

[3.3.3 Thiết kế giao diện 48](#_Toc75781756)

[CHƯƠNG 4: TRIỂN KHAI ỨNG DỤNG 57](#_Toc75781757)

[4.1 Triển khai các chức năng cho ứng dụng 57](#_Toc75781758)

[4.1.1 Các chức năng nghiệp vụ 57](#_Toc75781759)

[4.1.2 Các chức năng hệ thống 57](#_Toc75781760)

[4.2 Kiểm thử và triển khai ứng dụng 60](#_Toc75781761)

[4.2.1 Kiểm thử đơn vị 60](#_Toc75781762)

[4.2.2 Kiểm thử thủ công 60](#_Toc75781763)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 62](#_Toc75781764)

[Kết quả đạt được của đề tài 62](#_Toc75781765)

[Hạn chế của đề tài 62](#_Toc75781766)

[Hướng phát triển của đề tài 63](#_Toc75781767)

[TÀI LIỆU THAM KHẢO 64](#_Toc75781768)

DANH SÁCH CÁC THUẬT NGỮ

|  |  |  |
| --- | --- | --- |
| **Từ viết tắt** | **Từ đầy đủ** | **Giải thích** |
| OOAD | Object Oriented Analysis and Design | Phân tích và thiết kế hướng đối tượng |
| OOP | Object Oriented Programming | Lập trình hướng đối tượng |
| IDE | Integrated Development Environment | Môi trường phát triển tích hợp |
| DVM | Dalvik Virtual Machine | Một biến thể của Java Virtual Machine |
| SQLite | Structured Query Language Lite | Hệ quản trị cơ sở dữ liệu |
| UML | Unified Modeling Language | Ngôn ngữ mô hình hoá hợp nhất |

DANH SÁCH BẢNG BIỂU

[Bảng 3. 1: Danh sách các Actor 28](#_Toc75781769)

[Bảng 3. 2: Danh sách các Use-case 28](#_Toc75781770)

[Bảng 3. 3: Danh sách các lớp đối tượng và quan hệ 39](#_Toc75781771)

[Bảng 3. 4: Mô tả lớp CauHoi 40](#_Toc75781772)

[Bảng 3. 5: Mô tả lớp MucDo 40](#_Toc75781773)

[Bảng 3. 6: Mô tả lớp Diem 40](#_Toc75781774)

[Bảng 3. 7: Mô tả chi tiết bảng cauhoi 42](#_Toc75781775)

[Bảng 3. 8: Mô tả chi tiết bảng mucdo 42](#_Toc75781776)

DANH SÁCH HÌNH VẼ

[Hình 3. 1: Biểu đồ Use-case Tổng quát 29](#_Toc75781777)

[Hình 3. 2: Biểu đồ Use-case Luyện tập 30](#_Toc75781778)

[Hình 3. 3: Biểu đồ Use-case Kiểm tra 31](#_Toc75781779)

[Hình 3. 4: Biểu đồ Use-case Xem điểm 33](#_Toc75781780)

[Hình 3. 5: Biểu đồ Use-case Tính nhẩm 34](#_Toc75781781)

[Hình 3. 6: Biểu đồ Use-Case So sánh 36](#_Toc75781782)

[Hình 3. 7: Biểu đồ Use-case Tập đếm 37](#_Toc75781783)

[Hình 3. 8: Biểu đồ lớp 39](#_Toc75781784)

[Hình 3. 9: Lược đồ quan hệ 41](#_Toc75781785)

[Hình 3. 10: Biểu đồ lớp VOPC Kiểm Tra 43](#_Toc75781786)

[Hình 3. 11: Biểu đồ lớp VOPC Xem Điểm 43](#_Toc75781787)

[Hình 3. 12: Biểu đồ lớp VOPC Tính Nhẩm 44](#_Toc75781788)

[Hình 3. 13: Biểu đồ lớp VOPC So Sánh 44](#_Toc75781789)

[Hình 3. 14: Biểu đồ lớp VOPC Tập Đếm 45](#_Toc75781790)

[Hình 3. 15: Biểu đồ tuần tự UC Kiemtra 45](#_Toc75781791)

[Hình 3. 16: Biểu đồ tuần tự UC Xemdiem 46](#_Toc75781792)

[Hình 3. 17: Biểu đồ tuần tự UC TinhNham 46](#_Toc75781793)

[Hình 3. 18: Biểu đồ tuần tự UC SoSanh 47](#_Toc75781794)

[Hình 3. 19: Biểu đồ tuần tự UC TapDem 47](#_Toc75781795)

[Hình 3. 20: Giao diện chính 48](#_Toc75781796)

[Hình 3. 21: Giao diện Luyện tập 49](#_Toc75781797)

[Hình 3. 22: Giao diện Kiểm tra 50](#_Toc75781798)

[Hình 3. 23: Giao diện Xem điểm 51](#_Toc75781799)

[Hình 3. 24: Giao diện Chọn phạm vi 52](#_Toc75781800)

[Hình 3. 25: Giao diện Tính nhẩm 53](#_Toc75781801)

[Hình 3. 26: Giao diện So sánh 54](#_Toc75781802)

[Hình 3. 27: Giao diện Tập đếm 55](#_Toc75781803)

# MỞ ĐẦU

1.1 Lý do chọn đồ án

Trong bối cảnh đại dịch COVID-19 xâm nhập và lây lan trong nước khiến nhiều trường học phải đóng cửa để phòng dịch. Việc đóng cửa trường học như vậy dẫn đến việc học tập của học sinh các cấp, đặc biệt là học sinh lớp 1, bị ảnh hưởng rất nhiều. Các bạn phải học trực tuyến từ xa, đồng thời các kiến thức trên trường bị đứt gãy do nghỉ dịch nhiều ngày khiến các bạn học sinh lớp 1 không thể ôn tập nhiều các kiến thức đã học, nhất là với môn toán lớp 1.

Chính vì vậy, thấy được những khó khăn của các bạn, em nghĩ cần một ứng dụng để các bạn có thể ôn tập các bài tập toán lớp 1 thông qua smartphone của phụ huynh. Đồng thời phụ huynh cũng thể tham gia vào ôn tập, dạy con học thông qua ứng dụng đó như một cách củng cố kiến thức toán lớp 1 của các bạn. Từ đó mà ý tưởng về “Xây dựng ứng dụng trắc nghiệm ôn toán lớp 1” ra đời.

1.2 Mục tiêu của đồ án

1.2.1 Mục tiêu tổng quát

Trong học kỳ I của chương trình toán lớp 1, các bé được học về cách đếm các số từ 1 đến 10, đếm các hình tròn, hình vuông, hình tam giác và đếm các con vật, đồ vật; Các bé được học về phép cộng trừ 2 số trong phạm vi 10; Các bé được học về so sánh 2 số với nhau. Sang học kì II, các bé được học về các số trong phạm vi 100, cộng trừ các số trong phạm vi 100; Các bé được học về giải bài tập toán, so sánh 2 phép tính với nhau. Như vậy, có thể thấy xuyên suốt chương trình toán lớp 1, các bé đã được học 3 mảng kiến thức chính là tập đếm, tính nhẩm và so sánh. Qua đề tài này, em đặt ra mục tiêu là phát triển một ứng dụng trắc nghiệm ôn tập các dạng bài tập trong chương trình toán lớp 1 dành cho các bạn học sinh lớp 1, các bậc phụ huynh muốn ôn tập kiến thức toán lớp 1 cho con em mình có thể tham gia cùng con.

1.2.2 Mục tiêu cụ thể

* Về kiến thức
* Trình bày được các hoạt động trong quy trình phát triển một sản phẩm

phần mềm:

+ Trình bày được phương pháp phân tích thiết kế hướng đối tượng và các biểu đồ UML tương ứng trong các giai đoạn đặc tả yêu cầu, phân tích, thiết kế phần mềm và phương pháp chuyển từ thiết kế chi tiết sang mã nguồn triển khai.

+ Trình bày được các kiến thức về thiết kế và lập trình ứng dụng dị động trên nền tảng Android Studio sử dụng ngôn ngữ lập trinh Java.

+ Trình bày được các kiến thức về cơ sở dữ liệu, cách kết nối ứng dụng với CSDL SQLite.

* Về kỹ năng

+ Vận dụng được các kiến thức phân tích thiết kế hướng đối tượng với UML để xây dựng các biểu đồ ca sử dụng, lớp thực thể, lớp phân tích, biểu đồ tuần tự, lớp thiết kế chi tiết,… cho ứng dụng.

+ Sử dụng được hệ quản trị cơ sở dữ liệu SQLite để xây dựng và quản lý được cơ sở dữ liệu theo thiết kế.

+ Vận dụng được ngôn ngữ thiết kế và lập trình di động trên nền tảng Android Studio sử dụng ngôn ngữ lập trình java để xây dựng ứng dụng theo thiết kế.

+ Thực hiện kiểm thử và Debug được các lỗi trong ứng dụng.

+ Triển khai thử nghiệm được ứng dụng trên mạng Internet.

* Sản phẩm
* Cuốn báo cáo trình bày theo đúng mẫu và yêu cầu.
* Ứng dụng trắc nghiệm ôn toán lớp 1 cần đạt được những chức năng cụ thể sau:

+ Phải có danh sách câu hỏi và đáp án phù hợp với chương trình toán học lớp 1, các câu hỏi và đáp án phải phù hợp với nhau và phù hợp với từng chức năng khác nhau trong hệ thống. Nguồn câu hỏi cần lớn và phong phú để tránh việc lặp đi lặp lại các câu hỏi gây sự nhám chán cho bé trong lúc chơi.

+ Phải có âm thanh đúng, sai để thông báo lúc bé chọn câu hỏi.

+ Có chức năng luyện tập với ba khối kiến thức xuyên suốt chương trình toán lớp 1 là: tập đếm, tính nhẩm và so sánh.

+ Trong phần tập đếm cần có các hình ảnh sinh động về con vật, đồ vật, hoa quả để thu hút sự tập trung của bé.

+ Có chức năng kiểm tra để kiểm tra các kiến thức bé đã học.

+ Phải có chức năng xem điểm khi bé kiểm tra xong.

1.3 Giới hạn và phạm vi của đồ án

1.3.1 Đối tượng nghiên cứu

* Đối tượng nghiên cứu: nhu cầu ôn tập môn toán của học sinh lớp 1.
* Khách thể nghiên cứu: học sinh lớp 1.

1.3.2 Phạm vi nghiên cứu

* Phạm vi không gian: Trường Tiểu học Thị trấn Bần Yên Nhân.
* Phạm vi thời gian: thu thập thông tin trong tháng 3, tháng 4 năm 2021.
* Ý nghĩa khoa học và thực tiễn của đề tài: Xây dựng ứng dụng để tạo một không gian ôn tập kiến thức toán lớp 1 cho các bé thông qua các câu trắc nghiệm, nhằm giúp các bé học tập trong thời buổi dịch bệnh covid ảnh hưởng đến học tập môn toán lớp 1 của các bé. Phụ huynh có thể sử dụng ứng dụng trắc nghiệm ôn toán lớp 1 như một công cụ hữu ích khi muốn củng cố kiến thức toán lớp 1 của con em mình.

1.4 Nội dung thực hiện

* Nội dung thực hiện nghiên cứu cụ thể như sau:

+ Nghiên cứu công nghệ lập trình di động trên nền tảng Android Studio bằng ngôn ngữ lập trình Java.

+ Tìm hiểu hệ quản trị cơ sở dữ liệu SQLite.

+ Đặc tả yêu cầu người dùng.

+ Phân tích yêu cầu và đưa ra yêu cầu cho hệ thống.

+ Đặc tả chức năng, lớp cho hệ thống.

+ Thiết kế cơ sở dữ liệu.

+ Thiết kế giao diện cho ứng dụng.

+ Lập trình các chức năng, các ràng buộc như trong đặc tả cho hệ thống.

+ Kiểm thử hệ thống.

+ Triển khai thực hiện hệ thống trên mạng Internet.

1.5 Phương pháp tiếp cận

* Các phương pháp nghiên cứu đã sử dụng:

+ Phương pháp đọc tài liệu: nghiên cứu tài liệu về quy trình phát triển phần mềm, các phương pháp thiết kế hướng đối tượng. Nghiên cứu các kiến thức và kỹ năng lập trình ứng dụng di động trên nền tảng Android Studio bằng ngôn ngữ lập trình Java. Tìm hiểu các tài liệu về hệ quản trị cơ sở dữ liệu SQLite.

+ Phương pháp thực nghiệm: trải nghiệm thực tế quá trình làm một bài kiểm tra của học sinh lớp 1 và khảo sát một số ý kiến của giáo viên dạy toán lớp 1 của trường Tiểu học Thị Trấn Bần Yên Nhân về quá trình ôn tập toán lớp 1 cho các em học sinh.

* Tham khảo thông tin trên Internet.

# CƠ SỞ LÝ THUYẾT VỀ PHÁT TRIỂN ỨNG DỤNG

2.1 Phương pháp phát triển phần mềm hướng đối tượng

2.1.1 Phân tích và thiết kế hướng đối tượng (OOAD)

Phân tích và thiết kế hướng đối tượng là một kỹ thuật tiếp cận phổ biến dùng để phân tích, thiết kế một ứng dụng, hệ thống. Nó dựa trên bộ các nguyên tắc chung, đó là một tập các hướng dẫn để giúp chúng ta tránh khỏi một thiết kế xấu.

* Khái niệm về Phân tích và thiết kế hướng đối tượng (Object Oriented Analysis and Design: OOAD)

+ Trong kỹ nghệ phần mềm để sản xuất được một sản phẩm phần mềm người ta chia quá trình phát triển sản phẩm ra nhiều giai đoạn như thu thập và phân tích yêu cầu, phân tích và thiết kế hệ thống, phát triển (coding), kiểm thử, triển khai và bảo trì. Trong đó, giai đoạn phân tích, thiết kế bao giờ cũng là giai đoạn khó khăn và phức tạp nhất. Giai đoạn này giúp chúng ta hiểu rõ yêu cầu đặt ra, xác định giải pháp, mô tả chi tiết giải pháp. Nó trả lời 2 câu hỏi What (phần mềm này làm cái gì?) và How (làm nó như thế nào?).

+ Để phân tích và thiết kế một phần mềm thì có nhiều cách làm, một trong những cách làm đó là xem hệ thống gồm những đối tượng sống trong đó và tương tác với nhau. Việc mô tả được tất cả các đối tượng và sự tương tác của chúng sẽ giúp chúng ta hiểu rõ hệ thống và cài đặt được nó. Phương thức này gọi là Phân tích thiết kế hướng đối tượng (OOAD)

* Khái niệm về UML (Unified Modeling Language)

UML là ngôn ngữ mô hình hóa hợp nhất dùng để biểu diễn hệ thống. Nói một cách đơn giản là nó dùng để tạo ra các bản vẽ nhằm mô tả thiết kế hệ thống. Các bản vẽ này được sử dụng để các nhóm thiết kế trao đổi với nhau cũng như dùng để thi công hệ thống (phát triển), thuyết phục khách hàng, các nhà đầu tư.

* OOAD sử dụng UML
* OOAD cần các bản vẽ để mô tả hệ thống được thiết kế, còn UML là ngôn ngữ mô tả các bản vẽ nên cần nội dung thể hiện. Do vậy, chúng ta phân tích và thiết kế theo hướng đối tượng và sử dụng UML để biểu diễn các thiết kế đó nên chúng thường đi đôi với nhau.
* OOAD sử dụng UML bao gồm các thành phần sau:
* View (góc nhìn).
* Diagram (bản vẽ).
* Notations (các ký hiệu).
* Relationship (quan hệ).
* View (góc nhìn)

+ Use Case View: cung cấp góc nhìn về các ca sử dụng giúp chúng ta hiểu hệ thống có gì? ai dùng và dùng nó như thế nào.

+ Logical View: cung cấp góc nhìn về cấu trúc hệ thống, xem nó được tổ chức như thế nào. Bên trong nó có gì.

+ Process View: cung cấp góc nhìn động về hệ thống, xem các thành phần trong hệ thống tương tác với nhau như thế nào.

+ Component View: cũng là một góc nhìn về cấu trúc giúp chúng ta hiểu cách phân bổ và sử dụng lại các thành phần trong hệ thống ra sao.

+ Deployment View: cung cấp góc nhìn về triển khai hệ thống, nó cũng ảnh hưởng lớn đến kiến trúc hệ thống.

Tập hợp các góc nhìn này sẽ giúp chúng ta hiểu rõ hệ thống cần phân tích, thiết kế. Góc nhìn Use Case View nằm ở giữa và chi phối tất cả các góc nhìn còn lại. Chính vì thế chúng ta thường thấy các tài liệu nói về 4 view + 1 chứ không phải 5 view nhằm nhấn mạnh vai trò của Use Case View.

* Diagram (bản vẽ)

Các bản vẽ được dùng để thể hiện các góc nhìn của hệ thống.

* Notations (các kí hiệu)

Notations là các ký hiệu để vẽ, nó như từ vựng trong ngôn ngữ tự nhiên. Dưới đây là vài ví dụ về notation:

+ Ký hiệu về Usecase.

+ Ký hiệu về Class.

+ Ký hiệu về Actor.

Và còn rất nhiều ký hiệu nữa.

* Relations (quan hệ)

Relationship được sử dụng để kết nối giữa các đối tượng với nhau tạo nên bản vẽ Use Case. Nó các kiểu quan hệ cơ bản sau:

+ Association.

+ Generalization.

+ Include.

+ Extend.

2.1.2 Phân tích hướng đối tượng và thiết kế hướng đối tượng

Là giai đọan phát triển một mô hình chính xác và súc tích của vấn đề, có thành phần là các đối tượng và khái niệm đời thực, dễ hiểu đối với người sử dụng.

Trong giai đoạn OOA, vấn đề được trình bày bằng các thuật ngữ tương ứng với các đối tượng có thực. Thêm vào đó, hệ thống cần phải được định nghĩa sao cho người không chuyên Tin học có thể dễ dàng hiểu được.

Dựa trên một vấn đề có sẵn, nhà phân tích cần ánh xạ các đối tượng hay thực thể có thực như khách hàng, ô tô, người bán hàng,… vào thiết kế để tạo ra được bản thiết kế gần cận với tình huống thực. Mô hình thiết kế sẽ chứa các thực thể trong một vấn đề có thực và giữ nguyên các mẫu hình về cấu trúc, quan hệ cũng như hành vi của chúng. Nói một cách khác, sử dụng phương pháp hướng đối tượng chúng ta có thể mô hình hóa các thực thể thuộc một vấn đề có thực mà vẫn giữ được cấu trúc, quan hệ cũng như hành vi của chúng.

2.1.3 Lập trình hướng đối tượng OOP

OOP (viết tắt của Object Oriented Programming) – lập trình hướng đối tượng là một phương pháp lập trình dựa trên khái niệm về lớp và đối tượng. OOP tập trung vào các đối tượng thao tác hơn là logic để thao tác chúng.

Mục tiêu của OOP là tối ưu việc quản lý source code, giúp tăng khả năng tái sử dụng và quan trọng hơn hết là giúp tóm gọn các thủ tục đã biết trước tính chất thông qua việc sử dụng các đối tượng.

* 4 đặc tính cơ bản của OOP:

+ Tính đóng gói cho phép che giấu thông tin và những tính chất xử lý bên trong của đối tượng. Các đối tượng khác không thể tác động trực tiếp đến dữ liệu bên trong và làm thay đổi trạng thái của đối tượng mà bắt buộc phải thông qua các phương thức công khai do đối tượng đó cung cấp.Tính chất này giúp tăng tính bảo mật cho đối tượng và tránh tình trạng dữ liệu bị hư hỏng ngoài ý muốn.

+ Đây là tính chất được sử dụng khá nhiều. Tính kế thừa cho phép xây dựng một lớp mới (lớp Con), kế thừa và tái sử dụng các thuộc tính, phương thức dựa trên lớp cũ (lớp Cha) đã có trước đó. Các lớp Con kế thừa toàn bộ thành phần của lớp Cha và không cần phải định nghĩa lại. Lớp Con có thể mở rộng các thành phần kế thừa hoặc bổ sung những thành phần mới.

+ Tính đa hình trong lập trình OOP cho phép các đối tượng khác nhau thực thi chức năng giống nhau theo những cách khác nhau.

+ Tính trừu tượng giúp loại bỏ những thứ phức tạp, không cần thiết của đối tượng và chỉ tập trung vào những gì cốt lõi, quan trọng.

2.2 Công nghệ áp dụng

2.2.1 Lập trình trên Android Studio

Có nhiều công cụ để phát triển Android nhưng đến nay công cụ chính thức và mạnh mẽ nhất là Android Studio. Đây là IDE (Môi trường phát triển tích hợp) chính thức cho nền tảng Android, được phát triển bởi Google và được sử dụng để tạo phần lớn các ứng dụng mà bạn có thể sử dụng hàng ngày.

Android Studio lần đầu tiên được công bố tại hội nghị Google I/O vào năm 2013 và được phát hành cho công chúng vào năm 2014 sau nhiều phiên bản beta khác nhau. Trước khi được phát hành, các nhà phát triển Android thường sử dụng các công cụ như Eclipse IDE, một IDE Java chung cũng hỗ trợ nhiều ngôn ngữ lập trình khác.

Android Studio khiến việc tạo ứng dụng trở nên dễ dàng hơn đáng kể so với phần mềm không chuyên dụng. Đối với người mới bắt đầu, có rất nhiều thứ để học và nhiều thông tin có sẵn, thậm chí thông qua các kênh chính thức nhưng chúng có thể đã lỗi thời hoặc quá nhiều thông tin khiến họ cảm thấy choáng ngợp. Bài viết này sẽ giải thích ngắn gọn nhưng chi tiết về một số chức năng cơ bản của nó để bạn có thể nắm bắt được bước đầu tCác thành phần cơ bản của ứng dụng Android.

2.2.2 Activity trong Android Studio

Activity chính là một cửa sổ chưa giao diện ứng dụng của bạn mà người dùng có thể tương tác với nó. Mỗi màn hình trên ứng dụng của bạn sẽ là kế thừa của Activity Activity dùng các View để hiển thị giao diện ứng dụng người dùng và đáp ứng lại các yêu cầu từ người dùng.

* Có 3 dạng Activity

+ Floating Activity: là dạng Activity nổi có kích thước không phủ lấy toàn bộ màn hình.

+ Activity Group: là một nhóm các Activity nhúng trong một Activity khác

+ Full screen Activity: là Activity phu kinh toàn bộ màn hình, đây là loại Activity thường được sử dụng phổ biến nhất.

Vòng đời của một Activity sẽ diễn ra từ lúc khởi động cho đến khi kết thúc vòng đời của nó. Hệ thống Android sẽ được bắt đầu thực hiện một Activity tương ứng với từng trạng thái vòng đời của nó. Android cung cấp một số sự kiện để quản lý các trạng thái vòng đời của Activity như sau:

+ onCreate(): sự kiện này được gọi khi Activity được tạo lần đầu tiên, Trong sự kiện này bạn phải thực hiện những công việc như tạo giao diện, tải dữ liệu.

sau sự kiện này là onStart() được gọi

+ onRestart(): sự kiện được gọi khi muốn khói động lại một activity đã bị Stop.

Sau sự kiện này luôn là sự kiện onStart() được gọi.

+ onStart(): sự kiện nà được gọi khi giao diện của Activity hiển thị trên màn hình thiết bị.

+ onResume(): sự kiện này được gọi khi Activity bắt đầu tương tác với người dùng.

+ onPause(): khi một Activity khác dành quyền hiển thị và trạng thái focus thì Activity hiện hành sẽ gọi sự kiện onPause).

+ onStop(): sự kiện này được gọi khi Activity bị che khuất hoàn toàn bởi một Activity khác và Activity đó cũng dành được focus.

Vòng đời của 1 activity Khi một Activity bị paused hoặc stopped, trạng thái của một Activity vẫn sẽ được giữ lại Điều này là đúng vì mỗi đối tượng Activity vẫn còn được giữ lại trong bộ nhớ khi nó bị paused hoặc stopped Tất cả các thông tin về Activity cũng như trạng thái của nó vẫn còn tồn tại trong Vì vậy, bất cứ thay đổi nào của người dùng đối với Activity vẫn được giữ lại khi nó trở lại trạng thái hoạt động hoặc chế độ resume.

2.2.3 Thư viện Android (Android libraries) và Android Runtime

Phía trên tầng Linux kernel là tầng libraries, chứa những thư viện hỗ trợ. Một số có thể kể đến như là bộ máy trình duyệt web mã nguồn mở WebKit, thư viện libc, cơ sở dữ liệu SQLite tiện lợi cho việc lưu trữ và chia sẻ dữ liệu, thư viện hỗ trợ thu phát âm thanh và video, thư viện SSL giúp bảo mật mạng.

Tầng này chứa tất cả các thư viện Java, được viết đặc biệt cho Android như các thư viện framework, các thư viện xây dựng giao diện, đồ họa và cơ sở dữ liệu. Dưới đây là một số thư viện quan trọng mà các lập trình viên nên biết:

+ Android.app: Cung cấp quyền truy cập tới mô hình ứng dụng và là nền tảng của mọi ứng dụng Android applications.

+ Android.content: Cho phép truy cập nội dung, phát hành và tin hằn giữa các ứng dụng và các thành phần trong một ứng dụng.

+ Android database: Được dùng để truy cập vào dữ liệu được đưa ra bởi bộ phận cung cấp nội dung, bao gồm các lớp quan lý cơ sở dữ liệu SQLite.

+ Android opengl: Cho phép tương tác với thư viện đồ họa OpenGLES 3D.

+ Android OS: Giúp mg dụng truy cập những dịch vụ cơ bản của hệ điều hành bao gồm tên nhân dịch vụ hệ thống và liên lạc nội bộ (inter-process communication).

+ Android text: Được dùng để vẽ và thao tác văn bản trên màn hình,

+ Android view: Các khối xây dựng cơ bản của giao diện người dùng.

+ Androidwidget: Một tập hợp rất nhiều thành phần giao diện được xây dựng sẵn như nút bấm (button), nhãn (label), danh sách hiển thị (list Views), quan lý bố cục (layout managers)...

+ Android webkit. Tập hợp các lớp (classes) cho phép trình duyệt web được nhúng vào ứng dụng

+ Trong tầng này còn có một phần không kém phần quan trọng là Android untime, bây giờ chúng ta sẽ tìm hiểu các thư viện C/C++ trong tầng này.

+ Android Runtime chứa Dalvik Virtual Machine (DVM) một biến thể của Java Virtual Machine, đặc biệt thiết kế và tối ưu hóa cho Android. DVM giúp mỗi ứng dụng Android chạy trong chính trên trình (process) của nó với một đại diện (instance) của DVM.

+ Ngoài ra, Android Runtime cũng chứa tập hợp các thư viện quan trọng cho phép người lập trình viết ứng dụng sử dụng ngôn ngữ lập trình Java.

2.2.4 Ngôn ngữ lập trình Java

Java là một một ngôn ngữ lập trình hiện đại, bậc cao, hướng đối tượng, bảo mật và mạnh mẽ. và là một Platform.

Platform: Bất cứ môi trường phần cứng hoặc phần mềm nào mà trong đó có một chương trình chạy, thì được hiểu như là một Platform. Với môi trường runtime riêng cho mình (JRE) và API, Java được gọi là Platform.

Ngôn ngữ lập trình Java ban đầu được phát triển bởi Sun Microsystems do James Gosling khởi xướng và phát hành vào năm 1995. Phiên bản mới nhất của Java Standard Edition là Java SE 8. Với sự tiến bộ của Java và sự phổ biến rộng rãi của nó, nhiều cấu hình đã được xây dựng để phù hợp với nhiều loại nền tảng khác nhau. Ví dụ: J2EE cho các ứng dụng doanh nghiệp, J2ME cho các ứng dụng di động.

Các phiên bản J2 mới đã được đổi tên thành Java SE, Java EE và Java ME. Phương châm của java là "Write Once, Run Anywhere" - viết một lần chạy nhiều nơi, nghĩa là bạn chỉ cần viết một lần trên window chẳng hạn, sau đó vẫn chương trình đó bạn có thể chạy trên Linux, Android, các thiết bị J2ME,...

2.2.5 Hệ quản trị SQLite

SQLite là một thư viện nhỏ gọn. Với tất cả các tính năng được bật, kích thước thư viện có thể nhỏ hơn 600KB, tùy thuộc vào cài đặt tối ưu hóa nền tảng và trình biên dịch đích. (Mã 64 bit lớn hơn. Và một số tối ưu hóa trình biên dịch như hàm xâm lấn nội tuyến và không kiểm soát vòng lặp có thể khiến mã đối tượng lớn hơn nhiều). Có sự đánh đổi giữa việc sử dụng bộ nhớ và tốc độ. SQLite thường chạy nhanh hơn nhiều bộ nhớ bạn cung cấp cho nó. Tuy nhiên, hiệu suất thường khá tốt ngay cả trong môi trường bộ nhớ thấp. Tùy thuộc vào cách nó được sử dụng.

SQLite là một công cụ cơ sở dữ liệu SQL nhúng. Không giống như hầu hết các cơ sở dữ liệu SQL khác, SQLite không có quy trình máy chủ riêng biệt. SQLite đọc và ghi trực tiếp vào các tệp đĩa thông thường. Một cơ sở dữ liệu SQL hoàn chỉnh với nhiều bảng, chỉ mục, trình kích hoạt và khung nhìn, được chứa trong một tệp đĩa đơn. Định dạng tệp cơ sở dữ liệu là đa nền tảng - bạn có thể tự do sao chép cơ sở dữ liệu giữa các hệ thống 32 bit và 64 bit hoặc giữa các kiến trúc lớn cuối và cuối nhỏ. Các tính năng này làm cho SQLite trở thành một lựa chọn phổ biến dưới dạng định dạng tệp ứng dụng . Các tệp cơ sở dữ liệu SQLite là một định dạng lưu trữ được đề xuất bởi Thư viện Quốc hội Hoa Kỳ.

# PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

3.1 Phân tích yêu cầu bài toán

3.1.1 Khảo sát bài toán

Để đáp ứng yêu của người dùng, ứng dụng trắc nghiệm ôn toán lớp 1 sẽ cần ba chức năng chính đó là luyện tập, kiểm tra và xem điểm. Chức năng luyện tập sẽ ôn tâp các khối kiến thức thường xuyên sử dụng trong chương trình toán lớp 1 gồm kiến thức về tập đếm, kiến thức về tính nhẩm và kiến thức về so sánh. Chức năng kiểm tra sẽ giúp người dùng kiểm tra được kiến thức của mình sau khi ôn tập ở chức năng luyện tập. Chức năng xem điểm sẽ giúp người dùng xem được thành quả luyện tập của mình thông qua điểm kiểm tra.

* Yêu cầu về chức năng luyện tập

+ Cho phép người dùng ôn tập 3 khối kiến thức xuyên suốt chương trình toán lớp 1 thông qua ba chức năng nhỏ gồm: tập đếm, tính nhẩm và so sánh.

+ Chức năng tập đếm sẽ hiển thị các hình ảnh có số lượng về con vật, đồ vật, các hình khối cơ bản (hình tròn, hình vuông, hình tam giác) để người dùng đếm.

+ Chức năng tính nhẩm sẽ hiển thị các câu trắc nghiệm để cộng trừ hai giá trị trong phạm vi 0-100. Ở chức năng tính nhẩm này sẽ chia phạm vi 0-100 thành 3 phạm vi là 0-10, 0-20, 0-100 cho phù hợp với khả năng của người dùng.

+ Chức năng so sánh sẽ hiển thị các câu trắc nghiệm để người dùng so sánh hai giá trị với nhau. So sánh giá trị gồm so sánh hai số với nhau, hai phép tính với nhau hoăc một sô với một phép tính.

* Yêu cầu về chức năng kiểm tra

+ Chức năng cho phép người dùng kiểm tra kiến thức đã ôn tập của mình.

+ Chức năng sẽ có một đồng hồ đếm ngược thời gian kiểm tra là 10 phút. Khi đếm hết thời gian thì phần kiểm tra sẽ kết thúc.

+ Chức năng sẽ tính điểm tổng số câu làm đúng của người dùng, mỗi câu làm đúng được cộng 1 điểm.

* Yêu cầu về chức năng xem điểm

+ Chức năng sẽ hiển thị điểm vừa mới thi gần nhất và điểm cao nhất từng đạt được của người dùng.

3.1.2 Phân tích yêu cầu bài toán

* Chức năng luyện tập

Khi nhấn vào chức năng luyện tâp, hệ thống sẽ chuyển người dùng sang giao diện luyện tập. Ở đây, có 3 chức năng cho người dùng chọn: chức năng tính nhẩm, chức năng so sánh và chức năng tập đếm.

* Chức năng tính nhẩm

Người dùng nhấn vào chức năng tính nhẩm, hệ thống hiển thị 3 mức tính nhẩm cho người dùng chọn là phạm vi 0 - 10, phạm vi 0 - 20 và phạm vi 0 - 100. Chẳng hạn người dùng nhấn vào phạm vi 0 – 10, hệ thống sẽ chuyển sang giao diện phạm vi 0 – 10, ở đây sẽ hiển thị các câu hỏi và 4 đáp án A, B, C, D cho người dùng chọn (các câu hỏi được lấy từ CSDL của hệ thống và được lấy ngẫu nhiên). Nhiệm vụ của người dùng là đọc kỹ câu hỏi và chọn 1 trong 4 đáp án A, B, C, D đó. Sau khi người dùng chọn đáp án, hệ thống sẽ kiểm tra tính đúng/ sai của đáp án. Nếu là đáp án đúng/ sai, hệ thống sẽ phát ra âm thanh của đáp án đúng/ sai, đồng thời hệ thống sẽ hiển thị câu hỏi tiếp theo cho đến khi hết số lượng câu hỏi của phần tính nhẩm gồm 15 câu. Sau khi làm trắc nghiệm hết 15 câu, hệ thống sẽ hiển thị một thông báo để hiện số câu làm đúng của người dùng (chẳng hạn người dùng làm đúng 10 câu thì sẽ có thông báo là “chúc mừng bé đã trả lời đúng 10/15 câu”). Trong thông báo cũng hiện 2 nút “chơi lại” và “thoát”. Nếu nhấn “chơi lại” thì hệ thống sẽ reset lại danh sách câu hỏi và đáp án để bắt đầu chơi lại. Nếu nhấn “thoát” thì hệ thống sẽ thoát ra giao diện chọn phạm vi. Các phạm vi 0 – 20, 0 – 100 có cách chơi tương tự như phạm vi 0 – 10.

* Chức năng so sánh

Người dùng nhấn vào chức năng so sánh, hệ thống chuyển người dùng sang giao diện so sánh. Ở đây, hệ thống sẽ hiển thị các câu trắc nghiệm và 3 đáp án A, B, C tương ứng với 3 dấu so sánh “<”, “>” và “=”cho người dùng chọn (các câu hỏi được lấy từ CSDL của hệ thống và được lấy ngẫu nhiên). Sau khi chọn 1 trong các đáp án A, B, C, hệ thống sẽ kiểm tra tính đúng/ sai của đáp án. Nếu là đáp án đúng/ sai hệ thống sẽ phát ra âm thanh của đáp án đúng/ sai, đồng thời hệ thống sẽ chuyển sang câu hỏi tiếp theo cho đến khi hết 15 câu. Sau khi làm trắc nghiệm hết 15 câu, hệ thống sẽ hiển thị một thông báo để hiện số câu làm đúng của người dùng (chẳng hạn người dùng làm đúng 10 câu thì sẽ có thông báo là “chúc mừng bé đã trả lời đúng 10/15 câu”). Trong thông báo cũng hiện 2 nút “chơi lại” và “thoát”. Nếu nhấn “chơi lại” thì hệ thống sẽ reset lại danh sách câu hỏi và đáp án để bắt đầu chơi lại. Nếu nhấn “thoát” thì hệ thống sẽ thoát ra giao diện phần luyện tập.

* Chức năng tập đếm

Người dùng nhấn vào chức năng tập đếm, hệ thống chuyển người dùng sang giao diện tập đếm. Ở đây, hệ thống sẽ hiển thị các hình là các con vật, đồ vật, các hình khối cơ bản và 4 đáp án A, B, C, D (các câu hỏi được lấy từ CSDL của hệ thống và được lấy ngẫu nhiên). Nhiệm vụ của người dùng là đếm xem có bao nhiêu đồ vật, hình khối và chọn đáp án đúng. Sau khi chọn đáp án, hệ thống sẽ kiểm tra tính đúng/ sai của đáp án. Nếu là đáp án đúng/ sai thì hệ thống sẽ phát ra âm thanh của đáp án đúng/ sai, đồng thời hệ thống sẽ chuyến sang câu tiếp cho đến hết 10 câu. Sau khi làm trắc nghiệm hết 10 câu, hệ thống sẽ hiển thị một thông báo để hiện số câu làm đúng của người dùng (chẳng hạn người dùng làm đúng 10 câu thì sẽ có thông báo là “chúc mừng bé đã trả lời đúng 10/10 câu”). Trong thông báo cũng hiện 2 nút “chơi lại” và “thoát”. Nếu nhấn “chơi lại” thì hệ thống sẽ reset lại danh sách câu hỏi và đáp án để bắt đầu chơi lại. Nếu nhấn “thoát” thì hệ thống sẽ thoát ra giao diện phần luyện tập.

* Chức năng kiểm tra

Khi nhấn vào chức năng kiểm tra, hệ thống sẽ chuyển người dùng sang giao diện kiểm tra. Ở đây, hệ thống hiển thị các câu hỏi và 4 đáp án A, B, C, D. Kể từ thời điểm người dùng nhấn kiểm tra, hệ thống sẽ bắt đầu tính giờ làm bài. Thời gian làm bài là 10’ cho 20 câu hỏi. Mỗi khi chọn đáp án đúng người dùng được cộng 1 điểm, chọn sai không có điểm. Thời gian kiểm tra sẽ kết thúc khi người dùng hoàn thành hết 20 câu hỏi hoặc khi đồng hồ đếm giờ đếm đến 10 phút. Sau khi hoàn thành kiểm tra, hệ thống sẽ chuyển người dùng sang giao diện xem điểm. Ở giao diện xem điểm, hệ thống hiển thị điểm vừa thi của người dùng và điểm cao nhất từng đạt được. Trong trường hợp đang làm bài kiểm tra mà người dùng nhấn nút thoát thì sẽ có một thông báo hiện lên hỏi người dùng có chắc chắn muốn thoát không, nếu chọn “có” thì hệ thống sẽ ngay lập tức chuyển sang giao diện xem điểm và tính điểm dựa trên các câu mà người dùng đã làm trước đó, nếu chọn “không” thì bài làm sẽ được tiếp tục.

* Chức năng xem điểm

Khi nhấn vào chức năng xem điểm, hệ thống sẽ hiển thị giao diện xem điểm để người dùng xem được điểm vừa kiểm tra gần nhất và điểm kiểm tra cao nhất từng đạt được.

3.2 Đặc tả yêu cầu phần mềm

Sau khi xác định và phân tích yêu cầu bài toán thì em đã xác định được các Actor và Use-case như bảng dưới đây:

Bảng 3. 1: Danh sách các Actor

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên Actor** | **Ý nghĩa/Ghi chú** |
| 1 | Người dùng | Người sử dụng ứng dụng |

Bảng 3. 2: Danh sách các Use-case

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên Use-case** | **Ý nghĩa/Ghi chú** |
| 1 | Luyện tập | Ôn tập các dạng bài của chương trình toán 1 |
| 2 | Kiểm tra | Cho người dùng thi sau khi luyện tập |
| 3 | Xem điểm | Xem điểm vừa kiểm tra, điểm cao nhất |
| 4 | Tính nhẩm | Tổng hợp các câu hỏi cộng trừ từ 1 đến 100 |
| 5 | So sánh | Tổng hợp các câu so sánh hai số với nhau, hai phép tính với nhau hoặc một số với một phép tính |
| 6 | Tập đếm | Tổng hợp các hình ảnh có số lượng để đếm |

3.2.1 Các yêu cầu chức năng

Diagram

Description automatically generated

Hình 3. 1: Biểu đồ Use-case Tổng quát

* Đặc tả các Use-case của hệ thống
* Đặc tả Use-case “Luyện tập”
* ***Biểu đồ Use-case***

Diagram

Description automatically generated

Hình 3. 2: Biểu đồ Use-case Luyện tập

* ***Tóm tắt***

+ Use-case dành cho người dùng khi muốn vào luyện tập làm các bài toán lớp 1.

* ***Dòng sự kiện***

#### + Dòng sự kiện chính

1. Người dùng chọn chức năng luyện tập.
2. SYSTEM Hệ thống chuyển người dùng sang giao diện luyện tập.

* ***Các dòng sự kiện khác***

+ Không có.

* ***Các yêu cầu đặc biệt***

+ Không có.

* ***Trạng thái hệ thống khi bắt đầu thực hiện Use-case***

+ Khi người dùng khởi động ứng dụng sẽ hiện lên use case này.

* ***Trạng thái hệ thống sau khi thực hiện Use-case***

+ Không có.

* ***Điểm mở rộng***

+ Không có.

* Đặc tả Use-case “Kiểm tra”
* ***Biểu đồ Use-case***

Diagram

Description automatically generated

Hình 3. 3: Biểu đồ Use-case Kiểm tra

* ***Tóm tắt***

+ Use-case giúp người dùng kiểm tra và củng cố lại các kiến thức toán lớp 1 đã học.

* ***Dòng sự kiện***

#### + Dòng sự kiện chính

1. Người dùng chọn chức năng kiểm tra.
2. SYSTEM Hệ thống bắt đầu tính giờ kiểm tra, có 10 phút để làm bài.
3. SYSTEM Hệ thống lấy dữ liệu rồi hiển thị ra màn hình các câu hỏi và các đáp án A, B, C, D.
4. Người dùng chọn đáp án.
5. SYSTEM Hệ thống kiểm tra tính đúng/ sai của đáp án.
6. SYSTEM Hệ thống phát ra âm thanh của đáp án đúng/ sai, nếu đúng sẽ được cộng 1 điểm và chuyển sang câu tiếp theo, nếu sai không được cộng điểm và chuyển luôn sang câu tiếp.
7. Người dùng hoàn thành hết 20 câu.
8. SYSTEM Hệ thống chuyển sang giao diện xem điểm, hiển thị điểm vừa kiểm tra, điểm cao nhất và thông báo đã hoàn thành bài kiểm tra.

* ***Các dòng sự kiện khác***

#### + Dòng sự kiện 1

7a. Người dùng nhấn thoát.

SYSTEM Hệ thống hiện thông báo hỏi người dùng có chắc chắn muốn thoát, nếu chọn “có” thì hệ thống sẽ ngay lâp tức chuyển người dùng sang giao diện xem điểm và hiển thị số điểm dựa trên các câu đã làm, nếu chọn “không” thì hệ thống sẽ cho người dùng tiếp tục kiểm tra.

#### + Dòng sự kiện 2

7b. Thời gian kiểm tra kết thúc.

1. SYSTEM Hệ thống chuyển người dùng sang giao diện xem điểm, hiển thị điểm vừa kiểm tra, điểm cao nhất và thông báo đã hoàn thành bài kiểm tra.

* ***Các yêu cầu đặc biệt***

+ Không có.

* ***Trạng thái hệ thống khi bắt đầu thực hiện Use-case***

+ Khi người dùng vào phần kiểm tra sẽ vào use-case này.

* ***Trạng thái hệ thống sau khi thực hiện Use-case***

+ Không có.

* ***Điểm mở rộng***

+ Không có.

* Đặc tả Use-case “Xem điểm”
* ***Biểu đồ Use -case***

Diagram

Description automatically generated

Hình 3. 4: Biểu đồ Use-case Xem điểm

* ***Tóm tắt***

+ Use-case này cho phép người dùng vào xem điểm kiểm tra gần nhất và điểm cao nhất từng đạt được.

* ***Dòng sự kiện***

#### + Dòng sự kiện chính

1. Người dùng chọn chức năng xem điểm.
2. SYSTEM Hệ thống hiển thị điểm kiểm tra gần nhất và điểm cao nhất từng đạt được.

* ***Các dòng sự kiện khác***

+ Không có.

* ***Các yêu cầu đặc biệt***

+ Không có.

* ***Trạng thái hệ thống khi bắt đầu thực hiện Use-case***

+ Khi người dùng chọn xem điểm sẽ hiện lên use-case này.

* ***Trạng thái hệ thống sau khi thực hiện Use-case***

+ Không có.

* ***Điểm mở rộng***

+ Không có.

* Đặc tả Use-case “Tính nhẩm”
* ***Biểu đồ Use-case***

Diagram

Description automatically generated

Hình 3. 5: Biểu đồ Use-case Tính nhẩm

* ***Tóm tắt***

+ Use-case cho phép người dùng luyện tập các phép cộng trừ trong các phạm vi 0-100.

* ***Dòng sự kiện***

#### + Dòng sự kiện chính

1. Người dùng chọn chức năng tính nhẩm.
2. SYSTEM Hệ thống hiển thị 3 mức phạm vi 0-10, 0-20, 0-100 cho người dùng chọn.
3. Người dùng chọn mức mình muốn làm trắ nghiệm.
4. SYSTEM Hệ thống lấy dữ liệu rồi hiển thị các câu hỏi và các đáp án A, B, C, D.
5. Người dùng chọn đáp án.
6. SYSTEM Hệ thống kiểm tra tính đúng/ sai.
7. SYSTEM Hệ thống phát ra âm thanh của đáp án đúng/ sai, nếu đúng sẽ cộng 1 câu đúng và chuyển sang câu tiếp theo, nếu sai không được cộng câu đúng và chuyển luôn sang câu tiếp.
8. Người dùng hoàn thành 15 câu.
9. SYSTEM Hệ thống hiển thị thông báo chúc mừng đã trả lời đúng ??/15 câu và 2 nút “chơi lại”, “thoát”. Nếu chọn “chơi lại” thì hệ thống sẽ reset lại câu hỏi trong phần tính nhẩm và người dùng được chơi lại, nếu chọn “thoát” thì hệ thống chuyển người dùng về giao diện chọn mức.

|  |
| --- |
|  |

* ***Các dòng sự kiện khác***

#### + Dòng sự kiện

8a. Người dùng nhất thoát.

1. SYSTEM Hệ thống chuyển người dùng về giao diện chọn mức.

* ***Các yêu cầu đặc biệt***

+ Không có.

* ***Trạng thái hệ thống khi bắt đầu thực hiện Use-case***

+ Khi người dùng chọn vào tính nhẩm sẽ hiện lên use case này.

* ***Trạng thái hệ thống sau khi thực hiện Use-case***

+ Sau khi thực hiện use case này người dùng sẽ trở lại giao diện tính nhẩm.

* ***Điểm mở rộng***

+ Không có.

* Đặc tả Use-case “So sánh”
* ***Biểu đồ Use-case***

Diagram

Description automatically generated

Hình 3. 6: Biểu đồ Use-Case So sánh

* ***Tóm tắt***

+ Use-case giúp người dùng tập so sánh hai số với nhau, hai giá trị với nhau hoặc một số với một giá trị.

* ***Dòng sự kiện***

#### + Dòng sự kiện chính

1. Người dùng chọn chức năng so sánh.
2. SYSTEM Hệ thống lấy dữ liệu rồi hiển thị các câu hỏi và các đáp án A, B, C.
3. Người dùng chọn đáp án.
4. SYSTEM Hệ thống kiểm tra tính đúng/ sai.
5. SYSTEM Hệ thống phát ra âm thanh của đáp án đúng/ sai, nếu đúng sẽ cộng 1 câu đúng và chuyển sang câu tiếp theo, nếu sai sẽ không được cộng đâu đúng và chuyển luôn sang câu tiếp.
6. Người dùng hoàn thành 15 câu.
7. SYSTEM Hệ thống hiển thị thông báo chúc mừng đã trả lời đúng ??/15 câu và 2 nút “chơi lại”, “thoát”. Nếu chọn “chơi lại” thì hệ thống sẽ reset lại câu hỏi trong phần so sánh và người dùng được chơi lại, nếu chọn “thoát” thì hệ thống chuyển người dùng về giao diện luyện tập.

* ***Các dòng sự kiện khác***

#### + Dòng sự kiện

6a. Người dùng nhấn thoát.

1. SYSTEM Hệ thống chuyển người dùng về giao diện luyện tập.

* ***Các yêu cầu đặc biệt***

+ Không có.

* ***Trạng thái hệ thống khi bắt đầu thực hiện Use-case***

+ Khi người dùng chọn vào so sánh sẽ hiện lên use case này

* ***Trạng thái hệ thống sau khi thực hiện Use-case***

+ Sau khi thực hiện use-case này người dùng sẽ trở lại giao diện luyện tập.

* ***Điểm mở rộng***

+ Không có.

* Đặc tả Use-case “Tập đếm”
* ***Biểu đồ Use-case***

Diagram

Description automatically generated

Hình 3. 7: Biểu đồ Use-case Tập đếm

* ***Tóm tắt***

+ Use-case giúp người dùng tập đếm thông qua các hình ảnh con vật, đồ vật, hình khối cơ bản.

* ***Dòng sự kiện***

+ Dòng sự kiện chính

1. Người dùng chọn chức năng tập đếm.
2. SYSTEM Hệ thống lấy dữ liệu rồi hiển thị các câu hỏi và đáp án A, B, C, D.
3. Người dùng chọn đáp án.
4. SYSTEM Hệ thống kiểm tra tính đúng/ sai.
5. SYSTEM Hệ thống phát ra âm thanh của đáp án đúng/ sai, nếu đúng sẽ cộng 1 câu đúng và chuyển sang câu tiếp theo, nếu sai sẽ không được cộng câu đúng và chuyển luôn sang câu tiếp.
6. Người dùng hoàn thành 10 câu.
7. SYSTEM Hệ thống hiển thị thông báo chúc mừng đã trả lời đúng ??/10 câu và 2 nút “chơi lại”, “thoát”. Nếu chọn “chơi lại” thì hệ thống sẽ reset lại câu hỏi trong phần tập đếm và người dùng được chơi lại, nếu chọn “thoát” thì hệ thống chuyển người dùng về giao diện luyện tập.

* ***Các dòng sự kiện khác***

#### + Dòng sự kiện

6a. Người dùng nhấn thoát.

1. SYSTEM Hệ thống chuyển người dùng về giao diện luyện tập.

* ***Các yêu cầu đặc biệt***

+ Không có.

* ***Trạng thái hệ thống khi bắt đầu thực hiện Use-case***

+ Khi người dùng chọn vào tập đếm sẽ hiện ra use case này.

* ***Trạng thái hệ thống sau khi thực hiện Use-case***

+ Sau khi thực hiện use case này người dùng sẽ trở về giao diện luyện tập.

* ***Điểm mở rộng***

*+* Không có.

3.2.2 Biểu đồ lớp thực thể

* Biểu đồ lớp

Graphical user interface, application

Description automatically generated with medium confidence

Hình 3. 8: Biểu đồ lớp

## Danh sách các lớp đối tượng và quan hệ

Bảng 3. 3: Danh sách các lớp đối tượng và quan hệ

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên lớp/Quan hệ** | **Loại** | **Ý nghĩa/Ghi chú** |
| 1 | CauHoi | Lớp | Lớp đối tượng câu hỏi |
| 2 | MucDo | Lớp | Lớp đối tượng mức độ |
| 3 | Diem | Lớp | Lớp đối tượng điểm |
| 4 | CauHoi - MucDo | Quan hệ | Một câu hỏi chỉ có một mức độ, một mức độ có thể có nhiều câu hỏi. |

* Mô tả cho từng lớp đối tượng
* Lớp CauHoi

Bảng 3. 4: Mô tả lớp CauHoi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Ý nghĩa/Ghi chú** |
| 1 | maCauhoi | int | Duy nhất | Mã câu hỏi |
| 2 | cauHoi | String | not null | Câu hỏi |
| 3 | dapan\_a | String | not null | Đáp án A |
| 4 | dapan\_b | String | not null | Đáp án B |
| 5 | dapan\_c | String | not null | Đáp án C |
| 6 | dapan\_d | String | not null | Đáp án D |
| 7 | dapan\_dung | String | not null | Đáp án đúng |
| 8 | maMucdo | int | not null | Mã mức độ |
| 9 | Hinhanh | blog | not null | Hình ảnh |

* Lớp MucDo

Bảng 3. 5: Mô tả lớp MucDo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Ý nghĩa/Ghi chú** |
| 1 | maMucdo | int | Duy nhất | Mã mức độ |
| 2 | tenMucdo | String | not null | Tên mức độ |

* Lớp Diem

Bảng 3. 6: Mô tả lớp Diem

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Ý nghĩa/Ghi chú** |
| 1 | Diemvuathi | int | not null | Điểm vừa thi |
| 2 | Diemcaonhat | int | not null | Điểm cao nhất |

3.2.3 Các yêu cầu phi chức năng

* *Yêu cầu về giao diện của hệ thống*

+ Giao diện dễ nhìn, hiển thị rõ ràng các chức năng chính.

+ Màu sắc hài hoà, dễ nhìn, không gây khó chịu với mắt.

+ Các nút bấm to, rõ nét, dễ thao tác.

+ Các hình ảnh phù hợp với học sinh lớp 1.

+ Chuyển giao diện giữa các layout phải mượt mà, tránh giật lag.

+ Bộ cục ứng dụng phải sắp xếp hợp lý, khoa học, có thẩm mĩ.

* *Yêu cầu về bảo mật của hệ thống*

+ Vì ứng dụng không phân quyền, không lưu và sử dụng thông tin của người dùng nên không cần bảo mật.

3.3 Thiết kế hệ thống

3.3.1 Thiết kê cơ sở dữ liệu

* Mô hình dữ liệu quan hệ

Graphical user interface, diagram

Description automatically generated

Hình 3. 9: Lược đồ quan hệ

* Mô tả cấu trúc bảng
* Bảng cauhoi

Bảng 3. 7: Mô tả chi tiết bảng cauhoi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Ý nghĩa/Ghi chú** |
| 1 | macauhoi | int | Khoá chính | Mã câu hỏi |
| 2 | cauhoi | String | not null | Câu hỏi |
| 3 | dapanA | String | not null | Đáp án A |
| 4 | dapanB | String | not null | Đáp án B |
| 5 | dapanC | String | not null | Đáp án C |
| 6 | dapanD | String | not null | Đáp án D |
| 7 | dapandung | String | not null | Đáp án đúng |
| 8 | mamucdo | int | Khoá ngoại | Mã mức độ |
| 9 | hinhanh | byte | not null | Hình ảnh |

* Bảng mucdo

Bảng 3. 8: Mô tả chi tiết bảng mucdo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Ý nghĩa/Ghi chú** |
| 1 | mamucdo | int | Khoá chính | Mã mức độ |
| 2 | tenmucdo | String | not null | Tên mức độ |

3.3.2 Thiết kế lớp đối tượng

* **Biểu đồ VOPC**
* Biểu đồ lớp VOPC Kiểm Tra

Diagram

Description automatically generated

Hình 3. 10: Biểu đồ lớp VOPC Kiểm Tra

* Biểu đồ lớp VOPC Xem Điểm

Diagram

Description automatically generated

Hình 3. 11: Biểu đồ lớp VOPC Xem Điểm

* Biểu đồ lớp VOPC Tính Nhẩm

Diagram

Description automatically generated

Hình 3. 12: Biểu đồ lớp VOPC Tính Nhẩm

* Biểu đồ lớp VOPC So Sánh

Diagram

Description automatically generated

Hình 3. 13: Biểu đồ lớp VOPC So Sánh

* Biểu đồ lớp VOPC Tập Đếm

Diagram

Description automatically generated

Hình 3. 14: Biểu đồ lớp VOPC Tập Đếm

* Biểu đồ tuần tự

## Biểu đồ tuần tự cho Use-Case Kiểm Tra

Diagram

Description automatically generated

Hình 3. 15: Biểu đồ tuần tự UC Kiemtra

* Biểu đồ tuần tự cho Use-Case Xem Điểm

Diagram

Description automatically generated

Hình 3. 16: Biểu đồ tuần tự UC Xemdiem

* Biểu đồ tuần tự cho Use-Case Tính Nhẩm

Diagram

Description automatically generated

Hình 3. 17: Biểu đồ tuần tự UC TinhNham

* Biểu đồ tuần tự cho Use-Case So Sánh

Diagram

Description automatically generated

Hình 3. 18: Biểu đồ tuần tự UC SoSanh

* Biểu đồ tuần tự cho Use-Case Tập Đếm

Diagram

Description automatically generated

Hình 3. 19: Biểu đồ tuần tự UC TapDem

3.3.3 Thiết kế giao diện

a) Giao diện chính

A picture containing text, sign

Description automatically generated

Hình 3. 20: Giao diện chính

* Thao tác: Khi người dùng mở ứng dụng lên sẽ hiện giao diện này.
* Sự kiện:
* Nhấn vào luyện tập hệ thống sẽ chuyển sang activity luyện tập.
* Nhấn vào kiểm tra hệ thống sẽ chuyển sang activity kiểm tra và lấy dữ liệu ra.
* Nhấn vào xem điểm hệ thống sẽ chuyển sang activity xem điểm và hiển thị thông tin lên màn hình.

b) Giao diện Luyện tập

Graphical user interface, website

Description automatically generated

Hình 3. 21: Giao diện Luyện tập

* Thao tác: Khi người dùng nhấn vào nút luyện tập sẽ chuyển sang giao diện này.
* Sự kiện:
* Nhấn vào tính nhẩm hệ thống sẽ chuyển sang activity chọn phạm vi cho tính nhẩm.
* Nhấn vào so sánh hệ thống sẽ chuyển sang activity so sánh và lấy dữ liệu ra.
* Nhấn vào tập đếm hệ thống sẽ chuyển sang activity tập đếm và lấy dữ liệu ra.

c) Giao diện Kiểm tra

Graphical user interface, application

Description automatically generated

Hình 3. 22: Giao diện Kiểm tra

* Thao tác: Khi người dùng nhấn vào nút kiểm tra sẽ hiện giao diện này, cho người dùng làm bài kiểm tra bằng cách chọn đáp án.
* Sự kiện:
* Thơi gian tự động đếm ngược đến khi hết giờ.
* Điểm: Hiển thị số điểm đạt được dựa trên các câu trả lời đúng đã làm.
* Người dùng chọn 1 trong các đáp án A, B, C, D. Hệ thống sẽ kiểm tra tính đúng/ sai của đáp án. Nếu đúng thì cộng 1 điểm, phát ra âm thanh đúng. Nếu sai thì không cộng điểm, phát ra âm thanh sai. Sau khi chọn đáp án thì hệ thống sẽ chuyển sang câu tiếp theo cho đến hết.

d) Giao diện Xem điểm

Graphical user interface, text, application, chat or text message

Description automatically generated

Hình 3. 23: Giao diện Xem điểm

* Thao tác: Khi người dùng nhấn vào nút xem điểm sẽ hiển thị activity này.
* Sự kiện:
* Hệ thống hiển thị điểm vừa thi gần nhất và điểm cao nhất từng đạt được.

e) Giao diện Chọn phạm vi

Graphical user interface, text, application

Description automatically generated

Hình 3. 24: Giao diện Chọn phạm vi

* Thao tác: Khi người dùng nhấn vào nút tính nhẩm sẽ hiển thị giao diện này.
* Sự kiện:
* Nhấn vào phạm vi (0-10) hệ thống sẽ chuyển sang activity (0-10) và lấy dữ liệu ra.
* Nhấn vào phạm vi (0-20) hệ thống sẽ chuyển sang activity (0-20) và lấy dữ liệu ra.
* Nhấn vào phạm vi (0-100) hệ thống sẽ chuyển sang activity (0-100) và lấy dữ liệu ra.

f) Giao diện Tính nhẩm

Graphical user interface, application

Description automatically generated

Hình 3. 25: Giao diện Tính nhẩm

* Thao tác: Khi người dùng nhấn vào một trong các phạm vi (0-10), (0-20) hoặc (0-100) sẽ hiển thị giao diện này.
* Sự kiện:
* Hệ thống hiển thị các câu hỏi và đáp án cho người dùng chọn.
* Người dùng chọn 1 trong các đáp án A, B, C, D. Hệ thống sẽ kiểm tra tính đúng sai của đáp án. Nếu đúng thì cộng 1 câu đúng, phát ra âm thanh đúng, và hiển thị trên thanh thông báo. Nếu sai thì không được cộng câu đúng, phát ra âm thanh sai. Sau khi chọn đáp án thì thì hệ thống sẽ chuyển sang câu tiếp cho đến hết.
* Khi làm hết số câu trắc nghiệm hệ thống sẽ hiển thị 1 thông báo hiện ra số câu làm đúng của người dùng.

g) Giao diện So Sánh

Graphical user interface, application

Description automatically generated

Hình 3. 26: Giao diện So sánh

* Thao tác: Khi người dùng nhấn nút so sánh sẽ hiển thị giao diện này.
* Sự kiện:
* Hệ thống hiển thị các câu hỏi và đáp án cho người dùng chọn.
* Người dùng chọn 1 trong các đáp án A, B, C. Hệ thống sẽ kiểm tra tính đúng sai của đáp án. Nếu đúng thì cộng 1 câu đúng, phát ra âm thanh đúng, và hiển thị trên thanh thông báo. Nếu sai thì không được cộng câu đúng, phát ra âm thanh sai. Sau khi chọn đáp án thì thì hệ thống sẽ chuyển sang câu tiếp cho đến hết.
* Khi làm hết số câu trắc nghiệm hệ thống sẽ hiển thị 1 thông báo hiện ra số câu làm đúng của người dùng.

h) Giao diện Tập đếm

Graphical user interface, application

Description automatically generated

Hình 3. 27: Giao diện Tập đếm

* Thao tác: Khi người dùng nhấn nút tập đếm sẽ hiển thị giao diện này.
* Sự kiện:
* Hệ thống hiển thị các hình ảnh và đáp án cho người dùng chọn.
* Người dùng chọn 1 trong các đáp án A, B, C, D. Hệ thống sẽ kiểm tra tính đúng sai của đáp án. Nếu đúng thì cộng 1 câu đúng, phát ra âm thanh đúng, và hiển thị trên thanh thông báo. Nếu sai thì không được cộng câu đúng, phát ra âm thanh sai. Sau khi chọn đáp án thì thì hệ thống sẽ chuyển sang câu tiếp cho đến hết.
* Khi làm hết số câu trắc nghiệm hệ thống sẽ hiển thị 1 thông báo hiện ra số câu làm đúng của người dùng.

# TRIỂN KHAI ỨNG DỤNG

4.1 Triển khai các chức năng cho ứng dụng

4.1.1 Các chức năng nghiệp vụ

|  |  |  |
| --- | --- | --- |
| **Chức năng** | **Ý tưởng xây dựng** | **Xử lý nghiệp vụ** |
| Hiển thị câu hỏi và đáp án | Hệ thống lấy dữ liệu từ SQLite và hiển thị lên giao diện. | Khai báo biến databaseHelper, lấy dữ liệu từ SQLite và hiển thị. |

* Chức năng hiển thị câu hỏi và đáp án

+ Để hiển thị được câu hỏi và đáp án trong các chức năng tập đếm, tính nhẩm, so sánh và kiểm tra ta cần lấy dữ liệu từ cơ sở dữ liệu SQLite.

+ Tạo 1 class có tên DatabaseHelper để thực hiện việc rút trích dữ liệu.

+ Sau khi lấy được dữ liệu từ SQLite ta sẽ bỏ dữ liệu vào các đối tượng đã được định nghĩa.

+ Đưa các dữ liệu này vào các Adapter, tiếp đó lấy dữ liệu đã có đưa vào các TextView, ImageView đã được tạo để hứng dữ liệu khi thiết kế giao diện.

+ Sau đó sẽ đưa cả TextView, ImageView đã hứng dữ liệu và hiển thị lên layout của các chức năng cho người dùng thấy.

4.1.2 Các chức năng hệ thống

|  |  |  |
| --- | --- | --- |
| **Chức năng** | **Ý tưởng xây dựng** | **Xử lý nghiệp vụ** |
| Kiểm tra | Khi người dùng bắt đầu kiểm tra, hệ thống sẽ tính giờ 10 phút đếm ngược, lấy dữ liệu từ SQLite và hiển thị ra giao diện cho người dùng làm. Hệ thống hiện điểm số ở trên màn hình, khi người dùng làm đúng 1 câu, hệ thống sẽ cộng 1 điểm và update lại số điểm trên màn hình. Khi kết thúc bài kiểm tra, hệ thống chuyển sang giao diện xem điểm và hiển thị điểm vừa thi, điểm cao nhất. | + Sau khi người dùng nhấn vào nút kiểm tra, hệ thống sẽ bắt đầu tính giờ là 10 phút làm bài. Để có thể tính giờ ta cần tạo 1 hàm onTick trong class Kiemtra rồi tạo 2 biến int cho phút và giây để nó đếm ngược từ 10:00 về 00:00.  + Tiếp theo, hệ thống rút trích dữ liệu từ SQLite rồi đẩy câu hỏi và đáp án lên giao diện cho người dùng làm bài. Ta cần khao bảo 1 biến tên là databaseHelper để kết nối với SQLite.  + Ta tạo 1 hàm selectDapAn, nó có nhiệm vụ kiểm tra tính đúng/ sai của đáp án. Nếu nó check là đúng thi người dùng được cộng 1 điểm, có tiếng âm thanh đúng phát ra và chuyển sang câu tiếp theo, nếu nó check là sai thì người dùng không có điểm và chuyển sang câu tiếp.  + Sau khi làm hết 20 câu thì hệ thống sẽ chuyển người dùng sang giao diện xem điểm và hiển thị điểm vừa thi, điểm cao nhất. |
| Xem điểm | Khi người dùng nhấn vào xem điểm, hệ thống sẽ hiển thị điểm vừa thi và điểm cao nhất dựa vào điểm kiểm tra thi gần đây nhất của người dùng mà hệ thống đã lưu. | + Để xem điểm vừa thi và điểm cao nhất ta tạo một hàm 1 hàm onFinish và khai báo biến Intent, đặt tên nó là diemthi trong class Kiemtra,  + Sau người dùng tiến hành kiểm tra và có điểm thì số điểm sẽ được lưu ngay trong Intent diemthi.  + Khi người dùng nhấn vào xem điểm thì hệ thống sẽ lấy dữ liệu từ biến Intent diemthi và hiển thị số điểm vừa thi của người dùng. Trong các lần thi sau, nếu số điểm người dùng cao hơn các lần thi trước đó thì hệ thống sẽ so sánh 2 giá trị Intent và giá trị nào lớn hơn thì hệ thống sẽ truyền giá trị đó vào điểm cao nhất. |
| Luyện tập | Khi người dùng nhấn vào chức năng luyện tập sẽ hiển thị 3 chức năng nhỏ gồm tập đếm, tính nhẩm và so sánh cho người dùng chọn. Người dùng nhấn vào chức năng nào thì hệ thống sẽ hiển thị câu hỏi và đáp án của chức năng đó từ CSDL SQLite. Hệ thống hiện 1 thanh thông báo trên màn hình, khi người dùng làm đùng 1 câu thì thanh thông báo update lại số câu đúng. Khi làm hết số lượng câu trắc nghiệm thì hệ thống hiển thị 1 thông báo hiện ra số câu đúng mà người dùng làm được. | + Sau khi người dùng nhấn vào nút tập đếm/ tính nhẩm/ so sánh, hệ thống sẽ rút trích dữ liệu từ SQLite rồi đẩy câu hỏi và đáp án lên giao diện cho người dùng làm trắc nghiệm. Ta cần khai bảo 1 biến tên là databaseHelper để kết nối với SQLite.  + Ta tạo 1 hàm selectDapAn, nó có nhiệm vụ kiểm tra tính đúng/ sai của đáp án. Nếu nó check là đúng thì người dùng được cộng 1 câu đúng và nó sẽ hiển thị trong giao diện là “bạn đã làm làm đúng 1/15 câu”, có tiếng âm thanh đúng phát ra và chuyển sang câu tiếp theo, nếu nó check là sai thì không được cộng câu đúng và chuyển sang câu tiếp cho đến hết số lượng câu hỏi.  + Sau khi làm hết câu hỏi thì hệ thống hiển thị 1 thông báo “chúc mừng bé đã trả lời đúng ??/15” (trong đó ?? là số câu làm đúng). Tạo 1 biến score để lưu và update số câu đúng trong thanh thông báo. Dựa vào hàm selectDapAn và checkDapAn để biết có bao nhiêu câu đúng và hiển thị trên thông báo. |

4.2 Kiểm thử và triển khai ứng dụng

Kiểm thử là bước vô cùng quan trọng trong quá trình phát triển phần mềm. Trong thực tế, không có bất kì sản phẩm nào mà không có lỗi. Tuy nhiên, để hạn chế tối đa số lượng lỗi có thể xuất hiện trong ứng dụng, cần phải tiến hành kiểm thư để đem lại chất lượng tốt nhất cho ứng dụng.

4.2.1 Kiểm thử đơn vị

Để kiếm thử đơn vị cho ứng dụng, cần tiến hành viết các ca kiểm thử đơn vị cho các đối tượng trong ứng dụng để kiểm tra các hàm chức năng của chúng đã hoạt động đúng hay chưa, có hiêu quả không? Để có thể thuận tiện trong việc theo dõi và quản lý, các ca kiểm thử cần được phân chia thành các nhóm theo đối tượng.

+ Kiểm tra giá trị khởi tạo ban đầu của đối tượng, giá trị của đối tượng khi tăng lên 1 đơn vị và giá trị của đối tượng khi giảm xuống 1 đơn vị.

+ Kiểm tra kết nối ứng dụng với cơ sở dữ liệu SQLite: yêu cầu kiểm tra tinh toàn vẹn của dữ liệu khi tạo hoặc bất cứ chức năng nào có liên quan tới cơ sở dữ liệu. Khi rút trích dữ liệu thì kết quả trả về có tương ứng với kết quả nhìn thấy được trên giao diện không, dữ liệu được lấy về có đủng đủ không, có thiếu gì không?,...

4.2.2 Kiểm thử thủ công

Không sử dụng bất kì công cụ kiểm thử hay đoạn mã nào và đặt mình vào vị trí của người dùng cuối và tiến hành kiểm thử thủ công trên ứng dụng để xác định các lỗi không mong muốn từ phía người dùng.

* Môi trường kiểm thử

+ Điện thoại OPPA A3S, hệ điều hành Android 8.1, màn hình 6.2 inch.

+ Điện thoại Redmi Note 7, hệ điều hành Android 9.0, màn hình 6.7 inch.

+ Điện thoại Nokia 7.1 Plus, hệ điều hành Android 8.1, màn hình 6.18 inch.

* Các lỗi tìm thấy

+ Trong các chức năng tập đếm, tính nhẩm, so sánh nếu spam chọn đáp án liên tục thì khi hết số lượng câu trắc nghiệm, màn hình thông báo số câu trả lời đúng xảy ra hiện tượng giật lag.

+ Trong chức năng kiểm tra, nếu spam chọn đáp án liên tục thì khi làm hết 20 câu, giao diện xem điểm xảy ra hiện tượng giật lag.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Kết quả đạt được của đề tài

* *Về kiến thức*

+ Hiểu được các hoạt động trong quy trình phát triển một ứng dụng di động.

+ Hiểu được phương pháp phân tích thiết kế đối tượng và các biểu đồ UML tương ứng trong các giai đoạn đặc tả yêu cầu, phân tích, thiết kế phần mềm và chuyển từ thiết kế chi tiết sang mã nguồn triển khai.

+ Hiểu được các kiến thức về lập trình ứng dụng di động trên nền tảng Android Studio bằng ngôn ngữ lập trình Java.

+ Hiểu được các kiến thức về cơ sở dữ liệu, cách kết nối ứng dụng với CSDL SQLite.

* *Về sản phẩm*

+ Hoàn thành cuốn báo cáo trình bày theo đúng mẫu và yêu cầu.

+ Hoàn thành xây dựng ứng dụng trắc nghiệm ôn toán lớp 1 với đầy đủ yêu cầu về chức năng, ràng buộc như trong đặc tả.

Hạn chế của đề tài

* *Về sản phẩm*

+ Còn thiếu chức năng đăng nhập, chức năng quản lý người dùng, chức năng xếp hạng người dùng dựa vào ID và điểm số.

+ Các chức năng tập đếm, tính nhẩm, so sánh còn đơn giản chưa có hiệu ứng thu hút học sinh lớp 1.

+ Chức năng kiểm tra chưa kết nối được online để nhiều người dùng có thể cùng tham gia kiểm tra.

+ Giao diện ứng dụng còn đơn giản, thiếu phần độc đáo và thu hút người dùng.

+ Cơ sở dữ liệu vẫn chưa bao quát được toàn bộ kiến thức, bài tập của chương trình toán lớp 1.

* *Về kỹ năng*

+ Vận dụng các kiến thức về phân tích thiết kế hướng đối tượng với UML chưa có tính linh hoạt cao; các biểu ca sử dụng, lớp thực thể, lớp phân tích, biểu đồ tuần tự, lớp thiết kế chi tiết ở mức dùng ổn, chưa bao quát toàn bộ hàm ý muốn trình bày.

+ Các thuật toán sử dụng trong ứng dụng ở mức dùng được, chưa phải hiệu quả nhất.

+ Chưa kiểm thử được hết các cách kiểm thử.

Hướng phát triển của đề tài

* *Về sản phẩm*

+ Ứng dụng cần làm thêm chức năng đăng nhập, chức năng quản lý người dùng, chức năng xếp hạng người dùng dựa vào ID.

+ Chỉnh sửa các chức năng tập đếm, tính nhẩm, so sánh linh hoạt hơn, thu hút hơn.

+ Nâng cấp chức năng kiểm tra để có thể kết nối online để nhiều người dùng có thể cùng tham gia kiểm tra cùng lúc.

+ Nâng cấp giao diện ứng dụng sao cho độc đáo và thu hút người dùng hơn.

+ Cơ sở dữ liệu bao quát hết được toàn bộ kiến thức, bài tập của chương trình toán lớp 1.

* *Về kỹ năng*

+ Vận dụng tốt hơn các kiến thức về phân tích thiết kế hướng đối tượng với UML; các biểu ca sử dụng, lớp thực thể, lớp phân tích, biểu đồ tuần tự, lớp thiết kế chi tiết được vẽ chi tiết hơn, bao quát được toàn bộ hàm ý muốn trình bày.

+ Các thuật toán sử dụng trong ứng dụng ở mức hiệu quả nhất.

+ Kiểm thử được hết các cách kiểm thử.

TÀI LIỆU THAM KHẢO

*x*

1. *Giáo trính môn Chuyên đề 1: Lập trình Android cơ bản*, Bộ môn CNPM, Trường ĐHSPKTHY.
2. *Giáo trính môn lập trình hướng đối tượng với Java*, Bộ môn CNPM, Trường ĐHSPKTHY.
3. *Giáo trính môn phân tích thiết kế phần mềm*, Bộ môn CNPM, Trường ĐHSPKTHY.
4. *Giáo trính môn phân tích thiết kế hướng đối tượng với UML*, Bộ môn CNPM, Trường ĐHSPKTHY.
5. Ed Burnette, *Hello, Android Introducing Google’s Mobile Development Platform*, The Pragmatic Bookshelf, 2010.
6. Marziah Karch, *Android for Work Productivity for Professional*, Press, 2010.
7. J.F. DiMarzio, *Android A Programmer’s Guide*, The McGraw-Hill Companies, 2008.
8. Chris Haseman, *Android Essentials*, Apress, 2008.
9. Sams, *Android Application Development*, Lauren Darcey and Shane Conder, 2010.
10. stackoverflow.com, <https://stackoverflow.com>.

[11]. SQLite, https://www.sqlite.org.

[12]. Udemy, <https://www.udemy.com>.

[13]. Youtube, <https://www.youtube.com>.

[14]. *Sách giáo khoa Toán lớp 1*, Nhà xuất bản Giáo Dục Việt Nam, 2011.

[15]. *Vở bài tập Toán lớp 1*, Nhà xuất bản Giáo Dục Việt Nam, 2011.