

TRƯỜNG ĐẠI HỌC SƯ PHẠM HÀ NỘI
KHOA CÔNG NGHỆ THÔNG TIN



LUẬN VĂN TỐT NGHIỆP

ĐỀ TÀI:

**XÂY DỰNG PHẦN MỀM LUYỆN THI CHỨNG CHỈ TIN HỌC
TRÊN MOBILE**

Giảng viên HD: TS. Đỗ Trung Kiên

Sinh viên: Trần Hải Long

Lớp: K66C

Mã sinh viên: 665105037

Hà Nội, ngày.....tháng.....năm 2020

LỜI CAM ĐOAN

Tôi xin cam đoan nội dung và những kết quả của luận văn tốt nghiệp này là do tôi tự nghiên cứu dưới sự hướng dẫn của TS.Đỗ Trung Kiên. Trong toàn bộ nội dung của luận văn, những nội dung được trình bày là của cá nhân tôi hoặc được tổng hợp từ nhiều nguồn tài liệu khác. Tất cả các tài liệu tham khảo đều được trích dẫn rõ ràng ở phần cuối của luận văn.

Tôi xin cam đoan những lời trên là sự thật. Nếu sai tôi xin hoàn toàn chịu trách nhiệm.

Hà Nội, ngày.....tháng.....năm 2020
Sinh viên

Trần Hải Long

LỜI CẢM ƠN

Đầu tiên tôi xin chân thành cảm ơn TS.Đỗ Trung Kiên đã tận tình hướng dẫn và đóng góp những ý kiến quý báu về chuyên môn cũng như các phương pháp nghiên cứu khoa học để tôi có thể thực hiện được khóa luận tốt nghiệp này. Thầy cũng là tấm gương trong mọi mặt của cuộc sống để tôi học tập và noi theo.

Trong suốt quá trình học tập tại trường Đại học Sư phạm Hà Nội, tôi xin chân thành cảm ơn các thầy, cô giáo đã cung cấp cho tôi những kiến thức hữu ích, hỗ trợ cho quá trình phát triển sau này của tôi.

Cuối cùng tôi xin gửi lời cảm ơn tới gia đình tôi đã luôn ủng hộ tôi trên con đường học tập và nghiên cứu với nhiều khó khăn, vất vả. Mặc dù tôi đã cố gắng hết sức trong quá trình làm luận văn nhưng không thể tránh khỏi thiếu sót, rất mong nhận được những góp ý của thầy cô.

Hà Nội, ngày.....tháng.....năm 2020

Sinh viên

Trần Hải Long

MỤC LỤC

DANH MỤC CHỮ VIẾT TẮT.....	4
DANH MỤC HÌNH VẼ.....	3
DANH MỤC BẢNG BIỂU	4
LỜI NÓI ĐẦU.....	5
CHƯƠNG I: GIỚI THIỆU	11
1.1 Nghiên cứu hiện trạng các ứng dụng	12
1.2 Mục đích và phạm vi nghiên cứu	13
CHƯƠNG II: NGHIÊN CỨU GIẢI PHÁP CÔNG NGHỆ	14
2.1 Lựa chọn nền tảng hệ điều hành	15
2.1.1 Một số nền tảng hệ điều hành di động phổ biến	16
2.1.2 So sánh sự phát triển của ứng dụng trên các hệ điều hành di động	17
2.2 Các giải pháp kỹ thuật chủ yếu được nghiên cứu trong đề tài	18
2.2.1 Xây dựng ứng dụng trên nền tảng Android và Android Studio	19
2.2.2 Xây dựng ứng dụng bằng ngôn ngữ lập trình Java.....	20
2.2.3 Hệ quản trị cơ sở dữ liệu SQLite Studio.....	21
CHƯƠNG III: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG.....	22
3.1 Phân tích yêu cầu hệ thống.....	23
3.1.1 Yêu cầu chức năng.....	24
3.1.2 Yêu cầu phi chức năng.....	25
3.2 Thiết kế hệ thống	26
3.2.1 Thiết kế kiến trúc	27
3.2.2 Biểu đồ Use Case.....	28
3.2.3 Thiết kế cơ sở dữ liệu.....	29
3.2.4 Thiết kế giao diện hệ thống	29
CHƯƠNG IV: XÂY DỰNG HỆ THỐNG VÀ KẾT QUẢ ĐẠT ĐƯỢC.....	30
4.1 Quy trình xây dựng	31
4.2 Thử nghiệm.....	32
4.3 Đánh giá kết quả thử nghiệm.....	33
KẾT LUẬN	34
TÀI LIỆU THAM KHẢO.....	35

DANH MỤC CHỮ VIẾT TẮT

Chữ Viết Tắt	Điễn giải
CSDL	Cơ sở dữ liệu
HĐH	Hệ điều hành

DANH MỤC HÌNH VẼ

Hình 2.1	Hệ điều hành Android của Google	12
Hình 2.2	Hệ điều hành iOS của Apple	12
Hình 2.3	Hệ điều hành Windows Phone của Microsoft.....	12
Hình 2.4	Thị phần hđh cho thiết bị di động trên toàn thế giới – Tháng 1/2020	12
Hình 2.5	Kiến trúc của hệ điều hành Android	12
Hình 2.6	Lớp Linux Kernel	12
Hình 2.7	Lớp Libraries và Android Runtime	12
Hình 2.8	Lớp Application Framework	12
Hình 2.9	Lớp Applications	12
Hình 2.10	Môi trường phát triển Android Studio	12
Hình 2.11	Ngôn ngữ lập trình Java	12
Hình 2.12	Cơ sở dữ liệu SQLite	12
Hình 3.1	Kiến trúc của hệ thống	12
Hình 3.2	Biểu đồ Usecase cho nhóm chức năng làm bài thi	12
Hình 3.3	Biểu đồ Usecase cho nhóm chức năng ôn tập	12
Hình 3.4	Biểu đồ Usecase cho nhóm chức năng kết nối	12
Hình 3.5	Giao diện màn hình khởi động	12
Hình 3.6	Giao diện Menu chính	12
Hình 3.7	Giao diện Menu phụ	12
Hình 3.8	Giao diện chức năng thi thử	12
Hình 3.8.1	Giao diện làm bài thi	12
Hình 3.8.2	Giao diện kiểm tra bài làm	12
Hình 3.8.3	Giao diện kết quả bài thi	12
Hình 3.8.4	Giao diện chừa bài thi	12
Hình 3.8.5	Giao diện chừa và giải thích đáp án	12
Hình 3.9	Giao diện chức năng xem lịch sử bài thi	12
Hình 3.10	Giao diện chức năng ôn tập lý thuyết	12
Hình 3.11	Giao diện bài giảng ôn tập lý thuyết	12
Hình 3.12	Giao diện Menu ngân hàng câu hỏi	12
Hình 3.13	Giao diện nội dung câu hỏi trong ngân hàng câu hỏi	12
Hình 3.14	Giao diện danh sách câu hỏi trong ngân hàng câu hỏi	12
Hình 3.15	Giao diện liên kết cộng đồng	12
Hình 3.16	Giao diện gửi phản hồi cho nhà phát triển	12
Hình 3.17	Sơ đồ thực thể liên kết của hệ thống	12
Hình 3.18	Mô hình các bảng trong cơ sở dữ liệu	12
Hình 4.1	Hàm getQuestions lấy thông tin về câu hỏi trong CSDL	12

Hình 4.2	Lấy câu hỏi theo độ khó và Module để sinh đề ngẫu nhiên	12
Hình 4.3	Hàm lấy giá trị của đáp án được người dùng chọn	12
Hình 4.4	Hàm trả về ký tự A, B, C, D từ đáp án người dùng chọn	12
Hình 4.5	Hàm lưu kết quả chọn của người dùng	12
Hình 4.6	Hàm hiển thị đáp án và giải thích cho mỗi câu hỏi	12
Hình 4.7	Màn hình khởi động ứng dụng	12
Hình 4.8	Màn hình Menu chính	12
Hình 4.9	Menu Module để thi	12
Hình 4.10	Màn hình làm bài thi	12
Hình 4.11	Người dùng chọn đáp án	12
Hình 4.12	Người dùng kiểm tra lại bài trước khi nộp	12
Hình 4.13	Người dùng nộp bài và hệ thống tính điểm	12
Hình 4.14	Bài làm của người dùng sau khi hệ thống chừa	12
Hình 4.15	Hệ thống chừa bài và hiển thị giải thích đáp án	12
Hình 4.16	Màn hình lịch sử bài thi	12
Hình 4.17	Màn hình chi tiết lịch sử bài thi	12
Hình 4.18	Bài chừa và lựa chọn của người dùng được lưu lại	12

DANH MỤC BẢNG BIỂU

Bảng 1.1 So sánh các chức năng của một số ứng dụng luyện thi tin học hiện tại	12
Bảng 2.1 Thị phần hđh cho thiết bị di động trên toàn thế giới (2016-2017)	12
Bảng 2.2 So sánh giữa hệ điều hành Android và iOS	12
Bảng 3.1 Danh sách các tác nhân của hệ thống	12
Bảng 3.2 Danh sách các chức năng cần có của hệ thống.....	12
Bảng 3.3 Bảng Cau_hoi	12
Bảng 3.4 Bảng Bai_thi	12
Bảng 3.5 Bảng CT_Bai_thi	12
Bảng 4.1 So sánh ứng dụng thực nghiệm với các ứng dụng đã có hiện nay	12

LỜI NÓI ĐẦU

Trong những năm gần đây, các thiết bị di động thông minh có một sự tăng trưởng và phát triển mạnh mẽ với sự ra đời liên tục của một loạt các máy tính bảng hiện đại, điện thoại thông minh có tính cạnh tranh cao, do đó thị trường xây dựng các ứng dụng trên các thiết bị này trở nên vô cùng màu mỡ. Hai gian hàng ứng dụng phổ biến nhất hiện nay là App Store (hệ điều hành iOS) và Play Store (hệ điều hành Android). Apple đã mở App Store vào năm 2008 tại thời điểm ra mắt iPhone 3G và bắt đầu với chỉ 500 ứng dụng nhưng tính đến nay, số lượng ứng dụng đã lên đến 2,5 triệu ứng dụng và đạt khoảng 30 tỷ lượt tải về (Theo Forbes). Bên cạnh đó, Google Play cũng có sự tăng trưởng đáng kinh ngạc với 3,6 triệu ứng dụng và khoảng 150 tỷ lượt tải về. Sự tăng trưởng của ứng dụng di động gần như là tất yếu.

Thị trường sử dụng điện thoại di động tại Việt Nam từ đầu năm tới tháng 6/2018 do Appota phát hành có một vài con số đáng chú ý khiến nhiều người không khỏi giật mình. Theo đó, 72% dân số Việt Nam đang dùng điện thoại thông minh, 68% người Việt xem video và nghe nhạc mỗi ngày trên thiết bị di động, có nhiều người cùng lúc sử dụng hơn 2 chiếc điện thoại với tỷ lệ sở hữu máy trung bình là 1,7 máy/người. Nhiều người Việt lén mạng bằng điện thoại nhiều hơn cả máy tính, con số cụ thể là 68%. Với sự tăng trưởng và phát triển mạnh mẽ như vậy, có thể thấy lĩnh vực di động vẫn là một lĩnh vực rất tiềm năng.

Do là thiết bị di động nên chúng có đặc trưng nhỏ gọn, dễ dàng mang theo bên mình. Với đặc tính này, người dùng có thể học và luyện thi chứng chỉ IC3 mọi lúc, mọi nơi bất cứ khi nào có thời gian rảnh rỗi. Thay vì phải đến lớp học truyền thống hay mua tài liệu, giáo trình đắt tiền, người dùng chỉ việc ngồi ở nhà hoặc nghỉ giải lao ở nơi làm việc cũng có thể mở ứng dụng để ôn tập và làm bài thi.

IC3 (Digital Literacy Certification), đây là một chứng chỉ đánh giá khả năng sử dụng máy tính và Internet đầu tiên của quốc tế. Chứng chỉ này được cấp bởi tổ chức Certiport, đây là tổ chức hàng đầu trong việc cấp chứng chỉ tin học.

Cùng với sự phát triển như vũ bão của CNTT cũng như sự phát triển của E-Learning thì việc xây dựng một ứng dụng Mobile-Learning là vô cùng cần thiết và hữu ích không chỉ với những người có nhu cầu học chứng chỉ IC3 mà còn với rất nhiều lứa tuổi khác yêu thích và đam mê với CNTT. Trong tương lai ứng dụng không chỉ cung cấp nội dung về chứng chỉ IC3 mà còn có thể bổ sung thêm nhiều nội dung khác như Toán, Lý, Hóa, Tiếng Anh, Bằng lái xe, Pháp Luật Đại Cương,... để phục vụ nhu cầu học tập đa dạng của người dùng.

CHƯƠNG I: GIỚI THIỆU

1.1 Nghiên cứu hiện trạng các ứng dụng

Hiện nay có rất nhiều ứng dụng liên quan đến học tập như: Ứng dụng ôn thi giấy phép lái xe, ứng dụng học Toán, Lý, Hóa, Tiếng Anh, v...v. Hay như một số ứng dụng trắc nghiệm tin học đại cương: Trắc nghiệm Tin học (EscGroup), Chứng chỉ Tin học CNTT cơ bản (Tuhocvitinh.com), Trắc nghiệm Tin đại cương (HanPro). Nhưng chưa có một ứng dụng nào thực sự đi sâu vào nội dung của chứng chỉ tin học IC3.

Ứng dụng Trắc nghiệm Tin đại cương (HanPro): Là ứng dụng được phát triển để cung cấp cho người dùng chức năng ôn tập, kiểm tra và chữa bài. Tuy nhiên giao diện vẫn rất đơn giản, sơ sài, chưa trực quan với người dùng và chưa có cơ chế lưu điểm của các bài kiểm tra.

Ứng dụng Luyện thi chứng chỉ CNTT cơ bản (Tuhocvitinh.com): Là ứng dụng được phát triển để ôn tập lý thuyết và làm bài kiểm tra. Cũng giống như ứng dụng Trắc nghiệm Tin đại cương của HanPro là giao diện chưa thân thiện với người dùng nhưng đã cải tiến thêm chức năng lưu điểm của bài kiểm tra.

Ứng dụng Trắc nghiệm Tin học (EscGroup): Là ứng dụng được xây dựng để làm bài kiểm tra trắc nghiệm Tin học đại cương cho sinh viên năm nhất tại các trường đại học. Giao diện đẹp mắt, nội dung phong phú, có chức năng thống kê kết quả làm bài. Tuy nhiên ứng dụng chưa có chức năng lưu lại kết quả và ứng dụng sử dụng API có sẵn nên đôi khi người dùng không kết nối được với API sẽ sinh ra lỗi.

Tiêu Chí	Trắc nghiệm Tin đại cương (HanPro)	Luyện thi chứng chỉ CNTT cơ bản (Tuhocvitinh.com)	Trắc nghiệm Tin học (EscGroup)
Hệ Điều Hành			
Giao Diện Hợp Lý	X	X	✓
Câu Hỏi Ôn Tập	✓	✓	X
Tài Liệu Ôn Tập	X	✓	X
Làm Bài KT	✓	✓	✓
Chữa Bài KT	✓	✓	✓

Giải Thích Đáp Án	X	X	X
Lưu KQ Bài Làm	X	✓	✓
Xem Lại Bài Làm	X	X	X
Kết Nối Cộng Đồng	X	X	X

Bảng 1.1: So sánh các chức năng của một số ứng dụng luyện thi tin học hiện tại.

Với hiện trạng các ứng dụng di động luyện thi chứng chỉ Tin học nói chung và luyện thi chứng chỉ đánh giá khả năng sử dụng máy tính và Internet IC3 nói riêng, có thể thấy nhiều tính năng hỗ trợ người dùng trong việc luyện tập và thực hành còn chưa được khai thác hết. Việc xây dựng một ứng dụng với đầy đủ các chức năng hoàn chỉnh cho việc luyện tập và kiểm tra không chỉ hướng đến những đối tượng có nhu cầu thi chứng chỉ IC3 mà còn có thể thay đổi nội dung (Toán, Lý, Hóa, Tiếng Anh, Pháp Luật đại cương,...) để nhắm đến nhiều đối tượng khác như học sinh, sinh viên, người đi làm...

1.2 Mục đích và phạm vi nghiên cứu

Mục đích chính của luận văn là nghiên cứu và xây dựng Ứng dụng luyện thi chứng chỉ đánh giá khả năng sử dụng máy tính và Internet IC3 trên điện thoại di động. Để thực hiện được mục đích này, đầu tiên cần đánh giá và phân tích sự phát triển của điện thoại di động và sự phát triển của thị trường điện thoại di động hiện nay. Từ đó đánh giá môi trường phát triển ứng dụng trên hai nền tảng hệ điều hành điện thoại di động phổ biến là Android và IOS, so sánh để có thể lựa chọn nền tảng phù hợp nhất. Tiếp theo, luận văn sẽ nghiên cứu các giải pháp công nghệ phục vụ việc xây dựng ứng dụng (Công nghệ phát triển ứng dụng Android, ngôn ngữ lập trình Java, cơ sở dữ liệu SQLite). Cuối cùng sẽ thiết kế và xây dựng Ứng dụng luyện thi chứng chỉ đánh giá khả năng sử dụng máy tính và Internet IC3 trên điện thoại di động.

Đối với Ứng dụng luyện thi chứng chỉ IC3 trên điện thoại di động, các chức năng chính sẽ được xây dựng cụ thể như sau:

- Cung cấp ngân hàng câu hỏi để luyện tập, có hiển thị đáp án từng câu.
- Cung cấp tài liệu ôn tập ngay trên ứng dụng hoặc đường dẫn đến các tài liệu online.
- Chức năng làm bài thi, đề thi bám sát với đề thi thực tế.
- Chức năng tính điểm thi và chừa bài thi (Chừa từng câu và giải thích đáp án).
- Chức năng lưu lại kết quả bài thi và chi tiết bài làm (Lịch sử bài thi)
- Chia sẻ một số nguồn tài liệu, cộng đồng luyện thi IC3 tin cậy.

Về phía Client, lịch sử bài kiểm tra và chi tiết bài làm sẽ được lưu tại máy của người dùng để đảm bảo việc sử dụng ứng dụng là offline và private với từng thiết bị. Ngoài ra kết quả hiển thị đến người dùng cũng cần nhanh chóng và chính xác.

Dựa trên những mục tiêu đã nêu ra, luận văn chia thành các phần như sau:

Chương II có nội dung chính là nghiên cứu các giải pháp công nghệ được sử dụng để xây dựng ứng dụng. Đầu tiên sẽ phân tích, so sánh để lựa chọn được nền tảng hệ điều hành phát triển ứng dụng trên máy khách là iOS hay Android – hai nền tảng hệ điều hành thiết bị di động phổ biến nhất hiện nay. Đồng thời làm rõ đối với nền tảng được lựa chọn, môi trường phát triển và ngôn ngữ lập trình được sử dụng cũng như hệ quản trị cơ sở dữ liệu dùng cho hệ thống.

Ở chương III, luận văn tập trung vào việc phân tích và thiết kế hệ thống. Đầu tiên, phần phân tích yêu cầu sẽ làm rõ yêu cầu chức năng và phi chức năng của hệ thống. Sau đó, phần phân tích thiết kế sẽ bao gồm thiết kế kiến trúc, thiết kế biểu đồ ca sử dụng và thiết kế cơ sở dữ liệu.

Sau khi đã phân tích và thiết kế hệ thống, cuối cùng ở chương IV sẽ trình bày các bước xây dựng một số chức năng chính cho hệ thống và các kết quả đạt được của ứng dụng.

CHƯƠNG II: NGHIÊN CỨU GIẢI PHÁP CÔNG NGHỆ

2.1 Lựa chọn nền tảng hệ điều hành

Hệ điều hành trên di động có những bước phát triển rất nhanh chóng. Mỗi thời kỳ lại đánh dấu sự lên ngôi hay thoái trào của các nền tảng. Dưới đây luận văn đưa ra ba nền tảng hệ điều hành trên di động phổ biến nhất trên thế giới.

2.1.1 Một số nền tảng hệ điều hành di động phổ biến

a. Hệ điều hành Android

Android là một sản phẩm của tổng công ty Android (Android, Inc) với sự hỗ trợ tài chính của Google. Năm 2005 Google chính thức mua lại công ty này để phát triển hệ điều hành Android. Hệ điều hành này dựa trên nền tảng Linux, được viết để chạy trên các thiết bị smartphone cũng như máy tính bảng.

Android là nền tảng hệ điều hành mã nguồn mở có số lượng người dùng đông đảo nhất trên thế giới tính đến thời điểm hiện tại. Quý 1 năm 2020, thị phần của Android mặc dù có xu hướng giảm so với năm 2017 nhưng vẫn chiếm tới 74.3% và có gần 3 tỷ thiết bị

đã được kích hoạt. Không chỉ sử dụng trên các thiết bị di động, Android còn được tùy biến cho TV, máy chơi game, ô-tô và các thiết bị điện tử khác.



Hình 2.1: Hệ điều hành Android của Google.

Android cũng là hệ điều hành mã nguồn mở cho phép tùy biến rất sâu. Hệ điều hành này có một cộng đồng phát triển rộng lớn. Mỗi một người dùng đều có thể trở thành nhà phát triển. Giao diện thân thiện, số lượng ứng dụng dồi dào cùng với khả năng cá nhân hóa cao tạo ra sự phổ biến và hấp dẫn nhất định của Android đối với người dùng.

b. Hệ điều hành iOS

iOS là hệ điều hành do tập đoàn Apple phát triển, được mệnh danh là nền tảng thay đổi cách sử dụng điện thoại của người tiêu dùng. Ban đầu hệ điều hành này chỉ dành riêng cho sản phẩm Iphone của hãng. Tuy nhiên sau này iOS được tùy biến để phù hợp với các sản phẩm khác của Apple là Ipad và Ipod Touch. iOS là một hệ sinh thái đóng, tuy nhiên lại rất đa dạng về số lượng và chất lượng của ứng dụng.

Năm 2007, chiếc Iphone thế hệ đầu tiên được Apple cho ra mắt. Sự kiện này đánh dấu một bước chuyển mình lớn trong làng di động thế giới. Iphone cùng với nền tảng IOS đã làm thay đổi cách sử dụng điện thoại của người dùng. Điện thoại lúc này trở thành một thiết bị giải trí đa phương tiện.



Hình 2.2:Hệ điều hành iOS của Apple.

IOS là một hệ điều hành độc quyền chỉ được sử dụng trên các sản phẩm của Apple. Với sự tối ưu phần cứng và phần mềm rất tốt, nền tảng này cho người dùng trải nghiệm hết sức mượt mà và ổn định.

c. Hệ điều hành Windows Phone

Windows Phone là bản nâng cấp của Windows Mobile do tập đoàn Microsoft phát hành vào cuối năm 2010. Đây là một nền tảng đóng, mượt mà, ổn định và giá rẻ. Thương hiệu được biết đến nhiều nhất, gắn liền với Windows Phone chính là dòng sản phẩm Lumia của Nokia.

Hệ điều hành Windows Phone được xem là sự giao thoa giữa Android và IOS, bởi nó có được sự ổn định, mượt mà như IOS mà giá lại khá rẻ. Tuy nhiên sự nghèo nàn về ứng dụng cùng việc lập trình ứng dụng khó khăn tạo ra rào cản lớn cho cả người dùng và nhà phát triển.



Hình 2.3:Hệ điều hành Windows Phone của Microsoft.

Hệ điều hành Windows Phone nhanh chóng giảm thị phần và không đủ sức cạnh tranh với các nền tảng khác. Tháng 10/2017, Microsoft chính thức khai tử Windows Phone, đặt dấu chấm hết cho nền tảng 7 năm tuổi này. Sự thoái trào của Windows Phone dẫn đến sự sụp đổ của thương hiệu vốn được xem là biểu tượng của làng di động một thời: Nokia.

2.1.2 So sánh sự phát triển của ứng dụng trên các hệ điều hành di động

Hiện nay các ứng dụng trên điện thoại được phát triển chủ yếu trên hai nền tảng Android và iOS.

Thời điểm	Android	iOS	Windows Phone	Others
2016Q1	83.4%	15.4%	0.8%	0.4%
2016Q2	87.6%	11.7%	0.4%	0.3%
2016Q3	86.8%	12.5%	0.3%	0.4%
2016Q4	81.4%	18.2%	0.2%	0.2%
2017Q1	85.0%	14.7%	0.1%	0.2%

Bảng 2.1: Thị phần hệ điều hành cho thiết bị di động trên toàn thế giới (2016-2017)

Có thể thấy số lượng điện thoại thông minh chạy hệ điều hành Android và iOS phát triển mạnh, chiếm đến hơn 99% thị trường điện thoại di động thông minh hiện nay, trong

khi Windows Phone hiện giờ chỉ còn chiếm khoảng 0.12%, và những thiết bị di động thông minh chạy nền tảng khác chiếm 0.6%. Điều này cho thấy sự phát triển mạnh mẽ của Android và iOS cùng xu hướng hiện tại của người dùng.



Hình 2.4: Thị phần hệ điều hành cho thiết bị di động trên toàn thế giới – Tháng 1/2020

Nếu so sánh về số lượng thiết bị di động, nền tảng Android vẫn vượt trội hơn hẳn iOS do sự đa dạng về phân cấp, các hãng sản xuất,... Mặc dù vậy, nếu đánh giá về mặt ưu, nhược điểm đối với những nhà phát triển khi xây dựng ứng dụng trên hai nền tảng này thì iOS cũng có nhiều lợi thế nhất định.

Tiêu Chí	Android	iOS
Độ Phổ Biến	74.3% (Jan 2020)	24.98% (Jan 2020)
Mức Giá	Đa dạng	Cao
Ứng Dụng	Đa Dạng	Hạn Ché
Chất Lượng Ứng Dụng	Chưa tốt	Tốt
Độ Tùy Biến	Phong phú	Hạn chế
Bảo Mật	Dễ tấn công	Khó tấn công

Bảng 2.2: So sánh giữa hệ điều hành Android và iOS.

Tuy nhiên để phát triển một ứng dụng chạy trên hệ điều hành iOS sẽ mất nhiều thời gian hơn. Lý do đầu tiên giải thích cho việc đó là vì một ứng dụng Android được viết

bằng Java – một ngôn ngữ lập trình hướng đối tượng rất mạnh mẽ với các công cụ mã nguồn mở hỗ trợ đi kèm. Tiếp theo đó, Android còn là nền tảng mã nguồn mở với một cộng đồng phát triển đông đảo giúp cho nhà phát triển có thể thoải mái sáng tạo và xây dựng cho phù hợp với nhu cầu. Trong khi Apple sử dụng hệ sinh thái khép kín điều đó đồng nghĩa với việc nhà phát triển phải tuân theo những tiêu chuẩn nhất định để xây dựng và phát triển ứng dụng.

Không chỉ có vậy, Chi phí lưu trữ Mobile App trên AppStore sẽ có giá cao hơn (gần 200\$ trong khi ở PlayStore chỉ có 25\$). Các nhà phát triển ứng dụng có thể kiếm được 70% doanh thu trên App Store nhưng lại phải bỏ ra khoảng 99 USD để có quyền truy cập vào SDK iPhone. Riêng CH Play của Android, bạn chỉ cần đăng ký 1 lần 25 USD và kiếm được 70% doanh thu.

Bên cạnh đó, App Store phải mất 3-4 tuần để phê duyệt ứng dụng của bạn. Họ rất khắt khe và đặt ra nhiều quy chuẩn cho các nhà phát triển ứng dụng muốn đưa sản phẩm của mình lên cửa hàng này. Android phê duyệt ứng dụng dễ dàng hơn rất nhiều và bạn không cần phải cập nhật thường xuyên như trên IOS. Vì vậy, doanh nghiệp của bạn sẽ tiết kiệm được một khoản chi phí đáng kể. Điều đặc biệt hơn cả là số lượng người dùng smartphone trên nền tảng Android tại Việt Nam nhiều gấp đôi nền tảng iOS. Như vậy nếu chọn hệ điều hành Android thì cơ hội tiếp cận được nhiều đối tượng người sử dụng sẽ cao hơn đồng thời sẽ tiết kiệm được chi phí phát triển và vận hành.

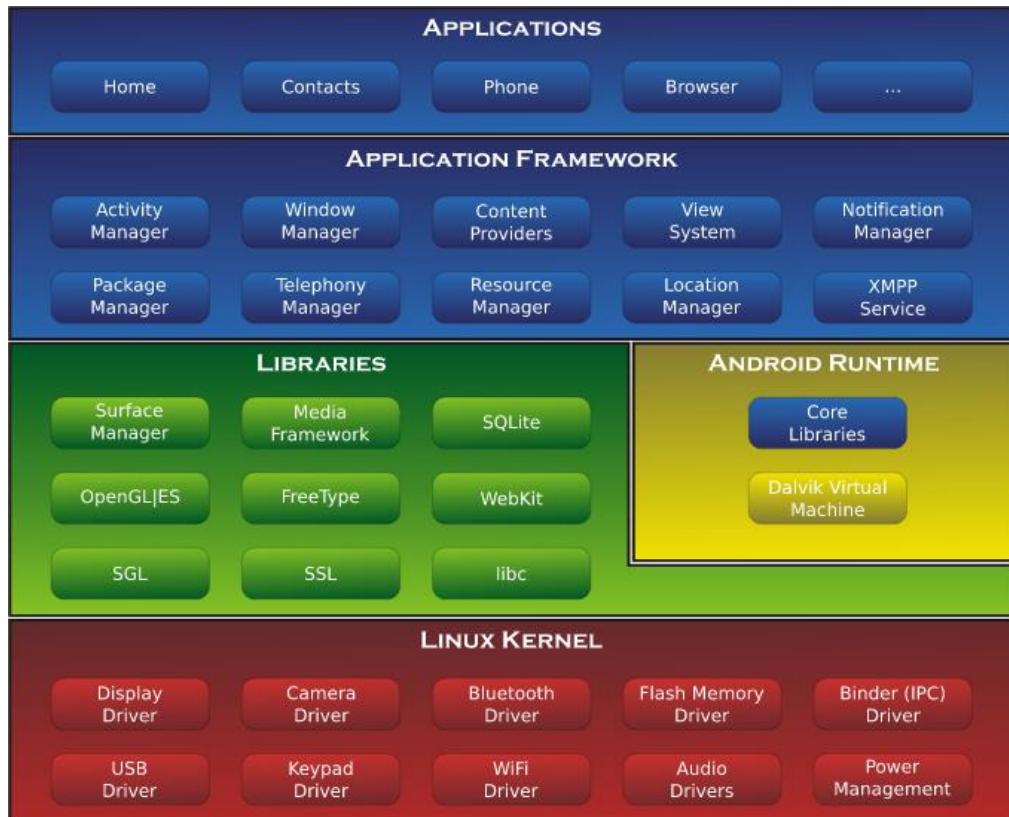
2.2 Các giải pháp kỹ thuật chủ yếu được nghiên cứu trong đề tài

Hiện nay có rất nhiều giải pháp kỹ thuật để xây dựng được một ứng dụng Android hoàn chỉnh chạy trên thiết bị di động. Trong luận văn này sẽ sử dụng một trong những cách phổ biến nhất để xây dựng ứng dụng di động nền tảng Android: Sử dụng Android Studio và ngôn ngữ lập trình Java, kết hợp với CSDL SQLite.

2.2.1 Xây dựng ứng dụng trên nền tảng Android

a. Kiến trúc hệ điều hành Android

Android được hình thành dựa trên nền tảng Linux nhân 2.6, từ phiên bản 4.0 sử dụng Linux nhân 3.x. Android gồm 3 thành phần chính: Middleware, Các thư viện và API viết bằng C, Các ứng dụng thực thi viết bằng Java. Sử dụng máy ảo Dalvik để biên dịch mã .dex (Dalvik Executable) sang Java bytecode.



Hình 2.5: Kiến trúc của hệ điều hành Android.

Kiến trúc của Android được chia làm nhiều tầng, mỗi tầng có nhiệm vụ khác nhau.



Hình 2.6: Lớp Linux Kernel

- **Linux Kernel:** Lõi chính của toàn hệ thống bao gồm các điều khiển phần cứng, bộ quản lý xử lý và bộ nhớ, bảo mật, kết nối mạng, bộ quản lý năng lượng



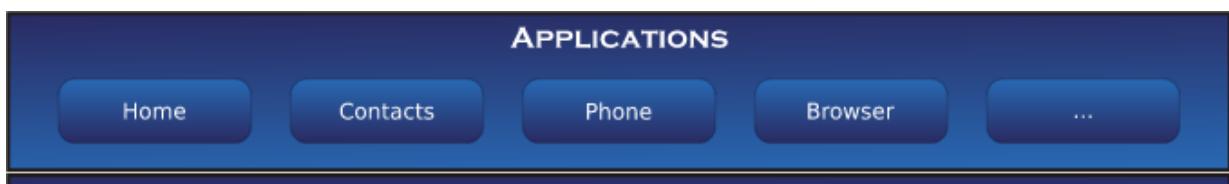
Hình 2.7: Lớp Libraries và Android Runtime.

- **Libraries** - Thực thi trên tầng nhân Linux, bao gồm các thư viện lõi khác nhau của C/C++ như libc và SSL. Có các dạng sau:
 - Thư viện hỗ trợ phát các tập tin đa truyền thông.
 - Bộ quản lý hiển thị
 - Thư viện hỗ trợ đồ họa OpenGL 2D và 3D
 - SQLite hỗ trợ lưu trữ cơ sở dữ liệu
 - SSL và WebKit cho phép tương tác với trình duyệt và bảo mật Internet.
- **Android Run Time:** Đây chính là điểm làm nên sự khác biệt giữa thiết bị Android và thiết bị Linux. Bên trong thành phần này bao gồm máy ảo Dalvik và thư viện lõi. Android Run Time ngoài tăng tốc độ cho ứng dụng còn làm nền cho tầng Application Framework kết nối đến.
- **Core Libraries:** Mặc dù hầu hết các ứng dụng Android viết bằng ngôn ngữ Java nhưng Dalvik không phải là máy ảo Java. Các thư viện lõi Android sẽ cung cấp hầu hết các chức năng chính có thể có trong thư viện Java cũng như thư viện riêng biệt của Android.
- **Dalvik VM:** Dạng máy ảo cho phép tối ưu hóa để có thể chạy được nhiều tiến trình một cách hiệu quả, dựa trên nhân Linux các máy ảo cho phép quản lý các tiến trình và quản lý bộ nhớ ở bậc thấp.



Hình 2.8: Lớp Application Framework.

- **Application Framework:** Cung cấp các lớp cho việc tạo ra các ứng dụng. Bên cạnh đó nó cũng chứa các lớp trùm tượng cho phép truy nhập phần cứng, quản lý giao diện người dùng và tài nguyên của ứng dụng.



Hình 2.9: Lớp Applications.

- **Application Layer:** Gồm các ứng dụng được tích hợp sẵn và các ứng dụng của hãng thứ ba. Tầng ứng dụng trong Android Run Time sử dụng các lớp từ tầng Application Framework để thực thi ứng dụng.

b. Môi trường phát triển và ngôn ngữ lập trình

Có rất nhiều giải pháp để phát triển ứng dụng Android và trong luận văn này sẽ sử dụng môi trường phát triển Android Studio và ngôn ngữ lập trình Java để phát triển ứng dụng. Chi tiết hơn sẽ được trình bày trong phần tiếp theo.

2.2.2 Xây dựng ứng dụng bằng Android Studio và ngôn ngữ lập trình Java

a. Môi trường phát triển Anroid 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. Điểm mạnh của Android Studio là hỗ trợ cả máy ảo cho phép nhà phát triển có thể chạy phần mềm để kiểm thử.

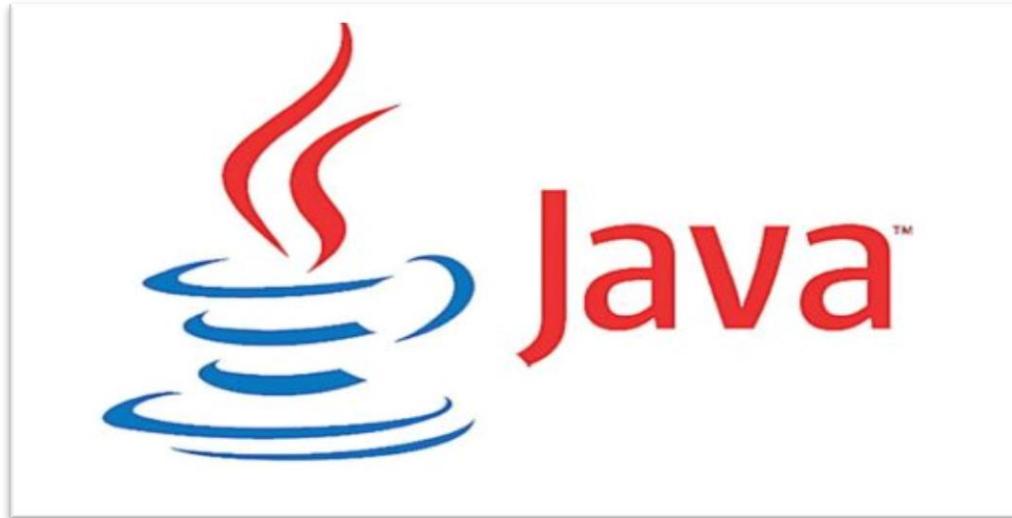


Hình 2.10: Môi trường phát triển Android Studio.

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.

b. Ngôn ngữ lập trình Java

Java là một trong những ngôn ngữ lập trình hướng đối tượng. Nó được sử dụng trong phát triển phần mềm, trang web, game hay ứng dụng trên các thiết bị di động.



Hình 2.11: Ngôn ngữ lập trình Java.

Java được khởi đầu bởi James Gosling và ban đồng nghiệp ở Sun MicroSystem năm 1991. Ban đầu Java được tạo ra nhằm mục đích viết phần mềm cho các sản phẩm gia dụng, và có tên là Oak. Java được phát hành năm 1994, đến năm 2010 được Oracle mua lại từ Sun MicroSystem.

Java được tạo ra với tiêu chí “*Viết một lần, thực thi khắp nơi*” (Write Once, Run Anywhere – WORA). Chương trình phần mềm viết bằng Java có thể chạy trên mọi nền tảng (platform) khác nhau thông qua một môi trường thực thi với điều kiện có môi trường thực thi thích hợp hỗ trợ nền tảng đó.

2.2.3 Hệ quản trị cơ sở dữ liệu SQLite Studio

SQLite là một thư viện phần mềm mà triển khai một SQL Database Engine, không cần máy chủ, không cần cấu hình, khép kín và nhỏ gọn. Nó là một cơ sở dữ liệu, không cần cấu hình, có nghĩa là giống như các cơ sở dữ liệu khác nhưng bạn không cần phải cấu hình nó trong hệ thống của mình.



Hình 2.12: Cơ sở dữ liệu SQLite

SQLite Studio là phần mềm quản lý cơ sở dữ liệu (DBMS) tương tự như Mysql, PostgreSQL... Đặc điểm của SQLite Studio là gọn, nhẹ, đơn giản. Chương trình gồm 1 file duy nhất vỏn vẹn chưa đến 400kB, không cần cài đặt, không cần cấu hình hay khởi động mà có thể sử dụng ngay. Dữ liệu của CSDL cũng được lưu ở một file duy nhất. Không có khái niệm user, password hay quyền hạn trong CSDL SQLite.

CHƯƠNG III: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

3.1 Phân tích yêu cầu hệ thống

Trong lĩnh vực phần mềm khái niệm “yêu cầu” là một trong những điều thường xuyên được nhắc đến. Trong đó, yêu cầu chức năng (functional) và yêu cầu phi chức năng (non-functional) là một trong những điều quan trọng nhất.

3.1.1 Yêu cầu chức năng đối với ứng dụng

Yêu cầu chức năng được định nghĩa là sự mô tả của chức năng hoặc dịch vụ của phần mềm hay hệ thống. Xác định những yêu cầu chức năng chính là việc xác định những tác nhân thực hiện những hành động hướng tới những chức năng cụ thể của hệ thống. Mỗi chức năng cần có mục đích rõ ràng, cụ thể và cần nêu rõ tác nhân hướng tới thực hiện chức năng đó.

STT	Tác Nhân	Mô Tả
1	Người dùng	Những người sử dụng ứng dụng bằng thiết bị di động

Bảng 3.1: Danh sách các tác nhân của hệ thống

STT	Nhóm Chức Năng	Tên Chức Năng	Mô Tả
1	Nhóm chức năng	Thi thử	Người dùng có thể chọn 1 trong 3 module thi của IC3. Hệ thống sẽ sinh đề ngẫu nhiên từ ngân hàng câu hỏi và bấm giờ làm bài. Mỗi đề có 50 câu và thời gian làm là 30p.
2	Nhóm chức năng làm bài thi	Tính điểm, hiển thị kết quả và chũa bài làm	Sau khi người dùng nộp bài, hệ thống sẽ phân tích và tính điểm thi trên bài làm của người dùng, đồng thời sẽ chẽa chi tiết từng câu trong đề và giải thích đáp án.
3		Xem lịch	

		sử bài thi	Sau thi người dùng nộp bài. Hệ thống sẽ lưu lại bài làm của người dùng vào CSDL. Người dùng có thể xem lại chi tiết bài làm của mình.
4	Nhóm chức năng ôn tập	Ôn tập với ngân hang câu hỏi	Hệ thống sẽ cung cấp chức năng cho phép người dùng được xem toàn bộ câu hỏi trong ngân hàng câu hỏi của hệ thống. Người dùng có thể xem đáp án và giải thích của từng câu.
5	Nhóm chức năng ôn tập	Ôn tập với slide bài giảng	Ngoài ôn tập với ngân hàng câu hỏi, hệ thống còn tích hợp bộ giáo trình điện tử để người dùng có thể tham khảo.
6	Nhóm chức năng kết nối	Liên kết đến cộng đồng luyện thi IC3 uy tín	Hệ thống cung cấp chức năng để đưa người dùng đến với cộng đồng luyện thi IC3 đông đảo và uy tín.
7	Nhóm chức năng kết nối	Gửi phản hồi cho nhà phát triển	Hệ thống cung cấp chức năng để người dùng có thể gửi phản hồi cho nhà phát triển thông qua Gmail của Google

Bảng 3.2: Danh sách các chức năng cần có của hệ thống

3.1.2 Yêu cầu phi chức năng đối với ứng dụng

Yêu cầu phi chức năng bao gồm tất cả những yêu cầu mà yêu cầu chức năng không có. Chúng chỉ ra những tiêu chí để đánh giá hoạt động của hệ thống thay vì hành vi. Ngoài yêu cầu chức năng, hệ thống cũng cần đảm bảo một số yêu cầu phi chức năng như sau:

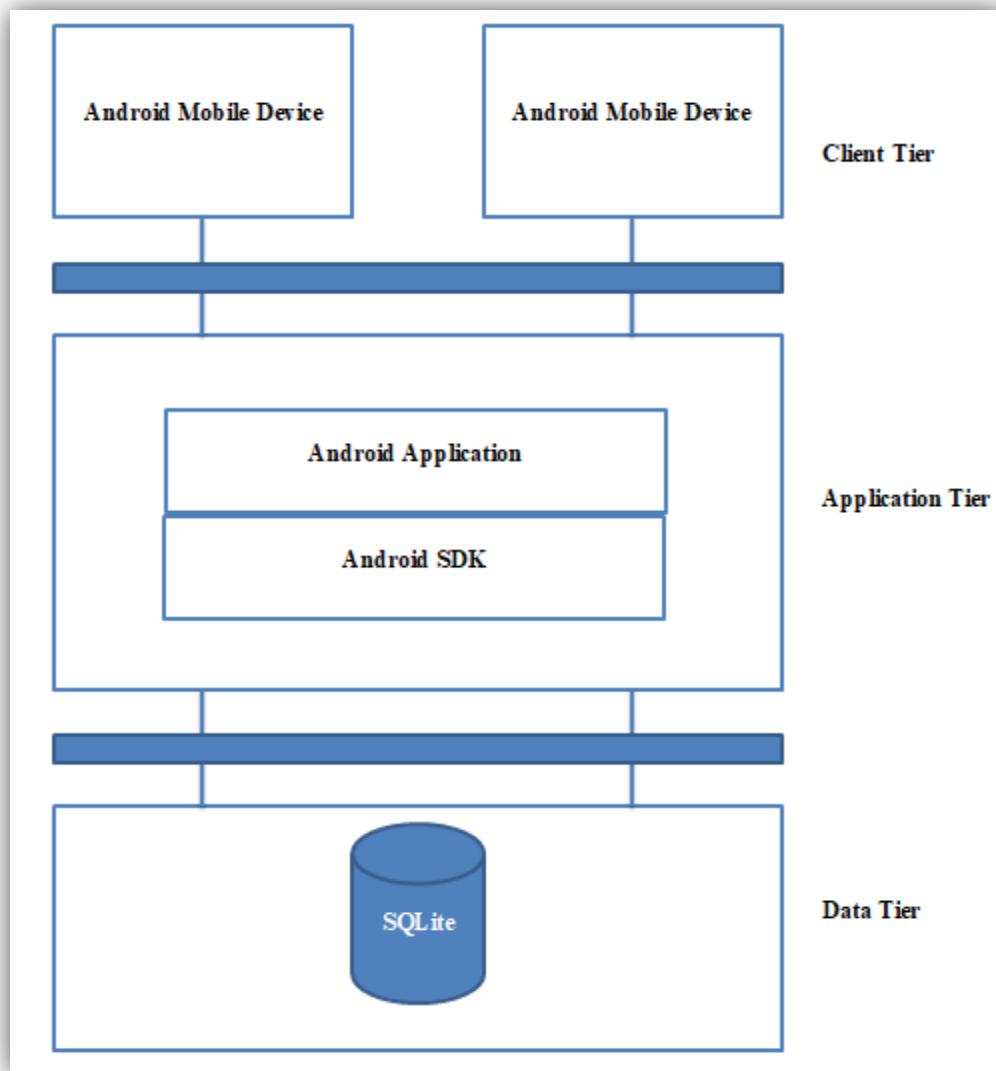
- Giao diện đẹp mắt, thân thiện với người dùng, tuân theo quy tắc thiết kế trên thiết bị di động.
- Hiệu suất ổn định, dễ sử dụng với hầu hết người dùng.
- Có thể chạy ổn định trên phần lớn các phiên bản Android.
- Hệ thống được xây dựng bằng ngôn ngữ lập trình Java.
- Sử dụng hệ quản trị CSDL SQLite để lưu trữ dữ liệu.

- Các chức năng của hệ thống đáp ứng được phần lớn những nhu cầu cơ bản của người sử dụng.

3.2 Thiết kế hệ thống

3.2.1 Thiết kế kiến trúc

Hệ thống hướng tới đối tượng là phần lớn những người sử dụng thiết bị di động, không đòi hỏi người dùng có nhiều kiến thức cũng như hiểu rõ sự hoạt động bên dưới của ứng dụng, cũng không đòi hỏi thiết bị của người dùng phải có kết nối Internet để sử dụng. Do vậy, hệ thống rất phù hợp với việc xây dựng dựa trên mô hình kiến trúc 3 tầng như sau:



Hình 3.1: Kiến trúc của hệ thống.

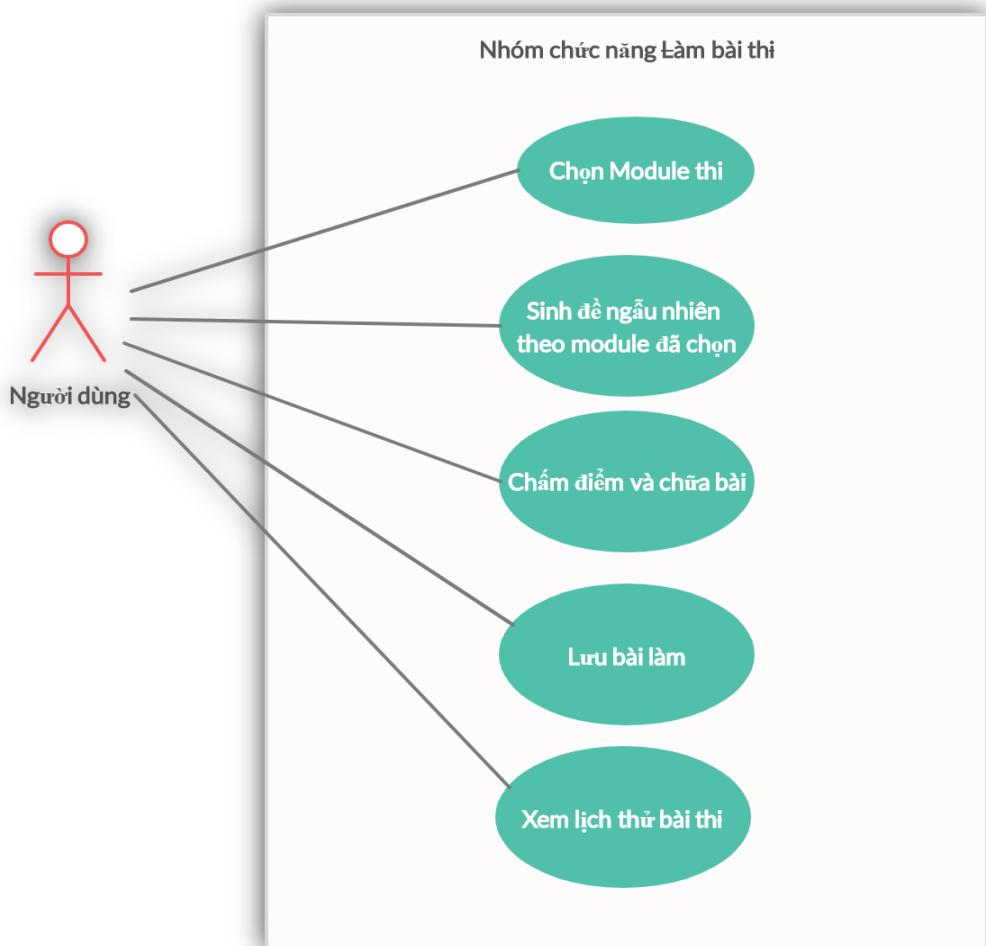
- **Tầng Client:** Các thiết bị di động sử dụng hệ điều hành Android đóng vai trò là client, chứa ứng dụng được xây dựng bằng ngôn ngữ Java. Có nhiệm vụ hiển thị

dữ liệu trong CSDL để phục vụ cho các chức năng như làm bài thi, ôn tập, xem lịch sử bài thi,...

- **Tầng ứng dụng:** Ở tầng này sẽ đảm nhận nhiệm vụ tương tác với client qua việc nhận dữ liệu, sau đó gọi xuống tầng dữ liệu để thực hiện truy vấn, cuối cùng sẽ gửi lại dữ liệu kết quả cho client.
- **Tầng dữ liệu:** Lưu trữ và thực hiện các thao tác truy cập, quản lý, truy vấn đến cơ sở dữ liệu, đóng vai trò trung gian giữa tầng ứng dụng và cơ sở dữ liệu. Ở tầng dữ liệu sẽ lưu trữ các thông tin về ngân hàng câu hỏi, chi tiết từng câu hỏi, lịch sử làm bài thi của người dùng.

3.2.2 Biểu đồ Use Case

Dựa vào các chức năng đã được liệt kê, có thể xây dựng biểu đồ Use Case như sau:



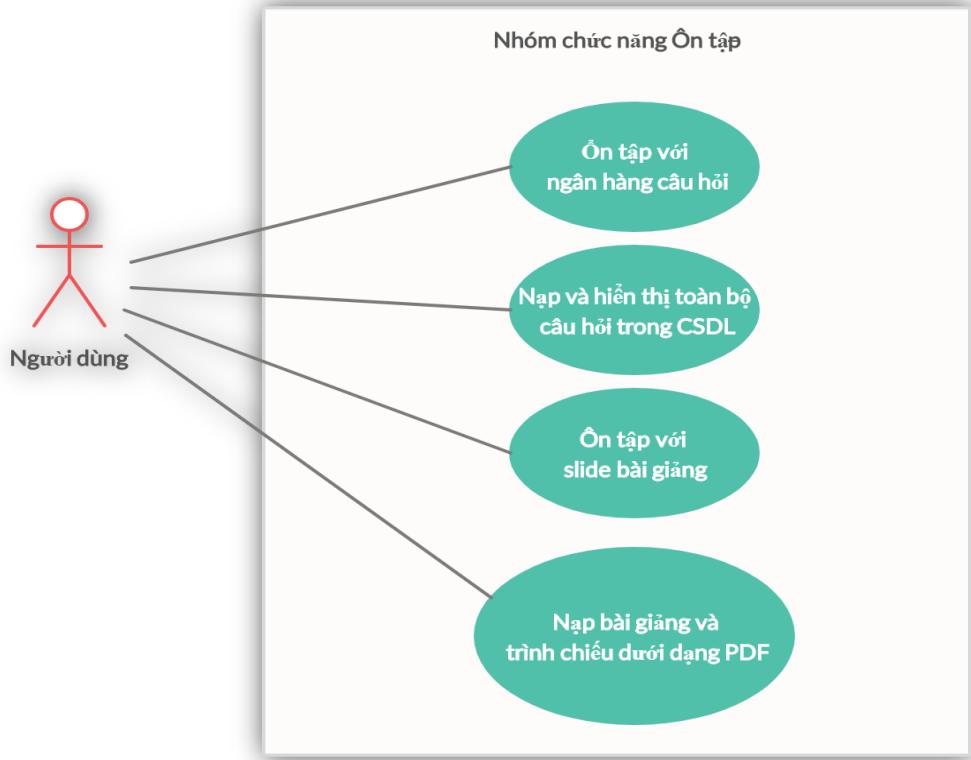
Hình 3.2: Biểu đồ Use Case cho nhóm chức năng Làm bài thi.

a. Thi thử

- **Tên Usecase:** Thi thử
- **Tác nhân chính:** Người dùng
- **Mô tả:** Chức năng có nhiệm vụ cho người dùng chọn 1 trong 3 Module thi. Sau khi người dùng chọn Module thi thì hệ thống sẽ sinh đề ngẫu nhiên bấm giờ làm bài.
- **Điều kiện bắt đầu:** Người dùng đã cài ứng dụng trên thiết bị di động
- **Điều kiện sau khi hoàn thành:** Hệ thống trả về điểm thi và chẽ bài làm chi tiết cho người dùng. Đồng thời lưu lại bài làm.
- **Kịch bản sử dụng chính:**
 - Người dùng mở ứng dụng trên điện thoại di động
 - Chọn chức năng “Thi Thử”
 - Chọn Module thi
 - Hệ thống sinh đề ngẫu nhiên và bấm giờ làm bài
 - Người dùng nộp bài và nhận kết quả thi cũng như bài chẽ chi tiết.
- **Kịch bản sử dụng phụ:**
 - Hết thời gian làm bài mà người dùng vẫn chưa nộp thì hệ thống sẽ tự động ngừng để tính điểm và chẽ bài chi tiết.

b. Xem lịch sử bài thi

- **Tên Usecase:** Lịch sử bài thi
- **Tác nhân chính:** Người dùng
- **Mô tả:** Chức năng có nhiệm vụ cho người dùng xem lại lịch sử bài thi. Người dùng có thể xem chi tiết bài làm của mình (Điểm số, bài làm, số câu đúng, số câu sai, ...) với từng mã đề.
- **Điều kiện sau khi hoàn thành:** Hệ thống trả về thông tin chi tiết lịch sử của từng bài thi.
- **Kịch bản sử dụng chính:**
 - Người dùng mở ứng dụng trên điện thoại di động
 - Chọn chức năng “Lịch sử bài thi”
 - Chọn 1 bài thi
 - Hệ thống hiển thị chi tiết về bài thi đã làm.
- **Kịch bản sử dụng phụ:**
 - Người dùng quay trở lại màn hình chính.



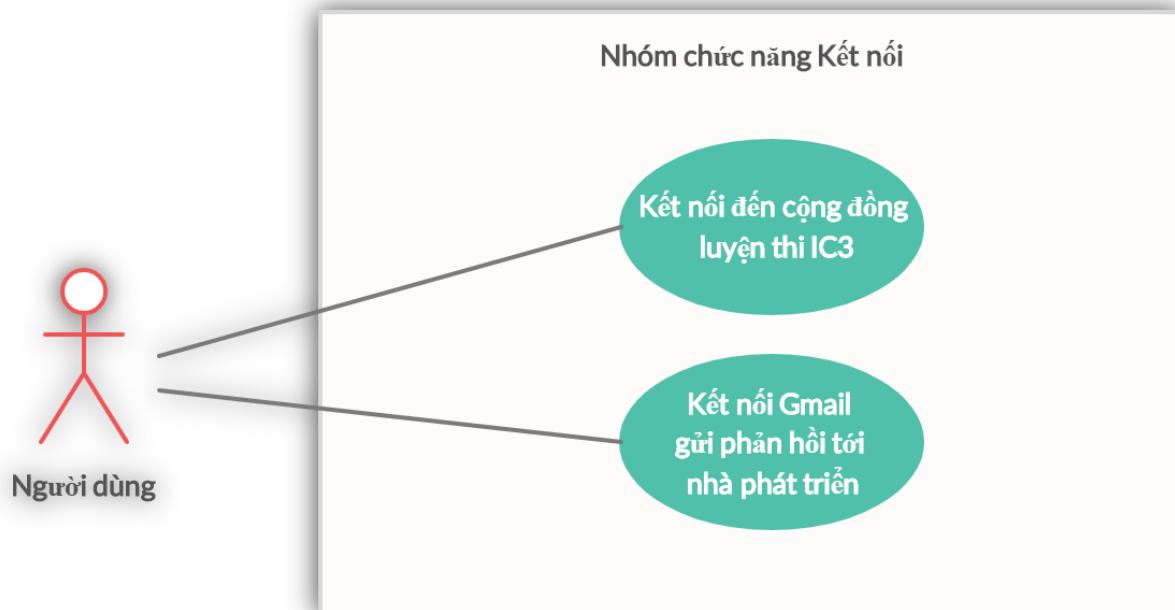
Hình 3.3: Biểu đồ Usecase cho nhóm chức năng Ôn tập.

c. Ôn tập với ngân hàng câu hỏi

- Tên Usecase:** Ôn tập với ngân hàng câu hỏi
- Tác nhân chính:** Người dùng
- Mô tả:** Chức năng có nhiệm vụ nạp ngân hàng câu hỏi trong CSDL và hiển thị cho người dùng. Người dùng xem được đáp án và giải thích chi tiết của từng câu.
- Điều kiện bắt đầu:** Người dùng đã cài ứng dụng trên thiết bị di động
- Điều kiện sau khi hoàn thành:** Hệ thống hiển thị được các câu hỏi lên và cho phép người dùng xem kết quả cũng như giải thích chi tiết của từng câu.
- Kịch bản sử dụng chính:**
 - Người dùng mở ứng dụng trên điện thoại di động
 - Chọn chức năng “Bộ câu hỏi”
 - Chọn Module câu hỏi
 - Hệ thống nạp toàn bộ câu hỏi thuộc lĩnh vực mà người dùng chọn
 - Người dùng có thể xem đáp án và giải thích của từng câu
- Kịch bản sử dụng phụ:**
 - Người dùng quay trở lại màn hình chính

d. Ôn tập với slide bài giảng

- **Tên Usecase:** Ôn tập với slide bài giảng
- **Tác nhân chính:** Người dùng
- **Mô tả:** Chức năng có nhiệm vụ nạp bài giảng được tích hợp sẵn trong ứng dụng và hiển thị với dạng file PDF.
- **Điều kiện bắt đầu:** Người dùng đã cài ứng dụng trên thiết bị di động
- **Điều kiện sau khi hoàn thành:** Hệ thống hiển thị danh sách các bài giảng cho người dùng chọn.
- **Kịch bản sử dụng chính:**
 - Người dùng mở ứng dụng trên điện thoại di động
 - Chọn chức năng “Ôn tập”
 - Chọn bài giảng
 - Hệ thống nạp bài giảng dưới dạng trình chiếu file PDF
- **Kịch bản sử dụng phụ:**
 - Người dùng quay trở lại màn hình chính



Hình 3.4: Biểu đồ Usecase cho nhóm chức năng Kết nối.

e. Kết nối đến cộng đồng IC3

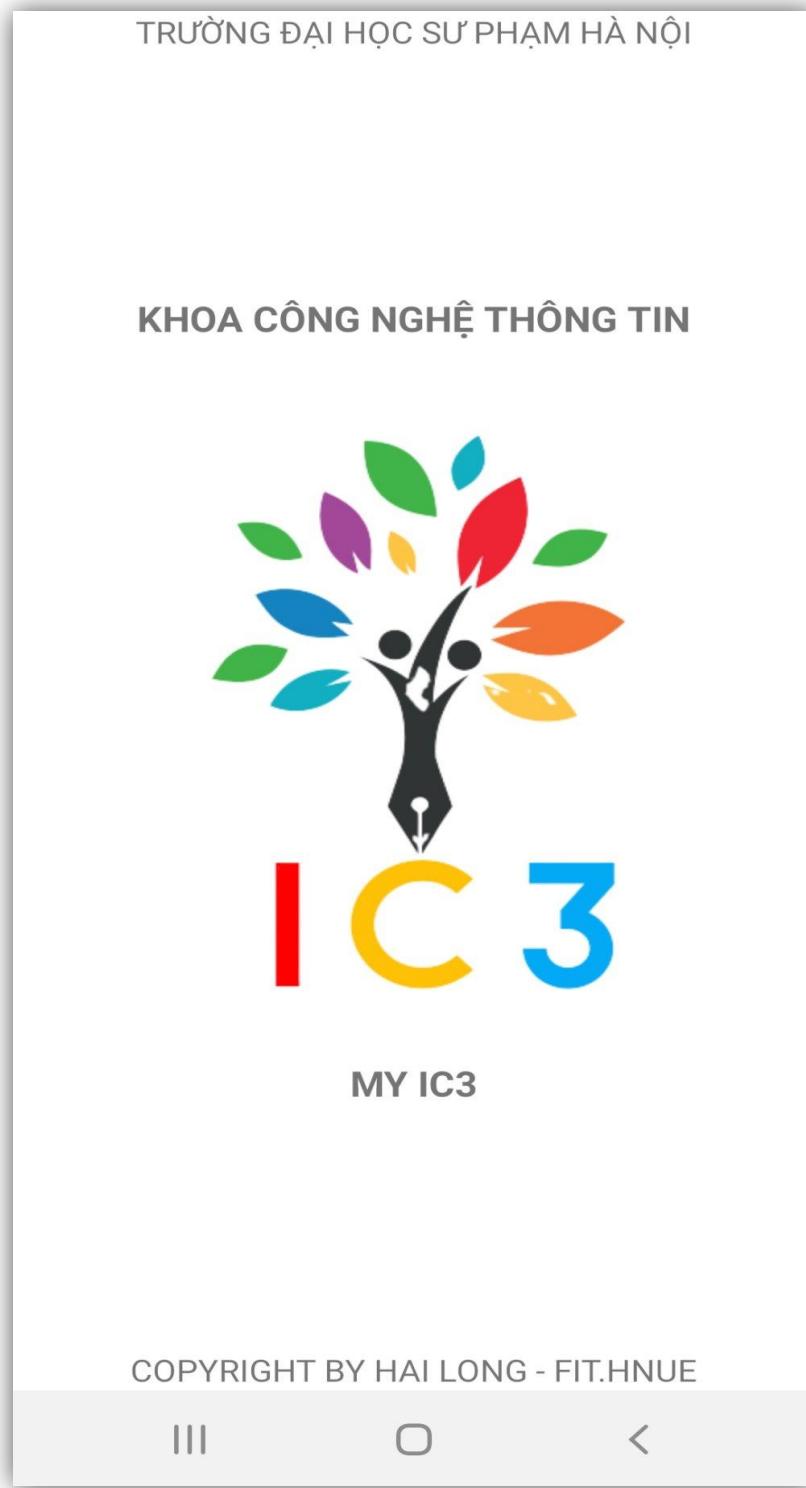
- **Tên Usecase:** Kết nối đến cộng đồng IC3.
- **Tác nhân chính:** Người dùng.
- **Mô tả:** Chức năng có nhiệm vụ mở trình duyệt web và đưa người dùng đến công đồng IC3.
- **Điều kiện bắt đầu:** Người dùng đã cài ứng dụng trên thiết bị di động và có kết nối Internet.
- **Điều kiện sau khi hoàn thành:** Hệ thống hiển thị trang web tới cộng đồng IC3.
- **Kịch bản sử dụng chính:**
 - Người dùng mở ứng dụng trên điện thoại di động.
 - Chọn chức năng “Cộng đồng”.
 - Hệ thống hiển thị trang web tới cộng đồng IC3.
- **Kịch bản sử dụng phụ:**
 - Người dùng quay trở lại màn hình chính.

f. Gửi phản hồi tới nhà phát triển

- **Tên Usecase:** Gửi phản hồi tới nhà phát triển.
- **Tác nhân chính:** Người dùng.
- **Mô tả:** Chức năng có nhiệm vụ mở ứng dụng Gmail tích hợp sẵn trên thiết bị di động và gửi mail vào địa chỉ của nhà phát triển.
- **Điều kiện bắt đầu:** Người dùng đã cài ứng dụng trên thiết bị di động và có kết nối Internet.
- **Điều kiện sau khi hoàn thành:** Hệ thống khởi động Gmail và điền sẵn địa chỉ mail của nhà phát triển.
- **Kịch bản sử dụng chính:**
 - Người dùng mở ứng dụng trên điện thoại di động.
 - Chọn chức năng “Gửi phản hồi”.
 - Hệ thống khởi động Gmail và điền sẵn địa chỉ mail của nhà phát triển.
- **Kịch bản sử dụng phụ:**
 - Người dùng quay trở lại màn hình chính.

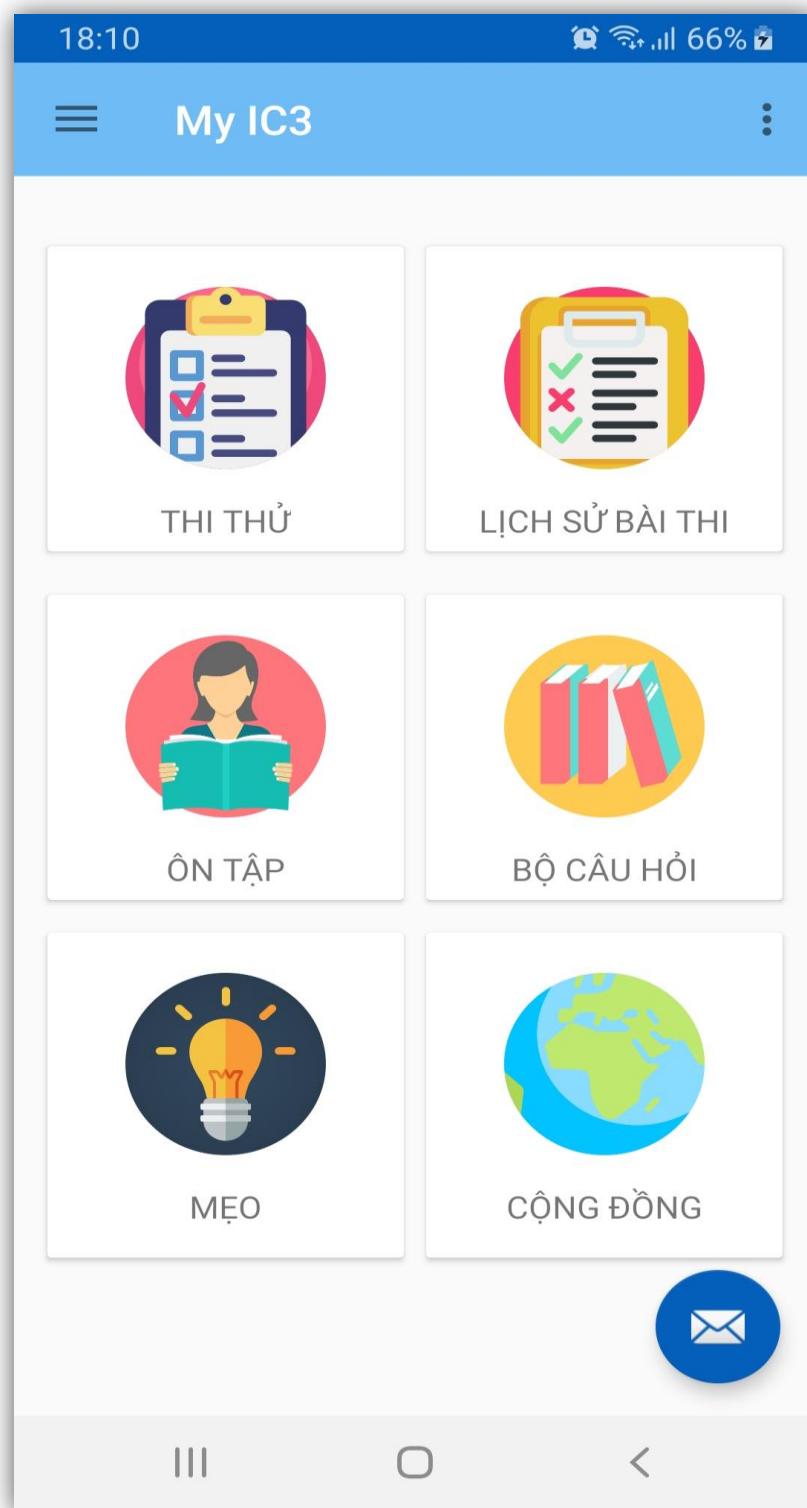
3.2.3 Thiết kế giao diện hệ thống

a. Màn hình khởi động



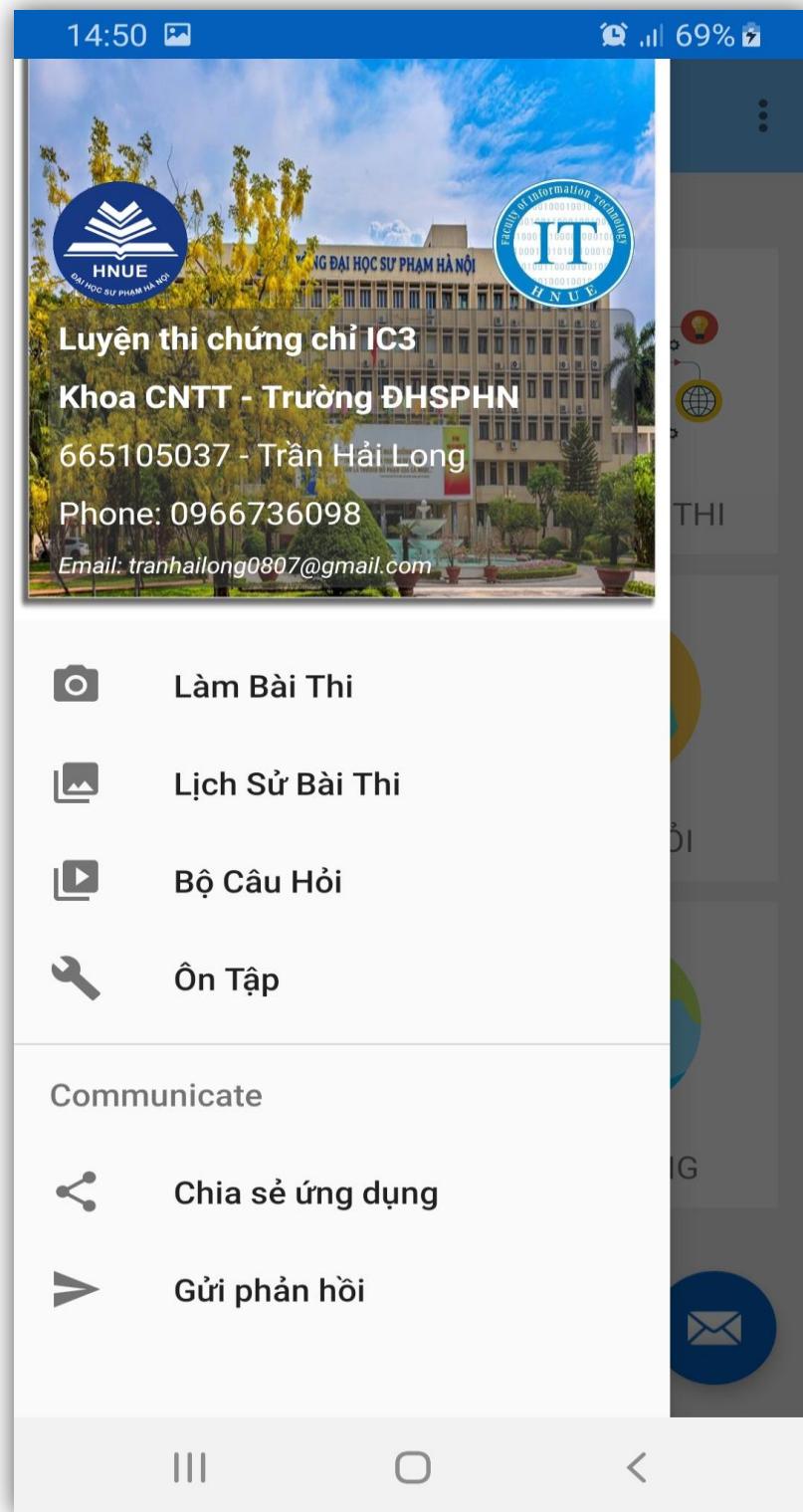
Hình 3.5: Giao diện màn hình khởi động

b. Giao diện Menu chính



Hình 3.6: Giao diện Menu chính

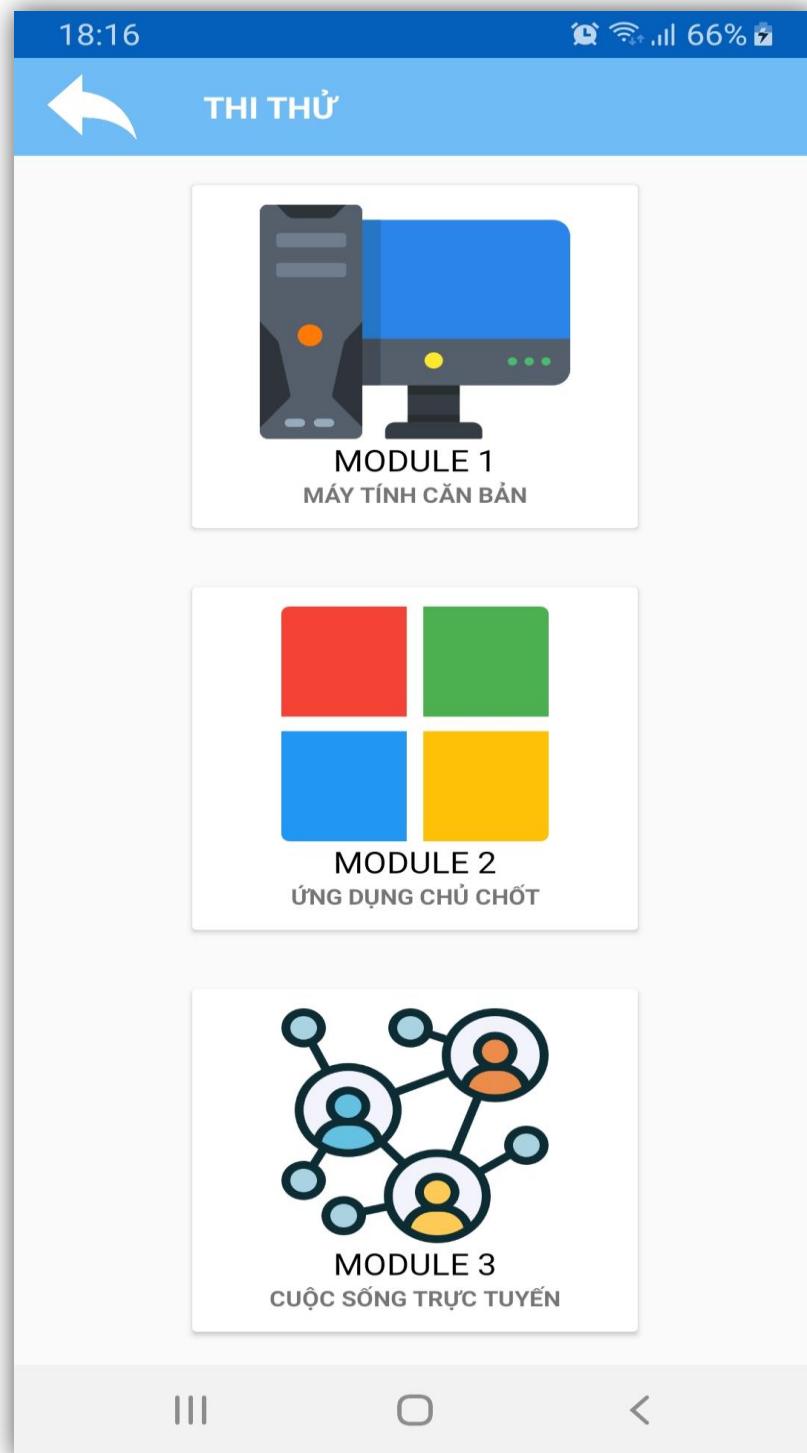
c. Giao diện Menu phụ



Hình 3.7: Giao diện Menu phụ.

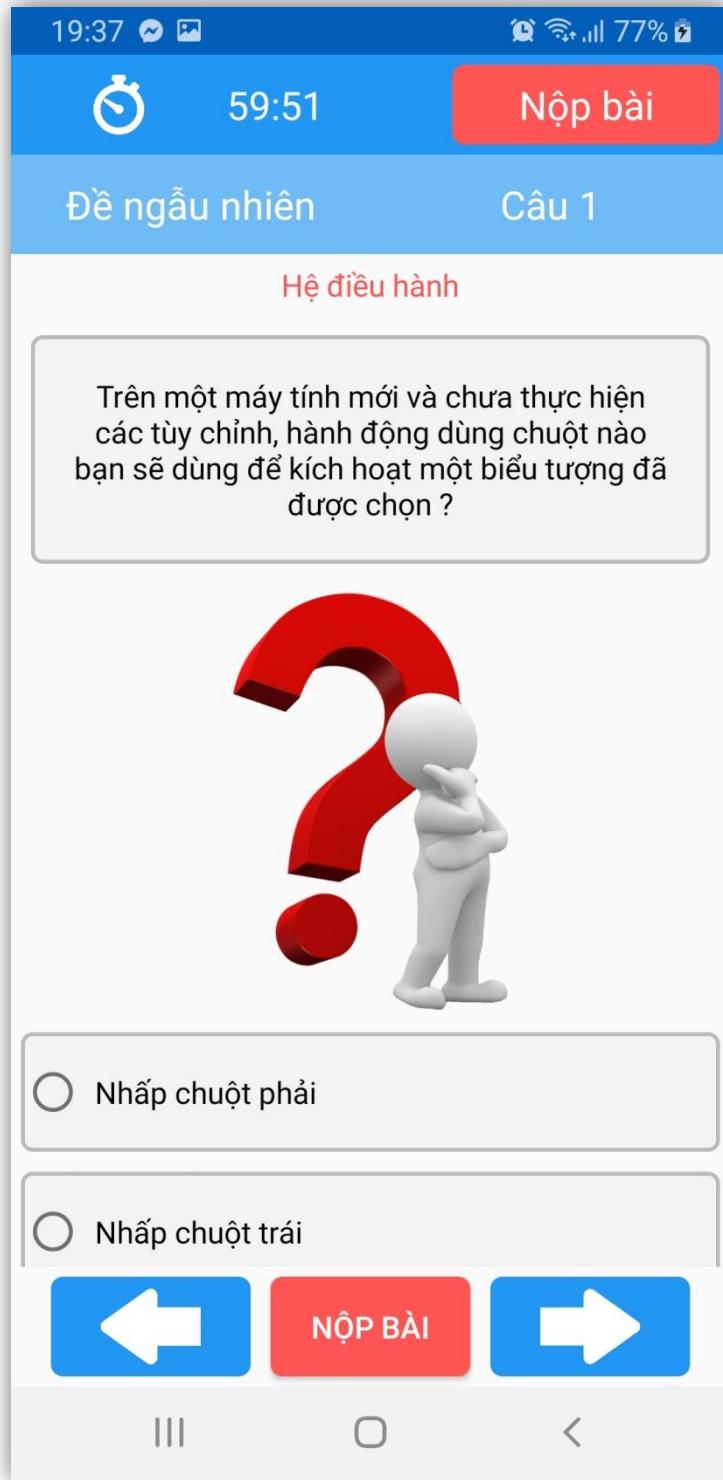
d. Giao diện chức năng thi thử

Khi người dùng chọn chức năng Thi Thử, Menu chính của chức năng sẽ hiện lên



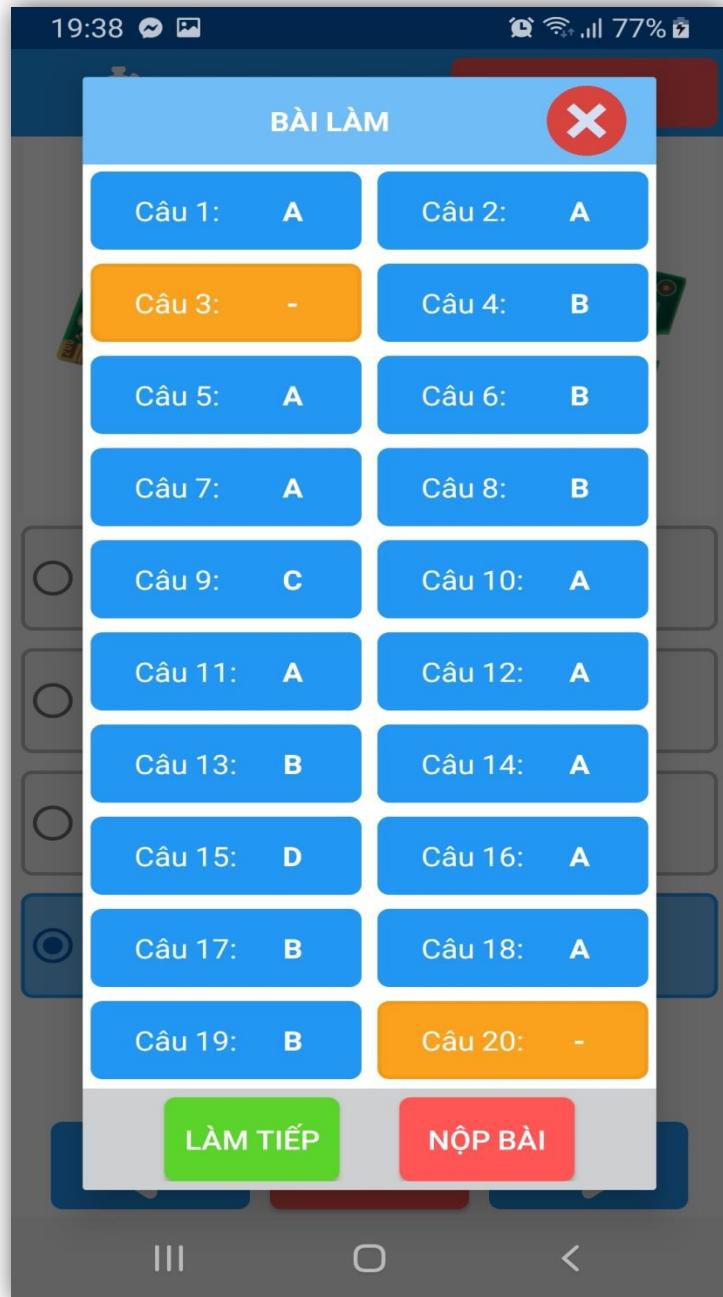
Hình 3.8: Giao diện chức năng thi thử.

Sau khi người dùng đã chọn Module, hệ thống sẽ sinh đề ngẫu nhiên từ ngân hàng câu hỏi và bấm giờ làm bài.



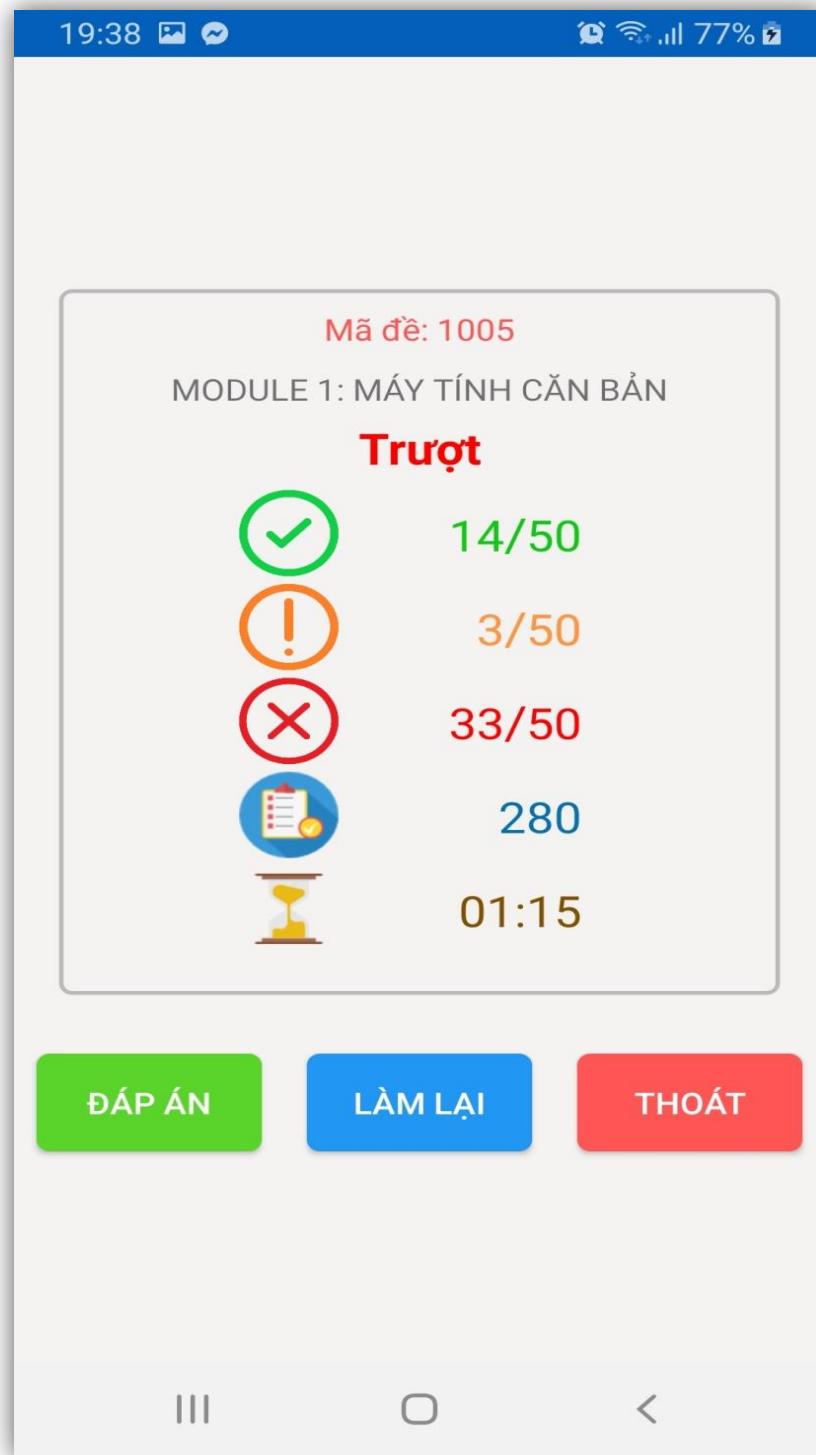
Hình 3.8.1: Giao diện làm bài thi.

Trước khi nộp bài, hệ thống sẽ hiển thị bài làm của người dùng lên để tiện cho việc kiểm tra lại.



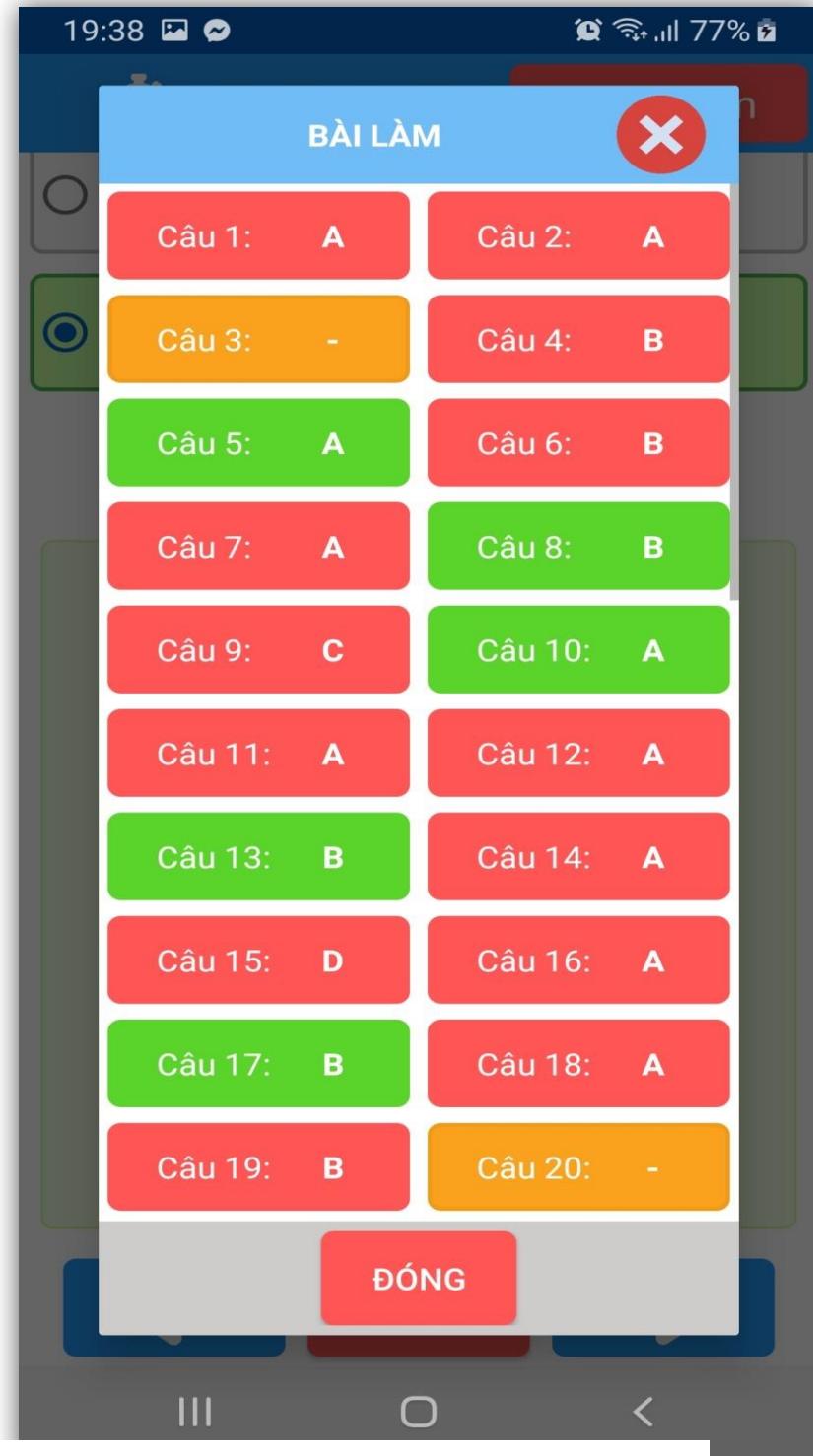
Hình 3.8.2: Giao diện kiểm tra bài làm.

Sau khi xác nhận nộp bài, hệ thống sẽ tính điểm và trả về kết quả cho người dùng.



Hình 3.8.3: Giao diện kết quả bài thi.

Sau khi biết kết quả thi, người dùng có thể xem lại bài làm của mình để biết mình sai ở đâu.



Hình 3.8.4: Giao diện chුa bài thi.

Hệ thống sẽ hiển thị giải thích đáp án cho người dùng.



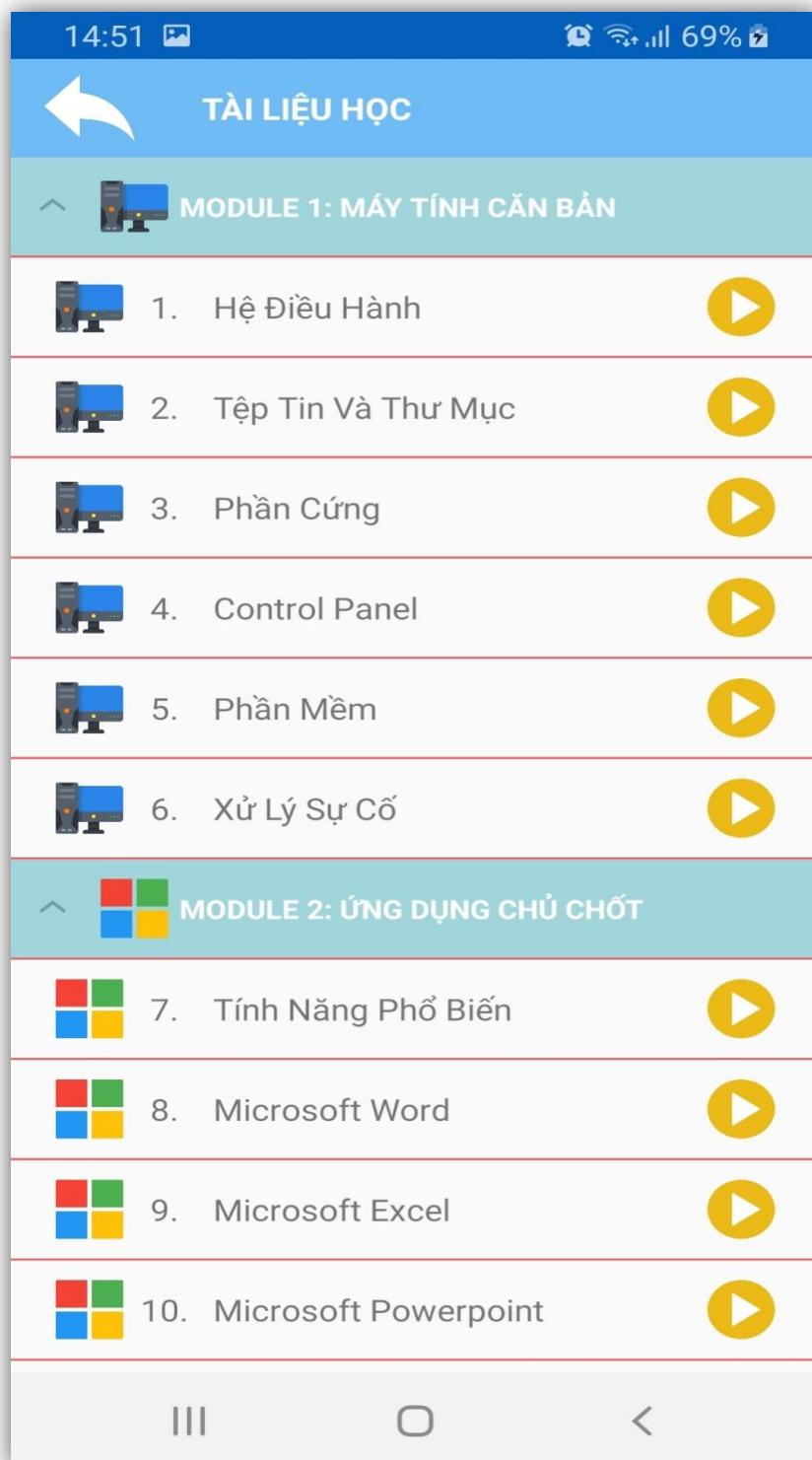
Hình 3.8.5: Giao diện chưa và giải thích đáp án.

e. Giao diện chức năng xem lịch sử bài thi

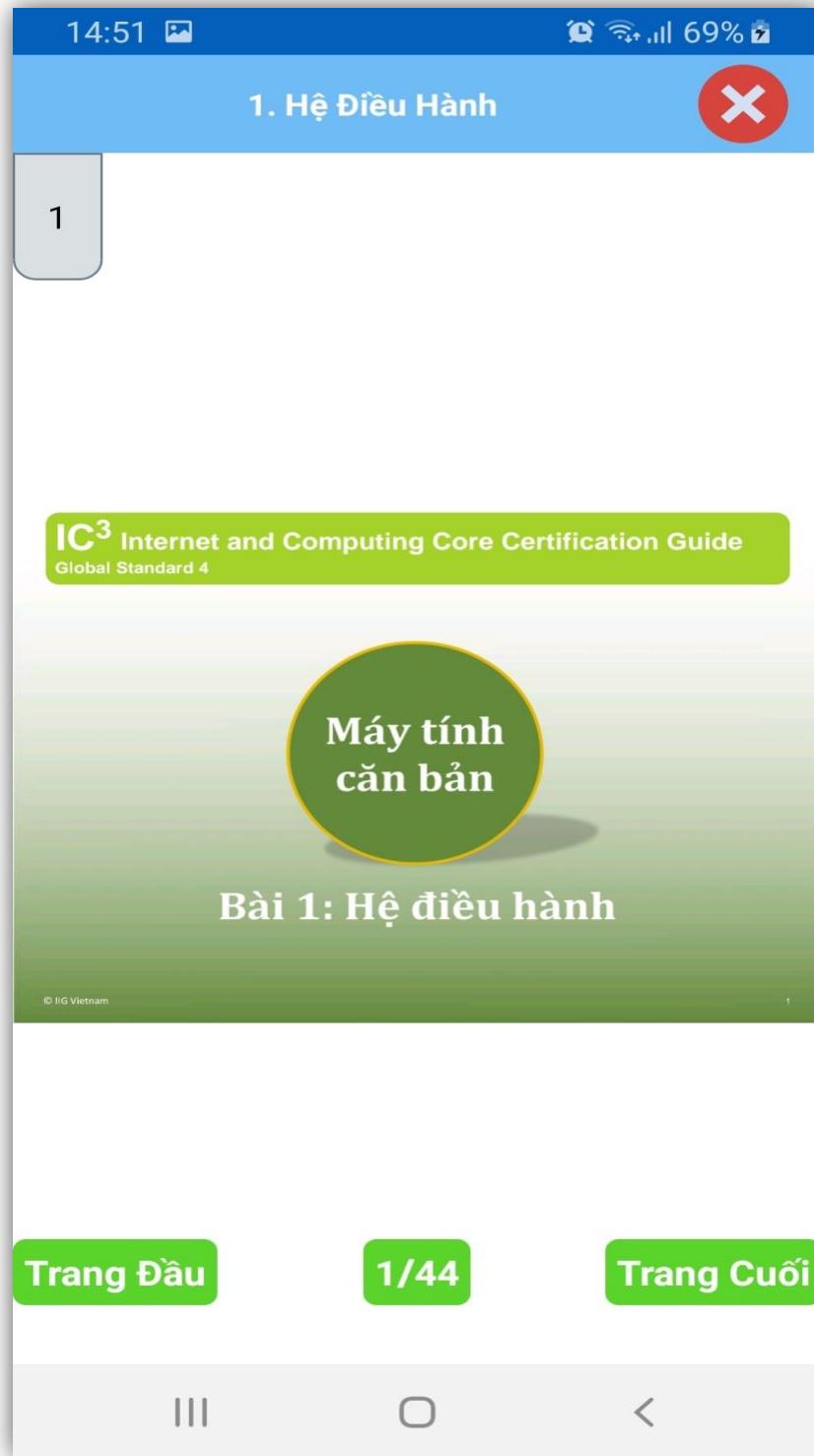


Hình 3.9: Giao diện chức năng xem lịch sử bài thi.

f. Giao diện chức năng ôn tập lý thuyết

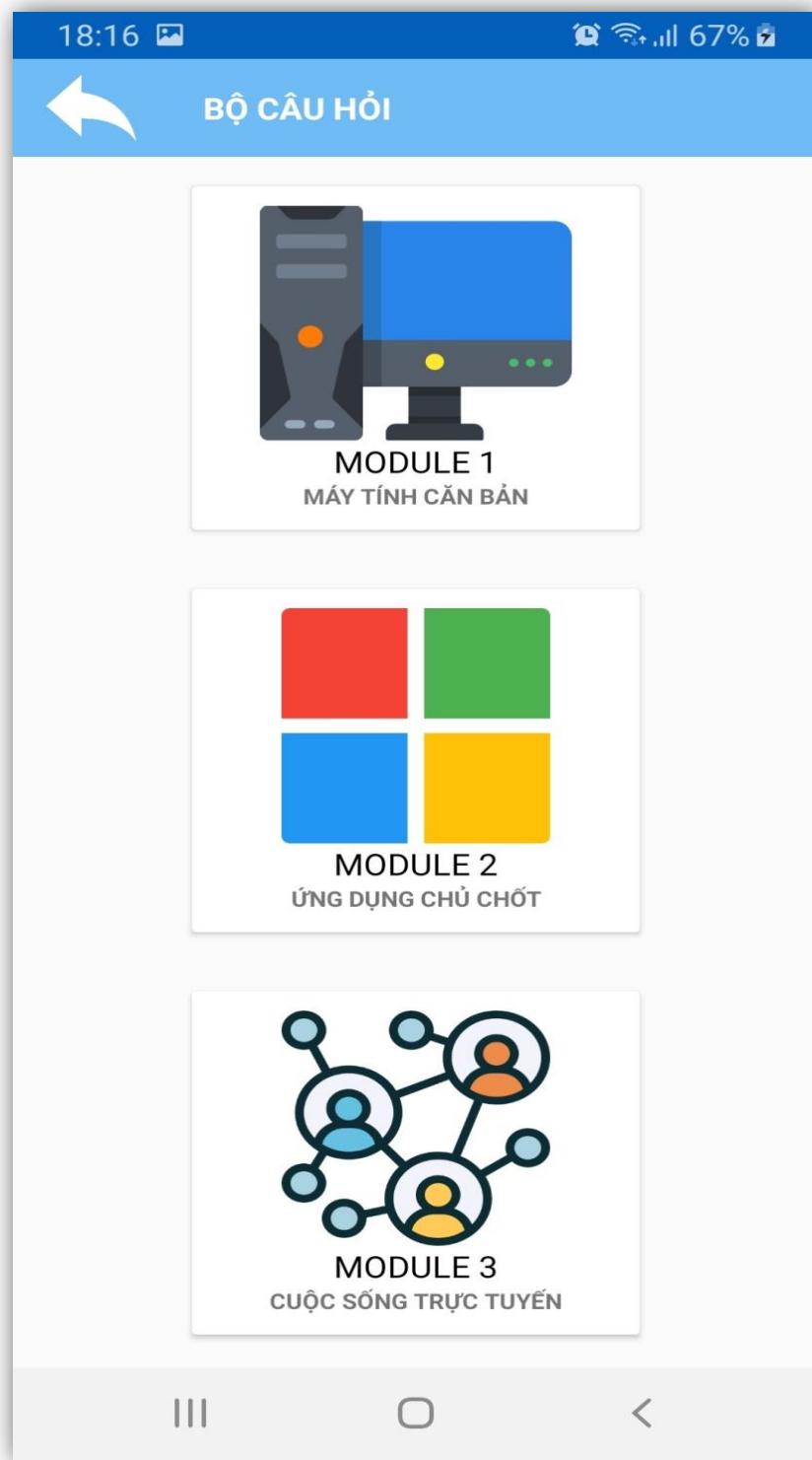


Hình 3.10: Giao diện chức năng ôn tập lý thuyết.

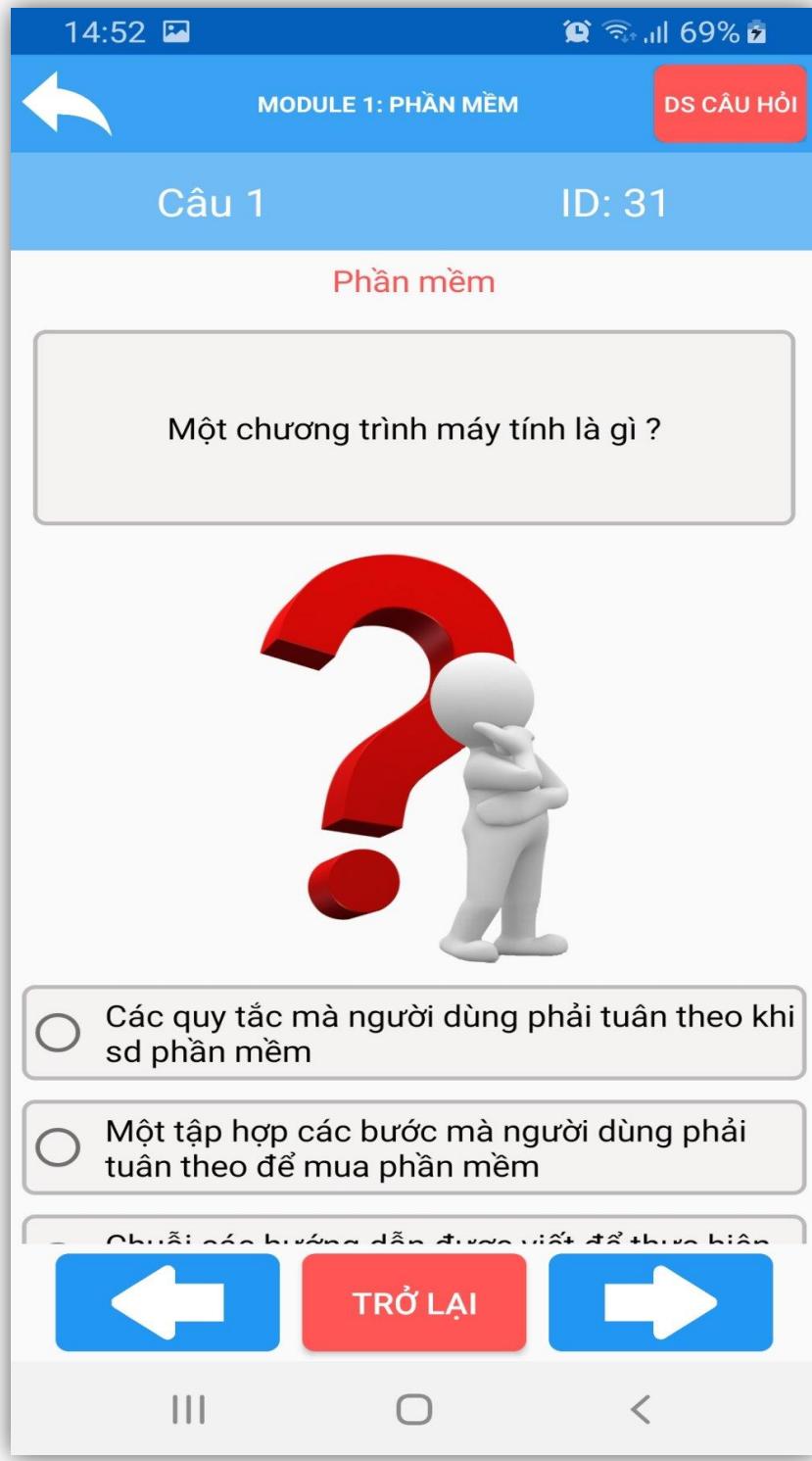


Hình 3.11: Giao diện bài giảng ôn tập lý thuyết.

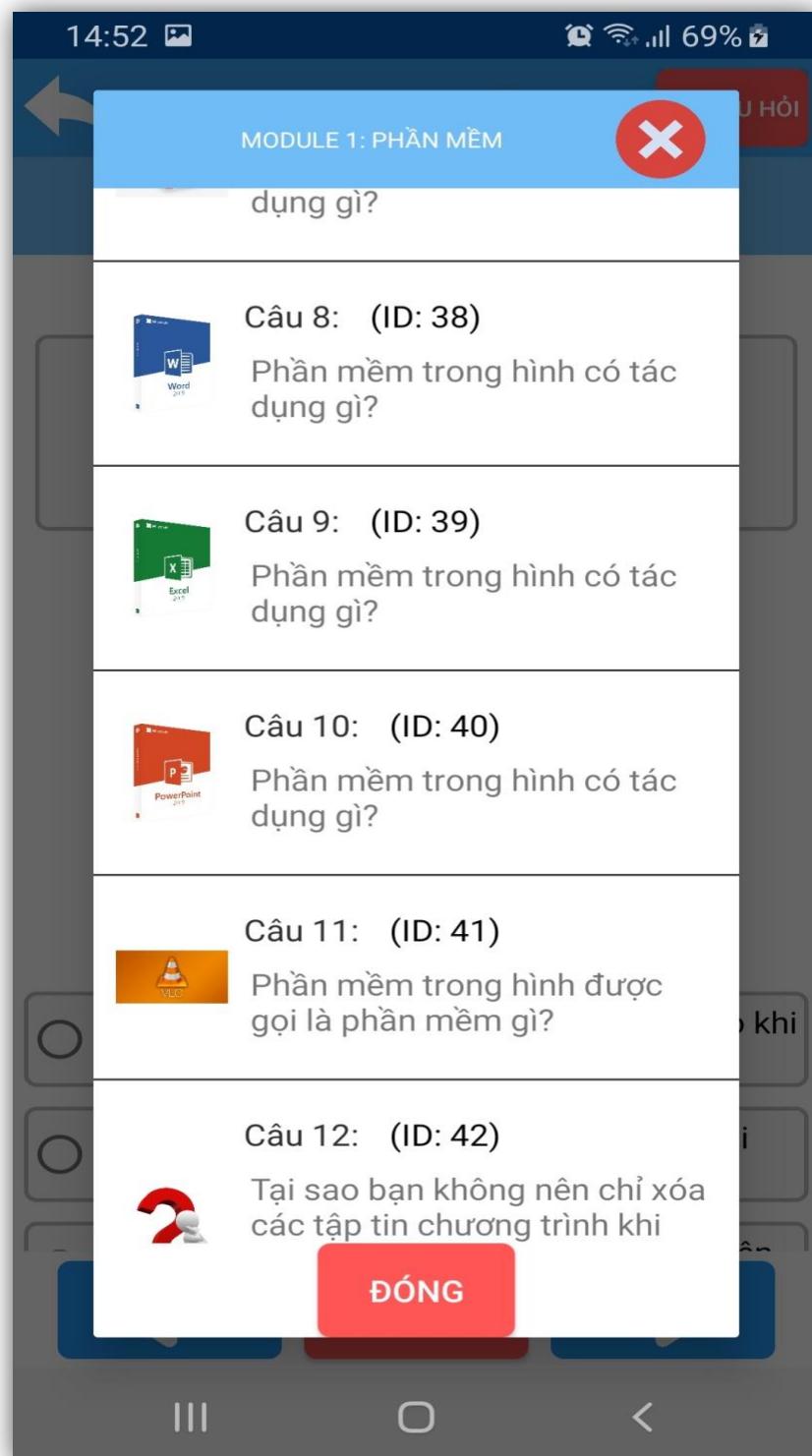
g. Giao diện chức năng ôn tập với ngân hàng câu hỏi



Hình 3.12: Giao diện Menu ngân hàng câu hỏi.

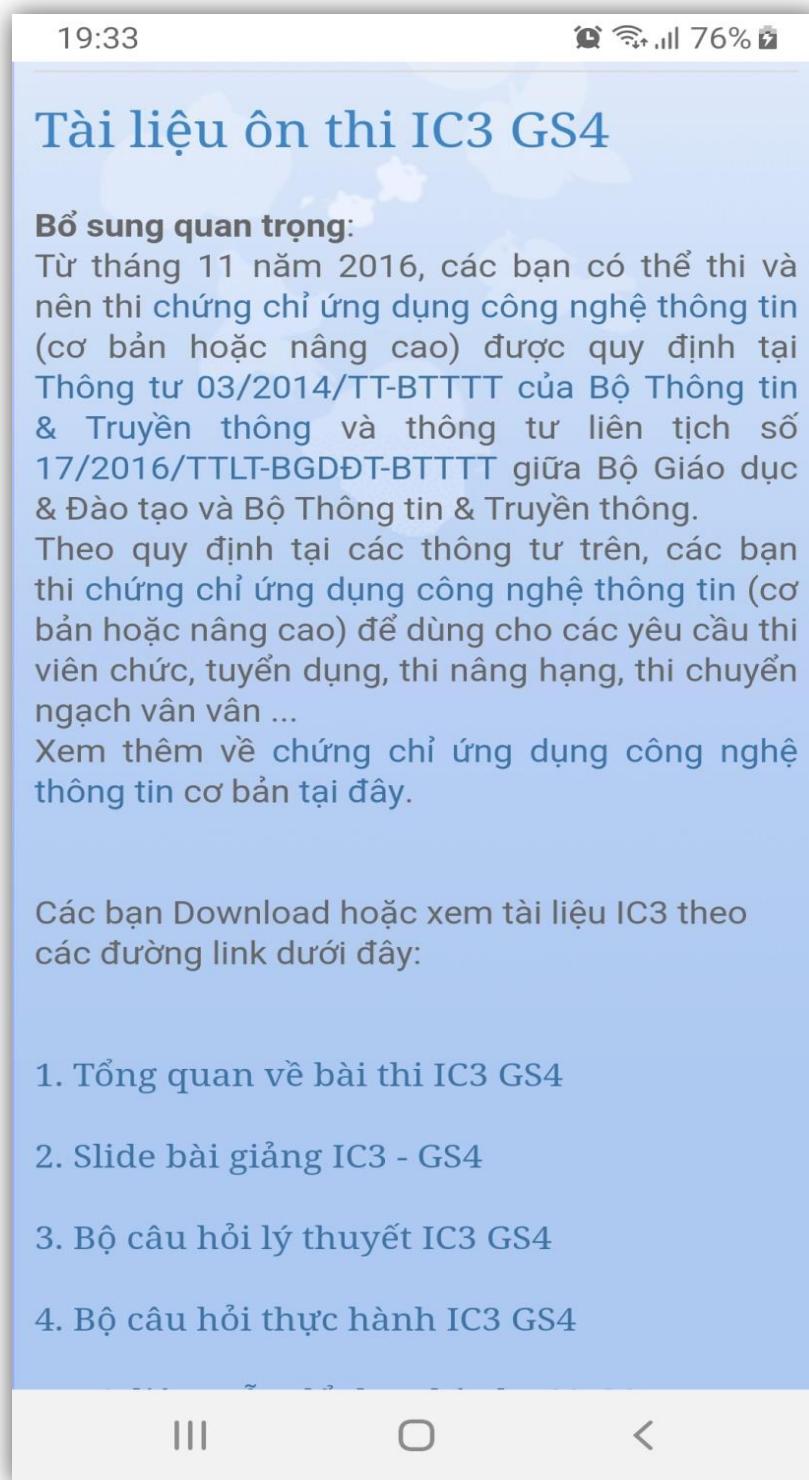


Hình 3.13: Giao diện nội dung câu hỏi trong ngân hàng câu hỏi.



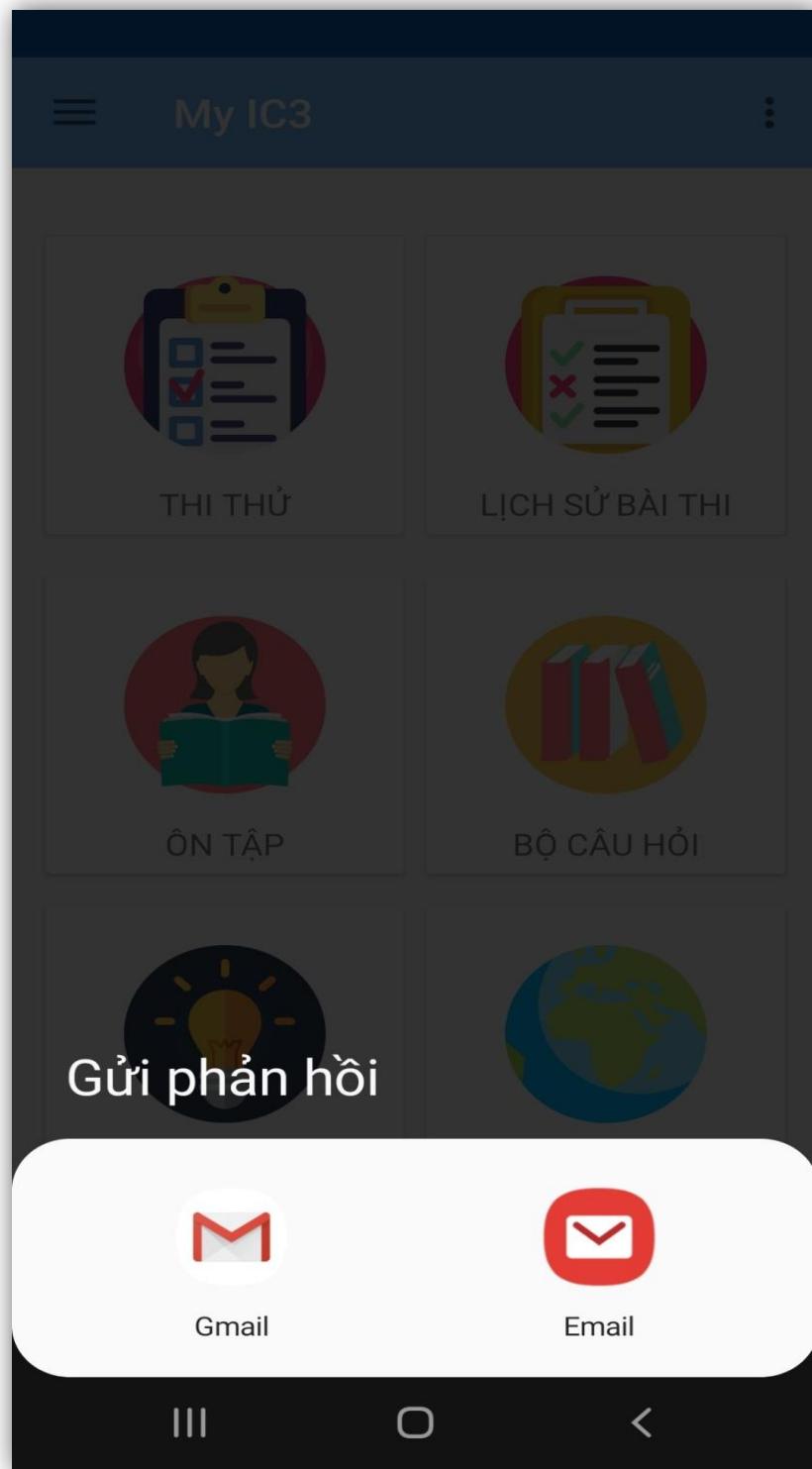
Hình 3.14: Giao diện danh sách câu hỏi trong ngân hàng câu hỏi.

h. Giao diện chức năng liên kết cộng đồng



Hình 3.15: Giao diện liên kết cộng đồng.

i. Giao diện chức năng gửi phản hồi tới nhà phát triển



Hình 3.16: Giao diện gửi phản hồi cho nhà phát triển.

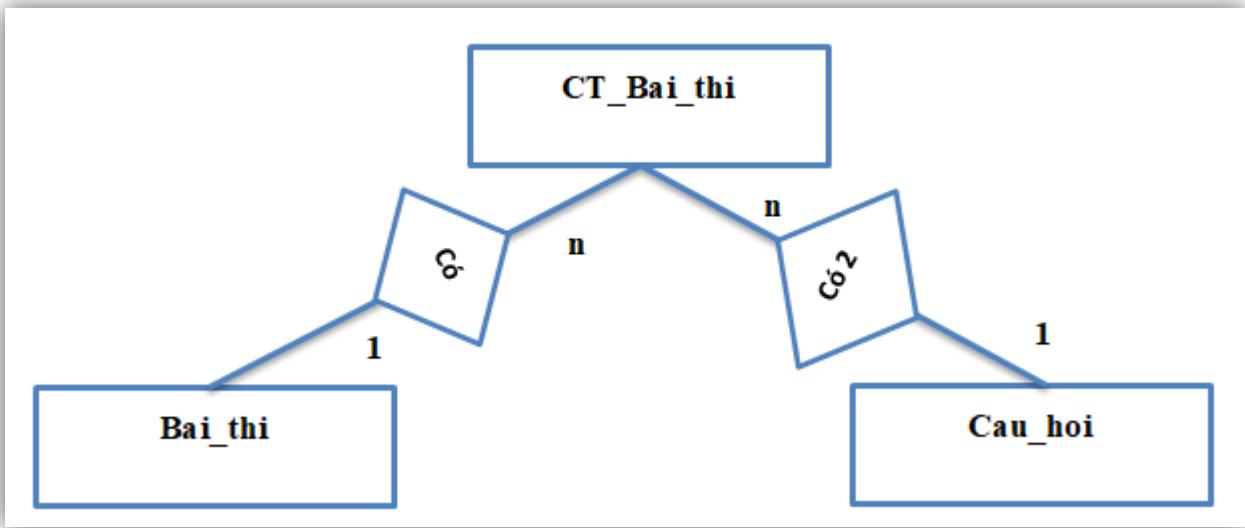
3.2.4 Thiết kế cơ sở dữ liệu

a. Phân tích

Dựa trên các chức năng của ứng dụng, cơ sở dữ liệu cần phải lưu trữ các thông tin của ngân hàng câu hỏi, kết quả bài thi và chi tiết bài thi. Tương ứng với đó là những thực thể được nêu ra dưới đây. Mỗi thực thể sẽ có các thuộc tính, các đặc trưng mô tả thực thể đó.

Các thuộc tính của các thực thể cụ thể như sau:

- **Câu Hỏi (Cau_hoi):** Lưu trữ các thuộc tính về thông tin một câu hỏi.
 - Mã câu hỏi
 - Nội dung câu hỏi
 - Nội dung đáp án A
 - Nội dung đáp án B
 - Nội dung đáp án C
 - Nội dung đáp án D
 - Đáp án đúng
 - Giải thích đáp án
 - Tên file ảnh minh họa
 - Mã Module
 - Mã chủ đề
 - Độ khó
- **Bài Thi (Bai_thi):** Lưu trữ thông tin của bài thi.
 - Mã bài thi
 - Mã Module
 - Ngày thi
 - Số câu trả lời đúng
 - Số câu trả lời sai
 - Số câu chưa trả lời
 - Điểm số
 - Thời gian làm bài
 - Kết quả
- **Chi Tiết Bài Thi (CT_Bai_thi):** Lưu trữ thông tin chi tiết bài làm của người dùng.
 - Mã bài thi
 - Mã câu hỏi
 - Nội dung câu hỏi
 - Nội dung đáp án A
 - Nội dung đáp án B
 - Nội dung đáp án C
 - Nội dung đáp án D
 - Đáp án đúng
 - Lựa chọn của người dùng
 - Giải thích đáp án
 - Tên file ảnh minh họa
 - Mã Module
 - Mã chủ đề
 - Độ khó



Hình 3.17: Sơ đồ thực thể liên kết của hệ thống.

Hình vẽ trên biểu thị mối liên kết giữa thực thể Bài Thi, Chi Tiết Bài Thi và thực thể Câu Hỏi. Mỗi liên kết giữa Bài Thi và Chi Tiết Bài Thi là liên kết 1:N, tức là một bài thi có nhiều chi tiết bài thi, nhưng một chi tiết bài thi chỉ thuộc một bài thi cụ thể. Sự tham gia của thực thể Chi Tiết Bài Thi là toàn bộ, chi tiết bài thi nào cũng do một bài thi nào đó được hoàn thành và lưu lại. Ngược lại, sự tham gia của thực thể Bài Thi là bộ phận, không phải bài thi nào cũng được lưu lại trong hệ thống.

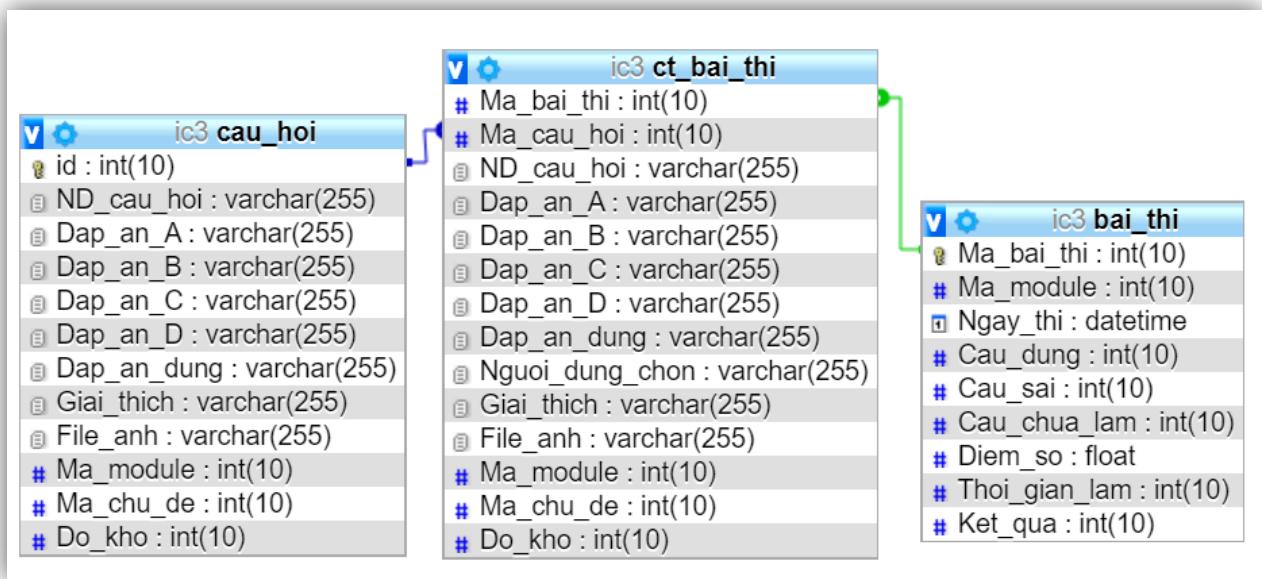
Mỗi liên kết giữa thực thể Câu Hỏi và Chi Tiết Bài Thi là liên kết 1:N, tức là một câu hỏi có thể xuất hiện trong nhiều bài thi, nhưng một chi tiết bài thi chỉ ứng với một câu hỏi cụ thể. Sự tham gia của thực thể Chi Tiết Bài Thi là toàn bộ, chi tiết bài thi nào cũng ứng với một câu hỏi nào đó. Ngược lại, sự tham gia của thực thể Câu Hỏi là bộ phận, không phải câu hỏi nào cũng có trong đề thi để có chi tiết về nó.

b. Thiết kế

Từ những phân tích về các thực thể và các liên kết thực thể nêu trên, có thể mô tả qua các lược đồ quan hệ như sau:

- Lược đồ quan hệ Câu Hỏi:** Cau_hoi(id, ND_cau_hoi, Dap_an_A, Dap_an_B, Dap_an_C, Dap_an_D, Dap_an_dung, Giai_thich, File_anh, Ma_module, Ma_chu_de, Do_kho) trong đó id là khóa chính.
- Lược đồ quan hệ Bài Thi:** Bai_thi(Ma_bai_thi, Ma_module, Ngay_thi, imagelon, Cau_dung, Cau_sai, Cau_chua_lam, Diem_so, Thoi_gian, Ket_qua) trong đó Ma_bai_thi là khóa chính, Diem_so là điểm của bài thi và Ket_qua là đánh giá đề thi đạt hay không đạt.

- Lược đồ quan hệ CT_Bai_thi:** CT_Bai_thi(Ma_bai_thi, Ma_cau_hoi, ND_cau_hoi, Dap_an_A, Dap_an_B, Dap_an_C, Dap_an_D, Dap_an_dung, Nguoi_dung_chon, Giai_thich, File_anh, Ma_module, Ma_chu_de, Do_kho) trong đó Ma_bai_thi và Ma_cau_hoi là khóa chính, Nguoi_dung_chon sẽ chứa thông tin đáp án của người dùng.



Hình 3.18: Mô hình các bảng trong cơ sở dữ liệu.

Tên thuộc tính	Mô tả	Kiểu giá trị	Độ dài
Id	Mã câu hỏi, NOT NULL, khóa chính, tự tăng	INT	10
ND_cau_hoi	Nội dung câu hỏi	VARCHAR	255
Dap_an_A	Đáp án A	VARCHAR	255
Dap_an_B	Đáp án B	VARCHAR	255
Dap_an_C	Đáp án C	VARCHAR	255
Dap_an_D	Đáp án D	VARCHAR	255
Dap_an_dung	Đáp án đúng	VARCHAR	255
Giai_thich	Giải thích đáp án	VARCHAR	255
File_anh	Tên file ảnh minh họa	VARCHAR	255
Ma_module	Mã module câu hỏi	INT	10
Ma_chu_de	Mã chủ đề câu hỏi	INT	10
Do_kho	Độ khó câu hỏi	INT	10

Bảng 3.3: Bảng Cau_hoi
49

Tên thuộc tính	Mô tả	Kiểu giá trị	Độ dài
Ma_bai_thi	Mã bài thi, NOT NULL, khóa chính, tự tăng	INT	10
Ma_module	Mã module bài thi	INT	10
Ngay_thi	Ngày làm bài thi	DATETIME	255
Cau_dung	Số câu trả lời đúng	INT	10
Cau_sai	Số câu chưa trả lời	INT	10
Cau_chua_lam	Số câu chưa làm	INT	10
Diem_so	Điểm số của đề thi	FLOAT	10
Thoi_gian	Thời gian làm bài	DATETIME	255
Ket_qua	Kết quả bài thi	INT	10

Bảng 3.4: Bảng Bai_thi.

Tên thuộc tính	Mô tả	Kiểu giá trị	Độ dài
Ma_bai_thi	Mã bài thi, NOT NULL, khóa chính	INT	10
Ma_cau_hoi	Mã câu hỏi, NOT NULL, khóa chính	INT	10
ND_cau_hoi	Nội dung câu hỏi	VARCHAR	255
Dap_an_A	Đáp án A	VARCHAR	255
Dap_an_B	Đáp án B	VARCHAR	255
Dap_an_C	Đáp án C	VARCHAR	255
Dap_an_D	Đáp án D	VARCHAR	255
Dap_an_dung	Đáp án đúng	VARCHAR	255
Nguoi_dung_chon	Đáp án người dùng chọn	VARCHAR	255
Giai_thich	Giải thích đáp án	VARCHAR	255
File_anh	Tên file ảnh minh họa	VARCHAR	255
Ma_module	Mã module câu hỏi	INT	10
Ma_chu_de	Mã chủ đề câu hỏi	INT	10
Do_kho	Độ khó câu hỏi	INT	10

Bảng 3.5: Bảng CT_Bai_thi.

CHƯƠNG IV: XÂY DỰNG HỆ THỐNG VÀ KẾT QUẢ ĐẠT ĐƯỢC

4.1 Quy trình xây dựng

a. Xây dựng hàm lấy ngẫu nhiên các câu hỏi từ bảng Câu Hỏi trong CSDL

Hàm có 3 tham số: Chủ đề của câu hỏi (Subject), Độ khó của câu hỏi (Level) và Số lượng tối đa lấy mỗi lần truy vấn (Limit). Kiểu trả về là một ArrayList<Question>

```
public ArrayList<Question> getQuestions(String subject, int limit, int level)
{
    ArrayList<Question> listQuestion = new ArrayList<>();
    SQLiteDatabase db = dbHelper.getReadableDatabase();
    Cursor cursor = db.rawQuery("SELECT * FROM cauhoi WHERE subject = "
        +
        subject + " and level = " + level + " ORDER BY RANDOM() LIMIT " +
        limit, null);
    cursor.moveToFirst();
    do{
        Question question;
        question = new Question(cursor.getInt(0),
            cursor.getString(1),
            cursor.getString(2),
            cursor.getString(3),
            cursor.getString(4),
            cursor.getString(5),
            cursor.getString(6),
            cursor.getString(7),
            cursor.getInt(8),
            cursor.getString(9),
            cursor.getInt(10),
            cursor.getString(11),
            "");
        listQuestion.add(question);
    }while (cursor.moveToNext());
    return listQuestion;
}
```

Hình 4.1: Hàm getQuestions lấy thông tin về câu hỏi trong CSDL

```

if(maModule.equals("1")){
    questions.addAll(questionController.getQuestions("pc", 10, 1));
    questions.addAll(questionController.getQuestions("pc", 5, 2));
    questions.addAll(questionController.getQuestions("pc", 5, 3));
    questions.addAll(questionController.getQuestions("pm", 5, 1));
    questions.addAll(questionController.getQuestions("pm", 5, 2));
    questions.addAll(questionController.getQuestions("pm", 5, 3));
    questions.addAll(questionController.getQuestions("hdh", 5, 1));
    questions.addAll(questionController.getQuestions("hdh", 5, 2));
    questions.addAll(questionController.getQuestions("hdh", 5, 3));
}
else if(maModule.equals("2")){
    questions.addAll(questionController.getQuestions("word", 10, 1));
    questions.addAll(questionController.getQuestions("word", 5, 2));
    questions.addAll(questionController.getQuestions("word", 5, 3));
    questions.addAll(questionController.getQuestions("excel", 5, 1));
    questions.addAll(questionController.getQuestions("excel", 5, 2));
    questions.addAll(questionController.getQuestions("excel", 5, 3));
    questions.addAll(questionController.getQuestions("ppt", 5, 1));
    questions.addAll(questionController.getQuestions("ppt", 5, 2));
    questions.addAll(questionController.getQuestions("ppt", 5, 3));
}
else if(maModule.equals("3")){
    questions.addAll(questionController.getQuestions("mvi", 10, 1));
    questions.addAll(questionController.getQuestions("mvi", 5, 2));
    questions.addAll(questionController.getQuestions("mvi", 5, 3));
    questions.addAll(questionController.getQuestions("ttdt", 5, 1));
    questions.addAll(questionController.getQuestions("ttdt", 5, 2));
    questions.addAll(questionController.getQuestions("ttdt", 5, 3));
    questions.addAll(questionController.getQuestions("sdi", 5, 1));
    questions.addAll(questionController.getQuestions("sdi", 5, 2));
    questions.addAll(questionController.getQuestions("sdi", 5, 3));
}
Collections.shuffle(questions);

```

Hình 4.2: Lấy câu hỏi theo độ khó và Module để sinh đẻ ngẫu nhiên

c. Xây dựng hàm lấy giá trị của 4 đáp án A, B, C, D

```
//Lấy giá trị của radButton
public String getCheckedRadBtnValue(int id)
{
    if(id == R.id.radA)
    {
        return radA.getText().toString();
    }
    if(id == R.id.radB)
    {
        return radB.getText().toString();
    }
    if(id == R.id.radC)
    {
        return radC.getText().toString();
    }
    if(id == R.id.radD)
    {
        return radD.getText().toString();
    }
    return "";
}
```

Hình 4.3: Hàm lấy giá trị của đáp án được người dùng chọn.

d. Xây dựng hàm trả về ký tự A, B, C, D từ đáp án người dùng chọn.

Hàm getAnswerKey với tham số là giá trị của đáp án người dùng chọn. Sau đó sẽ đếm đáp án được chọn ấy đi so sánh với các đáp án của câu hỏi trong CSDL để đưa ra được đáp án người dùng chọn dưới dạng A, B, C, D.

```

public String getAnswerKey(String result)
{
    if(result.equals(question.getAns_a()))
    {
        return "A";
    }
    if(result.equals(question.getAns_b()))
    {
        return "B";
    }
    if(result.equals(question.getAns_c()))
    {
        return "C";
    }
    if(result.equals(question.getAns_d()))
    {
        return "D";
    }
    return "";
}

```

Hình 4.4: Hàm trả về ký tự A, B, C, D từ đáp án người dùng chọn.

e. Hàm lưu kết quả chọn của người dùng.

```

//Lưu kết quả chọn của người dùng
radioGroup.setOnCheckedChangeListener(new
RadioGroup.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(RadioGroup radioGroup, int i) {
        dachondapan = 1;
        arr_question.get(mPageNumber).setChoiceID(i);
        arr_question.get(mPageNumber).setAnswer(getAnswerKey(getChoice(i)));
    }
})

```

Hình 4.5: Hàm lưu kết quả chọn của người dùng.

f. Hàm hiển thị đáp án và giải thích cho mỗi câu hỏi.

```
String result = question.getResult();
String answer = question.getAnswer();

disableRadioButton();
resetRads();
markCorrectAnswer(getKeyfromResult(getResult(question.getResult())));
if(!result.equals(answer))
{
    markWrongAnswer(getKeyfromResult(getChoice(question.getChoiceID())));
    tvThongbao.setText("Tiếc quá, bạn chọn sai rồi :(");
    tvThongbao.setTextColor(getResources().getColor(R.color.colorThongbaoSai));
    if(answer.equals(""))
    {
        tvThongbao.setText("Bạn chưa làm câu này :(");
        tvThongbao.setTextColor(getResources().getColor(R.color.colorChualam));
    }
}
else
{
    tvThongbao.setText("Chính xác!");
    tvThongbao.setTextColor(getResources().getColor(R.color.colorThongbaoCX));
}
```

Hình 4.6: Hàm hiển thị đáp án và giải thích cho mỗi câu hỏi.

4.2 Thủ nghiệm

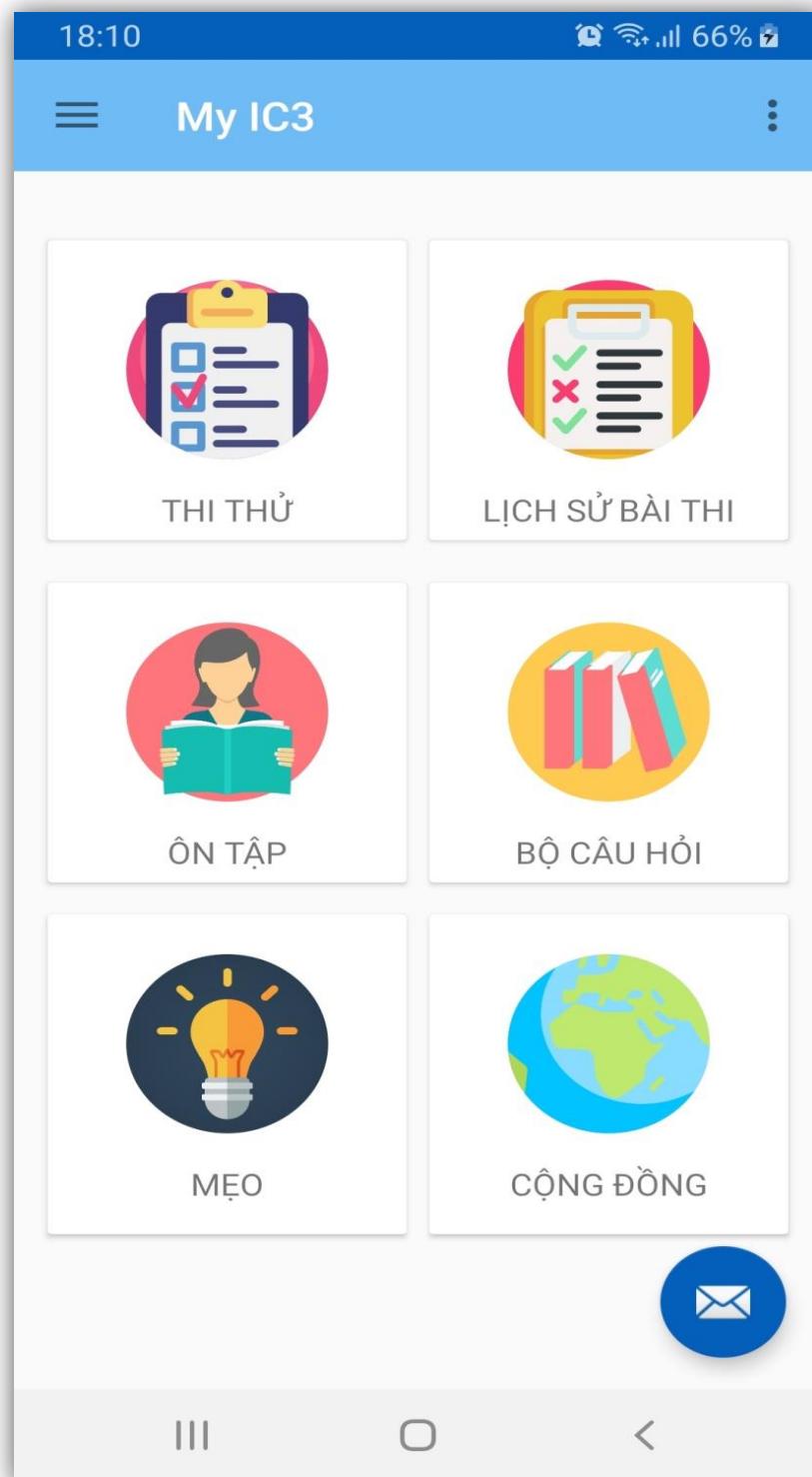
❖ Kịch bản thử nghiệm

1. Người dùng khởi động ứng dụng



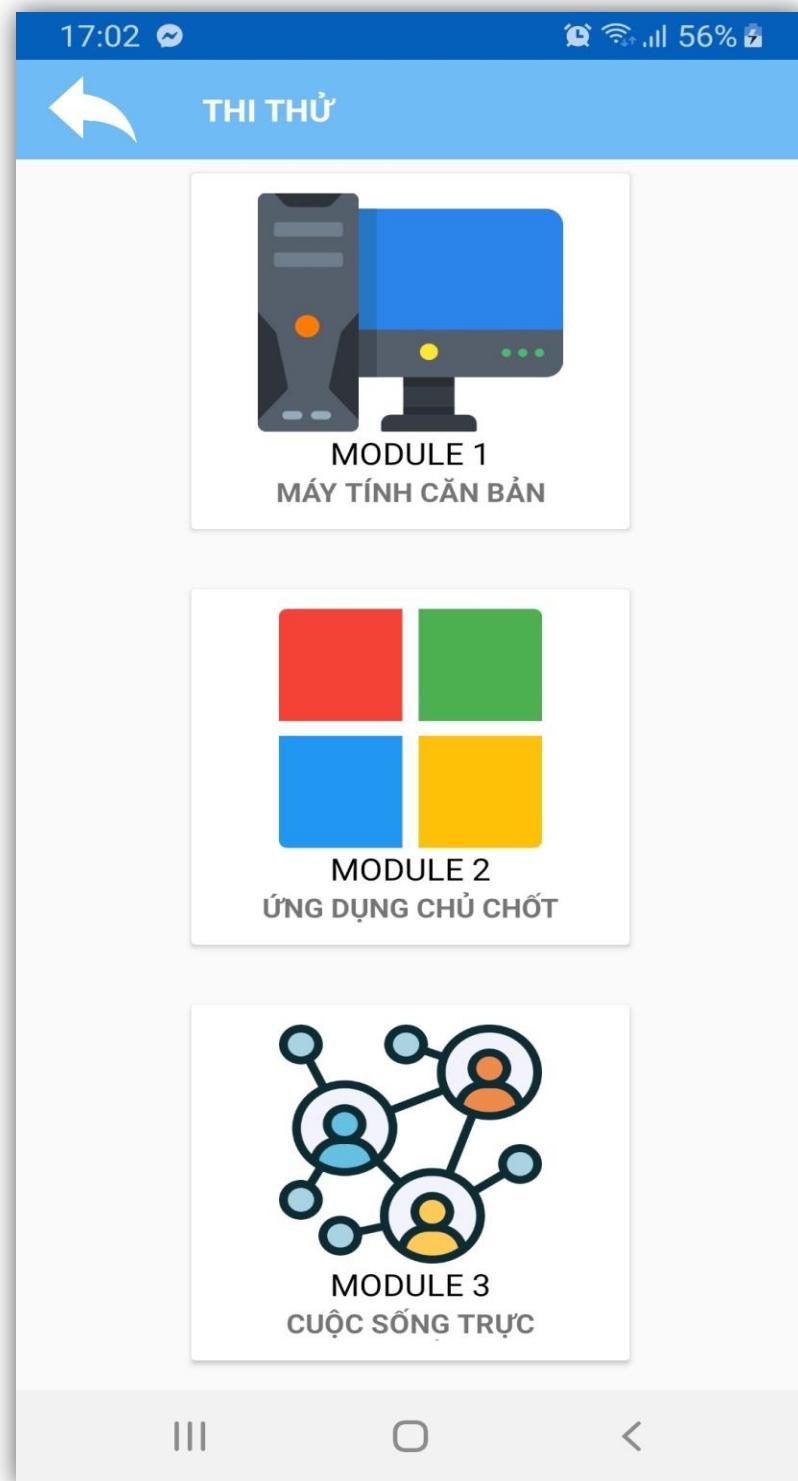
Hình 4.7: Màn hình khởi động ứng dụng.

2. Người dùng chọn chức năng Thi Thử



Hình 4.8: Màn hình Menu chính.

3. Người dùng chọn Module 2: Ứng dụng chủ chốt

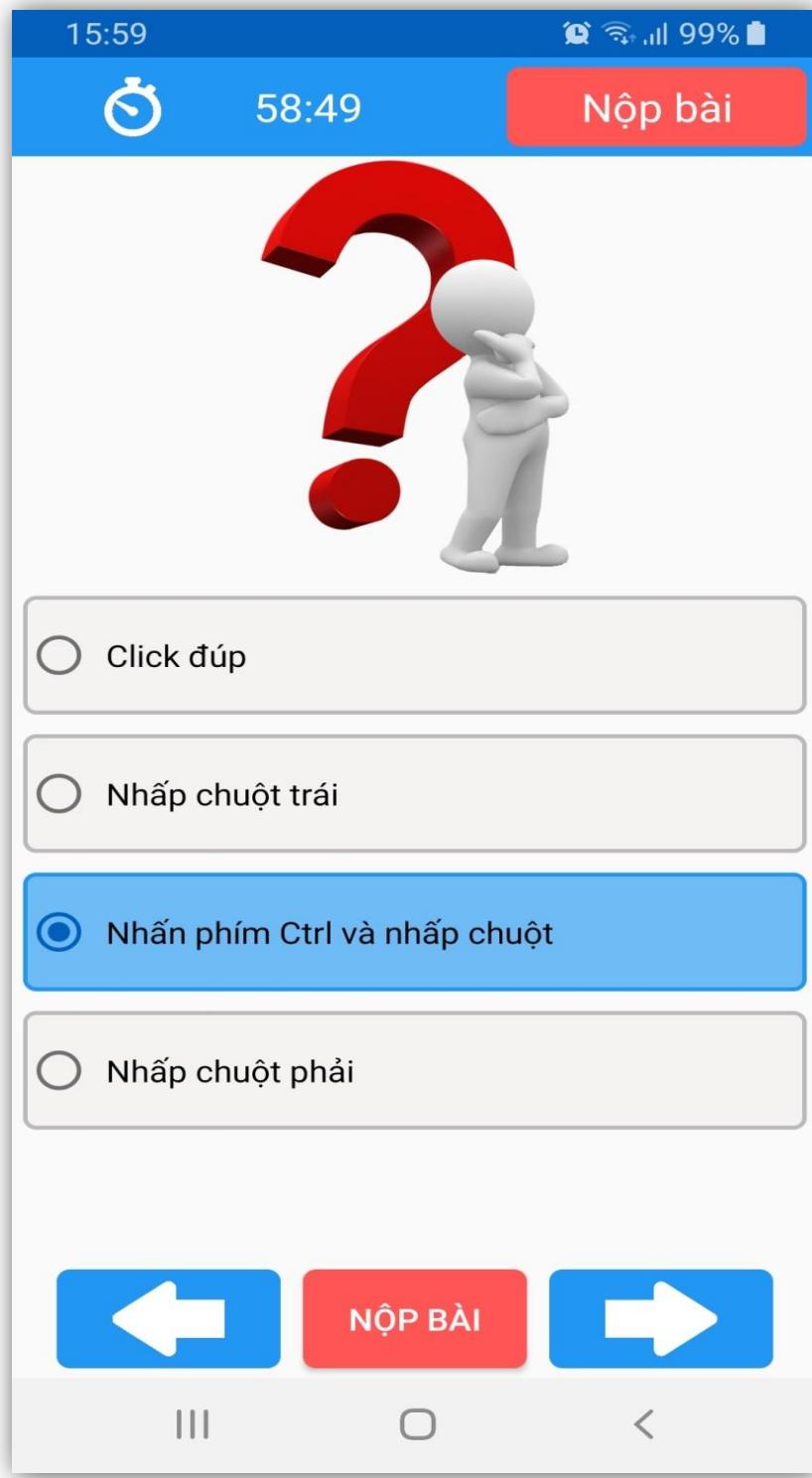


Hình 4.9: Menu Module để thi

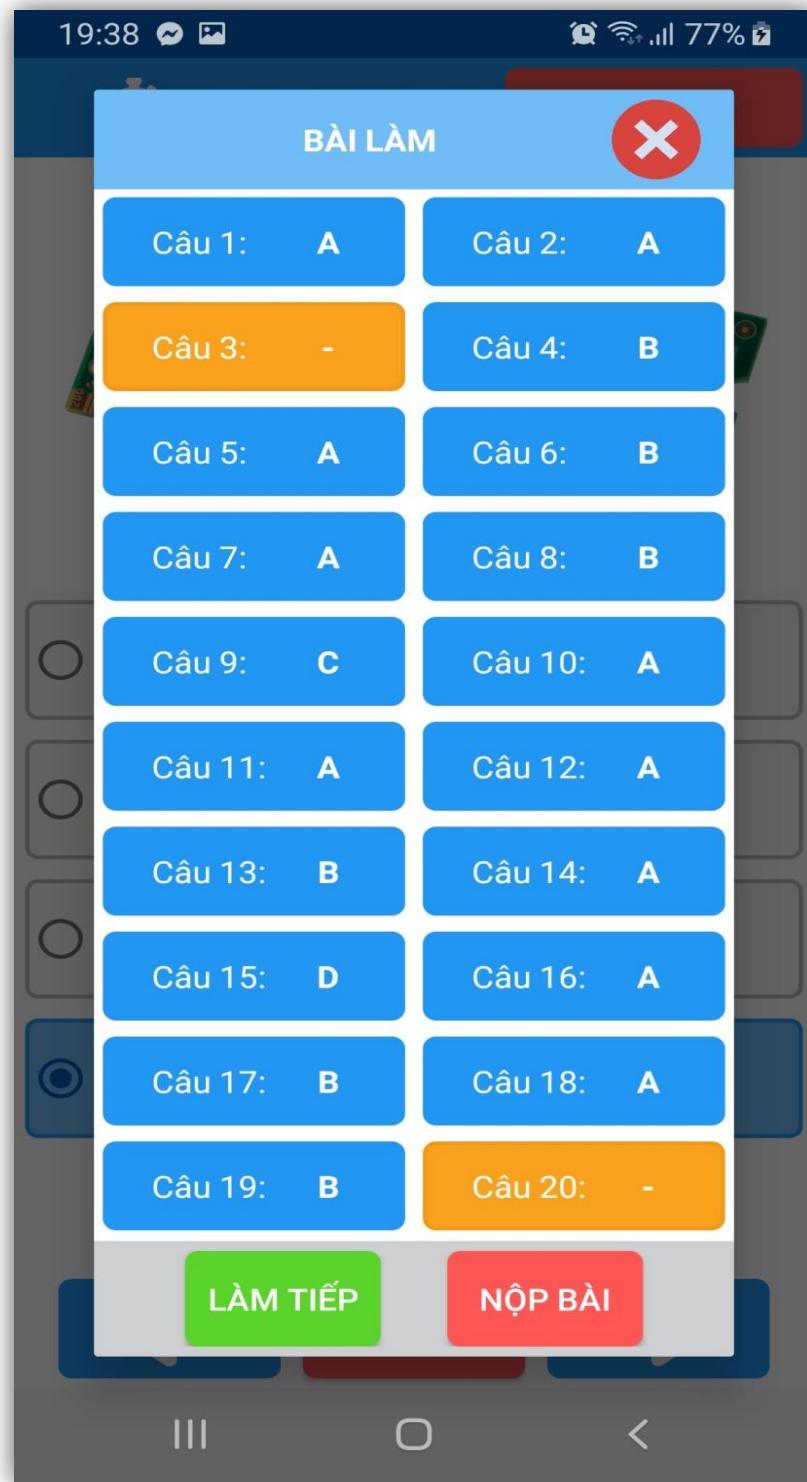
4. Người dùng làm bài thi và nộp bài



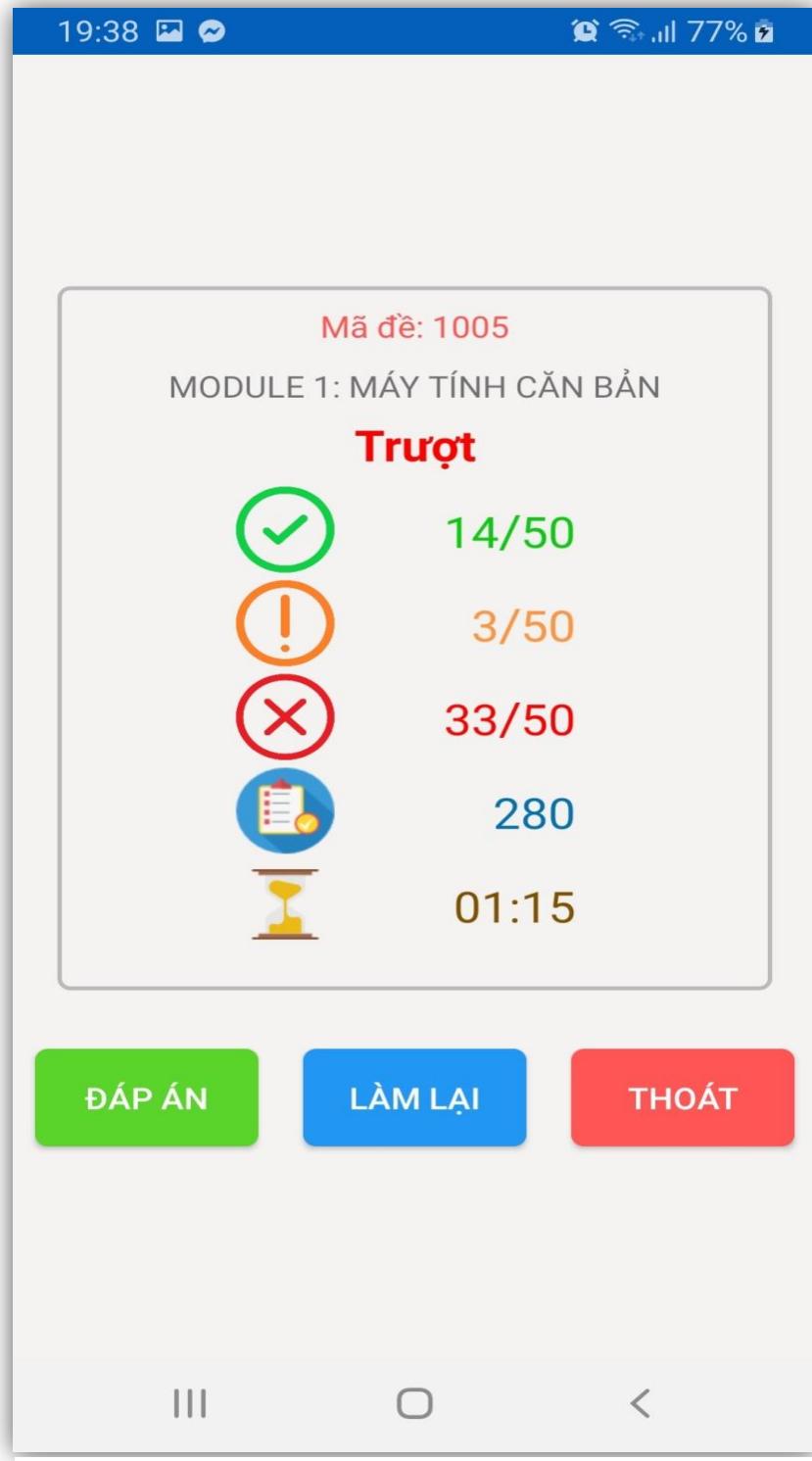
Hình 4.10: Màn hình làm bài thi.



Hình 4.11: Người dùng chọn đáp án.

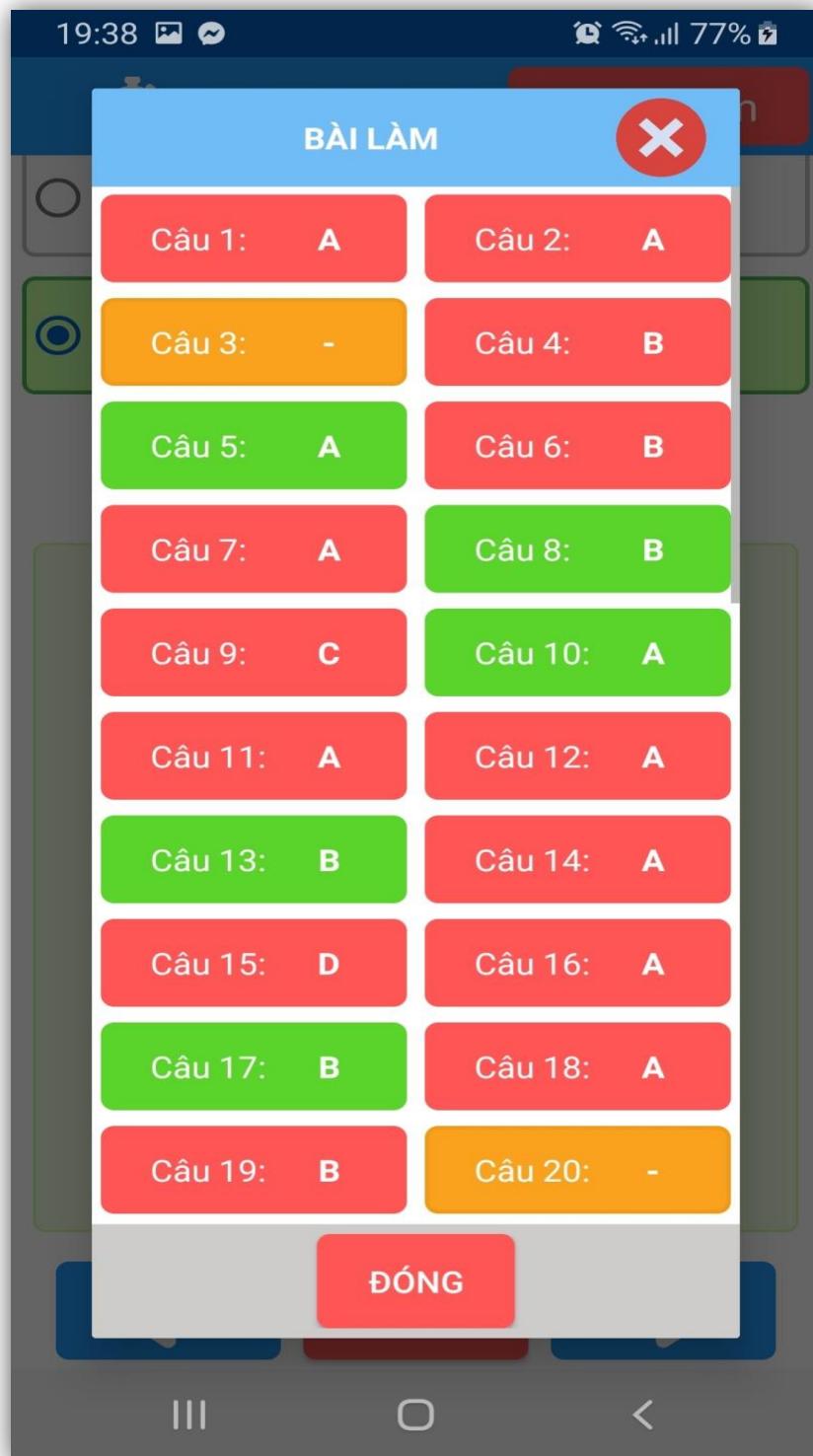


Hình 4.12: Người dùng kiểm tra lại bài làm trước khi nộp.



Hình 4.13: Người dùng nộp bài và hệ thống tính điểm.

5. Người dùng xem lại bài làm của mình

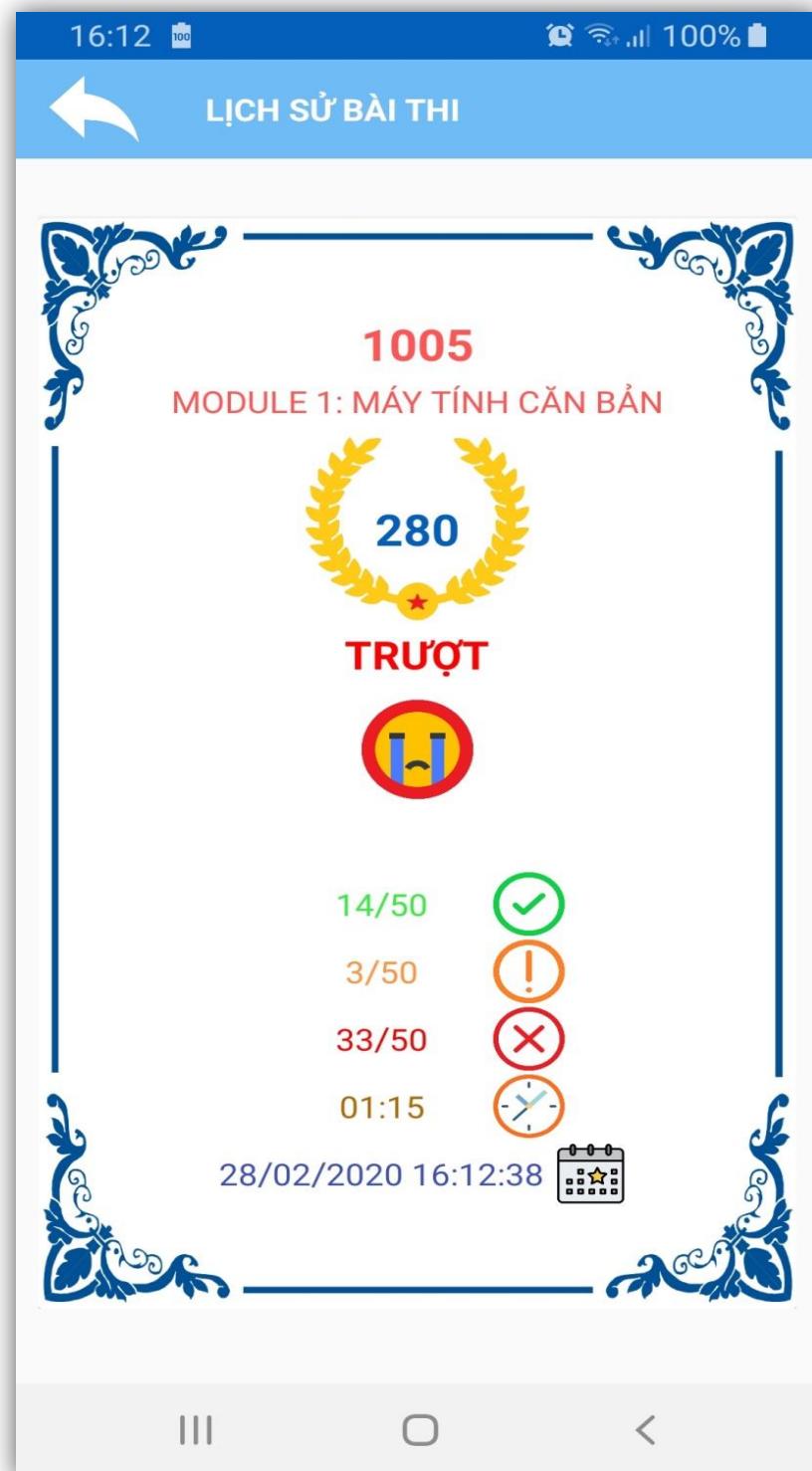


Hình 4.14: Bài làm của người dùng sau khi hệ thống chấm.



Hình 4.15: Hệ thống chia bài và hiển thị giải thích đáp án.

6. Người dùng xem lịch sử bài thi

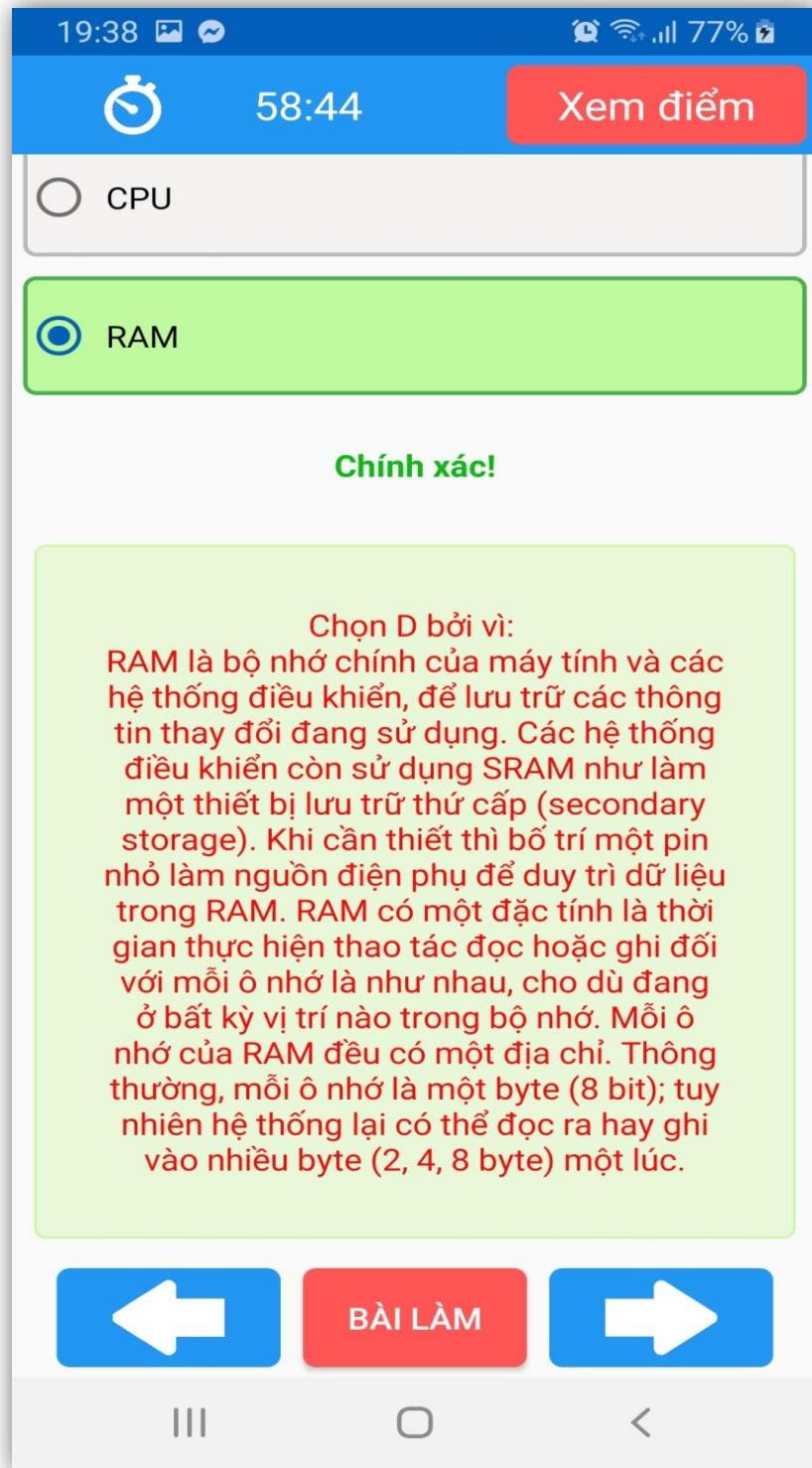


Hình 4.16: Màn hình lịch sử bài thi.

7. Người dùng xem chi tiết lịch sử bài thi



Hình 4.17: Màn hình chi tiết lịch sử bài thi.



Hình 4.18: Bài chữa và lựa chọn của người dùng được lưu lại.

4.3 Đánh giá kết quả thử nghiệm

Ứng dụng được xây dựng thử nghiệm thành công trên điện thoại di động nền tảng iOS, thực hiện được đầy đủ các chức năng như đã phân tích. So với những ứng dụng đã được phát triển hiện nay, ứng dụng được xây dựng trong phạm vi luận văn đã phát triển thêm các chức năng khác để hỗ trợ người dùng trong quá trình sử dụng.

Bảng 4.1 thực hiện việc so sánh ứng dụng thực nghiệm với các ứng dụng đã có hiện nay:

Tiêu Chí	Trắc nghiệm Tin đại cương (HanPro)	Luyện thi chứng chỉ CNTT cơ bản (Tuhocvitinh.com)	Trắc nghiệm Tin học (EscGroup)	Ứng dụng thực nghiệm
Hệ Điều Hành				
Giao Diện Hợp Lý	X	X	✓	✓
Câu Hỏi Ôn Tập	✓	✓	X	✓
Tài Liệu Ôn Tập	X	✓	X	✓
Làm Bài KT	✓	✓	✓	✓
Chữa Bài KT	✓	✓	✓	✓
Giải Thích Đáp Án	X	X	X	✓
Lưu KQ Bài Làm	X	✓	✓	✓
Xem Lại Bài Làm	X	X	X	✓
Kết Nối Cộng Đồng	X	X	X	✓

Bảng 4.1: So sánh ứng dụng thực nghiệm với các ứng dụng đã có hiện nay.

KẾT LUẬN

Dựa vào sự phát triển và phổ biến của công nghệ phần mềm trên nền tảng di động như hiện nay, rất nhiều ứng dụng Mobile-Learning được ra đời nhằm phục vụ nhu cầu học tập của người dùng. Tuy nhiên chưa thực sự có một ứng dụng nào xoáy sâu vào nội dung luyện thi chứng chỉ IC3. Trên kho ứng dụng của hai nền tảng phổ biến nhất hiện nay là iOS và Android cũng chỉ xuất hiện một vài ứng dụng về trắc nghiệm tin học đại cương, nhưng chủ yếu những ứng dụng này vẫn còn nghèo nàn về chức năng cũng như trải nghiệm của người dùng chưa thực sự tốt. Với tình hình đó, việc xây dựng một ứng dụng điện thoại di động phục vụ việc luyện thi chứng chỉ tin học IC3, cung cấp các chức năng cần thiết cho người dùng trong việc kiểm tra đánh giá là điều hoàn toàn cần thiết.

Về mặt công nghệ, luận văn đã nghiên cứu các nền tảng hệ điều hành phổ biến, so sánh sự phát triển ứng dụng trên các nền tảng này, qua đó lựa chọn nền tảng phù hợp nhất để xây dựng ứng dụng. Với những ưu điểm về nền tảng mã nguồn mở, cộng đồng phát triển đông đảo, số lượng người dùng trên thế giới nói chung và Việt Nam nói riêng là nhiều nhất trong tất cả các hệ điều hành, Android là lựa chọn phù hợp để xây dựng ứng dụng thực nghiệm. Bên cạnh đó, hệ quản trị CSDL SQLite với ưu điểm nổi bật hơn so với các hệ quản trị CSDL khác là không cần phải cấu hình, được lưu trữ trên một tập tin duy nhất, nhỏ gọn và truy xuất dữ liệu nhanh.

Về kết quả thực nghiệm, từ những công nghệ đã được nghiên cứu và lựa chọn, luận văn thực hiện phân tích thiết kế và xây dựng ứng dụng luyện thi chứng chỉ đánh giá khả năng sử dụng máy tính và Internet quốc tế IC3 trên nền tảng hệ điều hành Android. Ứng dụng được xây dựng và đã thử nghiệm thành công với các chức năng chính gồm: Thi Thủ, Xem Lịch Sử Bài Thi, Ôn Tập Với Slide Bài Giảng, Ôn Tập Với Ngân Hàng Câu Hỏi, Liên Kết Cộng Đồng IC3, Gửi Phản Hồi Cho Nhà Phát Triển.

Hướng nghiên cứu tiếp theo có thể phát triển từ các kết quả đã đạt được trong ứng dụng thực nghiệm được xây dựng trong luận văn, bổ sung các chức năng nâng cao hơn theo nhu cầu của người dùng như chức năng: Đăng nhập và quản lý người dùng, Tạo thêm nhiều kiểu câu hỏi và nhiều dạng bài tập phong phú,... Và hơn cả là có thể đưa những nội dung khác vào để phục vụ nhu cầu học tập của người dùng.

TÀI LIỆU THAM KHẢO

1. Hoàng Thu Trang, Nghiên cứu và xây dựng ứng dụng giám sát hành trình trên điện thoại di động, Trường Đại Học Công Nghệ - Đại Học Quốc Gia Hà Nội, 2017
2. <https://www.google.com/>
3. <https://developer.android.com/>
4. <https://github.com/barteksc/AndroidPdfViewer>
5. <https://github.com/chrisbanes/PhotoView>
6. <https://gs.statcounter.com/os-market-share/mobile/worldwide>
7. https://medium.com/@the_manifest/android-vs-ios-which-platform-to-build-your-app-for-first-22ea8996abe1
8. <https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/>