**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG DỮ LIỆU**



**BÁO CÁO CUỐI KÌ**

**PHÁT TRIỂN ỨNG DỤNG TRÊN**

**THIẾT BỊ DI ĐỘNG**

**Đề tài: APP QUẢN LÝ SINH VIÊN**

**Lớp: NT118.L11**

**Nhóm: 7**

**GVHD: Trần Hồng Nghi**

**Nhóm sinh viên thực hiện:**

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên** | **MSSV** |
| **1** | Lê Thị Hồng Ngọc | 18521155 |
| **2** | A Nguyễn Thị Yến Nhi | 18521198 |
| **3** | Ngô Thái Hưng | 18520062 |
| **4** | Nguyễn Vương Thịnh | 18520367 |

**□□NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN □□**

*……., ngày……...tháng……năm 2020*

**Người nhận xét**

*(Ký tên và ghi rõ họ tên****)***

**MỤC LỤC**

[MỞ ĐẦU 5](#_Toc61005423)

[KẾ HOẠCH CHI TIẾT 5](#_Toc61005424)

[a. NHÂN SỰ 5](#_Toc61005425)

[b. TỔNG QUAN 5](#_Toc61005426)

[c. PHÂN CÔNG CÔNG VIỆC 5](#_Toc61005427)

[CHƯƠNG I: GIỚI THIỆU ĐỀ TÀI 7](#_Toc61005428)

[1. Tổng quan 7](#_Toc61005429)

[2. Lý do chọn đề tài 7](#_Toc61005430)

[3. Phạm vi & đối tượng 7](#_Toc61005431)

[CHƯƠNG II: CƠ SỞ LÝ THUYẾT 8](#_Toc61005432)

[1. Hệ điều hành Android 8](#_Toc61005433)

[2. Công cụ lập trình Android Studio 8](#_Toc61005434)

[3. Cơ sở dữ liệu lưu trữ MySQL 10](#_Toc61005435)

[CHƯƠNG III: PHÂN TÍCH THIẾT KẾ HỆ THỐNG 12](#_Toc61005436)

[1. Yêu cầu 12](#_Toc61005437)

[a) Ngôn ngữ 12](#_Toc61005438)

[b) Giao Diện 12](#_Toc61005439)

[c) Chức năng 12](#_Toc61005440)

[2. MÔ HÌNH CHỨC NĂNG 13](#_Toc61005441)

[3. THIẾT KẾ LAYOUT 14](#_Toc61005442)

[a. Login 14](#_Toc61005443)

[b. Trang chủ 14](#_Toc61005444)

[c. Reset / forgot password 15](#_Toc61005445)

[d. Thông tin cá nhân 16](#_Toc61005446)

[e. Thông báo 16](#_Toc61005447)

[f. Góp ý 17](#_Toc61005448)

[h. Thông tin học phí 18](#_Toc61005449)

[i. Lịch học 18](#_Toc61005450)

[j. Lịch thi 19](#_Toc61005451)

[k. Điểm học tập 19](#_Toc61005452)

[m. Điểm danh 20](#_Toc61005453)

[n. Khảo sát 21](#_Toc61005454)

[o. Mẫu đơn 21](#_Toc61005455)

[p. Quiz 22](#_Toc61005456)

[4. CƠ SỞ DỮ LIỆU 23](#_Toc61005457)

[a) Bảng thuộc tính 23](#_Toc61005458)

[b) Sơ đồ thực thể liên kết 26](#_Toc61005459)

[CHƯƠNG IV: HIỆN THỰC ĐỀ TÀI 28](#_Toc61005460)

[**1.** **Cài đặt Web Server XAMPP** 28](#_Toc61005461)

[**2.** **Cấu hình gửi mail bằng PHP cho XAMPP** 28](#_Toc61005462)

[**3.** **Permission** 31](#_Toc61005463)

[**4.** **Using camera scan QR code** 31](#_Toc61005464)

[**5.** **Kết nối API từ frontend, lấy dữ liệu trên backend** 31](#_Toc61005465)

[**6.** **Mã hóa password** 32](#_Toc61005466)

[**7.** **Triển khai, phát triển các giao diện.** 33](#_Toc61005467)

[CHƯƠNG V: KIỂM THỬ, THỰC NGHIỆM ĐỀ TÀI 49](#_Toc61005468)

[CHƯƠNG VI: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 51](#_Toc61005469)

[TÀI LIỆU THAM KHẢO 52](#_Toc61005470)

# MỞ ĐẦU

# KẾ HOẠCH CHI TIẾT

# NHÂN SỰ

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên** | **MSSV** |
| 1 | Lê Thị Hồng Ngọc | 18521155 |
| 2 | A Nguyễn Thị Yến Nhi | 18521198 |
| 3 | Ngô Thái Hưng | 18520062 |
| 4 | Nguyễn Vương Thịnh | 18520367 |

# TỔNG QUAN

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Giai đoạn chính** | **Chịu trách nhiệm chính** | **Bắt đầu** | **Kết thúc** | **Tổng số ngày** |
| 1 | Khởi động, lên kế hoạch. | Ngọc | 10/09/2020 | 15/09/2020 | 5 |
| 2 | Thiết kế. | Hưng | 16/09/2020 | 20/09/2020 | 4 |
| 3 | Hiện thực ý tưởng. | Cả nhóm 7 | 20/09/2020 | 25/12/2020 | 95 |
| 4 | Kiểm thử, thực nghiệm. | Nhi | 26/12/2020 | 02/01/2021 | 7 |

# PHÂN CÔNG CÔNG VIỆC

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Task (Nhiệm vụ)** | **Nhân sự thực hiện** | **Bắt đầu** | **Kết thúc** | **Tổng số ngày** |
| **I** | **Khởi động lên kế hoạch** | **Ngọc** | **10/09/2020** | **15/09/2020** | **5** |
| 1 | Họp team lên ý tưởng thực hiện. | Nhóm 7 | 10/09/2020 | 11/09/2020 | 1 |
| 2 | Phân công công việc. | Ngọc | 11/09/2020 | 12/09/2020 | 1 |
| 3 | Xây dựng database. | Nhóm 7 | 12/09/2020 | 14/09/2020 | 2 |
| 4 | Khởi tạo project, quản lý mã nguồn. | Ngọc | 14/09/2020 | 15/09/2020 | 1 |
| **II** | **Thiết kế** | **Hưng** | **16/09/2020** | **20/09/2020** | **4** |
| 1 | Thiết kế các giao diện. | Hưng | 16/09/2020 | 18/09/2020 | 2 |
| 2 | Thảo luận với nhóm. (thay đổi nếu có) | Nhóm 7 | 18/09/2020 | 20/09/2020 | 2 |
| **III** | **Code giao diện, frontend, backend cho từng chức năng** | **Nhóm 7** | **20/09/2020** | **25/12/2020** | **95** |
| 1 | Đăng nhập, Quên mật khẩu. | Ngọc | 20/09/2020 | 30/09/2020 | 10 |
| 2 | Trang chủ, Đăng xuất. | Thịnh | 22/9/2020 | 2/10/2020 | 10 |
| 3 | Xem thông tin cá nhân. | Ngọc | 1/10/2020 | 11/10/2020 | 10 |
| 4 | Liên hệ - góp ý. | Thịnh | 5/10/2020 | 15/10/2020 | 10 |
| 5 | Thông báo. | Nhi | 1/10/2020 | 11/10/2020 | 10 |
| 6 | Học phí. | Hưng | 25/9/2020 | 5/10/2020 | 10 |
| 7 | Deadlines. | Ngọc | 15/11/2020 | 25/11/2020 | 10 |
| 8 | Điểm học phần. | Nhi | 12/10/2020 | 22/10/2020 | 10 |
| 9 | Điểm rèn luyện. | Nhi | 23/10/2020 | 3/11/2020 | 10 |
| 10 | Lịch thi. | Ngọc | 26/11/2020 | 6/12/2020 | 10 |
| 11 | Môn học. | Thịnh | 17/10/2020 | 27/10/2020 | 10 |
| 12 | Thời khóa biểu. | Ngọc, Thịnh | 6/12/2020 | 16/12/2020 | 10 |
| 13 | Làm trắc nghiệm. | Hưng | 10/10/2020 | 20/10/2020 | 10 |
| 14 | Điểm danh. | Ngọc | 23/11/2020 | 5/12/2020 | 13 |
| 15 | Mẫu đơn. | Hưng | 5/11/2020 | 15/11/2020 | 10 |
| 16 | Kết nối các chức năng, giao diện. | Ngọc | 10/12/2020 | 25/12/2020 | 15 |
| **IV** | **Kiểm thử, thực nghiệm** | **Nhi** | **26/12/2020** | **02/01/2021** | **7** |
| 1 | Kiểm thử các chức năng trong ứng dụng. | Nhi | 26/12/2020 | 31/12/2020 | 5 |
| 2 | Viết báo cáo. | Hưng | 01/01/2021 | 02/01/2021 | 2 |

* Trên đây là phân công công việc chi tiết của nhóm 7 chúng em. Sau đây chúng em xin trình bày nội dung cụ thể của đề tài thông qua 6 chương sau đây:
* Chương 1: Giới thiệu đề tài.
* Chương 2: Cơ sở lý thuyết.
* Chương 3: Phân tích thiết kế hệ thống.
* Chương 4: Hiện thực đề tài.
* Chương 5: Kiểm thử, thực nghiệm đề tài.
* Chương 6: Kết luận và hướng phát triển.

# CHƯƠNG I: GIỚI THIỆU ĐỀ TÀI

# Tổng quan

Ngày nay, công nghệ ngày càng phát triển, đi kèm theo đó là những nhu cầu phát sinh của sinh viên trong quá trình học tập và rèn luyện. Việc sinh viên học tập, trao đổi thông tin, sắp xếp lịch học cũng như thư giãn nghỉ ngơi được hỗ trợ bằng rất nhiều các phương tiện hiện đại. Nhưng có lẽ, lợi thế lớn nhất là chiếc smartphone mà đa phần các bạn sinh viên đều mang theo bên mình.

Hiện nay, sinh viên cần tra cứu rất nhiều thông tin khác nhau như lịch học, lịch thi, học bù, điểm rèn luyện, điểm danh,…Các chức năng khác nhau có thể nằm cùng hoặc nhiều trang web riêng biệt. Điều này gây khó khăn cho sinh viên trong việc tra cứu, tìm kiếm, cũng như cập nhật thông tin một cách chính xác và nhanh chóng nhất.

Vì vậy, một ứng dụng trên điện thoại góp phần tạo sự thuận tiện, chủ động cho sinh viên trong việc học tập, rèn luyện là một điều rất cần thiết. Trên cơ sở lý thuyết đã học, chúng em quyết định xây dựng đề tài: “App quản lý sinh viên”.

# Lý do chọn đề tài

Như đã trình bày ở trên, một ứng dụng cung cấp các thông tin cho sinh viên là rất cần thiết. Trên cơ sở lý thuyết đã học, chúng em quyết định chọn đề tài: “App quản lý sinh viên”. Đề tài này góp phần tạo ra một môi trường quản lý, cung cấp thông tin cho sinh viên tuy đơn giản nhưng hiệu quả. Ứng dụng này cần có một giao diện đơn giản, thân thiện, trực quan, dễ sử dụng, có nhiều chức năng cũng như cung cấp nhiều thông tin bổ ích cho học sinh, sinh viên cho học tập và rèn luyện tại trường Đại học Công nghệ Thông tin nói riêng và các trường học khác nói chung.

# Phạm vi & đối tượng

Đối với đề tài đã chọn, chúng em xác định phạm vi sử dụng ứng dụng trong nội bộ trường, mà cụ thể ở đây là trường Đại học Công nghệ Thông tin – ĐHQG TPHCM. Chính vì vậy, đối tượng hướng đến ở đây cũng chính là sinh viên đang học tập và rèn luyện tại Đại học Công nghệ Thông tin – ĐHQG TPHCM.

# CHƯƠNG II: CƠ SỞ LÝ THUYẾT

Đề tài: “App Quản lý sinh viên” là một ứng dụng hoạt động trên hệ điều hành Android bao gồm:

* Phần ứng dụng chạy trên điện thoại được phát triển trên nền Android SDK.
* Phần mềm cho phép sinh viên có thể xem lịch học, lịch thi, điểm, thời khóa biểu…tạo sự tiện lợi và nhanh chóng cho sinh viên xem thông tin.

Khi người dùng sử dụng ứng dụng để tra cứu, truy xuất thông tin, thông tin sẽ được lấy về thông qua WebService. Dưới đây là cơ sở lý thuyết được sử dụng để phát triển ứng dụng:

# Hệ điều hành Android

Android là một [hệ điều hành](https://vi.wikipedia.org/wiki/H%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh_di_%C4%91%E1%BB%99ng) dựa trên nền tảng [Linux](https://vi.wikipedia.org/wiki/Linux) được thiết kế dành cho các thiết bị di động có [màn hình cảm ứng](https://vi.wikipedia.org/wiki/M%C3%A0n_h%C3%ACnh_c%E1%BA%A3m_%E1%BB%A9ng) như [điện thoại thông minh](https://vi.wikipedia.org/wiki/%C4%90i%E1%BB%87n_tho%E1%BA%A1i_th%C3%B4ng_minh) và [máy tính bảng](https://vi.wikipedia.org/wiki/M%C3%A1y_t%C3%ADnh_b%E1%BA%A3ng). Ban đầu, Android được phát triển bởi [Android, Inc.](https://vi.wikipedia.org/wiki/Android,_Inc.) với sự hỗ trợ tài chính từ [Google](https://vi.wikipedia.org/wiki/Google) và sau này được chính Google mua lại vào năm 2005.

Android ra mắt vào năm 2007 cùng với tuyên bố thành lập [Liên minh thiết bị cầm tay mở](https://vi.wikipedia.org/wiki/Li%C3%AAn_minh_thi%E1%BA%BFt_b%E1%BB%8B_c%E1%BA%A7m_tay_m%E1%BB%9F): một hiệp hội gồm các công ty [phần cứng](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_c%E1%BB%A9ng), [phần mềm](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m) và [viễn thông](https://vi.wikipedia.org/wiki/Vi%E1%BB%85n_th%C3%B4ng) với mục tiêu đẩy mạnh các [tiêu chuẩn mở](https://vi.wikipedia.org/w/index.php?title=Ti%C3%AAu_chu%E1%BA%A9n_m%E1%BB%9F&action=edit&redlink=1) cho các thiết bị di động. Chiếc điện thoại đầu tiên chạy Android được bán vào năm 2008.

Android có [mã nguồn mở](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_ngu%E1%BB%93n_m%E1%BB%9F) và Google phát hành mã nguồn theo [Giấy phép Apache](https://vi.wikipedia.org/wiki/Gi%E1%BA%A5y_ph%C3%A9p_Apache). Chính mã nguồn mở cùng với một giấy phép không có nhiều ràng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên nhiệt huyết được điều chỉnh và phân phối Android một cách tự do. Ngoài ra, Android còn có một cộng đồng lập trình viên đông đảo chuyên viết các ứng dụng để mở rộng chức năng của thiết bị, bằng một loại ngôn ngữ lập trình [Java](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) có sửa đổi. Tháng 10 năm 2012, có khoảng 700.000 ứng dụng trên Android, và số lượt tải ứng dụng từ [Google Play](https://vi.wikipedia.org/wiki/Google_Play), cửa hàng ứng dụng chính của Android, ước tính khoảng 25 tỷ lượt.

Android chiếm 87,7% thị phần điện thoại thông minh trên toàn thế giới vào thời điểm quý 2 năm 2017, với tổng cộng 2 tỷ thiết bị đã được kích hoạt và 1,3 triệu lượt kích hoạt mỗi ngày. Sự thành công của hệ điều hành cũng khiến nó trở thành mục tiêu trong các vụ kiện liên quan đến bằng phát minh, góp mặt trong cái gọi là "[cuộc chiến điện thoại thông minh](https://vi.wikipedia.org/w/index.php?title=Cu%E1%BB%99c_chi%E1%BA%BFn_%C4%91i%E1%BB%87n_tho%E1%BA%A1i_th%C3%B4ng_minh&action=edit&redlink=1)" giữa các công ty công nghệ.

# Công cụ lập trình Android Studio

1. **Giới thiệu**

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.

Chức năng của Android Studio là cung cấp giao diện để tạo các ứng dụng và xử lý phần lớn các công cụ quản lý file phức tạp đằng sau hậu trường. Ngôn ngữ lập trình được sử dụng ở đây là [Java](https://quantrimang.com/java-la-gi-tai-sao-nen-chon-java-143130) và được cài đặt riêng trên thiết bị của bạn. Android Studio rất đơn giản, bạn chỉ cần viết, chỉnh sửa và lưu các dự án của mình và các file trong dự án đó. Đồng thời, Android Studio sẽ cấp quyền truy cập vào Android SDK.

Bạn cần sử dụng ngôn ngữ lập trình Java để viết các chương trình, Android SDK có nhiệm vụ kết nối các phần này lại với nhau. Cùng lúc đó Android Studio kích hoạt để chạy code, thông qua trình giả lập hoặc qua một phần cứng kết nối với thiết bị. Sau đó, bạn cũng có thể “gỡ rối” chương trình khi nó chạy và nhận phản hồi giải thích sự cố, v.v… để bạn có thể nhanh chóng giải quyết vấn đề.

1. **Các thành phần**

* Test modules

Những modules này chứa mã để kiểm tra ứng dụng và xây dựng bên trong ứng dụng kiểm tra mà nó chạy trên thiết bị. Mặc định, Android Studio sẽ tạo ra AndroidTest module chèn trong Junit tests.

* Library modules

Những modules này chứ các mã nguồn để có thể chia sẻ ứng dụng Android và tài nguyên đó ta có tham chiếu đến dự án Android. Điều này rất hữu ích khi ta muốn dùng lại mã nguồn. Library modules không được cài đặt trên thiết bị mà nó được nhúng trong file cài đặt .apk khi build tập tin .apk.

* App Engine module

Thuộc tính này cho phép chúng ta áp dụng các chức năng như sao lưu dữ liệu người dùng tới điện toán đám mây, nội dung máy chủ tới thiết bị di động, tương tác thời gian thực.

* Các tập tin trong ứng dụng Android
* .idea: Thư mục cài đặt IntelliJ IDEA.
* App: Thư mục chứa các modules và các tập tin.
* Build: Thư mục lưu trữ đầu ra khi build cho tất cả các modules của dự án.
* Gradle: Chức các tập tin gradler-wrapper.
* .gitignore: Sẽ bỏ qua các tập tin mà Git nên bỏ qua.
* Build.gradle: Cho phép chỉnh sửa các thuộc tính cho hệ thống build.
* Gradle.properties: Tập tin này để cài đặt các chế độ build ứng dụng và các thư viện.
* Gradlew: Khởi động cài đặt các tệp tin gradle cho Unix.
* Gradlew.bat: Khởi động cài đặt các tệp tin gradle cho Window.
* Local.properties: Tùy chỉnh các thuộc tính cụ thể trong máy tính cho hệ thống build như đường dẫn SDK.
* .iml: Tệp tin module được tạo bởi IntelliJ IDEA để lưu thông tin module.
* Settings.gradle: Cài đặt cụ thể các sub-projects để build
* Android Application Modules

Các modules của ứng dụng Android được bao gồm trong tệp tin .apk. Chúng chứa các thứ của ứng dụng như mã nguồn, các tệp tin nguồn. Hầu hết mã nguồn và các tệp tin nguồn được sinh ra bởi mặc định trong ứng dụng android gồm các thư mục và tệp tin sau:

1. libs/: Chứa các thư việc chỉ được dùng trong ứng dụng, lưu trữ module chính của ứng dụng.
2. src/ : Chứa mã nguồn của ứng dụng như các tệp tin Activity, fragment,…mà nó sẽ lưu dưới dạng src.main/java/activytyname.java và các tệp tin khác như .java hoặc .aidl.
3. main/java/com.project.app: Chứa mã nguồn java của ứng dụng.
4. main/jni/: Chứa mã nguyên thủy sử dụng Java Native Interface (JNI).
5. main/gen/: Chứa các tệp tin java được sinh ra bởi Android Studio như tệp tin r.java và các giao diện được tạo từ tệp AIDL.
6. main/assets/: Thư mục này rỗng, bạn có thể dùng để lưu trữ các tệp tin như Font, xmt, mp3… và sử dụng AssetManager để đọc các tệp tin này.
7. main/res/: Chứa nguồn tài nguyên của ứng dụng như các tệp tin drawable, các tệp tin layout và giá trị.
8. string.anim/: Chứa các tệp XML biên dịch các đối tượng animation.
9. color/: Chứa tệp tin XML mô tả color.
10. drawable/: Chứa các tệp hình ảnh (PNG,JPEG,GIF) và các tệp tin XML là các đối tướng Drawable chứa các trạng thái khác nhau.
11. Mimap/: Chứa icon launcher của ứng dụng.
12. layout/: Chứa các tệp tin về giao diện màn hình.
13. menu/: Chứa yteejp tin XML định nghĩa menu của ứng dụng.
14. raw/: Chứa các tập tin tương tự assets/.
15. Values/: Chứa các tệp tin XML định nghĩa nguồn tài nguyên, các khiểu phân tử XML. Không giống như các nguồn tài nguyên khác trong mục res/, nguồn tài nguyên trong values/ không tham chiếu bởi các tệp tin mặc định. Thay vì các phần tử XML điều khiển làm thế nào các nguồn tài nguyên định nghĩa trong tệp XML đặt trong lớp R
16. xml/: Chứa các tệp tin cấu hình ứng dụng, các tệp tin như PreferenceScreen, AppWidgetProviderInfo.
17. AndroidManifest.xml: Tệp tin điều khiển mô tả các thành phần của ứng dụng, nó mô tả các activities, service, intent receivers và các thành phần providers. Các quyền ứng dụng yêu cầu truy cập.

# Cơ sở dữ liệu lưu trữ MySQL

* + MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (Relational Database Management System, viết tắt là RDBMS) hoạt động theo mô hình client-server. [RDBMS](https://en.wikipedia.org/wiki/Relational_database_management_system) là một phần mềm hay dịch vụ dùng để tạo và quản lý các cơ sở dữ liệu (Database) theo hình thức quản lý các mối liên hệ giữa chúng.
  + MySQL là một trong số các phần mềm RDBMS. RDBMS và MySQL thường được cho là một vì độ phổ biến quá lớn của MySQL. [Các ứng dụng web lớn nhất](https://stackshare.io/mysql) như Facebook, Twitter, YouTube, Google, và Yahoo! đều dùng MySQL cho mục đích lưu trữ dữ liệu. Kể cả khi ban đầu nó chỉ được dùng rất hạn chế nhưng giờ nó đã tương thích với nhiều hạ tầng máy tính quan trọng như Linux, macOS, Microsoft Windows, và Ubuntu.
  + **MySQL – Database** là tập hợp dữ liệu theo cùng một cấu trúc được đặt trong một **bộ dữ liệu chung dataset**. **Database** được tổ chức sắp xếp giống như một bảng tính có sự liên kết chặt chẽ với nhau.

# CHƯƠNG III: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

# Yêu cầu

## Ngôn ngữ

* **Database: MySQL.**
* **Fronend: Java, xml.**
* **Backend: PHP.**

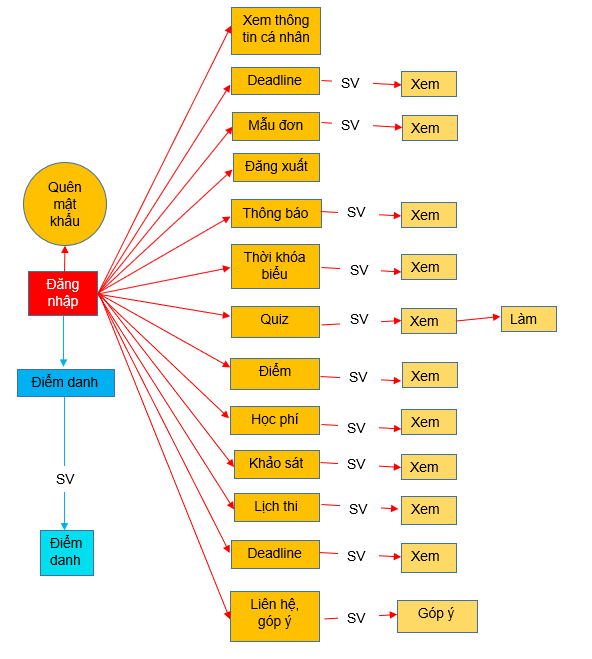
## Giao Diện

* **Thân thiện, đẹp mắt, dễ sử dụng.**
* **Phù hợp trên nhiều thiết bị.**

## Chức năng

* **Đăng nhập/ đăng xuất tài khoản.**
* **Thay đổi mật khẩu/ ảnh đại diện.**
* **Xem thông tin cá nhân/ lớp học.**
* **Xem bảng điểm học phần/ điểm rèn luyện cá nhân.**
* **Điểm danh.**
* **Chơi game.**
* **Liên hệ - góp ý.**
* **Xem thông báo.**
* **Xem thông tin học phí.**
* **Xem thời khóa biểu / lịch thi.**
* **Xem mẫu đơn thường dùng.**
* **Và một số tính năng khác.**

# MÔ HÌNH CHỨC NĂNG



SV

Xem

Điểm danh

# THIẾT KẾ LAYOUT

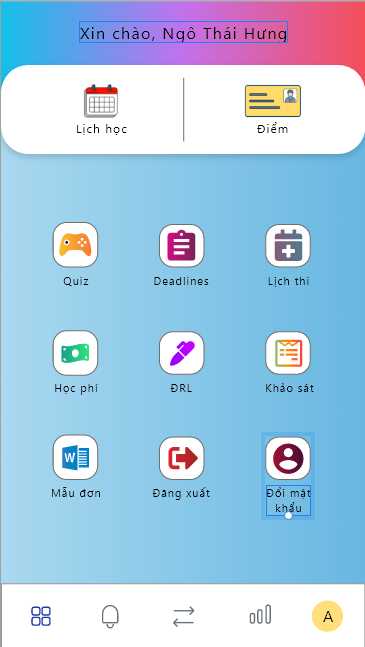
## Login

A picture containing graphical user interface, text

Description automatically generated

* Người dùng cần nhập username và password để có thể đăng nhập. Trong đó, password sẽ được lưu trữ ở dạng MD5. Khi người dùng nhập password, password sẽ chuyển về dạng MD5 và so sánh với mật khẩu được lưu trữ trong cơ sở dữ liệu.

## Trang chủ

* Sau khi người dùng đăng nhập thành công sẽ vào được trang chủ, bao gồm các button sau:
* Xem lịch học.
* Xem điểm.
* Xem deadlines.
* Xem lịch thi.
* Xem học phí.
* Xem điểm rèn luyện.
* Xem khảo sát.
* Quiz
* Đăng xuất.
* Thay đổi mật khẩu
* Ngoài ra, ở dưới thanh Bar có các nút lần lượt từ trái qua phải như sau:
* Trang chủ.
* Nút thông báo.
* Điểm danh.
* Góp ý
* Xem thông tin tài khoản.
* Trang home sẽ kết nối API lấy MSSV của sinh viên đăng nhập để sử dụng ở các trang sau

## Graphical user interface, text Description automatically generatedGraphical user interface Description automatically generatedA picture containing graphical user interface Description automatically generatedReset / forgot password

* Khi sinh viên quên mật khẩu hay muốn đổi mật khẩu, sinh viên cần nhập đúng MSSV đã được cấp.
* Ứng dụng sẽ gửi mã OTP về Email có đuôi @gm.uit.edu.vn của sinh viên để xác thực.
* Nhập đúng mã OTP thì sẽ chuyển sang trang thay đổi mật khẩu
* Sinh viên nhập 2 lần mật khẩu giống nhau thì có thể tiến hành cập nhật hoặc thay đổi mật khẩu.

## Thông tin cá nhân

A picture containing table

Description automatically generated

* Khi người dùng nhấn vào nút phải cùng ở thanh Bar bên dưới, ứng dụng sẽ hiển thị tất cả thông tin của sinh viên như:
  + Họ và tên.
  + Giới tính.
  + Lớp.
  + Ngày sinh.
  + Mã sinh viên.
  + Số điện thoại.
  + Email.
  + Địa chỉ.
  + Tình trạng.
  + Ảnh đại diện.

## Thông báo

Graphical user interface, text, application

Description automatically generated

Giao diện “Thông báo” sẽ hiển thị tất cả thông báo liên quan đến sinh viên như nghỉ học, lịch bù, lịch học và thông báo chung.

Với lịch nghỉ, lịch học và lịch bù, sinh viên chỉ nhận được thông báo của lớp sinh viên đang theo học.

Với thông báo chung, tất cả sinh viên đều sẽ nhận được thông báo.

## Góp ý

Graphical user interface, application

Description automatically generated

* Giao diện “Góp ý” cho phép sinh viên đóng góp ý kiến cá nhân về một vấn đề cụ thể nào đó đến nhà trường.
* Góp ý sẽ được gửi qua mail của quản trị viên để phải hồi kịp lúc đến sinh viên, đồng thời cũng được lưu vào CSDL để quản trị viên có thể lọc lại các góp ý dễ dàng hơn.
  1. **Deadline**

Text

Description automatically generated

* Tại giao diện “Deadline”,sinh viên có thể kiểm tra danh sách bài tập và hạn nộp cũng như tình trạng nộp bài.

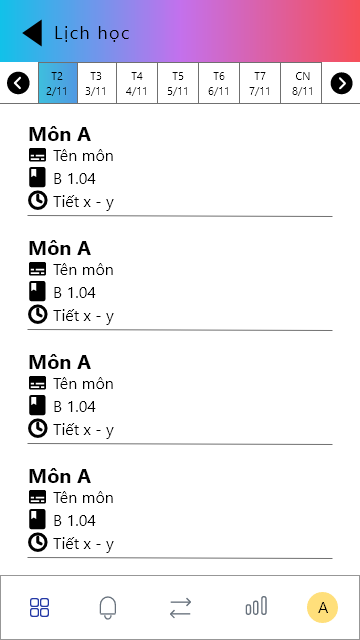
## Thông tin học phí

Table

Description automatically generated

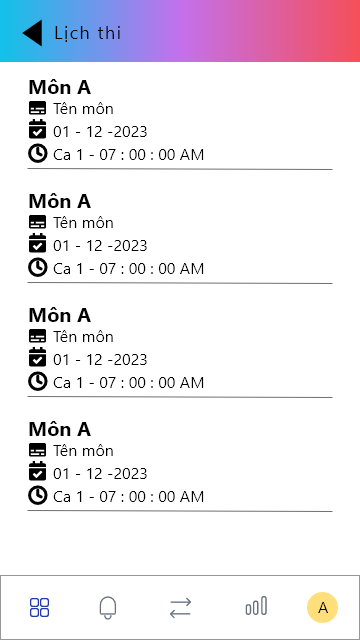
* Ở giao diện này, danh sách học phí từng học kỳ cũng như số tiền và tình trạng nộp sẽ được hiển thị một cách rõ ràng.

## Lịch học



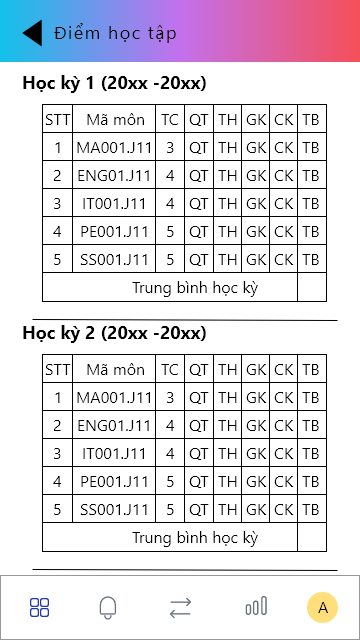
* Giao diện “Lịch học” sẽ hiển thị từng môn học, số phòng, số tiết theo từng ngày trong tuần với giao diện trực quan, dễ nhìn.

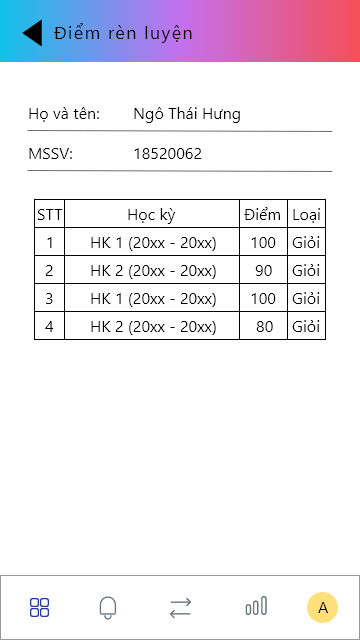
## Lịch thi



* Giao diện “Lịch thi” cho biết tên từng môn học, số phòng, ngày và giờ thi một cách cụ thể.

## Điểm học tập



* Giao diện “Điểm học tập” hiển thị chi tiết điểm của sinh viên theo từng học kỳ, từng môn học theo từng cột điểm.
  1. **Điểm rèn luyện**
* Giao diện “Điểm rèn luyện” hiển thị từng cột điểm của sinh viên theo từng học kỳ kèm theo xếp loại rèn luyện của học kỳ đó.

## Điểm danh



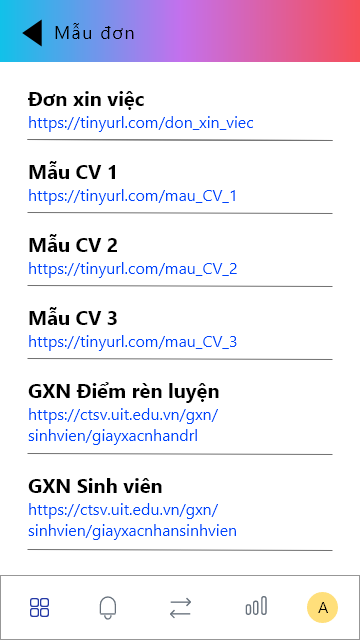
* Giao diện “Điểm danh” giúp sinh viên có thể đăng ký điểm danh buổi học từng môn khác nhau.

## Khảo sát



* Giao diện “Khảo sát” để sinh viên có thể tiến hành nhập các thông tin khảo sát của trường, giảng viên.

## Mẫu đơn



* Giao diện “Khảo sát” dùng để hiển thị các mẫu đơn thường gặp nhất. Điều này giúp sinh viên có thể thuận tiện hơn trong việc làm đơn, nộp đơn tuỳ vào nhu cầu của mình.

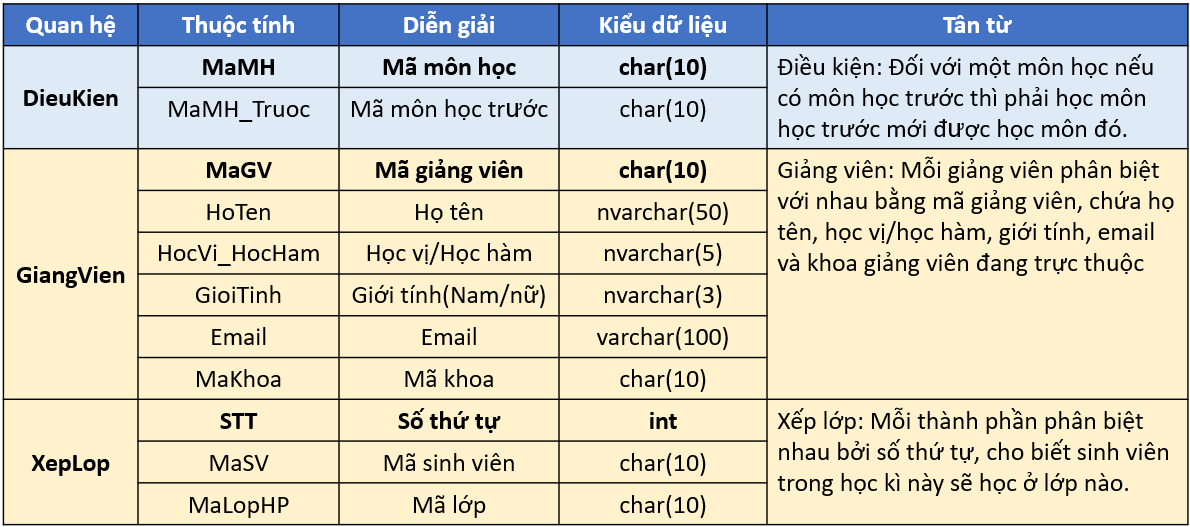
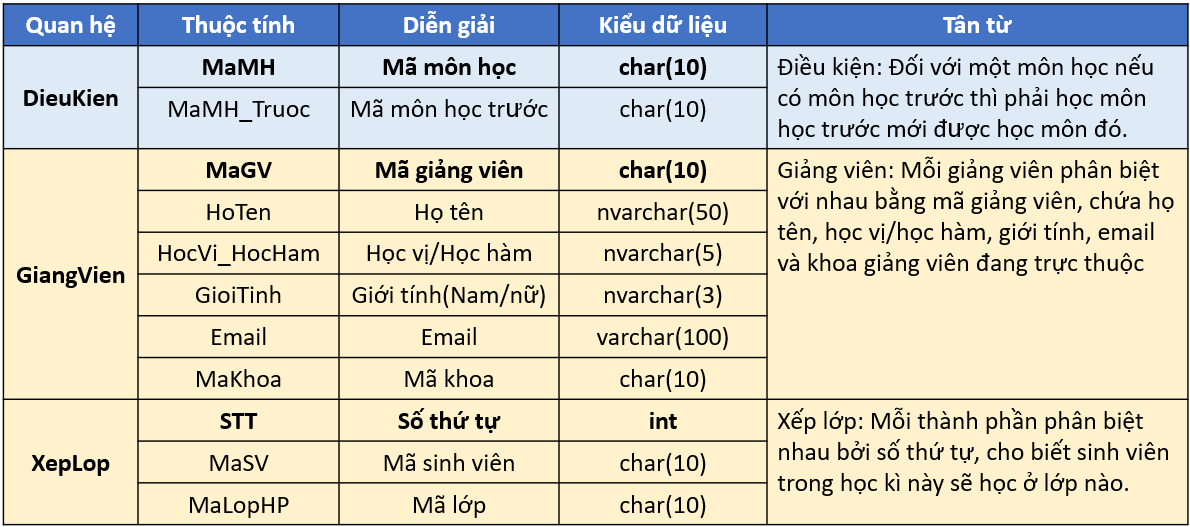
## Quiz

* Ở giao diện này, sinh viên cần nhập mã code mà giảng viên tạo để có thể tiến hành chơi game cũng như ôn lại các kiến thức đã học.

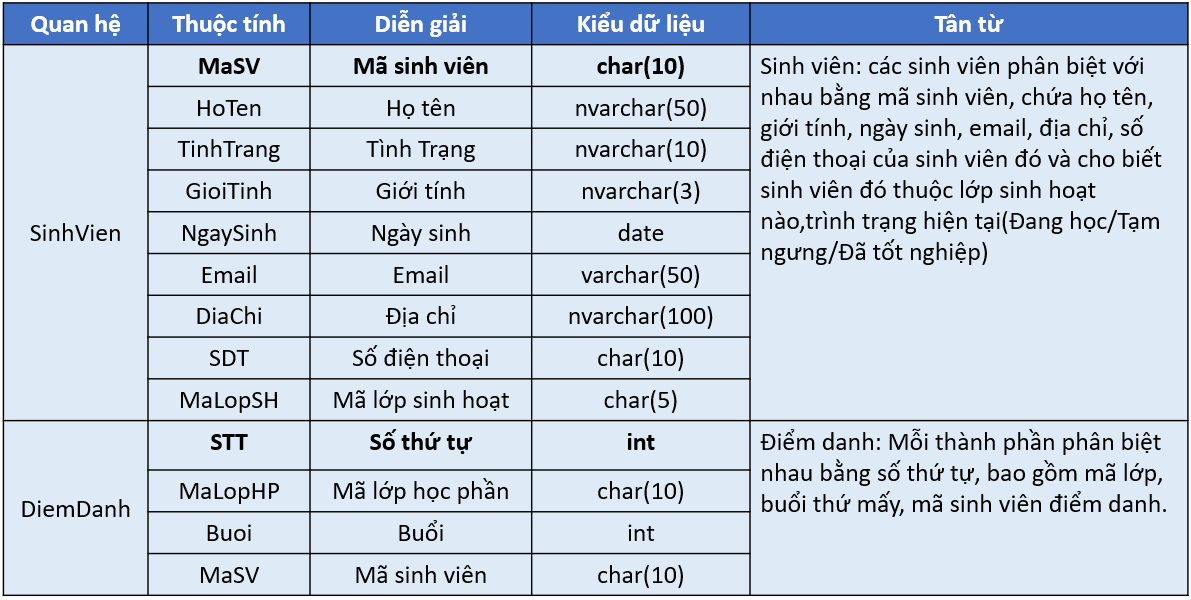
# CƠ SỞ DỮ LIỆU

## Bảng thuộc tính

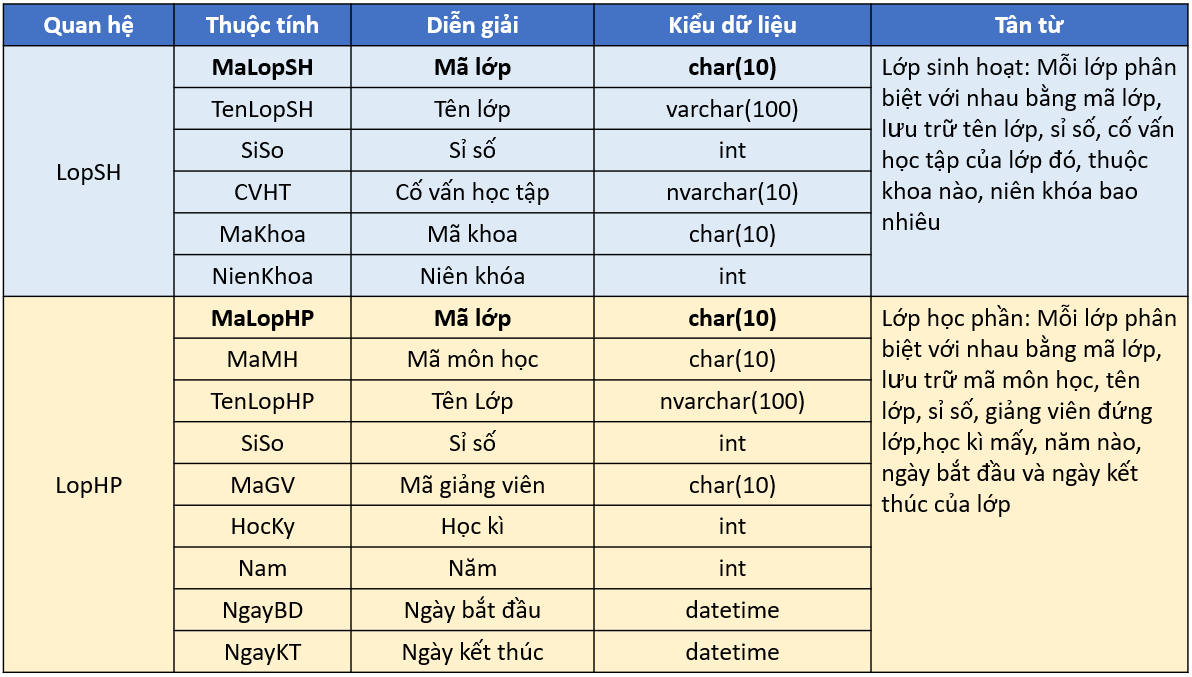
Bảng III.1: Mô hình cơ sở dữ liệu dành cho người dùng, khoa, môn học



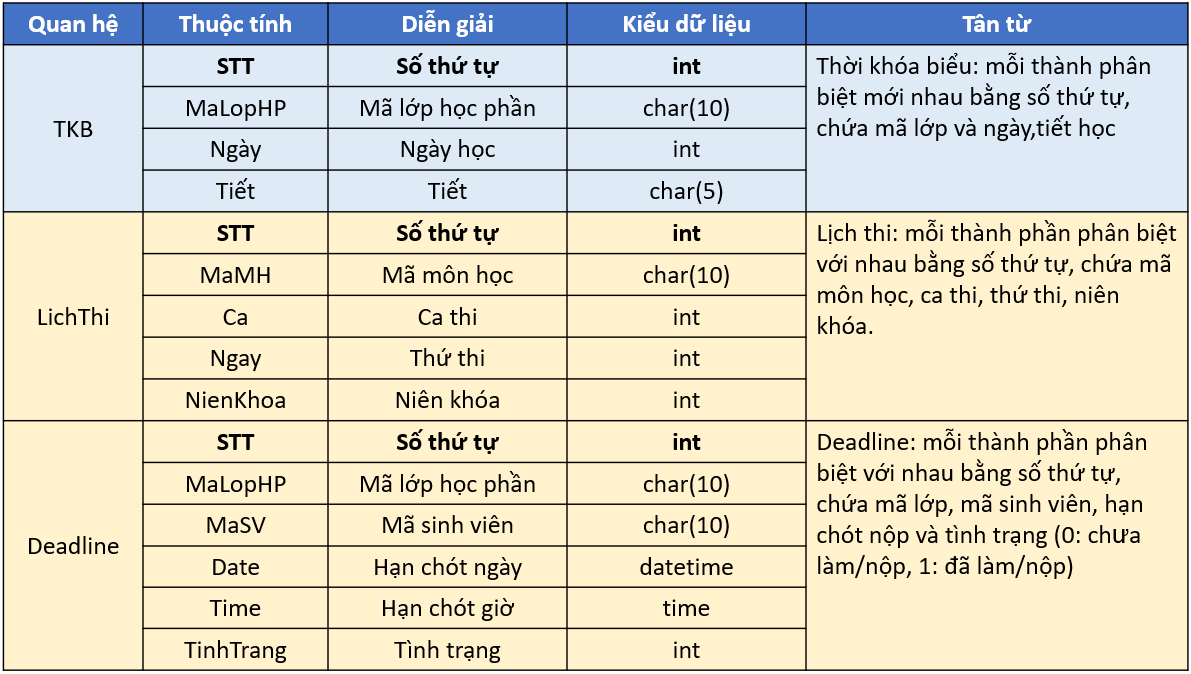
Bảng III.2: Mô hình cơ sở dữ liệu giảng viên và xếp lớp cho sinh viên.



Bảng III.3: Mô hình cơ sở dữ liệu sinh viên.



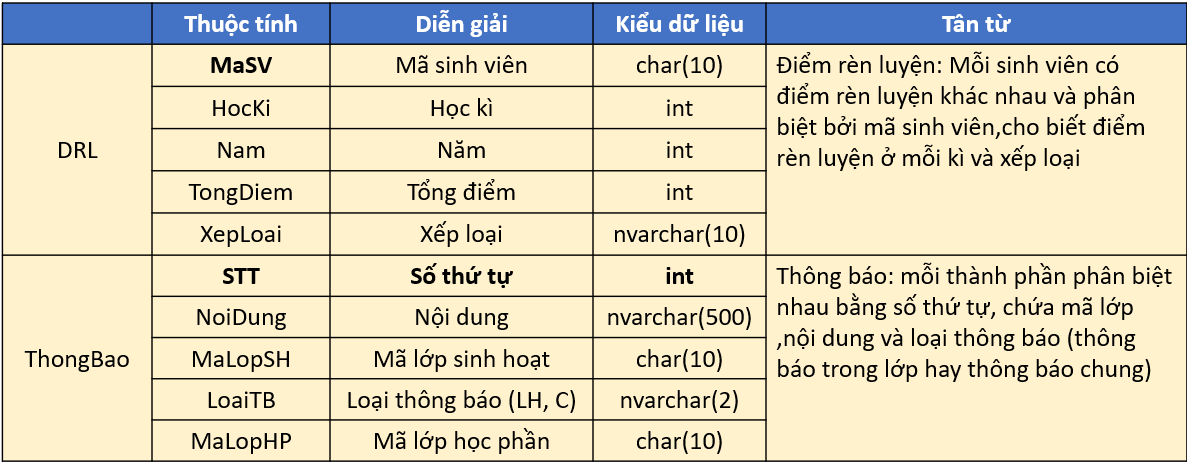
Bảng III.4: Mô hình cơ sở dữ liệu lưu trữ dữ liệu lớp sinh hoạt và lớp học phần.



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Thuộc tính | Diễn giải | Kiểu dữ liệu | Tân từ |
| Nộp bài | STT | Số thứ tự | Int | Nộp bài: Lưu trữ thông tin bài nộp của sinh viên phân biệt nhau bằng số thứ tự và số thứ tự deadline. |
| MaSV | Mã sinh viên | Char(10) |
| STTDealine | Số thứ tự deadline | Int |
| FileName | Tên file bài nộp | Varchar(100) |

Bảng III.5: Mô hình cơ sở dữ liệu lưu trữ dữ liệu thời khóa biểu, lịch thi và deadline.

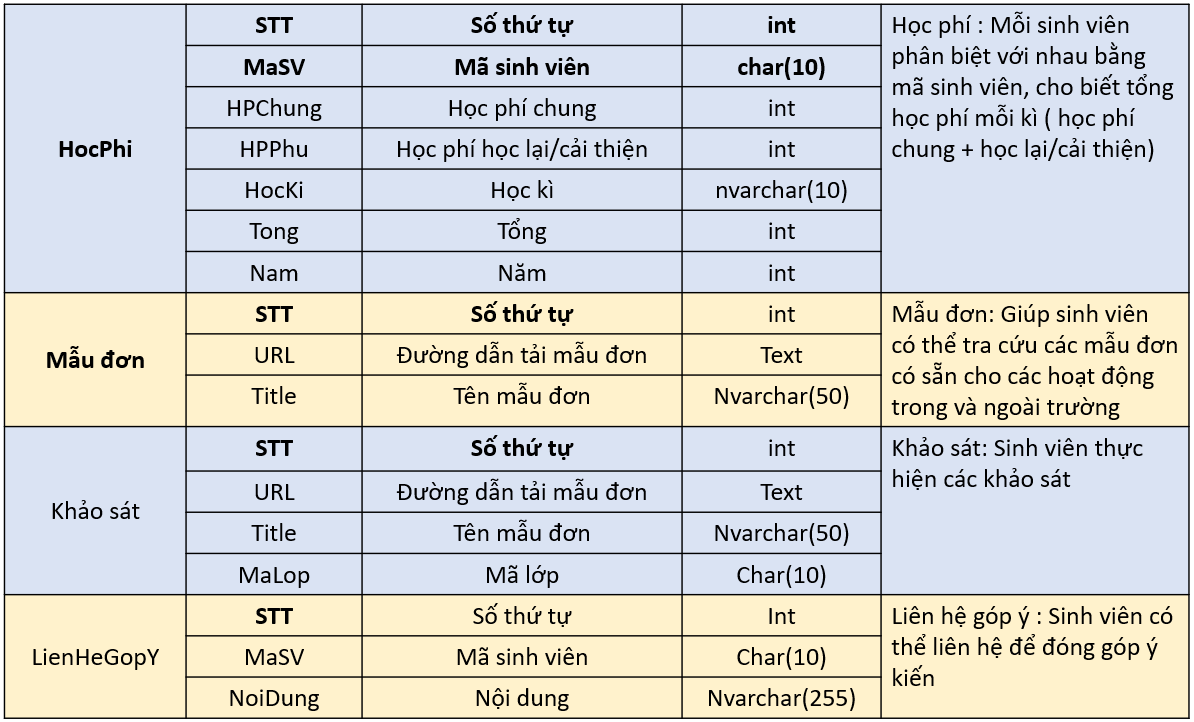
Bảng III.6: Mô hình cơ sở dữ liệu lưu trữ dữ liệu nộp bài.



Bảng III.7: Mô hình cơ sở dữ liệu lưu trữ dữ liệu điểm rèn luyện và thông báo.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Thuộc tính | Diễn giải | Kiểu dữ liệu | Tân từ |
| DiemDanh | STT | Số thứ tự | Int | Điểm danh: Lưu trữ thông tin điểm danh của sinh viên theo từng lớp, từng buổi học phân biệt nhau bằng số thứ tự. |
| MaLopHP | Mã lớp học phần | Char(10) |
| Buoi | Buổi học | Int |
| MaSV | Mã sinh viên | Char(10) |
| Code | Code kiểm tra | Int |

Bảng III.8: Mô hình cơ sở dữ liệu lưu trữ dữ liệu điểm danh.



Bảng III.9: Mô hình cơ sở dữ liệu lưu trữ dữ liệu học phí, mẫu đơn, khảo sát, liên hệ góp ý.

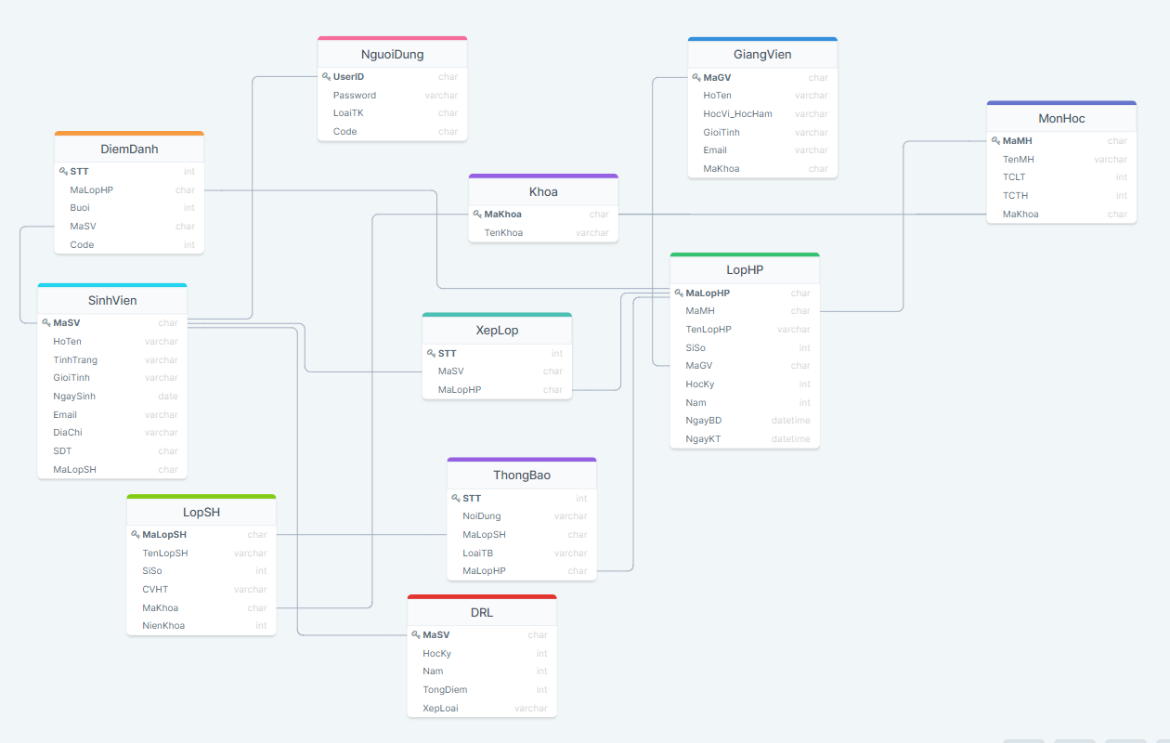
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Thuộc tính | Diễn giải | Kiểu dữ liệu | Tân từ |
| Game | **IDGame** | Mã game | char(10) | Game: Dùng để lưu trữ các game để sinh viên ôn lại kiến thức. |
| **TenGame** | Tên game | varchar(100) |
| **MaMH** | Môn mà game dành cho | char(10) |
| **TinhTrang** | Có 2 số: 0 (game đóng) và 1 (game mở) | char(2) |
| CauHoi | **STT** | Số thứ tự | int | Cauhoi: Lưu trữ câu hỏi cho các game, mỗi game gồm nhiều câu hỏi phân biệt nhau thông qua số thứ tự. |
| **IDGame** | Mã game | char(10) |
| **CauHoi** | Câu hỏi | varchar(200) |
| **CauA** | Đáp án A | varchar(200) |
| **CauB** | Đáp án B | varchar(200) |
| **CauC** | Đáp án C | varchar(200) |
| **CauD** | Đáp án D | varchar(200) |
| **CauDung** | Đáp án đúng | varchar(10) |
| SV\_Game | **STT** | Số thứ tự | int | Sv\_game: Lưu trữ thông tin các sinh viên đã chơi game. |
| **IDGame** | Mã game | char(10) |
| **MaSV** | Mã sinh viên đã chơi game | char(10) |
| **SoCauDung** | Số câu đúng | char(5) |

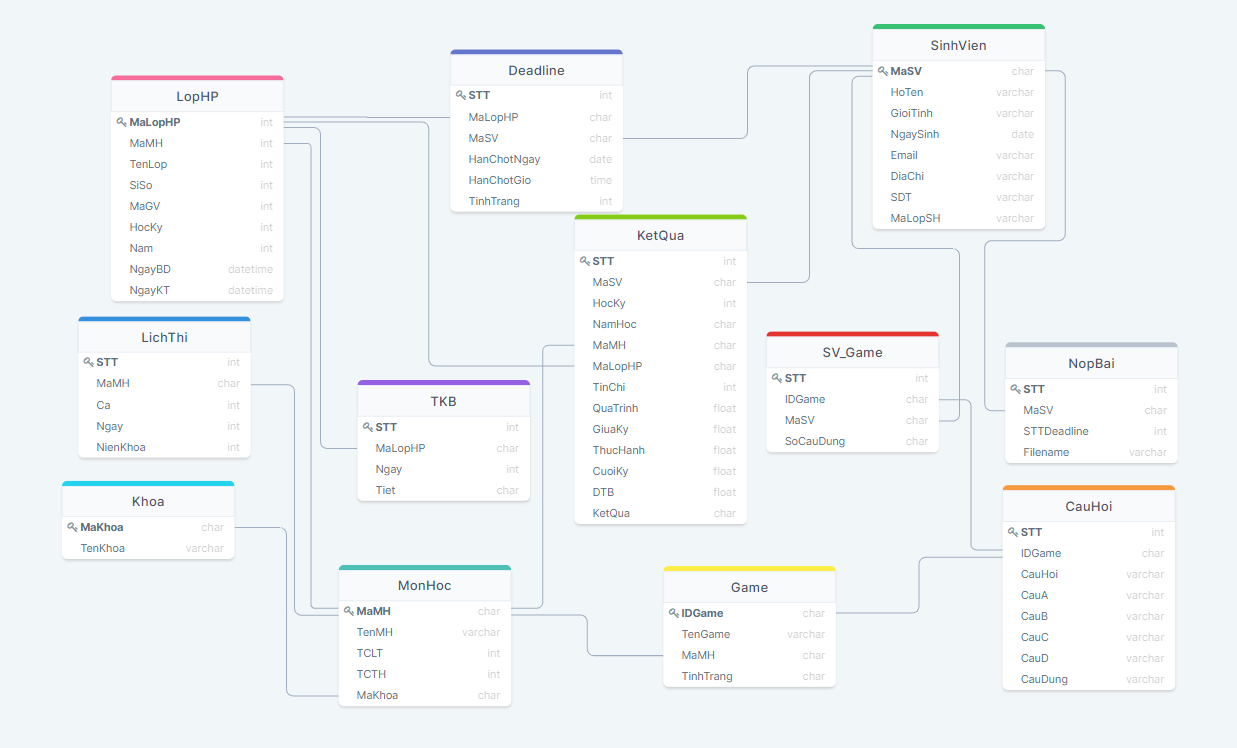
Bảng III.10: Mô hình cơ sở dữ liệu lưu trữ dữ liệu game, câu hỏi, sv\_game.

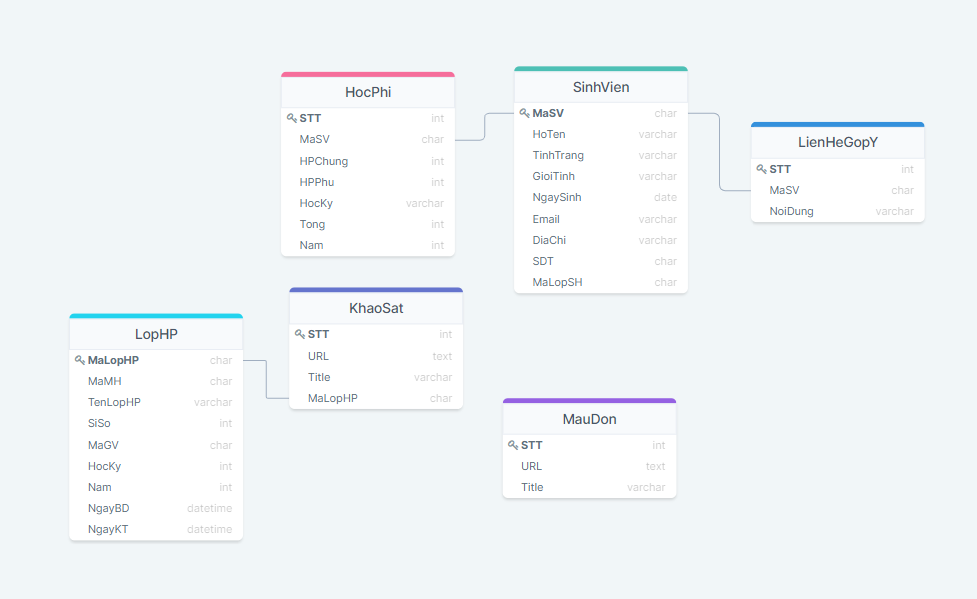
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Thuộc tính | Diễn giải | Kiểu dữ liệu | Tân từ |
| KetQua | STT | Số thứ tự | Int | KetQua: Lưu trữ thông tin điểm học tập mỗi môn học của sinh viên |
| MaSV | Mã số sinh viên | char(10) |
| HocKy | Học kỳ | Int |
| NamHoc | Năm học | char(10) |
| MaMH | Mã môn học | char(10) |
| MaLopHP | Mã lớp học phần | char(10) |
| TinChi | Số tín chỉ | int |
| QuaTrinh | Điểm quá trình | float |
| GiuaKy | Điểm giữa kỳ | float |
| ThucHanh | Điểm thực hành | float |
| CuoiKy | Điểm cuối kỳ | float |
| DTB | Điểm trung bình môn | float |
| KetQua | Kết quả | int |

Bảng III.11: Mô hình cơ sở dữ liệu lưu trữ dữ liệu game, câu hỏi, sv\_game.

## Sơ đồ thực thể liên kết







# CHƯƠNG IV: HIỆN THỰC ĐỀ TÀI

* + - 1. **Cài đặt Web Server XAMPP**
* Truy cập vào địa chỉ <https://www.apachefriends.org/download.html> và chọn phiên bản XAMPP phù hợp.
* Sau khi cài đặt xong, nhấn Start Apache và MySQL. Kết quả thành công như hình dưới.
* Nếu bị lỗi, nên đổi port khác vì có thể bị trùng với port 80 trên web



* + - 1. **Cấu hình gửi mail bằng PHP cho XAMPP**

1. **Mở file C:\xampp\sendmail\sendmail.ini**

Cấu hình các tham số như sau:

[sendmail]

smtp\_server=smtp.gmail.com

smtp\_port=587

smtp\_ssl=auto

default\_domain=localhost

error\_logfile=error.log

debug\_logfile=debug.log

auth\_username=[your\_gmail\_account\_username]

auth\_password=[your\_gmail\_account\_password]

force\_sender=[your\_gmail\_account\_username]

1. *Mở file C:\xampp\php\php.ini*

Cấu hình các tham số như sau:

[mail function]

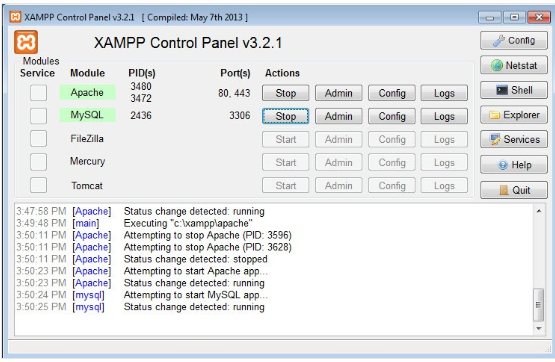
SMTP = stmp.gmail.com

smtp\_port = 587

sendmail\_from = [[your\_gmail\_account\_username]@gmail.com](mailto:your_gmail_account_username]@gmail.com)

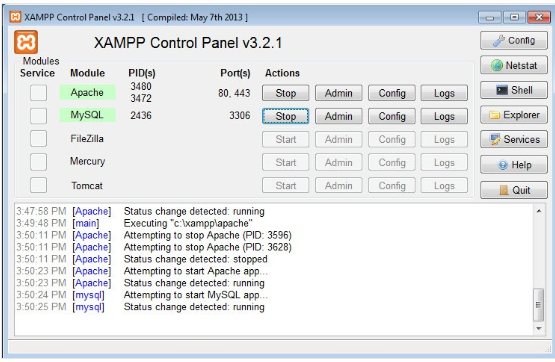
sendmail\_path = "\"C:\xampp\sendmail\sendmail.exe\" -t"

1. **Khởi động lại apache**

Mở XAMPP Control Panel lên, Nhấn “Stop” sau đó “Start” lại Apache

1. **Import CSDL vào XAMPP**

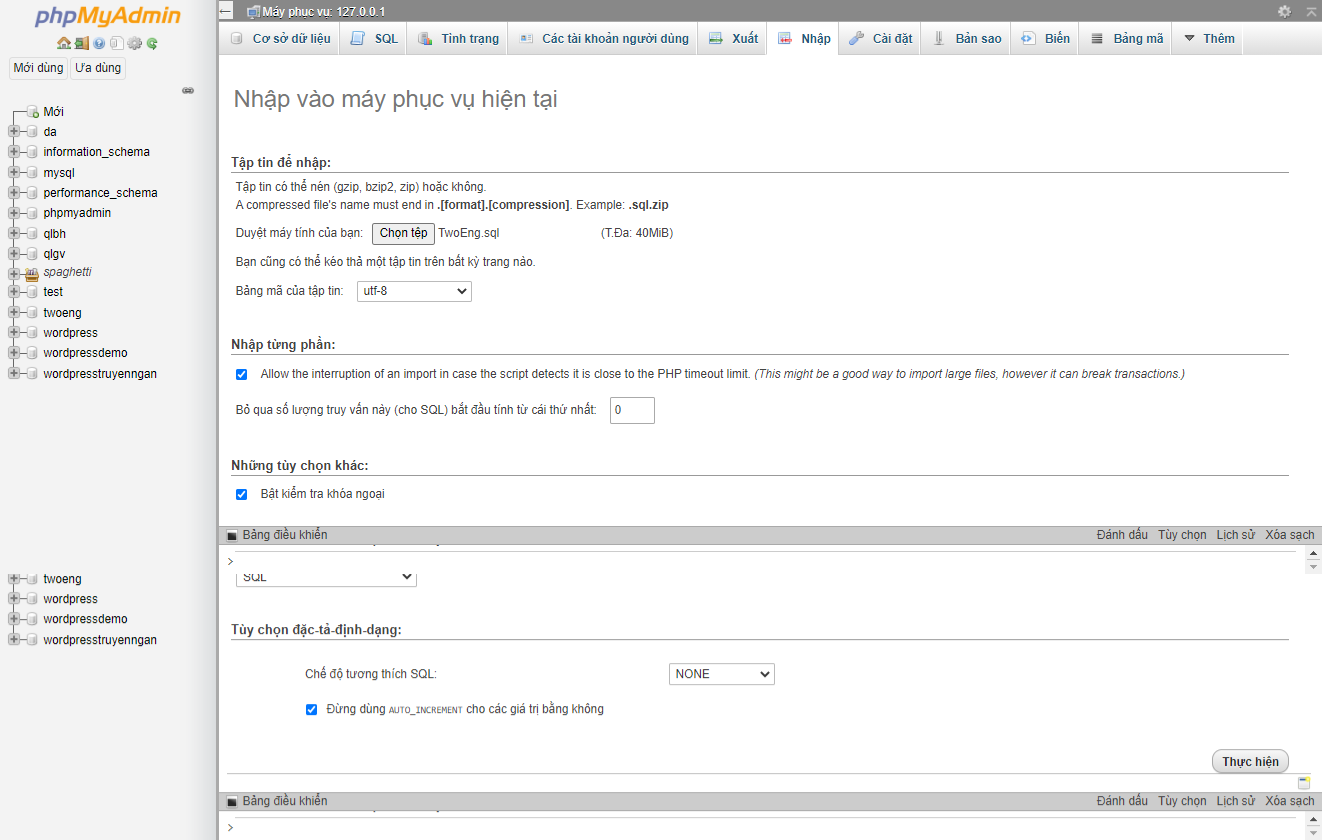
* Mở lại XAMPP Control Panel:



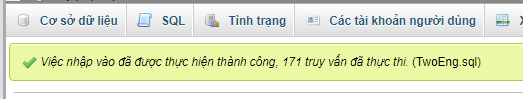
* Nhấn Admin tại dòng MySQL để mở phpMyAdmin trên trình duyệt:



* Chọn “Nhập” (Import) để Import CSDL vào.
* Sau đó nhấn “Chọn tệp”, mở thư mục htdocs của XAMPP: C:\xampp\htdocs\Nhom7\database, tìm chọn file QLGV.sql, nhấn OK.

F,{c844ba58-1c7e-41b0-bf1f-e9a6884afa76}{52},0.8541667,0.6666667

* Nhấn “Thực hiện” để Import.
* Import thành công:



* + - 1. **Permission**

**Cấu hình permission trong file AndroidManifest.xml**

1. **Kết nối internet**

<uses-permission android:name="android.permission.INTERNET"/>  
<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />

1. **Upload, download**  
   <uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE"/>  
   * + 1. **Using camera scan QR code**
2. **Trong file build.gradle(Module) thê các dòng bên dưới để sử dụng** BarcodeDetector và CameraSource

compile 'com.android.support:appcompat-v7:26.0.0-alphal'  
testCompile 'junit:junit:4.12'  
// barcode reader library  
implementation 'info.androidhive:barcode-reader:1.1.2'  
// google vision library  
implementation 'com.google.android.gms:play-services-vision:11.0.2'

1. **Trong fiel build.gradle(App) thêm dòng maven()**

allprojects **{**  
repositories **{**  
google()  
 jcenter()  
 maven()**{**  
url "https://maven.google.com"  
 **}**  
 **}**  
**}**

1. **Cấu hình trong file AndroidManifest.xml**

<uses-permission android:name="android.permission.CAMERA" />  
<uses-permission android:name="android.permission.VIBRATE" />

<meta-data android:name="com.google.android.gms.vision.DEPENDENCIES"  
android:value="barcode"></meta-data>

* + - 1. **Kết nối API từ frontend, lấy dữ liệu trên backend**

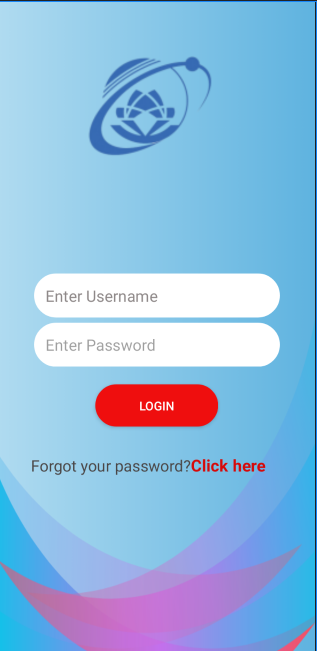
Dòng code kết nối API trên android studio, ở backend sử dụng phương thức POST để lấy dữ liệu và trả dữ liệu về bằng JSON là một JSONArray.

|  |
| --- |
| import org.json.JSONArray;  import java.io.BufferedReader; import java.io.DataOutputStream; import java.io.InputStreamReader; import java.net.HttpURLConnection; import java.net.URL; import java.nio.charset.StandardCharsets;  public class API {  public static JSONArray API(String item1, String item2, String item3, String item4, String URL){  BufferedReader reader=null;  JSONArray ja = null;  try  {  *// Defined URL where to send data* URL url = new URL(URL);  String urlParameters = "item1=" + item1 + "&item2=" + item2 + "&item3=" + item3 + "&item4=" + item4;  HttpURLConnection conn;  byte[] postData = urlParameters.getBytes( StandardCharsets.*UTF\_8* );  int postDataLength = postData.length;  conn= (HttpURLConnection) url.openConnection();  conn.setInstanceFollowRedirects( false );  conn.setRequestMethod( "POST" );  conn.setRequestProperty( "Content-Type", "application/x-www-form-urlencoded");  conn.setRequestProperty( "charset", "utf-8");  conn.setRequestProperty( "Content-Length", Integer.*toString*( postDataLength ));  conn.setUseCaches( false );  conn.setDoInput( true );  conn.setDoOutput( true );    *// Send POST data request* DataOutputStream wr = new DataOutputStream( conn.getOutputStream());  wr.write( postData );  wr.flush();  wr.close();   *// Get the server response* reader = new BufferedReader(new InputStreamReader(conn.getInputStream()));  StringBuilder sb = new StringBuilder();  String line = null;    *// Read Server Response* while((line = reader.readLine()) != null)  {  *// Append server response in string* sb.append(line + "\n");  }  String response = sb.toString();  ja = new JSONArray(response);   }  catch(Exception ex)  {   }  finally  {  try  {  reader.close();  }   catch(Exception ex) {}  }  return ja;  } } |

* + - 1. **Mã hóa password**

Mã hóa: $hash = password\_hash($password, PASSWORD\_BCRYPT, array("cost" => 10));

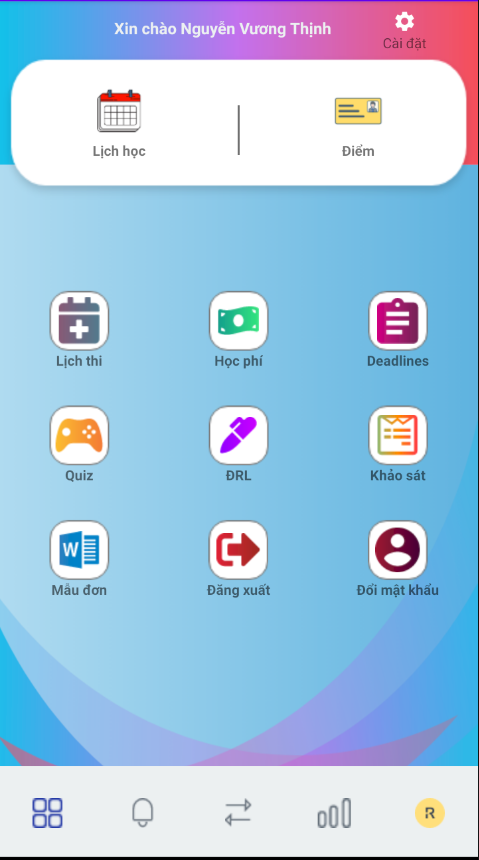
So sánh password: password\_verify($hash, $row['Password'])

* + - 1. **Triển khai, phát triển các giao diện.**
         1. Giao diện đăng nhập, đổi mật khẩu

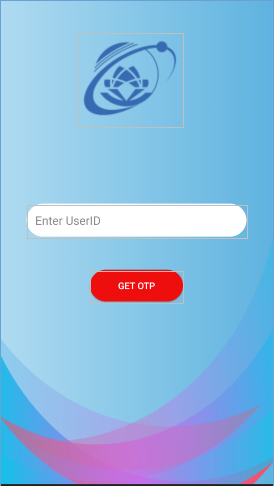
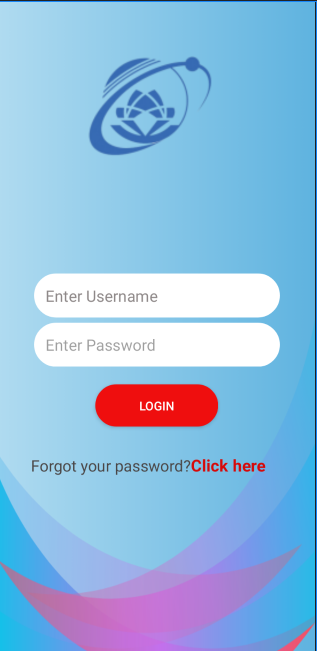
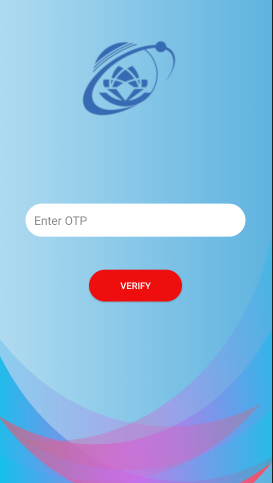
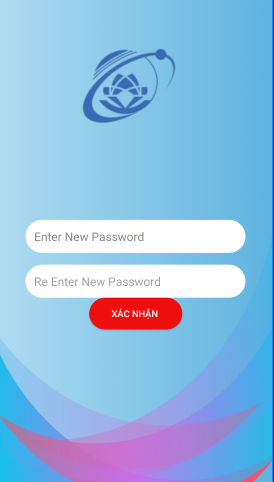
Người dùng cần nhập đúng username và password đã cung cấp trước đó để đăng nhập vào ứng dụng.

Username, mật khẩu sẽ được so sánh với Username, mật khẩu đã lưu ở database để kiểm tra. Trong đó, username và mật khẩu kiểm tra được lấy từ database thông qua Webservice. Nếu có tồn tại tài khoản sẽ trả về file JSON chứa username và password.

Hình IV.1: Giao diện đăng nhập

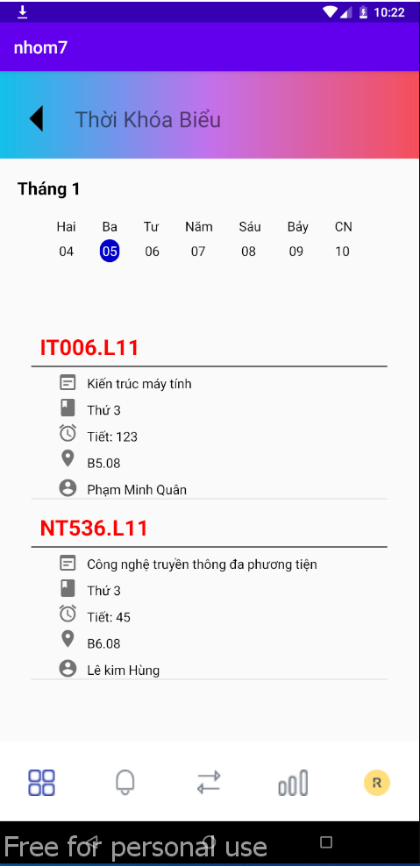
* + - * 1. Giao diện trang chủ
* Màn hình trang chủ có có đầy đủ các chức năng như:
* Xem lịch học.
* Xem điểm.
* Xem deadlines.
* Xem lịch thi.
* Xem học phí.
* Xem điểm rèn luyện.
* Xem khảo sát.
* Quiz
* Đăng xuất.
* Đổi mật khẩu
* Ngoài ra, ở dưới thanh Bar có các nút lần lượt từ trái qua phải như sau:
* Trang chủ.
* Nút thông báo.
* Điểm danh.
* Góp ý.
* Xem thông tin cá nhân.
* Trang chủ lưu trữ MSSV của sinh viên đăng nhập đế sử sụng cho các trang sau.

Hình IV.2: Giao diện trang chủ

* + - * 1. Giao diện đổi mật khẩu
* Khi sinh viên quên mật khẩu hay muốn đổi mật khẩu, sinh viên cần nhập đúng MSSV đã được cấp.

Hình IV.3,4,5,6: Giao diện từng bước đổi mật khẩu

* Ứng dụng sẽ gửi mã OTP về Email có đuôi @gm.uit.edu.vn của sinh viên để xác thực.
* Nhập đúng mã OTP thì sẽ chuyển sang trang thay đổi mật khẩu
* Sinh viên nhập 2 lần mật khẩu giống nhau thì có thể tiến hành cập nhật hoặc thay đổi mật khẩu.
  + - * 1. Giao diện lịch học

Đối với mỗi sinh viên khác nhau, sẽ có lịch học khác nhau được phân biệt dựa theo mã số sinh viên.

Thời khóa biểu sẽ cung cấp cho sinh viên các thông tin về các ca học trong ngày gồm:

* + Mã môn
  + Tên môn
  + Thứ
  + Tiết
  + Phòng
  + Giảng viên

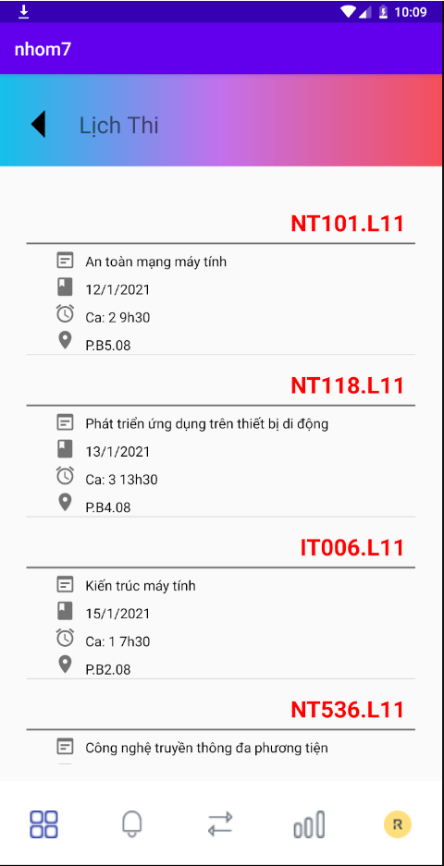
Thời khóa biểu sẽ tự động add ngày từ thứ Hai đến Chủ nhật phù hợp với ngày thực tế.

Khi bấm bào button thời khóa biểu ở trang home để chuyển sang trang thời khóa biểu, trang thời khóa biểu sẽ tự động hiển thị ngày/thứ học hôm đó lên đầu tiên.

Với các ngày không có lịch học, sẽ hiển thị thông báo “Không có dữ liệu”. Thời khóa biểu sẽ được sắp xếp từ trên xuống dưới dựa theo ca học, ca học nào sớm hơn sẽ được xếp lên trên.

Hình IV.7: Giao diện trang lịch học

* + - * 1. Giao diện lịch thi

Đối với mỗi sinh viên khác nhau, sẽ có lịch thi khác nhau được phân biệt dựa theo mã số sinh viên.

Giao diện lịch thi sẽ cung cấp cho sinh viên các thông tin của các ca thi bao gồm:

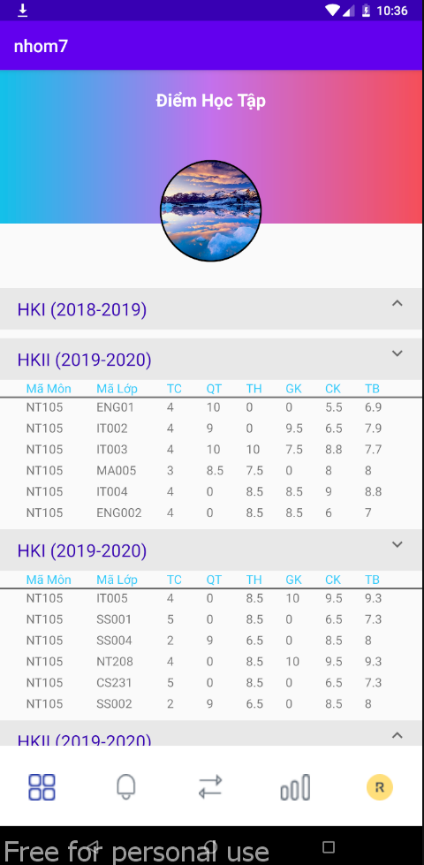
* Mã môn
* Tên môn
* Ca thi
* Giờ thi
* Phòng Thi

Lịch thi sẽ được sắp xếp từ trên xuống dưới dựa theo ngày thi, ngày thi gần nhất sẽ được xếp lên trên

Sau khi kết thúc ca thi, ca thi đó sẽ tự động loại bỏ ra khỏi trang lịch thi.

Hình IV.8: Giao diện trang lịch thi

* + - * 1. Giao diện điểm học tập



Đối với mỗi sinh viên khác nhau, sẽ có điểm học tập khác nhau được phân biệt dựa theo mã số sinh viên.

Điểm học tập của sinh viên được xây dựng dưới dạng dropdown, hiển thị thông tin điểm học tập của sinh viên từ học kì đầu tiên đến học kì hiện tại gồm:

* + Ảnh đại diện sinh viên
  + Học kỳ / Năm học
  + Mã môn
  + Mã lớp
  + Số tín chỉ
  + Điểm quá trình
  + Điểm thực hành
  + Điểm giữa kì
  + Điểm cuối kì
  + Điểm trung bình

Hình IV.9: Giao diện trang điểm học tập

* + - * 1. Giao diện điểm rèn luyện

Đối với mỗi sinh viên khác nhau, sẽ có điểm rèn luyện khác nhau được phân biệt dựa theo mã số sinh viên.

Điểm rèn luyện sinh viên chứa thông tin điểm rèn luyện của sinh viên từ học kì đầu tiên đến học kì hiện tại gồm các thông tin:

* + Số thứ tự
  + Học kỳ
  + Năm học
  + Điểm
  + Xếp loại

Điểm rèn luyện được đánh giá bằng thang điểm 100. Trong đó:

* Từ 90 đến 100 điểm: loại xuất sắc;
* Từ 80 đến dưới 90 điểm: loại tốt;
* Từ 65 đến dưới 80 điểm: loại khá;
* Từ 50 đến dưới 65 điểm: loại trung bình;
* Từ 35 đến dưới 50 điểm: loại yếu;

Hình IV.7: Giao diện trang điểm rèn luyện

* + Dưới 35 điểm: loại kém
    - * 1. Giao diện thông tin tài khoản

Giao diện sẽ hiển thị thông tin cá nhân từng sinh viên khác nhau. Mỗi sinh viên khác nhau sẽ có các thông tin được hiển thị khác nhau.

Gồm các thông tin:

* + - * + Ảnh đại diện
        + Họ và tên
        + MSSV
        + Giới tính
        + Ngày sinh
        + Số điện thoại
        + Địa chỉ
        + Email
        + Mã lớp sinh hoạt

Hình IV.8: Giao diện trang thông tin cá nhân

* + - * 1. Giao diện thông báo
  + Giao diện này sẽ hiển thị các thông báo cho sinh viên.

Gồm các loại thông báo chứa các thông tin:

+ Thông báo nghỉ học

Mã lớp

Tên lớp

Ngày nghỉ

+ Thông báo học bù

* Mã lớp
* Tên lớp
* Lịch học
* Tiết học
* Phòng học

+ Thông báo lịch học

Mã lớp

Tên lớp

Lịch học

Tiết học

Phòng học

+ Thông báo chung

Hình IV.9: Giao diện trang thông báo

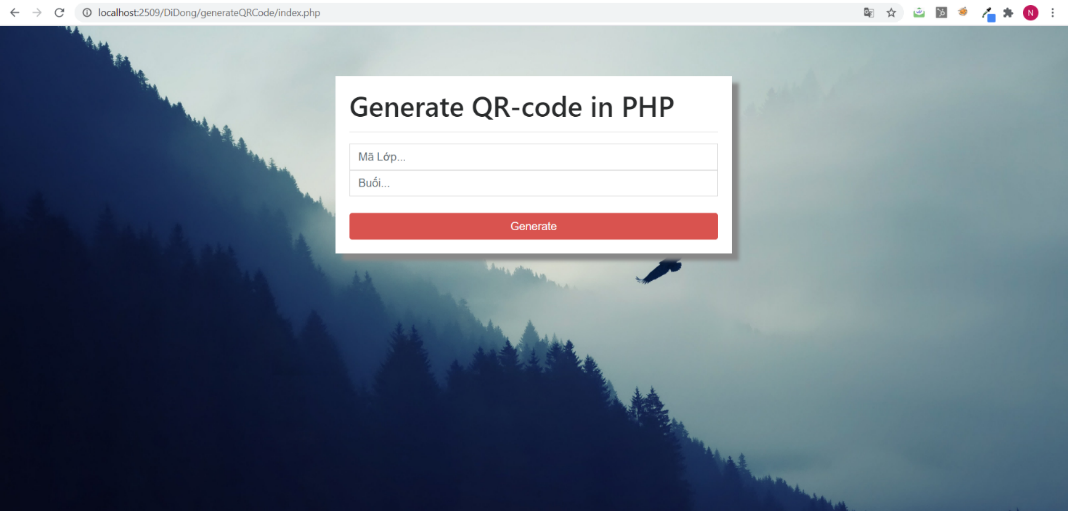
Nội dung thông báo

Với các loại thông báo thông báo nghỉ, bù, lịch học , sinh viên thuộc lớp mới nhận được thông báo, các sinh viên không thuộc lớp sẽ không nhận được.

Với thông báo chung, tất cả sinh viên đều nhận được.

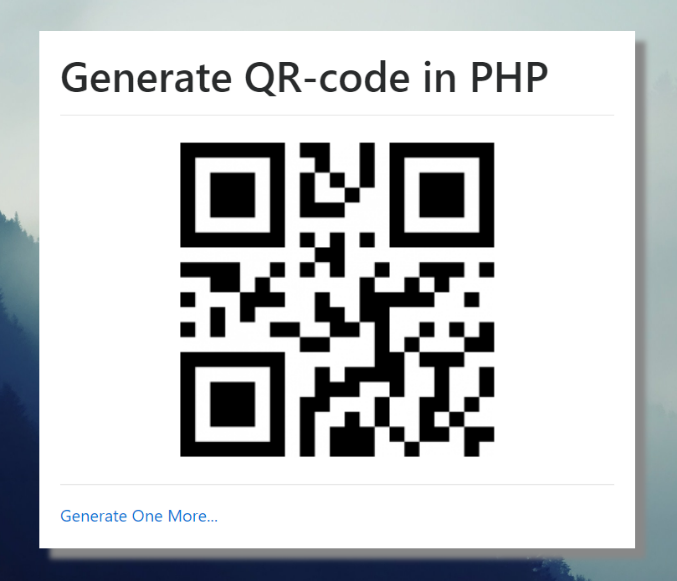
* + - * 1. Giao diện điểm danh

Thay thế cho việc điểm danh thông qua link mặc định, chúng em đã thay thế bằng cách điểm danh bằng cách quét mã QR.

Giao điện tạo mã QR trên web brower, Giảng viên nhập Mã lớp và buổi, web sẽ tự động tạo ra một mã code và mã QR.

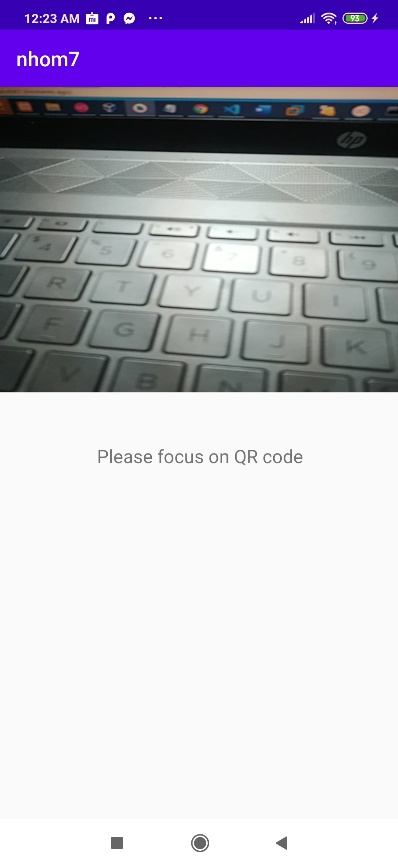
Hình IV.10: Giao diện trang thông tin cá nhân

Trong đó, sau khi scan mã QR ứng dụng sẽ so sánh code lấy được với code đã lưu trong database để kiểm tra, ví dụ code.



Hình IV.11: Tạo mã QR-code

Trên android app, sinh viên điểm danh với giao diện như hình dưới, cho phép camera quét mã QR



Hình IV.12: Giao diện quét mã QR-code

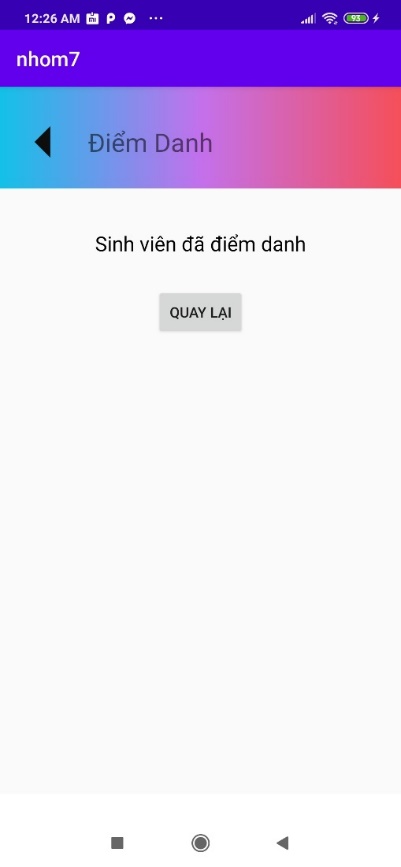
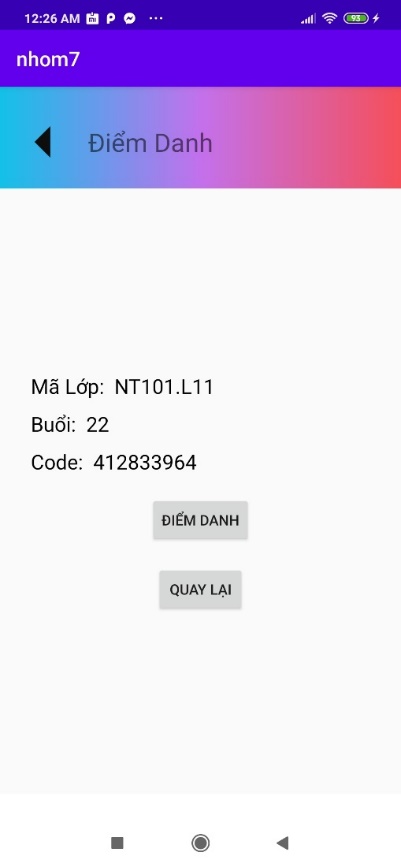
Khi camera nhận được mã QR sẽ chuyển tiếp sang giao diện bên dưới. Nếu mã QR phù hợp sẽ hiển thị thông tin lớp học, buổi học và mã code, sinh viên nhấn điểm danh để tiến hành điểm danh

Với trường hợp sinh viên không thể điểm danh:

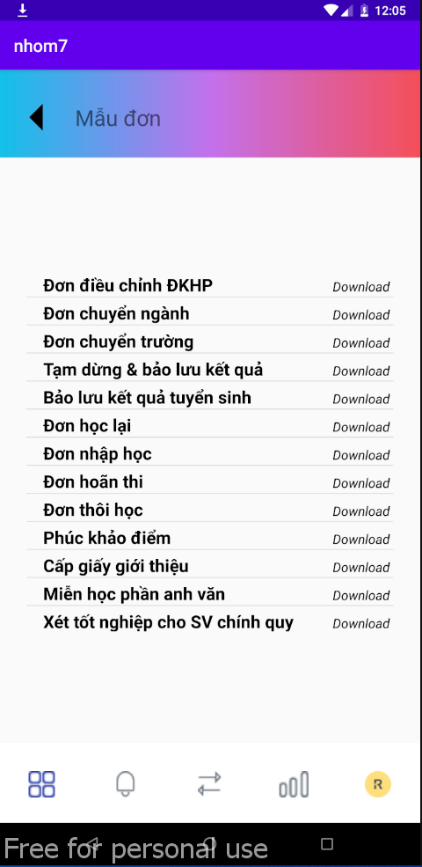
+ Sinh viên đã điểm danh

+ Không tìm thấy mã code hoặc code hết hạn

+ Sinh viên không phải là thành viên của lớp



Hình IV.13,14: Giao diện sau khi đã quét mã QR

* + - * 1. Giao diện mẫu đơn
* Giao diện này sẽ cung cấp cho sinh viên các mẫu đơn của UIT thuận tiện nhất gồm:
* Đơn điều chỉnh đăng kí học phần.
* Đơn chuyển ngành.
* Đơn chuyển trường.
* Đơn tạm dừng và bảo lưu kết quả.
* Đơn bảo lưu kết quả tuyển sinh.
* Đơn học lại.
* Đơn nhập học.
* Đơn hoạn thi.
* Đơn thôi học.
* Đơn phúc khảo điểm.
* Đơn cấp giấy giới thiệu.
* Đơn miễn học phần anh văn.
* Đơn xét tốt nghiệp cho sinh viên chính quy.
* Khi sinh viên nhấn vào mẫu đơn lựa chọn, mẫu đơn sẽ được download về máy.

Hình IV.15: Giao diện mẫu đơn

* + Code download mẫu đơn

|  |
| --- |
| lvMaudon.setOnItemClickListener(new AdapterView.OnItemClickListener() {  @Override  public void onItemClick(AdapterView<?> arg0, View arg1, int arg2, long arg3) {  arg0.getChildAt(arg2).setBackgroundColor(Color.*GRAY*);  if (*save* != -1 && *save* != arg2){  arg0.getChildAt(*save*).setBackgroundColor(Color.*WHITE*);  }  *save* = arg2;  url = item.get(arg2).getLink();  if(Build.VERSION.*SDK\_INT* >=Build.VERSION\_CODES.*M* ){  if(checkSelfPermission(Manifest.permission.*WRITE\_EXTERNAL\_STORAGE*) == PackageManager.*PERMISSION\_DENIED*){  *// permission denied, request it* String[] permission = {Manifest.permission.*WRITE\_EXTERNAL\_STORAGE*};  requestPermissions(permission, *PERMISSION\_STORAGE\_CODE*);  }  else{  *// permission already granted, perform download* startDownloading();  }  }  else {  *// system os is less than marshmallow, perform download* startDownloading();  }  } });  private void startDownloading(){  DownloadManager.Request request = new DownloadManager.Request(Uri.*parse*(url));  request.setAllowedNetworkTypes(DownloadManager.Request.*NETWORK\_WIFI* | DownloadManager.Request.*NETWORK\_MOBILE*);  request.setTitle("Download");  request.setDescription("Downloading file...");  request.allowScanningByMediaScanner();  request.setNotificationVisibility(DownloadManager.Request.*VISIBILITY\_VISIBLE\_NOTIFY\_COMPLETED*);  request.setDestinationInExternalPublicDir(Environment.*DIRECTORY\_DOWNLOADS*, ""+System.*currentTimeMillis*());   DownloadManager manager = (DownloadManager)getSystemService(Context.*DOWNLOAD\_SERVICE*);  manager.enqueue(request);  }  // handle permission result   @Override public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {  switch (requestCode){  case *PERMISSION\_STORAGE\_CODE*: {  if (grantResults.length > 0 && grantResults[0] == PackageManager.*PERMISSION\_GRANTED*){  startDownloading();  }  else {  Toast.*makeText*(this, "Permisson deny...!", Toast.*LENGTH\_SHORT*).show();  }  }  } } |

* + - * 1. Giao diện liên hệ- góp ý



Giao diện liên hệ - góp ý giúp nhà phát triển thu thập nhiều ý kiến để phát triển ứng dụng hoặc giải đáp thắc mắc của sinh viên đến nhà trường

Ở đây sẽ hiện thị thông tin của sinh viên như:

* + Họ tên
  + MSSV
  + Email
  + Nội dung góp ý

Góp ý của sinh viên sẽ được gửi qua email của quản trị viên để quản trị viên có thể kịp thời giải đáp. Ngoài ra góp ý của sinh viên cũng được lưu lại vào CSDL để quản trị viên có thể dễ dàng tìm lại

Hình IV.16: Giao diện liên hệ - góp ý

* + - * 1. Giao diện deadline



Đối với mỗi sinh viên khác nhau, sẽ có deadline khác nhau được phân biệt dựa theo mã số sinh viên.

Các deadline sẽ được sắp xếp từ trên xuống dưới, deadline mới nhất sẽ được sắp xếp lên đầu tiên gồm các thông tin:

* Mã môn
* Yêu cầu
* Hạn chót
* Tình trạng deadline
* Tình trạng nộp bài của sinh viên
* File bài nộp của sinh viên (nếu có)

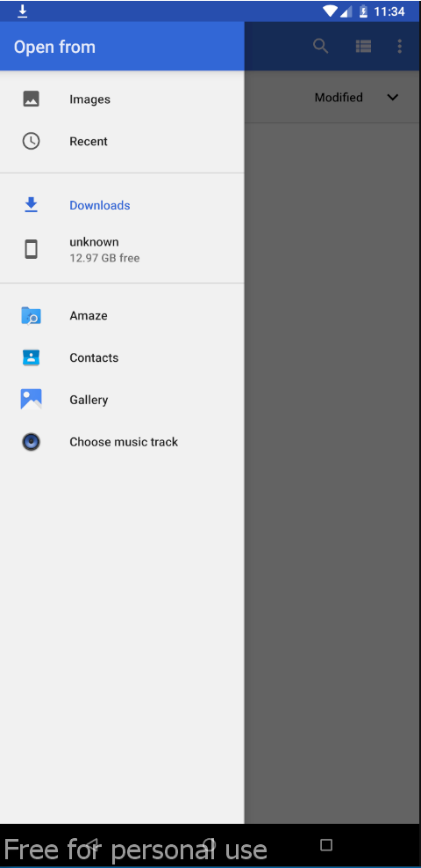
Ở giao diện deadline sinh viên có thể nộp và xóa bài nộp khi deadline còn hạn.

Các deadline hết hạn sinh viên sẽ không thể thao tác với nó.

Khi sinh viên nhấn xóa, thông tin bài nộp đó của sinh viên sẽ bị xóa ra khỏi CSDL.

Hình IV.17: Giao diện dealine

Khi sinh viên chọn nộp bài, file bài nộp sinh viên sẽ lựa chọn trong bộ nhớ máy như hình bên dưới:



Hình IV.18: Chọn file tải lên

* Code lấy file từ bộ nhớ và upload lên server:

|  |
| --- |
| public void onBrowse() {  Intent intent = new Intent();  *//sets the select file to all types of files* intent.setType("\*/\*");  *//allows to select data and return it* intent.setAction(Intent.*ACTION\_GET\_CONTENT*);  *//starts new activity to select file and return data* startActivityForResult(Intent.*createChooser*(intent, "Choose File to Upload.."), *PICK\_FILE\_REQUEST*); }  protected void onActivityResult(int requestCode, int resultCode, Intent data) {  super.onActivityResult(requestCode, resultCode, data);   if (resultCode == Activity.*RESULT\_OK*) {  if (requestCode == *PICK\_FILE\_REQUEST*) {  if (data == null) {  *//no data present* return;  }  Uri selectedFileUri = data.getData();  String name = getFileName(selectedFileUri);  upload\_file(selectedFileUri, name);  }  } }  private void upload\_file(final Uri uri, final String name) {  final Thread thread = new Thread(new Runnable() {   @Override  public void run() {  try {  uploadFile(uri, name);  JSONArray jaNopBai = API.*API*(UserID, sttBai, name, "", Constants.*addBaiNop\_URL*);  try {  JSONObject jo = jaNopBai.getJSONObject(0);   if(jo.getString("code").equals("ok")){  item.get(positionAdd).setNopbai("Đã Nộp Bài");  item.get(positionAdd).setFilename(name);  item.get(positionAdd).setSttbainop(jo.getString("SttBaiNop"));  Toast.*makeText*(deadline.this, "Nộp bài thành công!", Toast.*LENGTH\_SHORT*).show();  adapter.notifyDataSetChanged();  lvDeadline.invalidateViews();  lvDeadline.refreshDrawableState();  }  else {  Toast.*makeText*(deadline.this, "Nộp bài không thành công!", Toast.*LENGTH\_SHORT*).show();  }  } catch (JSONException e) {  e.printStackTrace();  }   } catch (Exception e) {  e.printStackTrace();  }  }  });  thread.start(); }  public String getFileName(Uri uri) {  String result = null;  if (uri.getScheme().equals("content")) {  Cursor cursor = getContentResolver().query(uri, null, null, null, null);  try {  if (cursor != null && cursor.moveToFirst()) {  result = cursor.getString(cursor.getColumnIndex(OpenableColumns.*DISPLAY\_NAME*));  }  } finally {  cursor.close();  }  }  if (result == null) {  result = uri.getPath();  int cut = result.lastIndexOf('/');  if (cut != -1) {  result = result.substring(cut + 1);  }  }  return result; }  public int uploadFile(Uri selectedFileUri, String name) throws IOException {  HttpURLConnection connection;  DataOutputStream dataOutputStream;  String lineEnd = "\r\n";  String twoHyphens = "--";  String boundary = "\*\*\*\*\*";  int bytesRead, bytesAvailable, bufferSize;  byte[] buffer;  int maxBufferSize = 1 \* 1024 \* 1024;   InputStream inputStream = getContentResolver().openInputStream(selectedFileUri);   try {  URL url = new URL(Constants.*uploadBaiNop\_URL*);  connection = (HttpURLConnection) url.openConnection();  connection.setDoInput(true);*//Allow Inputs* connection.setDoOutput(true);*//Allow Outputs* connection.setUseCaches(false);*//Don't use a cached Copy* connection.setRequestMethod("POST");  connection.setRequestProperty("Connection", "Keep-Alive");  connection.setRequestProperty("ENCTYPE", "multipart/form-data");  connection.setRequestProperty("Content-Type", "multipart/form-data;boundary=" + boundary);  connection.setRequestProperty("uploaded\_file", name);   *//creating new dataoutputstream* dataOutputStream = new DataOutputStream(connection.getOutputStream());   *//writing bytes to data outputstream* dataOutputStream.writeBytes(twoHyphens + boundary + lineEnd);  dataOutputStream.writeBytes("Content-Disposition: form-data; name=\"uploaded\_file\";filename=\""  + name + "\"" + lineEnd);   dataOutputStream.writeBytes(lineEnd);   *//returns no. of bytes present in fileInputStream* bytesAvailable = inputStream.available();  *//selecting the buffer size as minimum of available bytes or 1 MB* bufferSize = Math.*min*(bytesAvailable, maxBufferSize);  *//setting the buffer as byte array of size of bufferSize* buffer = new byte[bufferSize];   *//reads bytes from FileInputStream(from 0th index of buffer to buffersize)* bytesRead = inputStream.read(buffer, 0, bufferSize);   *//loop repeats till bytesRead = -1, i.e., no bytes are left to read* while (bytesRead > 0) {  *//write the bytes read from inputstream* dataOutputStream.write(buffer, 0, bufferSize);  bytesAvailable = inputStream.available();  bufferSize = Math.*min*(bytesAvailable, maxBufferSize);  bytesRead = inputStream.read(buffer, 0, bufferSize);  }   dataOutputStream.writeBytes(lineEnd);  dataOutputStream.writeBytes(twoHyphens + boundary + twoHyphens + lineEnd);   serverResponseCode = connection.getResponseCode();  String serverResponseMessage = connection.getResponseMessage();   Log.*i*("testLan1", "Server Response is: " + serverResponseMessage + ": " + serverResponseCode);   *//response code of 200 indicates the server status OK* if (serverResponseCode == 200) {  runOnUiThread(new Runnable() {  @Override  public void run() {  *//tvFileName.setText("File Upload completed.\n\n You can see the uploaded file here: \n\n" + "http://coderefer.com/extras/uploads/" + name);* }  });  }  *//closing the input and output streams* inputStream.close();  dataOutputStream.flush();  dataOutputStream.close();  } catch (FileNotFoundException e) {  e.printStackTrace();  runOnUiThread(new Runnable() {  @Override  public void run() {  *//Toast.makeText(NewClassPdf.this,"File Not Found",Toast.LENGTH\_SHORT).show();* }  });  } catch (MalformedURLException e) {  e.printStackTrace();  *//Toast.makeText(NewClassPdf.this, "URL error!", Toast.LENGTH\_SHORT).show();* } catch (IOException e) {  e.printStackTrace();  *//Toast.makeText(NewClassPdf.this, "Cannot Read/Write File!", Toast.LENGTH\_SHORT).show();* }  dialog.dismiss();  return serverResponseCode; } |

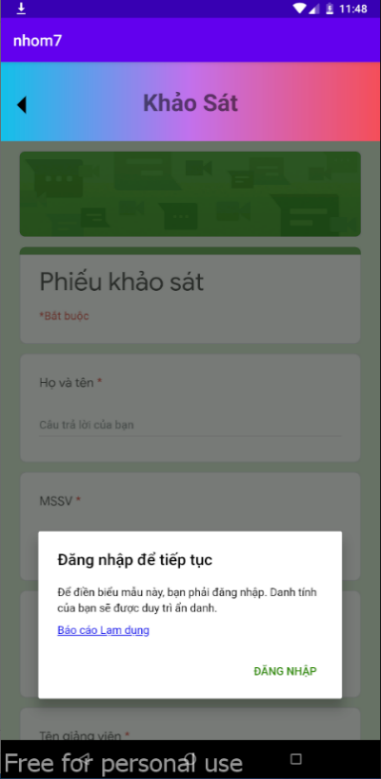
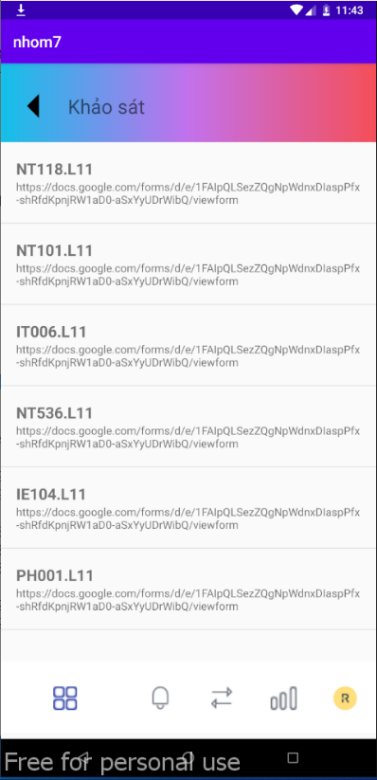
* + - * 1. Giao diện khảo sát

Giao diện này cung cấp các đường link về các vấn đề nhà trường muốn khảo sát như khảo sát chung, khảo sát từng môn học.

Đối với khảo sát từng môn học, sinh viên đang học môn nào mới nhận được phiếu khảo sát của môn học đó.

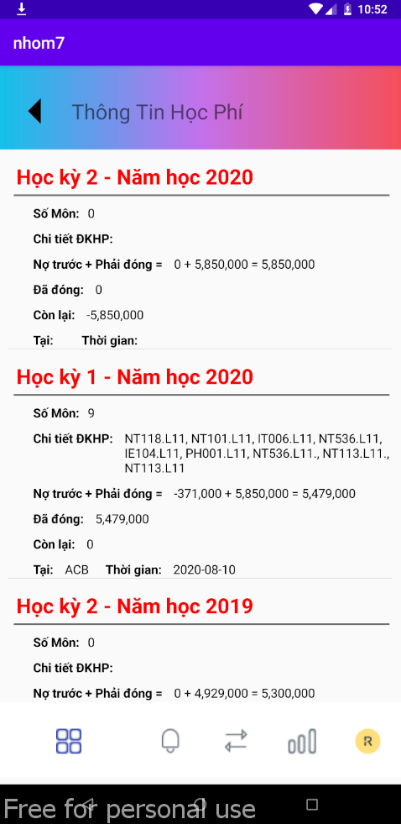
Ở mỗi mục khảo sát có chứa đường link trang khảo sát.

* + Khi nhấn vào từng phiếu khảo sát sẽ được chuyển hướng sang giao diện WebView để tiến hành khảo sát



Hình IV.18,19: Chọn và điền phiếu khảo sát

* + - * 1. Giao diện học phí

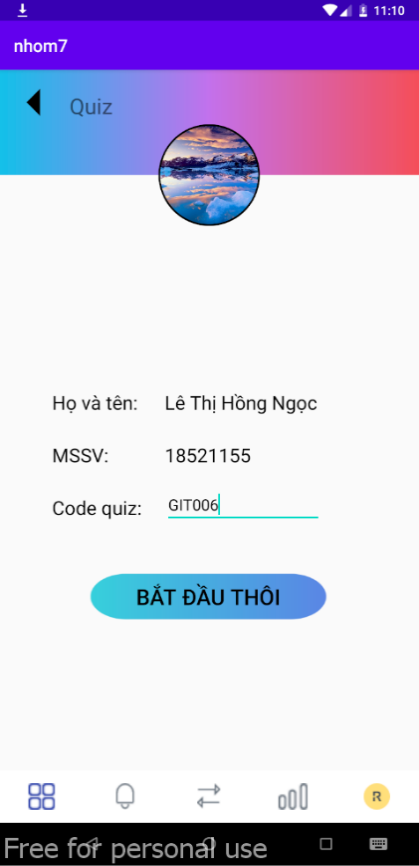
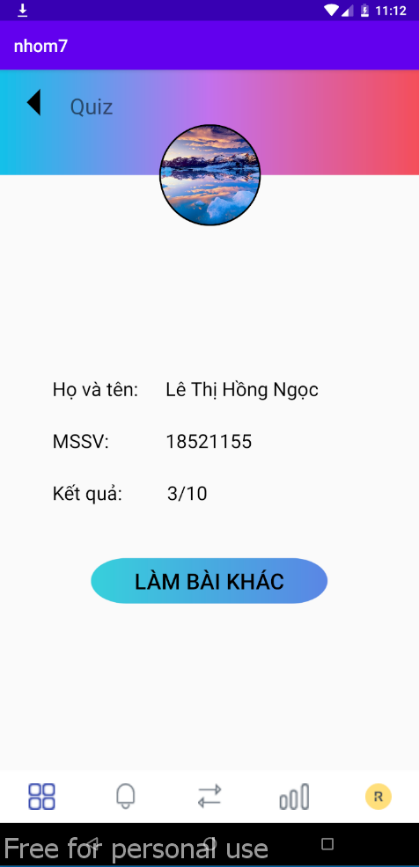
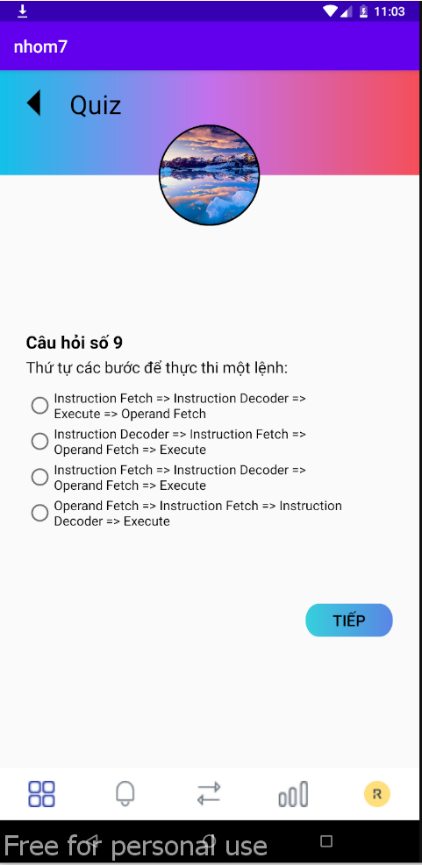
Giao diện học phí sẽ cung cấp cho sinh viên thông tin học phí cho từng kì, từ học kì đầu tiên đến học kì hiện tại gồm:

* Học kì
* Năm học
* Số môn đăng kí
* Chi tiết đăng kí học phần
* Nợ trước
* Phải đóng ở học kì này
* Tổng học phí cần đóng
* Đã đóng
* Còn lại
* Đóng tại ngân hàng nào
* Thời gian đóng học phí

Học phí sẽ được sắp xếp theo thứ tự từ trên xuống, học kì gần hết sẽ được xếp lên trên

Phần nợ trước sẽ được cộng dồn học phí của các kì trước còn nợ cho học kì hiện tại.

Hình IV.20: Giao diện học phí

* + - * 1. Giao diện quiz

Hình IV.21,22,23: Giao diện Quiz theo từng bước

* Quiz để giảng viên có thể đưa ra các câu hỏi ôn tập cho sinh viên, đồng thời cũng có thể thay thế chứ năng điểm danh, sinh viên khi làm quiz sẽ trả về kết quả ngay lập tức, đồng thời thông tin sinh viên làm quiz cũng sẽ được lưu lại vào CSDL, gồm có 3 giao diện:

+ Nhập Code

+ Làm Quiz

+ Kết quả

* Ở trang Nhập code: sinh viên sẽ nhập mã code do giảng viên đưa ra để tham gia Quiz. Sinh viên sẽ không thể tham gia quiz trường hợp quiz hết hạn, code sai hoặc sinh viên không thuộc thành viên của lớp đó.
* Ở trang Làm quiz: Mỗi lần sẽ hiển thị câu hỏi, khi sinh viên lựa chọn câu hỏi phù hợp thì bấm next để chuyển qua câu hỏi khác. Không được bỏ trống
* Sau khi hoàn tất các câu hỏi, sinh viên chọn Kết thúc để xem kết quả đạt được tại trang Kết quả. Ở đây thông tin sinh viên sẽ được lưu lại để giảng viên dễ dàng theo dõi gồm các thông tin của sinh viên như: Tên sinh viên, MSSV và kết quả test.
  + - 1. **Kết quả thu được**
         1. Ứng dụng Android

Bảng IV.1: Kết quả thu được ứng dụng Androidthu được ứng dụng Android

|  |  |  |
| --- | --- | --- |
| Tên chức năng/giao diện | Tình trạng | Ghi chú |
| Đăng nhập | Hoàn thành |  |
| Đăng xuất | Hoàn thành |  |
| Đổi mật khẩu | Hoàn thành |  |
| Trang chủ | Hoàn thành |  |
| Lịch học | Hoàn thành |  |
| Lịch thi | Hoàn thành |  |
| Điểm rèn luyện | Hoàn thành |  |
| Điểm học tập | Hoàn thành |  |
| Thông tin tài khoản | Hoàn thành |  |
| Thông báo | Hoàn thành |  |
| Điểm danh | Hoàn thành |  |
| Mẫu đơn | Hoàn thành |  |
| Liên hệ - góp ý | Hoàn thành |  |
| Khảo sát | Hoàn thành |  |
| Học phí | Hoàn thành |  |
| Quiz | Hoàn thành |  |
| Đăng ký giấy xác nhận | Hoàn thành |  |
| Một số cài đặt khác | Hoàn thành |  |

* + - * 1. Web service

Bảng IV.2: Các chức năng webservice và data base được ứng dụng Android

|  |  |  |
| --- | --- | --- |
| Tên chức năng | Tình trạng | Ghi chú |
| Cơ sở dữ liệu | Hoàn thành |  |
| Các webservice cung cấp cho ứng dụng | Hoàn thành |  |

# CHƯƠNG V: KIỂM THỬ, THỰC NGHIỆM ĐỀ TÀI

Chạy source code của từng layout trên máy ảo Genymotion, kiểm tra giao diện và các chức năng được thực hiện trên layout. Đặt các Log ở những dòng code khả nghi nếu có lỗi xảy ra và xem ở LogCat khi chạy layout. Sau khi kiểm tra từng layout tiến hành chạy ứng dụng trên điện thoại và kiểm thử.

1. **Kiểm thử trên nhiều thiết bị Android**

Bảng V.1: Kiểm thử Android

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Độ phân giải | Tên thiết bị | Phiên bản | GPU | Ram | Vi xử lí | Kết quả |
| 1080x2340 | Xiaomi Redmi Note 8 | Android 9 (Pie) | Adreno 610 | 4 GB | Snapdragon 665 8 nhân | Tốt |
| 720x1480 | Samsung Galaxy J8 | Android 8  (Oreo) | Adreno 506 | 3 GB | Snapdragon 450 8 nhân | Tốt |
| 1080x2340 | Xiaomi Mi 9T Pro | Android 10 (Q) | Adreno 640 | 6 GB | Snapdragon 855 8 nhân | Tốt |

1. **Kiểm tra các chức năng**

|  |  |  |
| --- | --- | --- |
| STT | Chức năng | Kết quả |
| **1** | **Đăng nhập.** | Tốt |
| **2** | **Đăng xuất tài khoản.** | Tốt |
| **3** | **Thay đổi mật khẩu.** | Tốt |
| **4** | **Thay đổi ảnh đại diện.** | Tốt |
| **5** | **Xem mẫu đơn thường dùng.** | Tốt |
| **6** | **Xem thời khóa biểu.** | Tốt |
| **7** | **Xem lịch thi.** | Tốt |
| **8** | **Xem thông tin học phí.** | Tốt |
| **9** | **Xem thông báo.** | Tốt |
| **10** | **Liên hệ - góp ý.** | Tốt |
| **11** | **Chơi game.** | Tốt |
| **12** | **Điểm danh.** | Tốt |
| **13** | **Xem điểm học phần.** | Tốt |
| **14** | **Xem điểm rèn luyện.** | Tốt |
| **15** | **Xem thông tin cá nhân.** | Tốt |
| **16** | **Xem thông tin lớp học.** | Tốt |

Bảng V.2: Kiểm thử chức năng

1. **Danh sách các lỗi**
2. Khi kết nối thiết bị thật với android studio: D/NetworkSecurityConfig: No Network Security Config specified, using platform default

Cách sửa lỗi:

+ Thêm dòng **android:usesCleartextTraffic="true"** vào file AndroidManifest.xml

1. Trang quét QR, scan nhiều lần

Cách sử lỗi:

+ Hide SurfaceView sau khi scan xong, **surfaceView.setVisibility(View.GONE);**

# CHƯƠNG VI: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

1. **Kết luận**
2. Những điều thu được

* Trong quá trình thực hiện, chúng em đã có cơ hội tìm tòi, nghiên cứu các công nghệ và áp dụng để xây dựng ứng dụng Android. Chúng em đã có thể nắm bắt được tổng quan công nghệ Android.
* Xây dựng thành công các chức năng đã được trình bày của ứng dụng ở phần trên.
* Giao diện thân thiện, tương thích với nhiều kích cỡ màn hình khác nhau, dễ sử dụng.
* Ứng dụng đáp ứng được các yêu cầu về độ bảo mật.
* Triển khai được trên thiết bị thật trong thực tế.
* Hiểu được kiến thức thiết kế giao diện trên XML.
* Hiểu biết nhiều hơn các kỹ thuật lập trình trong Java.
* Truy vấn cơ sở dữ liệu thông qua việc phân tích đối tượng JSON từ webservice trả về.
* Truy cập được Camera trên thiết bị, kết nối được internet thông qua giao thức http.
* Cấu trúc dữ liệu tương đối đầy đủ và gần gũi với thực tế giúp người dùng nắm bắt và sử dụng một cách dễ dàng.
* Nâng cao tinh thần tự học và tự nghiên cứu.

1. Những điều chưa đạt được

* Do thời gian có hạn nên nhiều chức năng chưa hoàn thiện, đối tượng rộng và phong phú nên không tránh khỏi một số thiếu sót.
* Chỉ mới hỗ trợ trên nền tảng Android, chưa hỗ trợ các nền tảng khác như IOS.
* Ứng dụng chỉ tương thích với một số loại màn hình nhất định.
* Ngôn ngữ hỗ trợ ứng dụng chỉ có tiếng Việt.
* Đây là ứng dụng dựa trên mô hình client/server với số lượng người dùng khá lớn. Để đáp ứng điều này, đòi hỏi cần một máy chủ đủ mạnh và ổn định. Tuy nhiên, việc xây dựng và vận hành máy chủ sẽ rất tốn công sức và chi phí.

1. **Hướng phát triển**

* Để tiếp tục phát triển đề tài này và có thể áp dụng trong trong thực tế ,chúng em nhận thấy cần phải tiếp tục chỉnh sửa, cải tiến một số chức năng cũng như tiếp tục thực hiện một số công việc sau:
* Phát triển hệ thống trên nhiều nền tảng di động hơn như IOS.
* Xử lý các lỗi chặt chẽ hơn để sử dụng một cách trơn tru.
* Hoàn thiện các chức năng phù hợp với thực tế.
* Phát triển ứng dụng với nhiều ngôn ngữ khác nhau.
* Mở rộng thêm một số chức năng dành cho Giảng viên.
* Phát triển ứng dụng để có thể tương thích với nhiều loại màn hình điện thoại.

# TÀI LIỆU THAM KHẢO

1. **Khái niệm hệ điều hành Android:** <https://vi.wikipedia.org/wiki/Android_(h%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh>)
2. **Tổng quan về Android Studio:** <https://www.codehub.com.vn/Gioi-Thieu-Ve-Android-Studio>
3. **Tổng quan về Xampp:** <https://monamedia.co/phan-mem-xampp-la-gi-huong-dan-cai-dat-xampp/>
4. **Sách Android Studio Essentials:** https://books.google.com.vn/books?id=IXJuBgAAQBAJ&lpg=PP1&dq=editions%3AZYxGmgJl0toC&hl=vi&pg=PP1#v=onepage&q&f=false
5. **Sách XML and Java: Developing Web Applications:**

<https://books.google.com.vn/books?id=Qu17E2bsfLUC&lpg=PA299&dq=xml&hl=vi&pg=PA299#v=onepage&q=xml&f=false>

1. **Google android:**

<https://developer.android.com/>

1. **Sách High Performance MySQL: Optimization, Backups, Replication, and More**

https://books.google.com.vn/books?id=BL0NNoFPuAQC&lpg=PA154&dq=mysql&hl=vi&pg=PA154#v=onepage&q=mysql&f=false