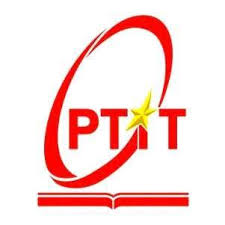
**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**TP.HỒ CHÍ MINH**

⁕⁕⁕⁕⁕⁕⁕⁕⁕⁕



**PHÁT TRIỂN ỨNG DỤNG CHO CÁC THIẾT BỊ DI ĐỘNG**

**Đề tài: QUẢN LÝ SINH VIÊN**

***Giảng viên hướng dẫn: Trương Bá Thái***

Sinh viên thực hiện: PHẠM NGUYÊN TÁNH N15DCCN137

NGUYỄN LÊ THẾ SINH N15DCCN159

ĐỖ THỊ MỸ LINH N15DCCN135

PHẠM TRƯỜNG THIÊN ÂN N15DCCN173

HOÀNG THỊ BÌNH N15DCCN142

Lớp: D15CQCP01-N

1. **Giới thiệu đề tài**

Ngày nay, tin học đã có những bước tiến nhanh chóng về ứng dụng của nó trong mọi lĩnh vực của cuộc sống trên phạm vi toàn thế gi ới nói chung và Việt Nam nói riêng.Tin học được người ta quan tâm và nhắc đến nhiều hơn bao giờ hết vì nó là một phần không thể thiếu trong cuộc sống văn minh,góp phần đẩy mạnh công cuộc công nghiệp hoá hiện đ ại hoá đất nước, tiến đến nền kinh tế tri thức. Máy vi tính cùng với những phần mềm là công cụ đắc lực giúp ta quản lý, tổ chức, sắp xếp và xử lý công việc một cách nhanh chóng và chính xác. Ở Việt Nam hiện nay, máy tính điện tử đặc biệt là máy vi tính trong nhiều năm qua đã được sử dụng rất rộng rãi. Sự phát triển của tin học, các công nghệ phần mềm, phần cứng, các tài liệu tham khảo đã đưa chúng ta từng bước tiếp cận với công nghệ thông tin trong mọi lĩnh vực nhằm đáp ứng nhu cầu của con người. Quản lý sinh viên là một đề tài không còn mới mẻ với các bài toán quản lý. Việc đưa tin học vào ứng dụng để quản lý là rất hữu ích, vì chúng ta phải bỏ ra rất ít thời gian mà lại thu được hiệu quả cao, rất chính xác và tiện lợi nhanh chóng.Trong phạm vi bài nhóm chúng em đã được đề cập đến vấn đề “Quản lý sinh viên”.

1. **Giới thiệu về Android**
2. **Khái niệm Android**

* Android là hệ điều hành điện thoại mã nguồn mở miễn phí do Google phát triển dựa trên nền tảng Linux. Bất kì một hãng sản xuất phần cứng nào cũng đều có thể tự do sử dụng hệ điều hành Android cho thiết bị của mình, miễn là các thiết bị phải đáp ứng được các tiêu chuẩn cơ bản do Google đặt ra.
* Các nhà sản xuất có thể tự do thay đổi phiên bản Android trên máy của mình một cách tự do mà không cần xin phép hay trả bất kỳ một khoản phí nào nhưng phải đảm bảo tính tương thích ngược của phiên bản chế riền đó.
* Android là nền tảng cho thiết bị di động bao gồm một hệ điều hành, midware và một số ứng dụng chủ đạo. Bộ công cụ Android SDK cung cấp các công cụ và bộ thư viện các hàm API cần thiết để phát triển ứng dụng cho nền tảng Android sử dụng ngôn ngữ lập trinh Java.

1. **Các thành phần cơ bản của project Android**

**2.1. Activity** : Là một trong những thành phần chính của bất kỳ một ứng dụng Android nào. Activity được dùng để hiển thị một màn hình, chỉ một việc mà người dùng có thể thực hiện trong một ứng dụng Android, nó tương tác với người sử dụng.   
**2.2. Intent** : Là một cấu trúc dữ liệu mô tả cách thức, đối tượng thực hiện của một Activity, là cầu nối giữa các Activity, giúp ta triệu gọi cũng như truyền các dữ liệu cần thiết để thực hiện một Activity từ một Activity khác. Ứng dụng Android thường bao gồm nhiều Activity, mỗi Activity hoạt động độc lập với nhau và thực hiện những công việc khác nhau.   
**2.3. Bundle** : Ta thường dùng đối tượng Bundle để đóng gói dữ liệu cho việc truyền tải qua các Activity khác nhau. Bundle có 2 loại phương thức: putXXX để cho bên Bundle gửi key đi, ứng với kiểu dữ liệu nào thì put đúng kiểu đó. Ở bên nhận Bundle thì dùng các phương thức getXXX tương ứng để lấy dữ liệu theo key bên gửi.

1. **Thư viện Volley trong Android**

Volley được Google giới thiệu vào khoảng tháng 6/2013, là một thư viện dùng để send và receive response từ Server sử dụng giao thức HTTP.

**3.1.Volley có các điểm nổi bật dưới đây:**

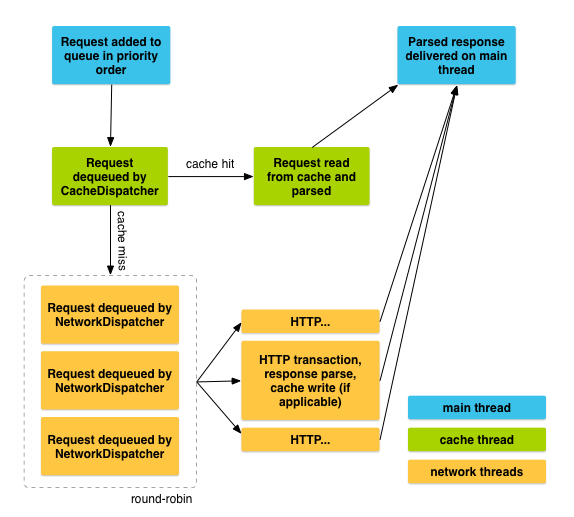
+ Tự động lập lịch (scheduling) cho các request.

+ Caching Response.

+ Hỗ trợ set độ ưu tiên cho các request.

+ Hỗ trợ nhiều kết quả trả về (String, JSONObject, JSONArray, Bitmap,…)

+ Có thể hủy Request.

**3.2. Mô hình làm việc của Volley**

**3.3. Sử dụng thư viện Volley**

Trước tiên chúng ta phải import thư viện này vào Android Studio. Copy và paste dòng dưới đây vào **dependencies** trong file **build.gradle** của module **app**:



Nhấn **Async Now** để Android Studio download và nạp thư viện vào project.

Để sử dụng Volley chúng ta phải cấp quyền Internet trong

AndroidManifest.xml như sau:



**3.4. RequestQueue Volley**

Việc sử dụng Volley rất đơn giản, các bạn chỉ cần tạo một RequestQueue bằng lệnh



Phương thức này về một RequestQueue. Khi đã có RequestQueue chúng ta chỉ cần các loại Request và add các request đó vào RequestQueue.



**3.5. StringRequest Volley**

**GET StringRequest Volley:**



**POST StringRequest Volley:**



Cách tạo request đối với JSONObjectRequest, JSONArrayObject, ImageRequest đều tương tự như với StringRequest. Contructor sẽ có những thành phần dưới đây:

+ Method POST, GET.

+ Contructor truyền vào url của request

+ Hai Listener là Response.Listener và Response.ErrorListener

## **3.6. JSONObjectRequest Volley**

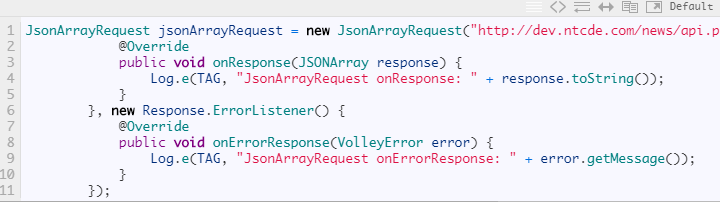
Request có response trả về là JSONObject.

Nếu Request thành công chúng ta có thể lấy ra đối tượng JSONObject trong phương thức onResponse. Ngược lại chúng ta sẽ có thông tin lỗi trong phương thức onErrorResponse.

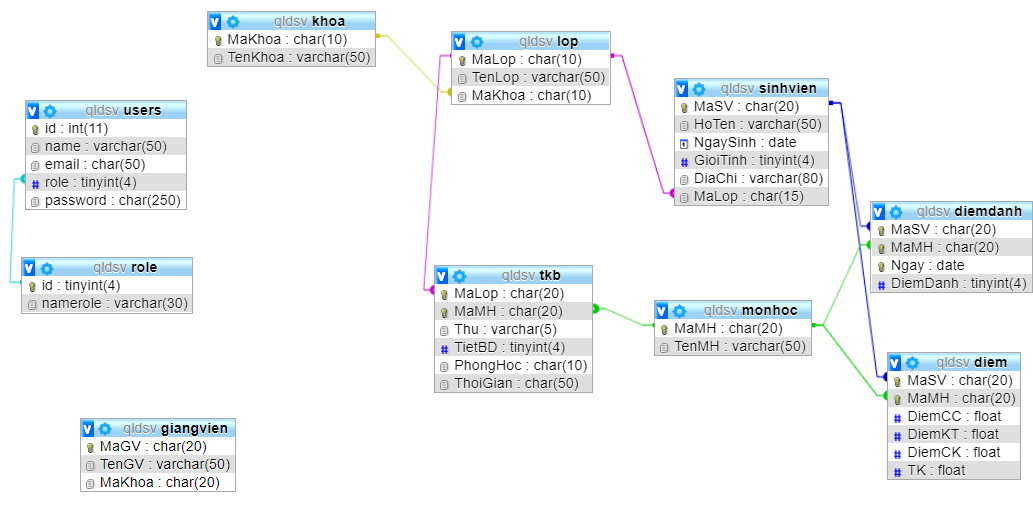


## **3.7. JSONArrayRequest Volley**

Tương tự như JSONObjectRequest nhưng JSONArrayRequest trả về một JSONArray nếu thành công trong phương thức onResponse.



1. **Phân tích và thiết kế cơ sở dữ liệu**
2. **Diagram**



1. **Phân tích và thiết kế**
   1. **Đề bài :** Quản lý sinh viên.
   2. **Yêu cầu**

* Giảng viên: Nhập điểm, điểm danh.
* Sinh viên: Xem điểm, xem thời khóa biểu, xem thông báo.
* Giáo vụ: Cập nhật thời khóa biểu, thông báo realtime, điểm, quản lý thông tin các sinh viên.
* Các chức năng chính đối : Hiển thị giao diện và dữ liệu, các hành động thêm , xóa , hiệu chỉnh dữ liệu.
* Dựa vào yêu cầu bài toán ta có thiết kế database như sau
* Vì có đến 3 đối tượng là giảng viên, sinh viên, giáo vụ nên ta sẽ có màn hình đăng nhập với username và password được sử dụng trong bảng users. Có 3 đối tượng ta sẽ phân quyền cho từng users sử dụng bảng role liên kết với bảng users.
* Trong bảng users, username là mã sinh viên, mã giảng viên hoặc phòng giáo vụ để có thể biết đang là ai đăng nhập vào ứng dụng.
  1. **Phân tích, thiết kế**
* Đối tượng sinh viên.

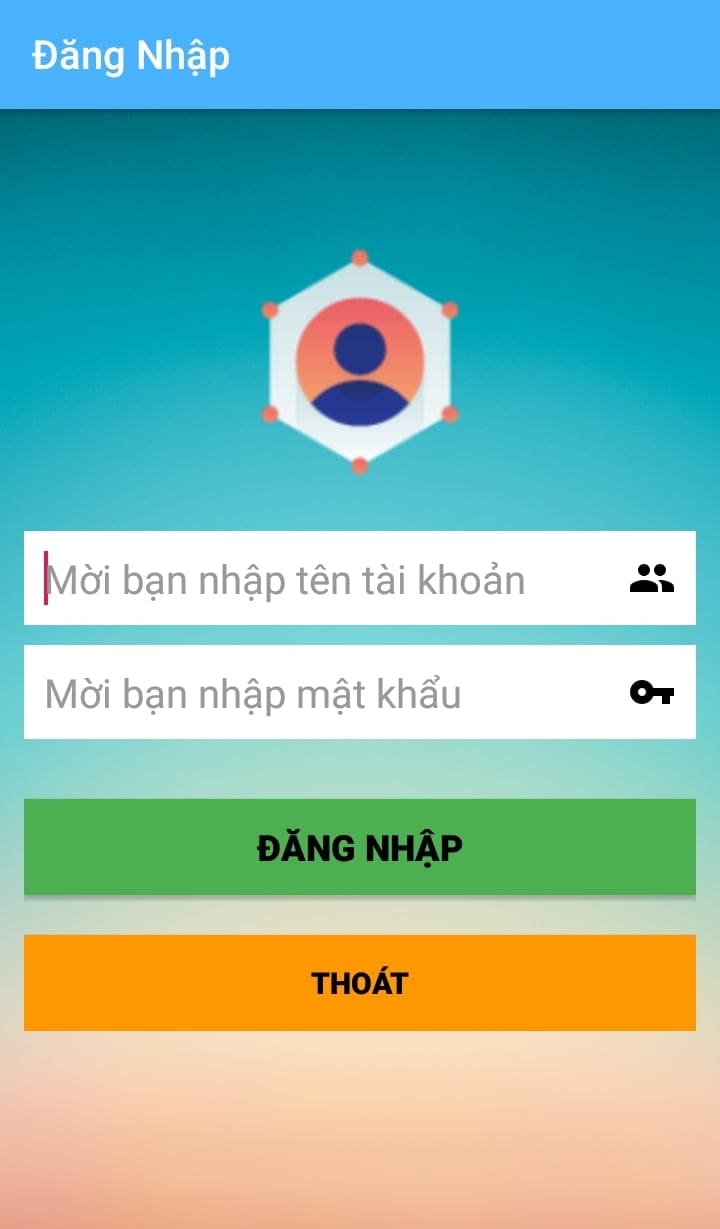
Sinh viên sẽ có thông tin: mã sinh vien, họ tên , ngày sinh , quê quán, giới tính , địa chỉ, sinh viên thuộc lớp nào vì vậy ta sẽ có bảng sinh viên với những thành phần trên và thêm MaLop để biết của lớp nào. Vì vậy ta sẽ cần bảng Lớp , và lớp thuộc một khoa nên ta sẽ có thêm bảng Khoa. Một khoa có nhiều lớp , một lớp có nhiều sinh viên nên bảng Lớp liên kết 1 – nhiều với bảng SinhViên , và bảng Khoa liên kết 1 – nhiều với bảng Lớp.

Một sinh viên sẽ có thêm điểm, điểm danh , thời khóa biểu nên ta có bảng Diem, bảng DiemDanh và bảng TKB. Một sinh viên có nhiều lần điểm danh và nhiều điểm nên bảng SinhVien sẽ liên kết 1- nhiều với bảng DiemDanh sẽ liên kết 1- nhiều với bảng Điểm.

Sinh viên của lớp nên sẽ liên kết 1 – nhiều với bảng TKB.

* Đối tượng giảng viên: có bảng GiangVien đi kèm
* Đối tượng giáo vụ: có nhiệm vụ thông báo realtime nên ta sử dụng firebase để thông báo.

1. **Phân tích và thiết kế hệ thống ứng dụng**
   1. **Màn hình Đăng nhập**
   2. **Giao diện**

****

* 1. **SRS**
* Khi chạy chương trình lên sẽ xuất hiện màn hình với hiệu ứng để giới thiệu ứng dụng có tên : “QUẢN LÝ SINH VIÊN”.
* Sau 4s, sẽ xuất hiện màn hình Đăng Nhập với 2 ô text box yêu cầu nhập tài khoản và mật khẩu.
* Khi nhập xong tài khoản, mật khẩu bấm “ĐĂNG NHẬP”, tùy vào tài khoản thuộc quyền nào để hiển thị ra trang chủ tương ứng. Nếu không muốn dử dụng chương trình nữa thì bấm nút “THOÁT”.
  1. **SDS**

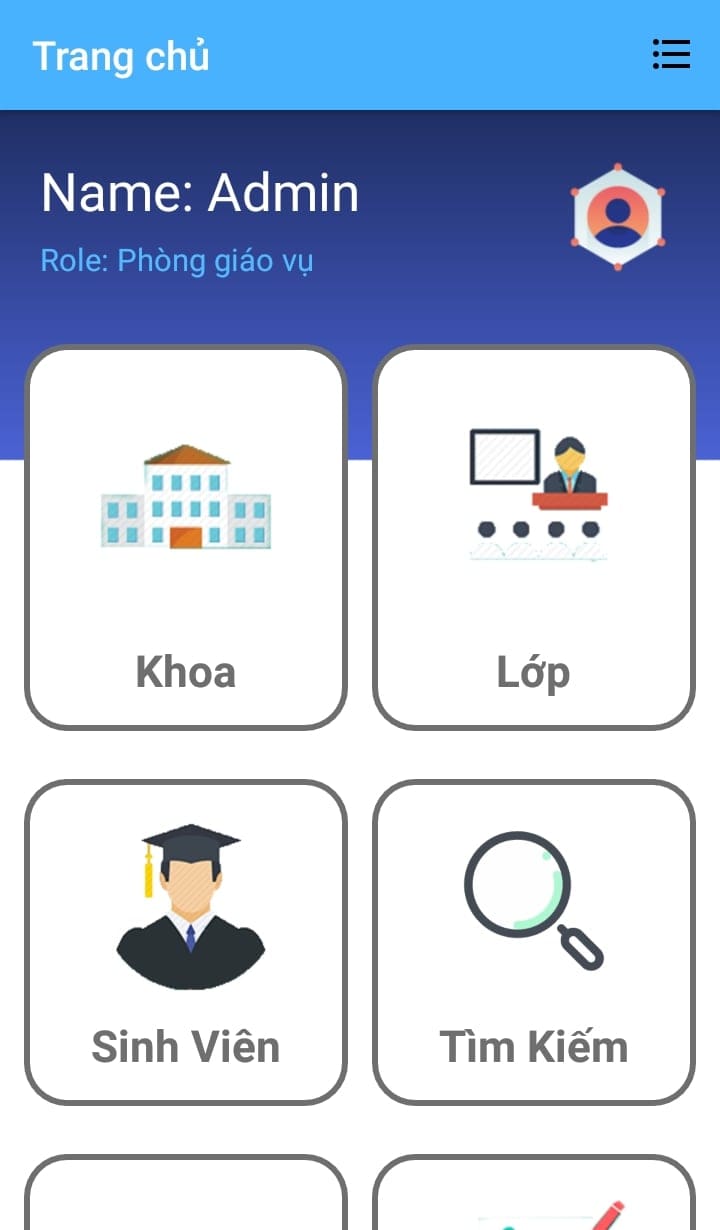
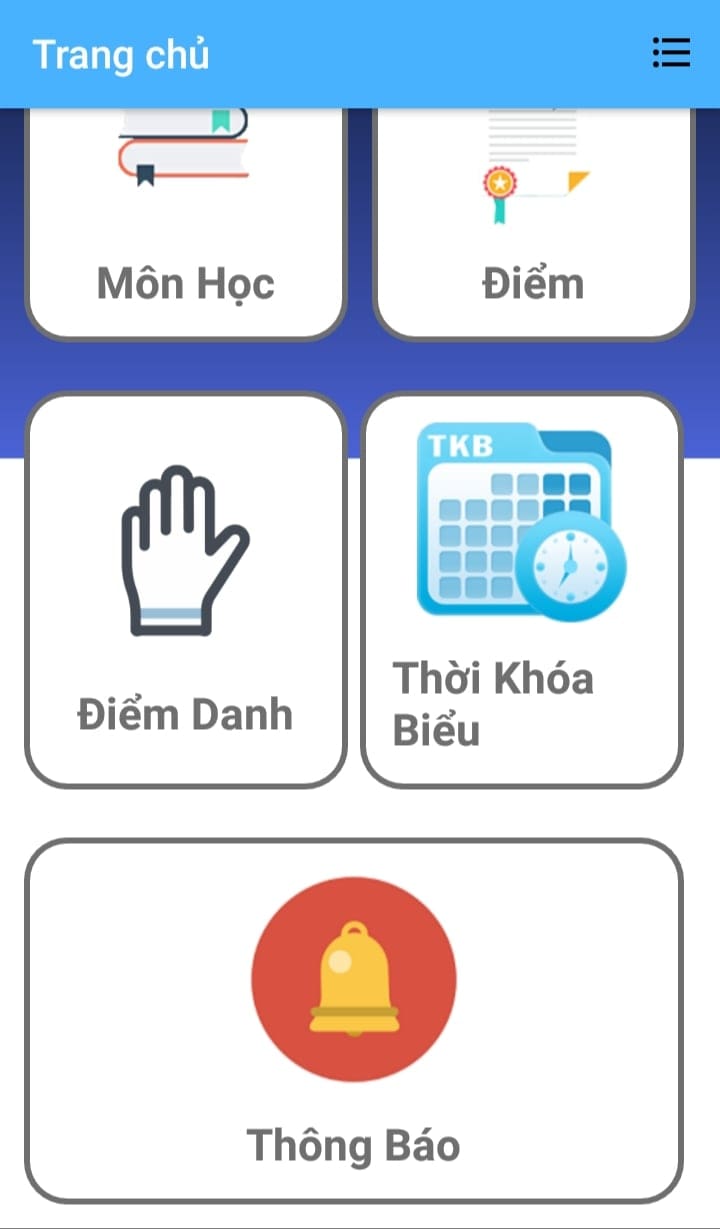
- Nhập vào username và password, kiểm tra username và password không được trống và gửi lên service.

- Trên service sẽ kiểm tra username và password có giống trong database. Nếu giống sẽ trả về quyền của user đó. Nếu không sẽ trả về nhắc nhở kiểm tra lại tài khoản mật khẩu.

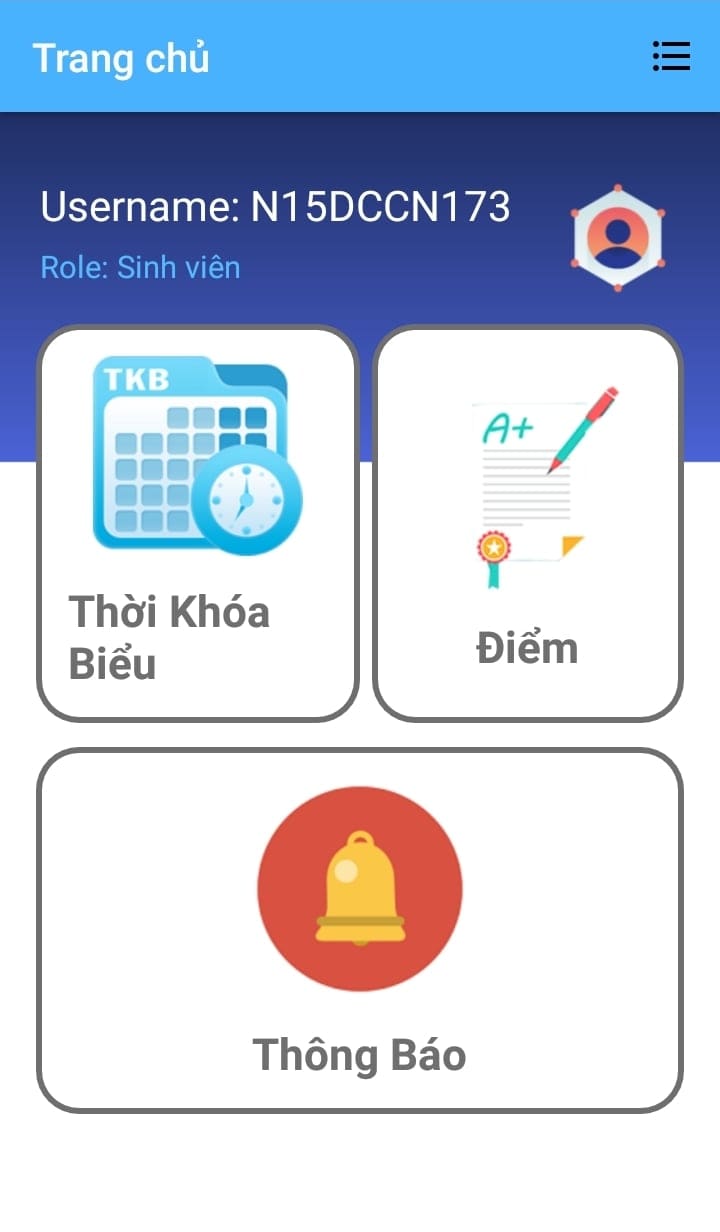
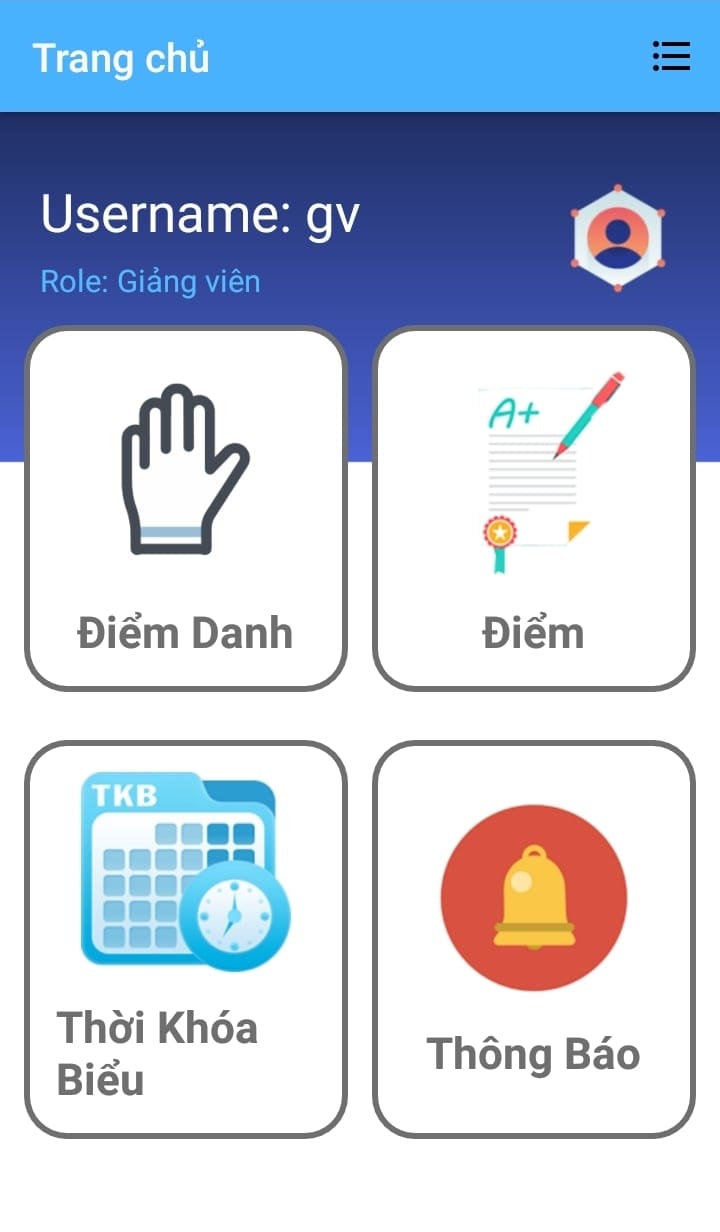
- Khi nhận được kết quả: Nếu nhận được quyền nào thì sẽ chuyển trang theo quyền được nhận: quyền sinhvien sẽ chuyển sang homesv, quyền admin sẽ chuyển sang trang home, quyền giangvien sẽ chuyển sang trang homegv. Nếu không trùng quyền nào thì sẽ giữ nguyên trang login và hiển thị thông báo lỗi khi đăng nhập.

* 1. **Màn hình Home**
  2. **Giao diện**

1. **Quyền Phòng giáo vụ**

**** ****

1. **Quyền Giảng Viên và Sinh Viên**

****

* 1. **SRS**

1. **Đối với quyền Phòng giáo vụ**

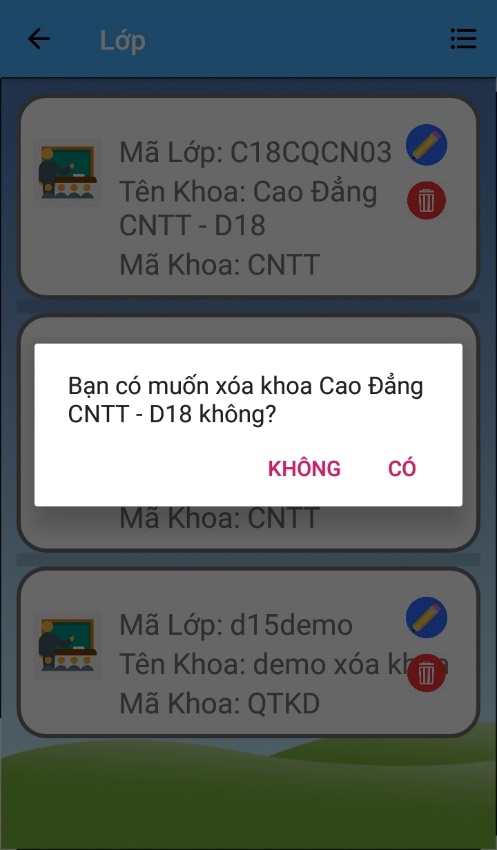
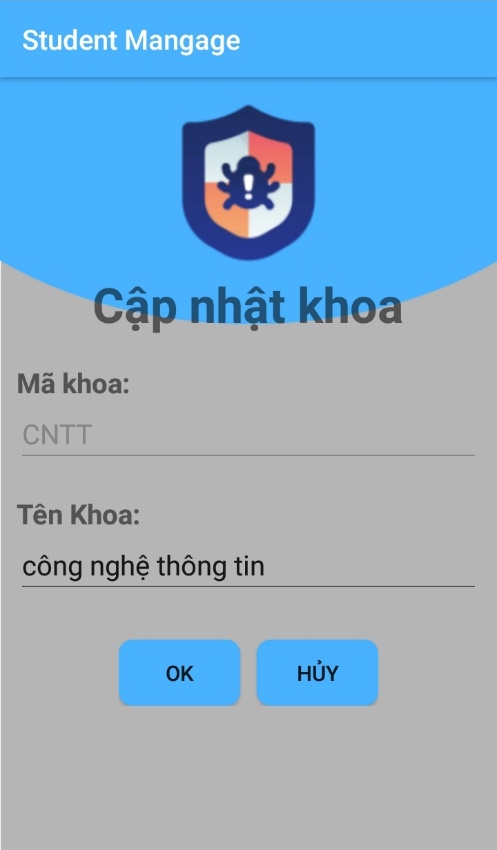
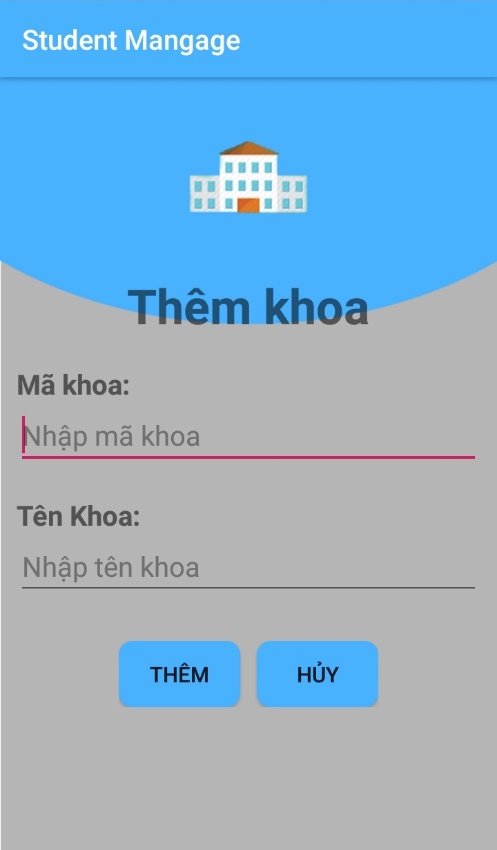
* Sẽ hiển thị user với quyền Phòng Giáo vụ phía trên cùng của màn hình.
* Màn hình chính sẽ có các mục : Khoa, Lớp, Sinh Viên , Tìm Kiếm, Môn học, Điểm, Điểm Danh, Thời Khóa Biểu, Thông Báo.
* Khi “click” vào một mục trong giao diện sẽ hiển thị một giao diện mới đúng với mục đã lựa chọn.

1. **Đối với quyền Giáo Viên**

* Sẽ hiển thị user với quyền Giáo Viên phía trên cùng của màn hình.
* Màn hình chính sẽ có các mục : Điểm, Điểm Danh, Thời Khóa Biểu, Thông Báo.
* Khi “click” vào một mục trong giao diện sẽ hiển thị một giao diện mới đúng với mục đã lựa chọn.

1. **Đối với quyền Sinh Viên**

* Sẽ hiển thị user với quyền Sinh Viên phía trên cùng của màn hình.
* Màn hình chính sẽ có các mục : Điểm, Thời Khóa Biểu, Thông Báo.
* Khi “click” vào một mục trong giao diện sẽ hiển thị một giao diện mới đúng với mục đã lựa chọn.
  1. **SDS**
* Từ username và password lúc đăng nhập sẽ tìm ra được quyền của user: Nếu nhận được quyền nào thì sẽ chuyển trang theo quyền được nhận: quyền sinhvien sẽ chuyển sang homesv, quyền admin sẽ chuyển sang trang home, quyền giangvien sẽ chuyển sang trang homegv. Nếu không trùng quyền nào thì sẽ giữ nguyên trang login và hiển thị thông báo lỗi khi đăng nhập.
  1. **Màn hình Khoa**
  2. **Giao diện**

****

* 1. **SRS**
* Màn hình này sẽ hiển thị tất cả các Khoa.
* Phía góc trên bên phải màn hình có 1 menu nhỏ ẩn bên trong, click vào đó sẽ hiện ra 1 menu, có các chức năng : Thêm, trở về Trang chủ và Đăng xuất.
* Khi bấm vào Thêm sẽ hiện ra một màn hình giao diện mới là Thêm Khoa.
* Bên cạnh mỗi Khoa sẽ có 2 icon Sửa và Xóa. Khi click vào icon sẽ hiện ra màn hình giao diện xử lý tương ứng, khi click vào Sửa sẽ hiện ra một màn hình giao diện mới là Cập nhật Khoa, khi click vào Xóa sẽ hiện ra một MessageBox xác nhận xóa Khoa.
  1. **SDS**

1. **Hiển thị dữ liệu**

* Khi vào khoa sẽ gửi yêu cầu lên service
* Trên service sẽ lấy tất cả các khoa và trả về.
* Khi nhận được kết quả:Nhận dữ liệu kiểu json và chuyển sang từng object của khoa và hiển thị 1 list dữ liệu tương ứng lên màn hình.

1. **Chức năng thêm khoa**

- Nhập mã khoa và tên khoa, kiểm tra mã khoa và tên khoa không được trống và gửi lên service.

- Trên service sẽ kiểm tra mã khoa và tên khoa có bị rỗng không , kiểm tra có mã khoa trong database giống với mã khoa vừa nhập không. Nếu mã khoa và tên khoa rỗng thì sẽ trả về phần tử nào chưa được nhập. Nếu mã khoa đã có trong database sẽ trả về mã khoa bị trùng và không cho thêm mới. Nếu mã khoa chưa có trong database sẽ cho thêm một khoa mới với mã khoa và tên vừa nhập trả về thêm thành công.

- Khi nhận được kết quả: Nếu nhận được quyền nào thì sẽ chuyển trang theo quyền được nhận: quyền sinhvien sẽ chuyển sang homesv, quyền admin sẽ chuyển sang trang home, quyền giangvien sẽ chuyển sang trang homegv. Nếu không trùng quyền nào thì sẽ giữ nguyên trang login và hiển thị thông báo lỗi khi đăng nhập.

1. **Chức năng sửa khoa**

- Khi nhấp vào 1 khoa sẽ nhận được dữ liệu tương ứng của khoa đó là mã khoa và tên khoa đó.

- Tiến hành sửa tên khoa, tên khoa không được trống và gửi lên service, mã khoa không được sửa.

- Trên service sẽ kiểm tra mã khoa và tên khoa có bị rỗng không. Nếu mã khoa và tên khoa rỗng thì sẽ trả về phần tử nào chưa được nhập. Nếu mã khoa không rỗng : tìm kiếm khoa với mã khoa tương ứng và sửa dữ liệu của khoa đó. Nếu sửa thành công sẽ trả về sửa khoa thành công. Nếu không thành công sẽ trả về thông báo xảy ra lỗi.

- Khi nhận được kết quả: Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới. Nếu không thành công sẽ hiển thị thông báo lỗi.

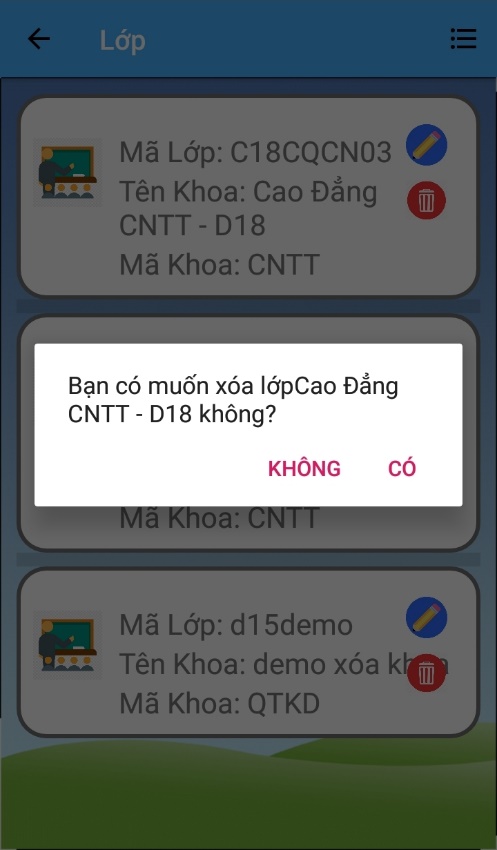
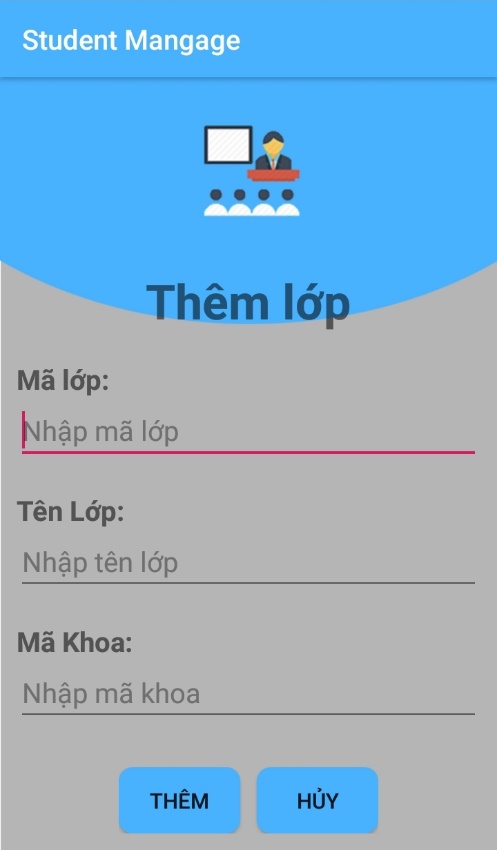
1. **Chức năng xóa khoa**

- Khi nhấp vào xóa 1 khoa, sẽ gửi mã khoa lên service.

- Trên service sẽ kiểm tra mã khoa gửi lên có trống không. Nếu mã khoa rỗng thì sẽ trả về mã khoa rỗng. Nếu không rỗng thì kiểm tra có mã có tồn tại không. Nếu mã khoa không tồn tại sẽ trả về thông báo mã không tồn tại. Nếu mã khoa tồn tại thì sẽ kiểm tra có lớp nào thuộc khoa này không. Nếu có lớp tồn tại trong khoa này sẽ không cho xóa và gửi thông báo không được xóa. Trong trường hợp không có lớp nào thuộc khoa này sẽ tìm khoa với mã khoa được nhận và tiến hành xóa. Nếu xóa thành công sẽ trả về thông báo xóa thành công. Nếu không thành công sẽ trả về thông báo lỗi.

- Khi nhận được kết quả sẽ hiển thị thông báo và chạy lại dữ liệu.

* 1. **Màn hình Lớp**
  2. **Giao diện**

****

* 1. **SRS**
* Màn hình này sẽ hiển thị tất cả các Lớp.
* Phía góc trên bên phải màn hình có 1 menu nhỏ ẩn bên trong, click vào đó sẽ hiện ra 1 menu, có các chức năng : Thêm, trở về Trang chủ và Đăng xuất.
* Khi bấm vào Thêm sẽ hiện ra một màn hình giao diện mới là Thêm Lớp.
* Bên cạnh mỗi Lớp sẽ có 2 icon Sửa và Xóa. Khi click vào icon sẽ hiện ra màn hình giao diện xử lý tương ứng, khi click vào Sửa sẽ hiện ra một màn hình giao diện mới là Cập nhật Lớp, khi click vào Xóa sẽ hiện ra một MessageBox xác nhận xóa Lớp.
  1. **SDS**

1. **Hiển thị dữ liệu**

* Khi vào lớp sẽ gửi yêu cầu lên service
* Trên service sẽ lấy tất cả các lớp và trả về.
* Khi nhận được kết quả: Nhận dữ liệu kiểu json và chuyển sang từng object của lớp và hiển thị 1 list dữ liệu tương ứng lên màn hình.

1. **Chức năng thêm Lớp**

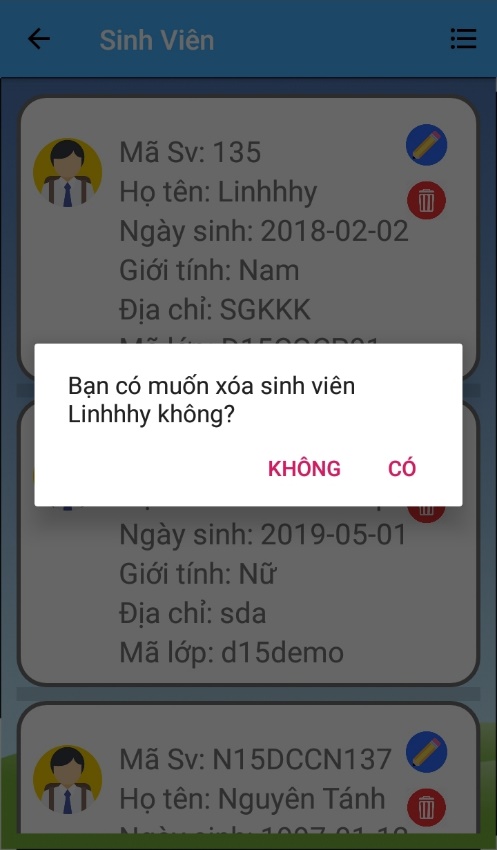
* Nhập mã lớp, tên lớp và mã khoa gửi lên service.
* Trên service sẽ kiểm tra mã lớp, tên lớp và mã khoa có bị rỗng không . Nếu mã lớp ,tên lớp và mã khoa rỗng thì sẽ trả về phần tử nào chưa được nhập. Trường hợp không rỗng sẽ kiểm tra có mã lớp có tồn tại trong cơ sở dữ liệu. Dữ liệu đã tồn tại thì trả về thông báo đã tồn tại , còn không tồn tại thì kiểm tra mã khoa vừa nhập có tồn tại hay không. Nếu mã khoa không tồn tại thì trả về thông báo tương ứng. Trong trường hợp mã khoa tồn tại thì sẽ cho thêm lớp với dữ liệu gửi lên. Thêm thành công trả về thông báo thành công. Nếu bị lỗi sẽ trả về thông báo xảy ra lỗi
* Khi nhận được kết quả: Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới. Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.

1. **Chức năng sửa Lớp**

* Khi nhấp vào 1 lớp sẽ nhận được dữ liệu tương ứng của lớp đó là mã lớp, tên lớp và mã khoa đó.
* Tiến hành sửa tên lớp ,mã khoa và không được trống ,gửi lên service, mã lớp không được sửa.
* Trên service sẽ kiểm tra mã lớp,tên lớp và mã khoa có bị rỗng không. Nếu mã khoa ,mã lớp và tên lớp rỗng thì sẽ trả về phần tử nào chưa được nhập. Trong trường hợp mã khoa ,mã lớp và tên lớp không rỗng: kiểm tra mã khoa khi sửa có tồn tại không. Nếu mã khoa không tồn tại : trả về thông báo mã khoa không tồn tại. Nếu mã khoa tồn tại : tìm kiếm lớp với mã lớp tương ứng và sửa dữ liệu của lớp đó. Sửa thành công sẽ trả về sửa khoa thành công. Không thành công sẽ trả về thông báo xảy ra lỗi.
* Khi nhận được kết quả: Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới . Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.

1. **Chức năng xóa Lớp**

* Khi nhấp vào xóa 1 lớp, sẽ gửi mã lớp lên service.
* Trên service sẽ kiểm tra mã lớp gửi lên có trống hay không? Nếu mã lớp rỗng thì sẽ trả về mã lớp rỗng. Trong trường hợp mã lớp không rỗng thì kiểm tra có mã lớp đó có tồn tại không. Mã lớp không tồn tại sẽ trả về thông báo mã không tồn tại. Ngược lại, mã lớp tồn tại thì sẽ kiểm tra có sinh viên nào thuộc lớp này không. Nếu có sinh viên tồn tại trong khoa này sẽ không cho xóa và gửi thông báo không được xóa. Nếu không có sinh viên nào thuộc lớp này sẽ tìm lớp với mã lớp được nhận và tiến hành xóa. Khi xóa thành công sẽ trả về thông báo xóa thành công. Còn nếu không thành công sẽ trả về thông báo lỗi.
* Khi nhận được kết quả: Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới. Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.
  1. **Màn hình Sinh Viên**
  2. **Giao diện**

****

* 1. **SRS**
* Màn hình này sẽ hiển thị tất cả các Sinh Viên.
* Phía góc trên bên phải màn hình có 1 menu nhỏ ẩn bên trong, click vào đó sẽ hiện ra 1 menu, có các chức năng : Thêm, trở về Trang chủ và Đăng xuất.
* Khi bấm vào Thêm sẽ hiện ra một màn hình giao diện mới là Thêm Sinh Viên.
* Bên cạnh mỗi Sinh Viên sẽ có 2 icon Sửa và Xóa. Khi click vào icon sẽ hiện ra màn hình giao diện xử lý tương ứng, khi click vào Sửa sẽ hiện ra một màn hình giao diện mới là Cập nhật Sinh Viên, khi click vào Xóa sẽ hiện ra một MessageBox xác nhận xóa Sinh Viên.
  1. **SDS**

1. **Hiển thị dữ liệu**

* Khi vào sinh viên sẽ gửi yêu cầu lên service
* Trên service sẽ lấy tất cả các sinh viên và trả về.
* Khi nhận được kết quả:Nhận dữ liệu kiểu json và chuyển sang từng object của sinh viên và hiển thị 1 list dữ liệu tương ứng lên màn hình.

1. **Chức năng thêm Sinh Viên**

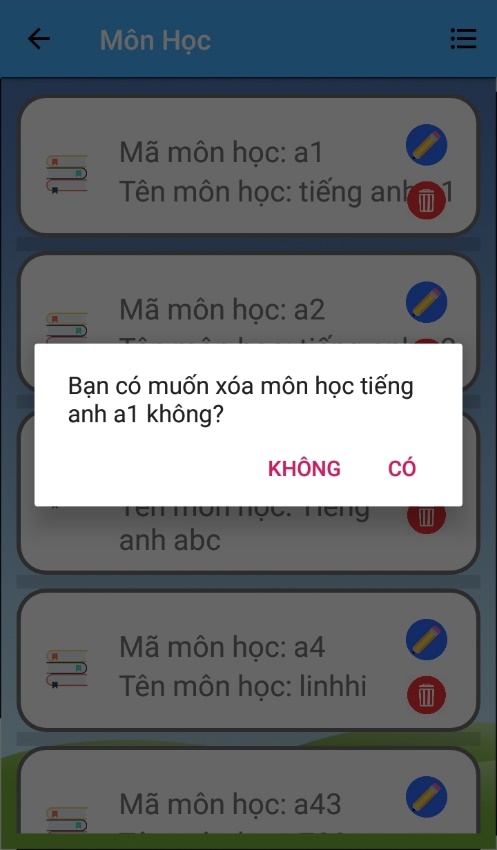
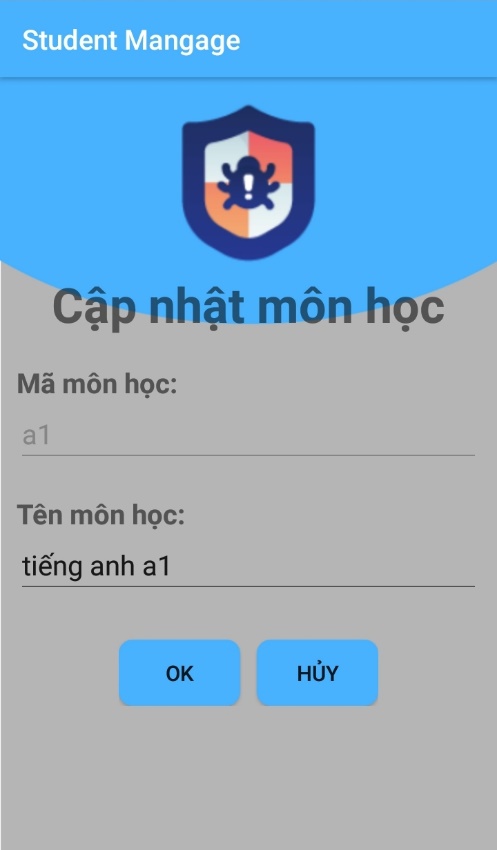
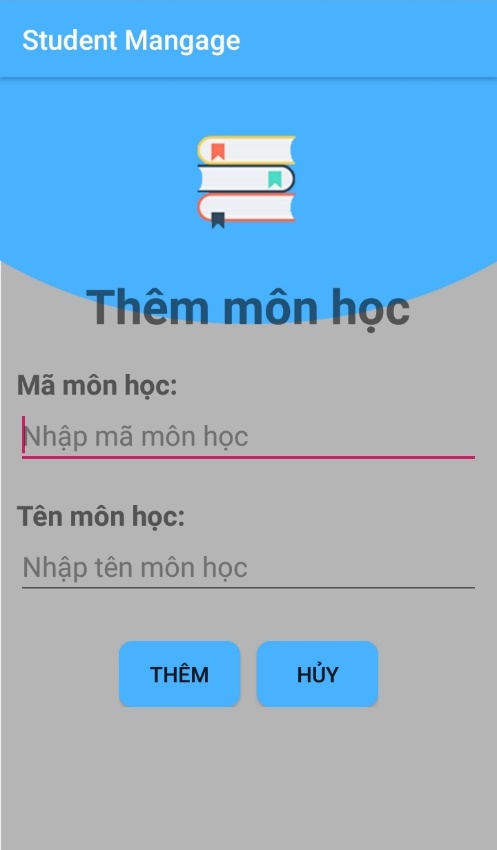
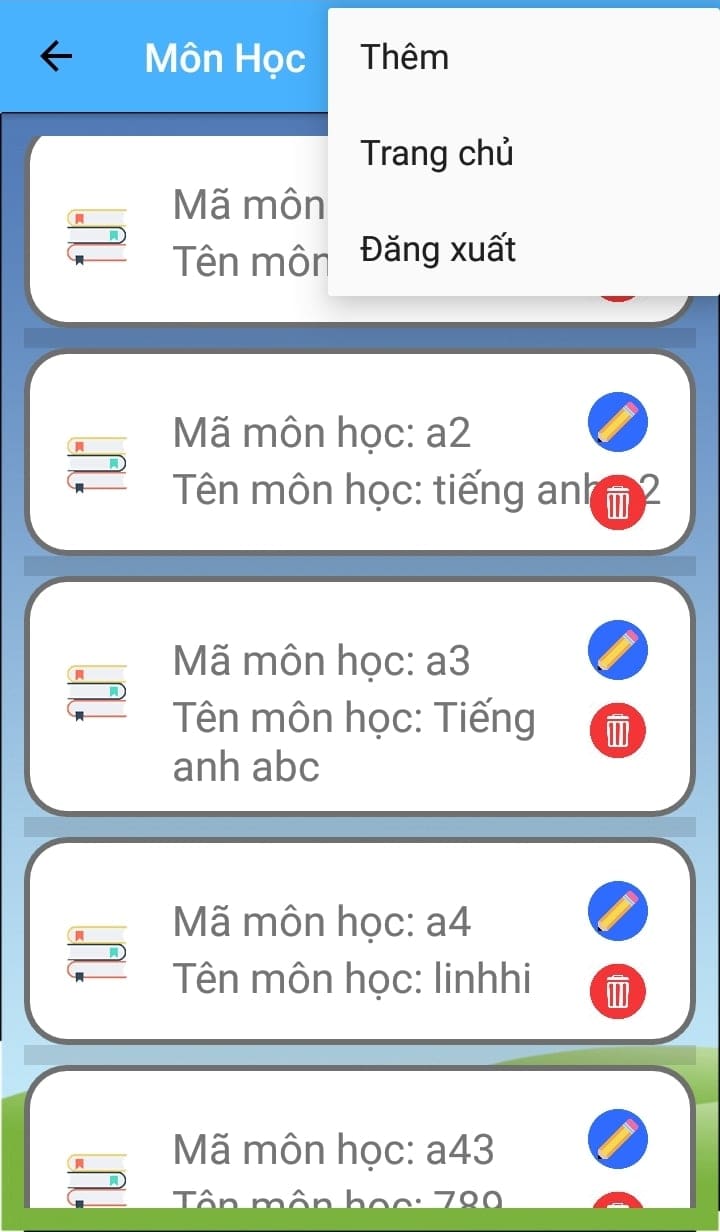
* Nhập mã sinh viên, tên sinh viên,ngày sinh,giới tính , địa chỉ và mã lớp gửi lên service
* Trên service sẽ kiểm tra dữ liệu gửi lên có bị rỗng không, khi có phần tử rỗng thì sẽ trả về phần tử nào chưa được nhập, dữ liệu không rỗng sẽ kiểm tra có mã sinh viên có tồn tại trong cơ sở dữ liệu. Nếu dữ liệu đã tồn tại thì trả về thông báo đã tồn tại . Nếu dữ liệu không tồn tại thì kiểm tra mã lớp vừa nhập có tồn tại hay không. Kiểm tra mã lớp không tồn tại thì trả về thông báo tương ứng. Nếu mã lớp tồn tại thì sẽ cho thêm sinh viên với dữ liệu gửi lên. Khi thêm thành công trả về thông báo thành công. Nếu bị lỗi sẽ trả về thông báo xảy ra lỗi
* Khi nhận được kết quả: Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới. Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.

1. **Chức năng sửa Sinh Viên**
   * + Khi nhấp vào 1 sinh viên sẽ nhận được dữ liệu tương ứng của sinh viên đó.
     + Tiến hành sửa tên sinh viên,ngày sinh,giới tính , địa chỉ ,mã lớp và không được trống ,gửi lên service, mã sinh viên không được sửa.
     + Trên service sẽ kiểm tra dữ liệu gửi lên có bị rỗng không. Khi dữ liệu rỗng thì sẽ trả về phần tử nào chưa được nhập, khi dữ liệu không rỗng: kiểm tra mã sinh viên khi sửa có tồn tại không, nếu mã sinh viên không tồn tại: trả về thông báo sinh viên không tồn tại , nếu mã sinh viên tồn tại : tìm kiếm sinh viên với mã sinh viên tương ứng và sửa dữ liệu của lớp đó. Nếu sửa thành công sẽ trả về thành công, còn không thành công sẽ trả về thông báo xảy ra lỗi.

* Khi nhận được kết quả: Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới. Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.

1. **Chức năng xóa Sinh Viên**

* Khi nhấp vào xóa 1 sinh viên, sẽ gửi mã sinh viên lên service.
* Trên service sẽ kiểm tra mã sinh viên gửi lên có trống không? Nếu mã sinh viên rỗng thì sẽ trả về mã sinh viên rỗng, còn không rỗng thì kiểm tra có mã có tồn tại không. Nếu mã sinh viên không tồn tại sẽ trả về thông báo mã không tồn tại. Trong trường hợp mã sinh viên tồn tại thì sẽ kiểm tra có sinh viên có điểm môn nào. Sinh viên có điểm môn học sẽ không cho xóa và gửi thông báo không được xóa, còn sinh viên chưa có điểm môn học sẽ tìm sinh viên với mã sinh viên được nhận và tiến hành xóa. Nếu xóa thành công sẽ trả về thông báo xóa thành công, nếu không thành công sẽ trả về thông báo lỗi.
* Khi nhận được kết quả: Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới. Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.
  1. **Màn hình Môn Học**
  2. **Giao diện**

****

* 1. **SRS**
* Màn hình này sẽ hiển thị tất cả các Môn học.
* Phía góc trên bên phải màn hình có 1 menu nhỏ ẩn bên trong, click vào đó sẽ hiện ra 1 menu, có các chức năng : Thêm, trở về Trang chủ và Đăng xuất.
* Khi bấm vào Thêm sẽ hiện ra một màn hình giao diện mới là Thêm Môn học.
* Bên cạnh mỗi Môn học sẽ có 2 icon Sửa và Xóa. Khi click vào icon sẽ hiện ra màn hình giao diện xử lý tương ứng, khi click vào Sửa sẽ hiện ra một màn hình giao diện mới là Cập nhật Môn học, khi click vào Xóa sẽ hiện ra một MessageBox xác nhận xóa Môn học.
  1. **SDS**

1. **Hiển thị dữ liệu**

* Khi vào môn học sẽ gửi yêu cầu lên service
* Trên service sẽ lấy tất cả các môn học và trả về.
* Khi nhận được kết quả: Nhận dữ liệu kiểu json và chuyển sang từng object của môn học và hiển thị 1 list dữ liệu tương ứng lên màn hình.

1. **Chức năng thêm Môn Học**

- Nhập mã môn học và tên môn học, kiểm tra mã môn học và tên môn học không được trống và gửi lên service.

- Trên service sẽ kiểm tra mã môn học và tên môn học có bị rỗng không nếu mã môn học và tên môn học rỗng thì sẽ trả về phần tử nào chưa được nhập kiểm tra có mã môn học trong database giống với mã môn học vừa nhập không.Nếu mã môn học đã có trong database sẽ trả về mã môn học bị trùng và không cho thêm mới. nếu mã môn học chưa có trong database sẽ cho thêm một môn học mới với mã môn học và tên vừa nhập trả về thêm thành công.Nếu thêm thành công sẽ trả về thông báo thành công. Nếu không thành công sẽ trả về thông báo lỗi.

- Khi nhận được kết quả:nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới.Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.

1. **Chức năng sửa Môn Học**

- Khi nhấp vào 1 môn học sẽ nhận được dữ liệu tương ứng của môn học đó là mã môn học và tên môn học đó.

- Tiến hành sửa tên môn học, tên môn học không được trống và gửi lên service, mã môn học không được sửa.Trên service sẽ kiểm tra mã môn học và tên môn học có bị rỗng không .Nếu mã môn học và tên môn học rỗng thì sẽ trả về phần tử nào chưa được nhập. Nếu mã môn học không rỗng : tìm kiếm môn học với mã môn học tương ứng và sửa dữ liệu của môn học đó. Nếu sửa thành công sẽ trả về sửa môn học thành công. Nếu không thành công sẽ trả về thông báo xảy ra lỗi.

- Khi nhận được kết quả:Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới.Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.

1. **Chức năng xóa Môn Học**

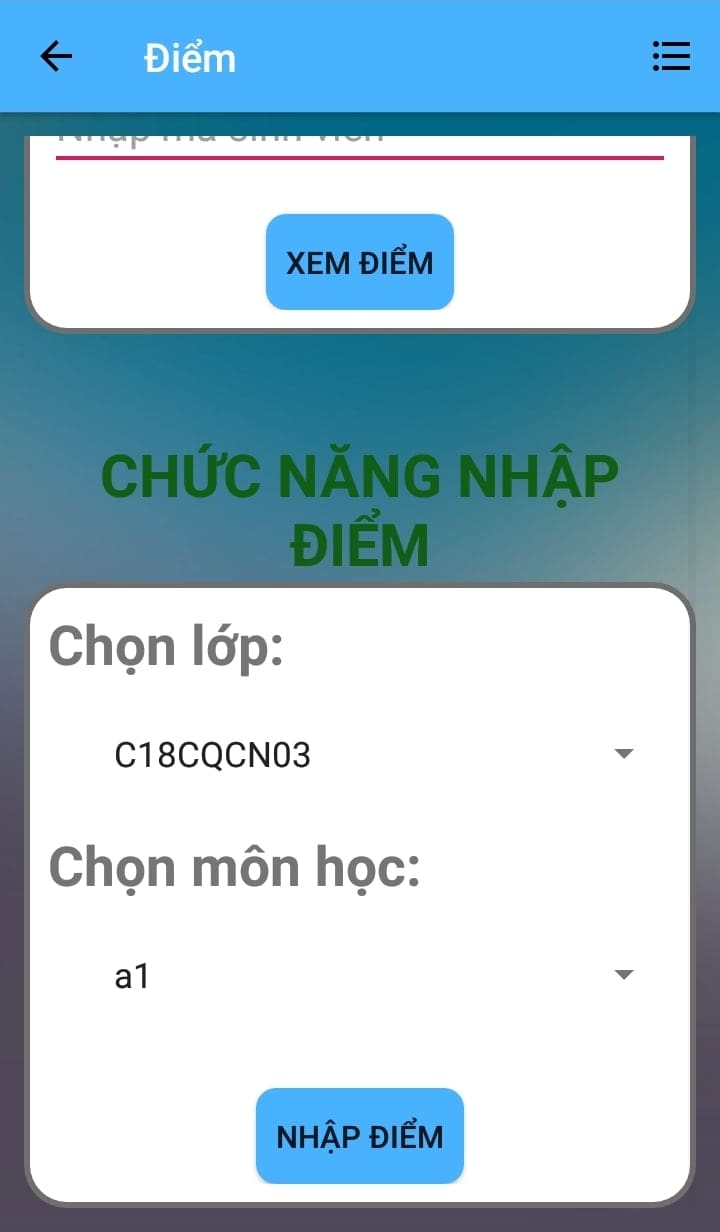
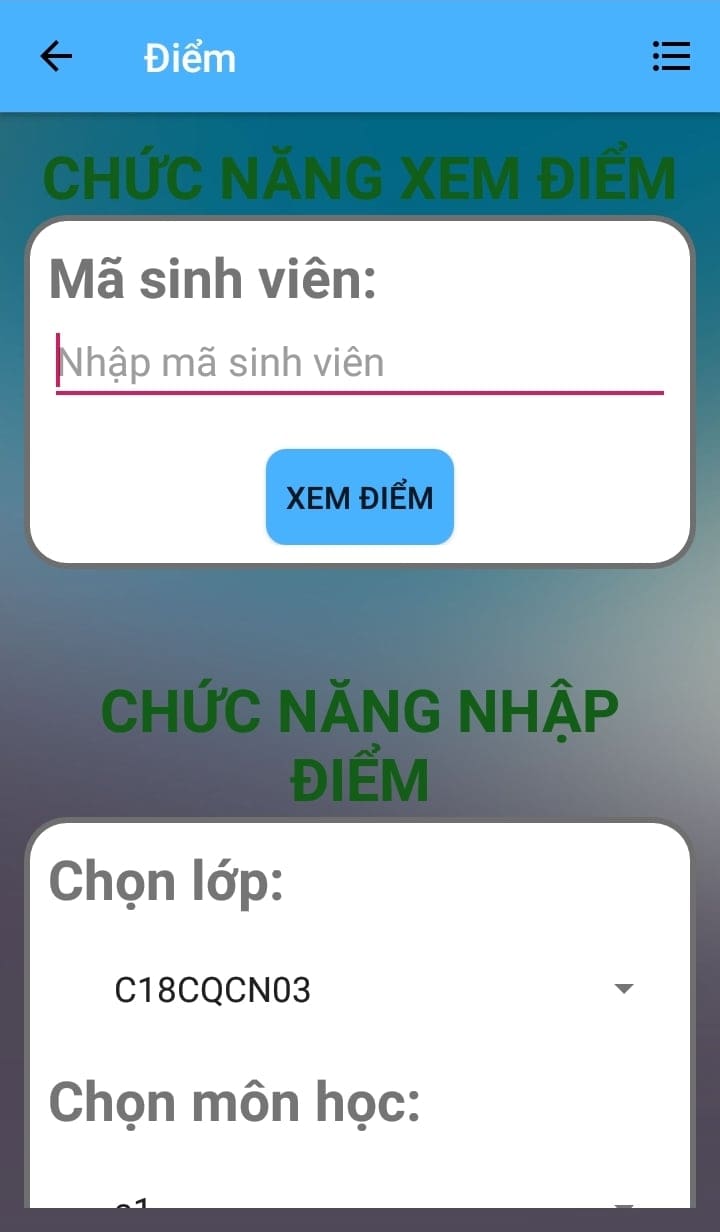
- Khi nhấp vào xóa 1 môn học, sẽ gửi mã môn học lên service.

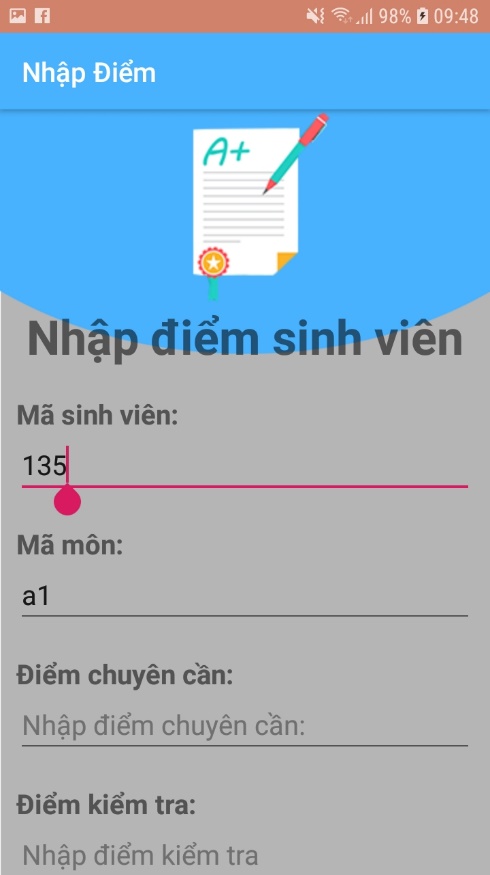
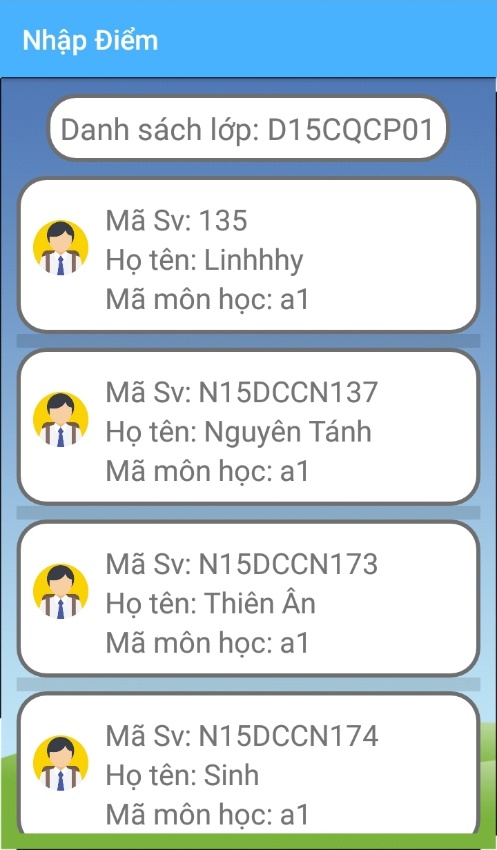
-Trên service sẽ kiểm tra mã môn học gửi lên có trống không? Nếu mã môn học rỗng thì sẽ trả về mã môn học rỗng.Nếu không rỗng thì kiểm tra có mã có tồn tại không.Nếu mã môn học không tồn tại sẽ trả về thông báo mã không tồn tại. Nếu mã môn học tồn tại thì sẽ kiểm tra có sinh viên nào có điểm môn học này không. Nếu có sinh viên có điểm môn học này sẽ không cho xóa và gửi thông báo không được xóa.Nếu không có sinh viên nào có điểm môn học này sẽ tìm môn học với mã môn học được nhận và tiến hành xóa. Nếu xóa thành công sẽ trả về thông báo xóa thành công. Nếu không thành công sẽ trả về thông báo lỗi.

- Khi nhận được kết quả:Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới. Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.

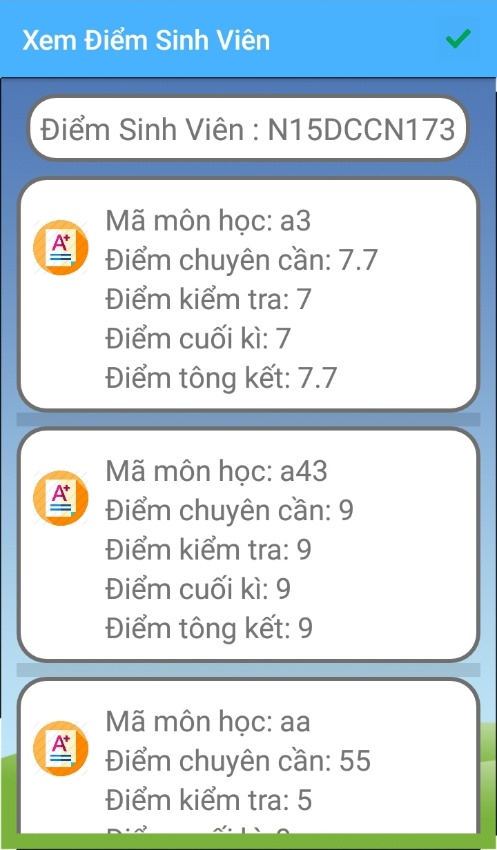
* 1. **Màn hình Điểm**
  2. **Giao diện**

1. **Đối với Giáo Viên**

****



1. **Đối với Sinh Viên**

****

* 1. **SRS**

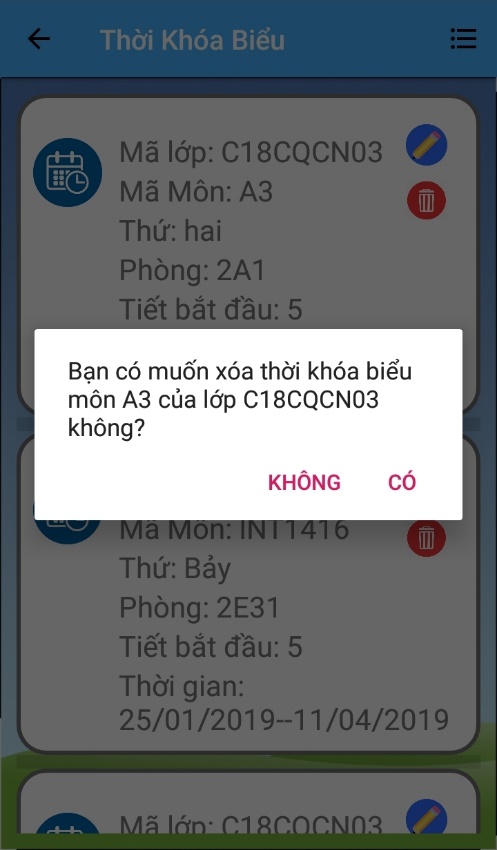
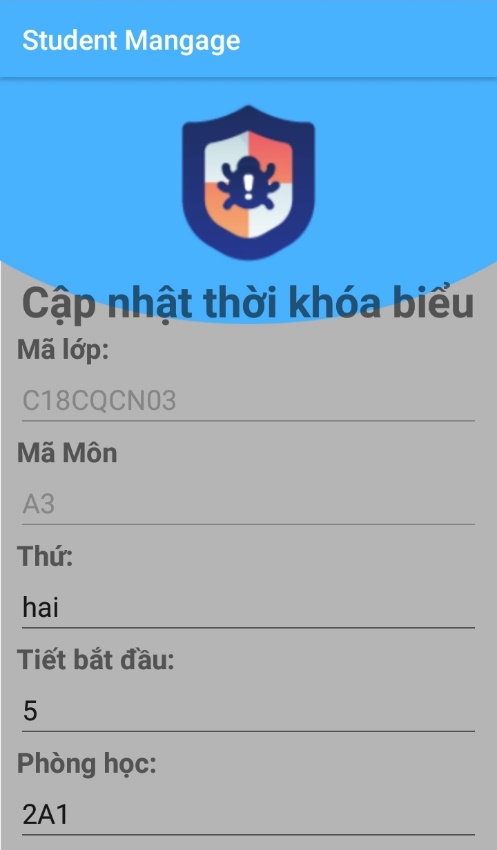
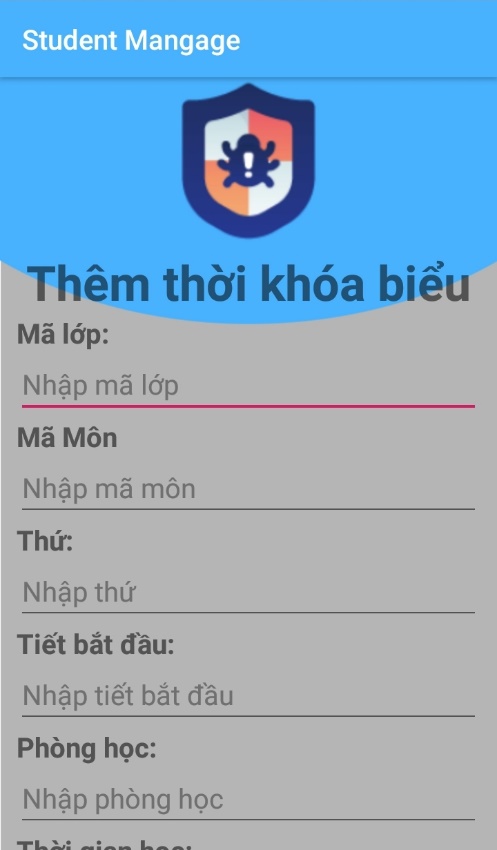
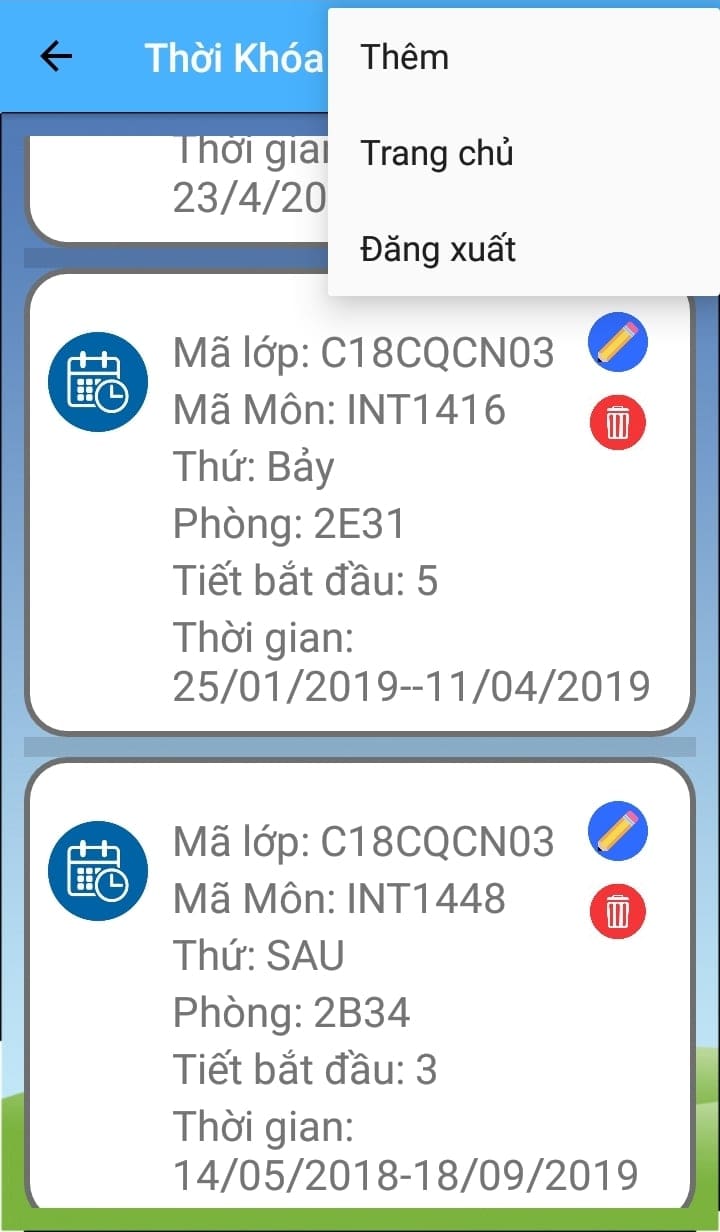
1. **Đối với quyền Giáo Viên**

* Màn hình sẽ cho phép người dùng xem điểm của bất kì sinh viên nào bằng cách nhập vào MSSV và cho phép lựa chọn lớp muốn nhập điểm. Sau khi chọn được lớp, màn hình sẽ hiện thị danh sách lớp. Người dùng sẽ chọn lớp để nhập điểm cho lớp đó. Sau khi chọn được lớp sẽ chuyển tới màn hình hiển thị danh sách các Sinh viên trong lớp. Khi đã chọn được Sinh Viên muốn nhập điểm sẽ chuyển tới màn hình cho phép nhập điểm cho sinh viên đó. Sau khi nhập xong bấm nút OK để hoàn tất việc nhập điểm.

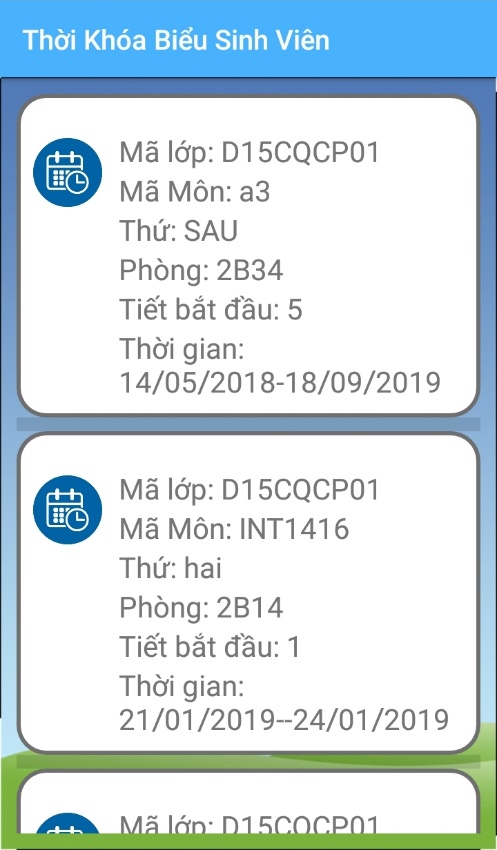
1. **Đối với quyền Sinh Viên**

* Màn hình sẽ hiển thị danh sách điểm của các môn học mà Sinh Viên trong lớp đó học.
  1. **SDS**
* Nhập mã mã sinh viên, mã môn học,điểm chuyên cần, điểm kiểm tra, điểm cuối kỳ,điểm tổng kết, kiểm tra dữ liệu không được trống và gửi lên service.
* Trên service sẽ kiểm tra dữ liệu có bị rỗng không, nếu dữ liệu rỗng thì sẽ trả về phần tử nào chưa được nhập. Nếu dữ liệu đã đầy đủ, kiểm tra sinh viên và môn học có tồn tại không, còn không tồn tại thì sẽ trả về thông báo phần tử nào không tồn tại. Nếu tồn tại rồi thì kiểm tra sinh viên đã có điểm môn này chưa, khi đã có điểm môn này sẽ trả về thông báo, còn chưa có điểm môn, kiểm tra điểm là số và từ 0-10 điểm, không thể là số hoặc <0,>10 sẽ trả về thông báo. Dữ liệu đúng sẽ cho thêm điểm một sinh viên với mã môn học và dữ liệu điểm kèm theo.Nếu thêm thành công sẽ trả về thông báo thành công. Nếu không thành công sẽ trả về thông báo lỗi.
* Khi nhận được kết quả: Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới. Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.
  1. **Màn hình Thời Khóa Biểu**
  2. **Giao diện**

1. **Quyền Phòng giáo vụ và Giáo Viên**

****

1. **Quyền Sinh Viên**

****

* 1. **SRS**

1. **Đối với quyền Phòng Giáo Vụ**

* Màn hình này sẽ hiển thị tất cả các Thời Khóa Biểu.
* Phía góc trên bên phải màn hình có 1 menu nhỏ ẩn bên trong, click vào đó sẽ hiện ra 1 menu, có các chức năng : Thêm, trở về Trang chủ và Đăng xuất.
* Khi bấm vào Thêm sẽ hiện ra một màn hình giao diện mới là Thêm Thời khóa biểu.
* Bên cạnh mỗi Thời khóa biểu sẽ có 2 icon Sửa và Xóa. Khi click vào icon sẽ hiện ra màn hình giao diện xử lý tương ứng, khi click vào Sửa sẽ hiện ra một màn hình giao diện mới là Cập nhật Thời khóa biểu, khi click vào Xóa sẽ hiện ra một MessageBox xác nhận xóa Thời khóa Biểu.

1. **Đối với quyền Giảng Viên và Sinh Viên**

* Màn hình này sẽ hiển Thời Khóa Biểu của chính Giảng Viên và Sinh Viên đăng nhập vô dựa vào mã của Giảng Viên và Sinh Viên đó lúc đăng nhập.
  1. **SDS**

1. **Chức năng thêm thời khóa biểu**

* Nhập mã lớp, mã môn học,thứ,tiết bắt đầu, phòng học và thời gian học học, kiểm tra dữ liệu không được trống và gửi lên service.
* Trên service sẽ kiểm tra dữ liệu có bị rỗng không. Nếu dữ liệu rỗng rỗng thì sẽ trả về phần tử nào chưa được nhập. Kiểm tra lớp đã có thời khóa biểu của môn học này. Đã có thời khóa biểu cho môn này trả về thông báo lớp này đã có lịch học của môn này. Kiểm tra lớp này đã học môn gì vào thứ và tiết được nhập. Nếu có lịch môn khác vào tiết cùng ngày thì sẽ trả về thông báo lớp đã có lịch học. Và sẽ cho thêm một thời khóa biểu mới với mã lớp,mã môn học và các dữ liệu còn lại. Nếu thêm thành công sẽ trả về thông báo thành công. Nếu không thành công sẽ trả về thông báo lỗi.
* Khi nhận được kết quả: Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới. Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.

1. **Chức năng sửa thời khóa biểu**

* Khi nhấp vào 1 thời khóa biểu sẽ nhận được dữ liệu tương ứng của thời khóa biểu đó.
* Sửa lại thông tin mã lớp, mã môn học,thứ,tiết bắt đầu, phòng học và thời gian học học, kiểm tra dữ liệu không được trống và gửi lên service.
* Trên service sẽ kiểm tra dữ liệu có bị rỗng không. Nếu dữ liệu rỗng rỗng thì sẽ trả về phần tử nào chưa được nhập. Kiểm tra lớp đã có thời khóa biểu của môn học này.Đã có thời khóa biểu cho môn này trả về thông báo lớp này đã có lịch học của môn này.Kiểm tra lớp này đã học môn gì vào thứ và tiết được nhập. Nếu có lịch môn khác vào tiết cùng ngày thì sẽ trả về thông báo lớp đã có lịch học và sẽ cho thêm một thời khóa biểu mới với mã lớp,mã môn học và các dữ liệu còn lại. Nếu thêm thành công sẽ trả về thông báo thành công. Nếu không thành công sẽ trả về thông báo lỗi.
* Khi nhận được kết quả: Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới. Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.

1. **Chức năng xóa thời khóa biểu**

* Khi nhấp vào xóa thời khóa biểu một môn của một lớp, sẽ gửi mã lớp và mã môn lên service.
* Trên service sẽ kiểm tra mã lớp và mã môn gửi lên có trống không? Nếu rỗng thì sẽ trả về mã lớp và mã môn rỗng, còn không rỗng thì kiểm tra 2 mã đó có tồn tại không. Nếu mã không tồn tại sẽ trả về thông báo mã không tồn tại và tiến hành xóa.Nếu xóa thành công sẽ trả về thông báo xóa thành công, nếu không thành công sẽ trả về thông báo lỗi.
* Khi nhận được kết quả: Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới. Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.
  1. **Tìm kiếm**
  2. **Giao diện**

****

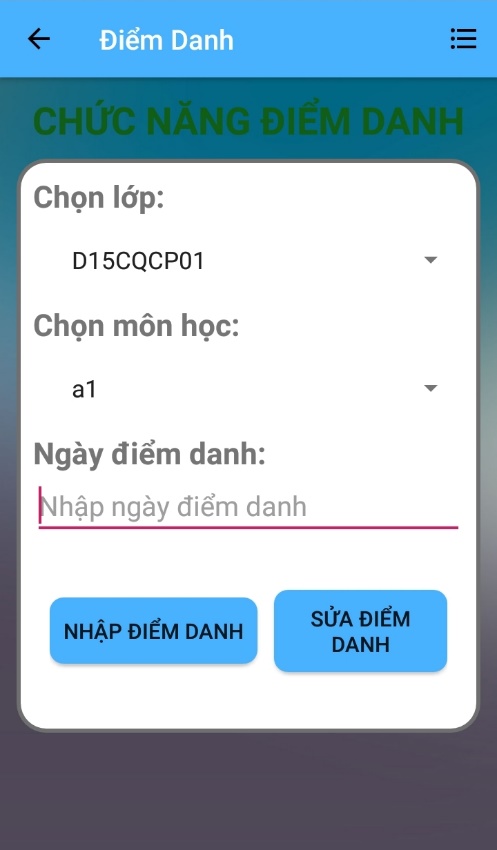
* 1. **SRS**
* Màn hình sẽ hiển thị một thanh tìm kiếm cho phép người dùng nhập bất cứ thứ gì muốn tìm kiếm.
* Sau khi nhập xong sẽ bấm vào nút Tìm, nếu có dữ liệu nào có trong cơ sở dữ liệu sẽ hiển thị ra màn hình dưới dạng Object.
  1. **SDS**

- Nhập bất kỳ dữ liệu gì, kiểm tra dữ liệu không được trống và gửi lên service.

- Trên service sẽ kiểm tra dữ liệu có bị rỗng không, nếu dữ liệu rỗng thì sẽ trả về dữ liệu chưa được nhập, sẽ tìm kiếm tất cả sinh viên có dữ liệu tương ứng với dữ liệu nhập các thông tin như họ tên, mã sinh viên, mã lớp, địa chỉ và trả về kết quả vừa tìm.

- Khi nhận được kết quả: Nhận dữ liệu kiểu json và chuyển sang từng object của sinh viên và hiển thị 1 list dữ liệu tương ứng lên màn hình.

* 1. **Màn hình Điểm danh**
  2. **Giao diện**

****

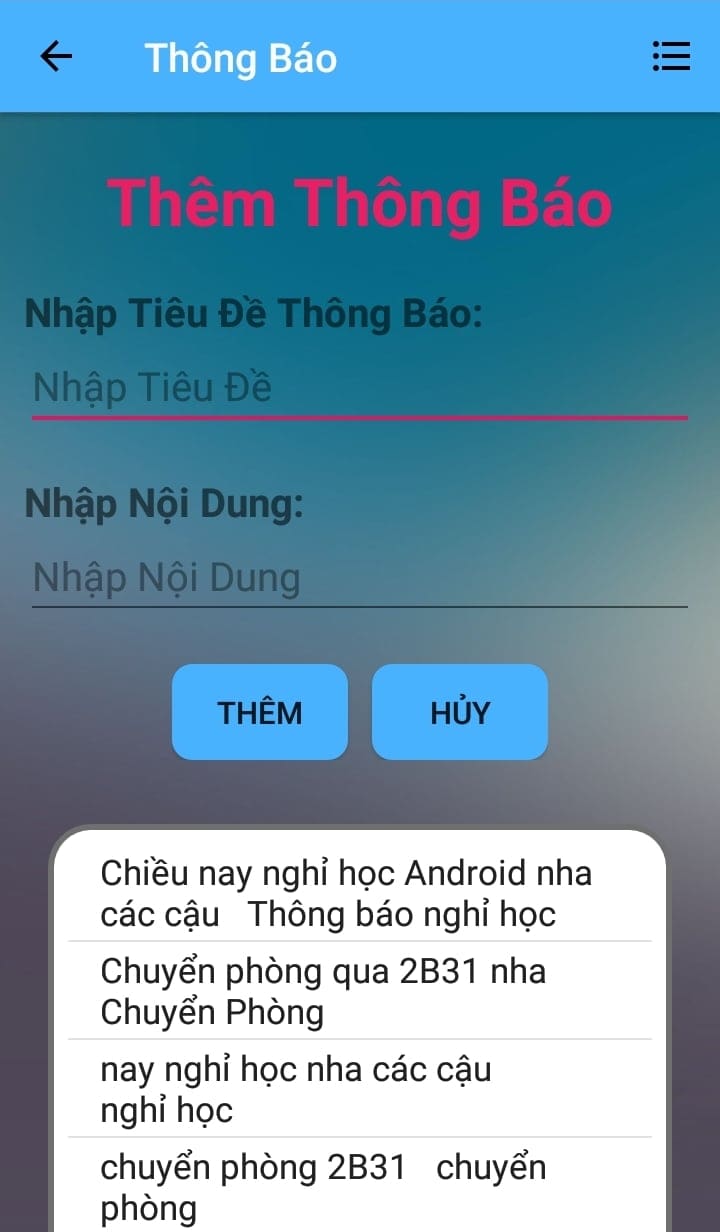
* 1. **SRS**
* Màn hình hiển thị cho phép chọn lớp muốn điểm danh và chọn môn sẽ điểm danh, người dùng cần nhập vào ngày điểm danh. Nếu đã chọn được lớp và môn thì bấm vào “NHẬP ĐIỂM DANH”
* Sau khi bấm vào nút “NHẬP ĐIỂM DANH” sẽ hiển thị danh sách Sinh Viên của lớp với môn đã chọn. Nếu Sinh Viên nào có mặt tại lớp sẽ có một check box bên cạnh và tích vào check box đó.
  1. **SDS**

- Nhập danh sách điểm danh của một lớp với môn học và ngày,kiểm tra dữ liệu không được trống và gửi lên service.

- Trên service sẽ kiểm tra dữ liệu có bị rỗng không, nếu dữ liệu rỗng thì sẽ trả về thông báo dữ liệu chưa được nhập, nếu dữ liệu không rỗng, chạy vòng dữ liệu để lấy điểm danh của từng sinh viên. Tạo điểm danh của 1 sinh viên với mã môn , ngày kèm điểm danh và thêm vào cơ sở dữ liệu. Tiếp tục thêm cho đến hết và trả về thành công. Nếu lỗi trả về lỗi.

- Khi nhận được kết quả: Nếu thành công sẽ hiển thị thông báo thành công, và chạy dữ liệu mới. Nếu không thành công sẽ hiển thị thông báo lỗi,và giữ nguyên trang.

* 1. **Màn hình Thông Báo**
  2. **Giao diện**

****

* 1. **SRS**
* Đối với quyền đăng nhập là Sinh Viên và Giáo Viên có thể xem những thông báo khẩn cấp mà phòng Giáo Vụ đưa ra để cập nhật thông tin nhanh nhất. Đối với quyền giáo vụ cho phép nhập thông báo để gủi tới Sinh Viên và Giáo Viên. Màn hình có ô nhập tiêu đề và Nôi Dung của thông báo, Nếu ấn nút Thêm thì ngay lập tức thông báo sẽ được gửi tới Sinh Viên và Giáo Viên , còn ấn nút hủy có chức năng thoát nhập không thông báo nữa.
* Chức năng: Sinh Viên và Giáo Viên có thể xem thông báo từ phòng Giáo Vụ đưa ra.
  1. **SDS**

1. **Chức năng**

* Realtime Database Firebase là một dịch vụ cơ sở dữ liệu thời gian thực hoạt động trên nền tảng đám mây được cung cấp bởi Google nhằm giúp các lập trình phát triển nhanh các ứng dụng có tương tác cơ sở dữ liệu (CRUD) một cách nhanh chóng và ngay tức thời (Realtime).
* Thực ra ta cũng nên bày vẽ dịch Realtime Database Firebase ra Cơ sơ dữ liệu thời gian thực firebase

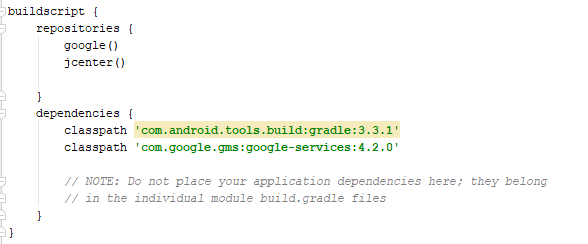
1. **Tạo và cài đặt fire base**

**Implement Path**

* Bước 1: Cài đặt FCM SDK thông qua dependency
* Bước 2: Mở Notification Console soạn và gửi notification tới người dùng
* Bước 3 (Optional) : Xử lý notification trên client-app

### **Cài đặt Notifications SDK trên Android**

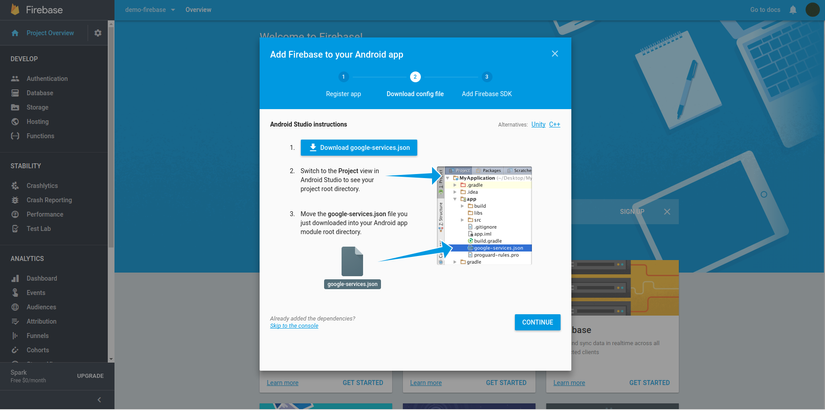
* Trước hết bạn phải cài đặt Firebase SDK vào project. Thêm dependency classpath vào file build.gradle của project.



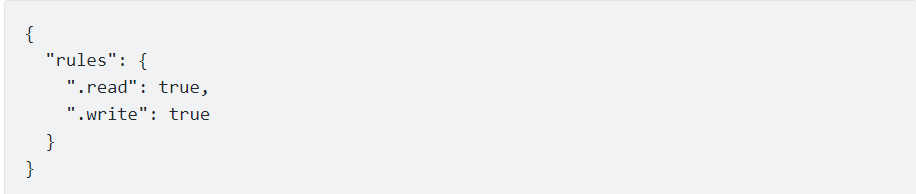
* Sau đó apply plugin google play service, thêm đoạn sau vào dòng cuối cùng file build.gradle của app, sau cả block dependencies.

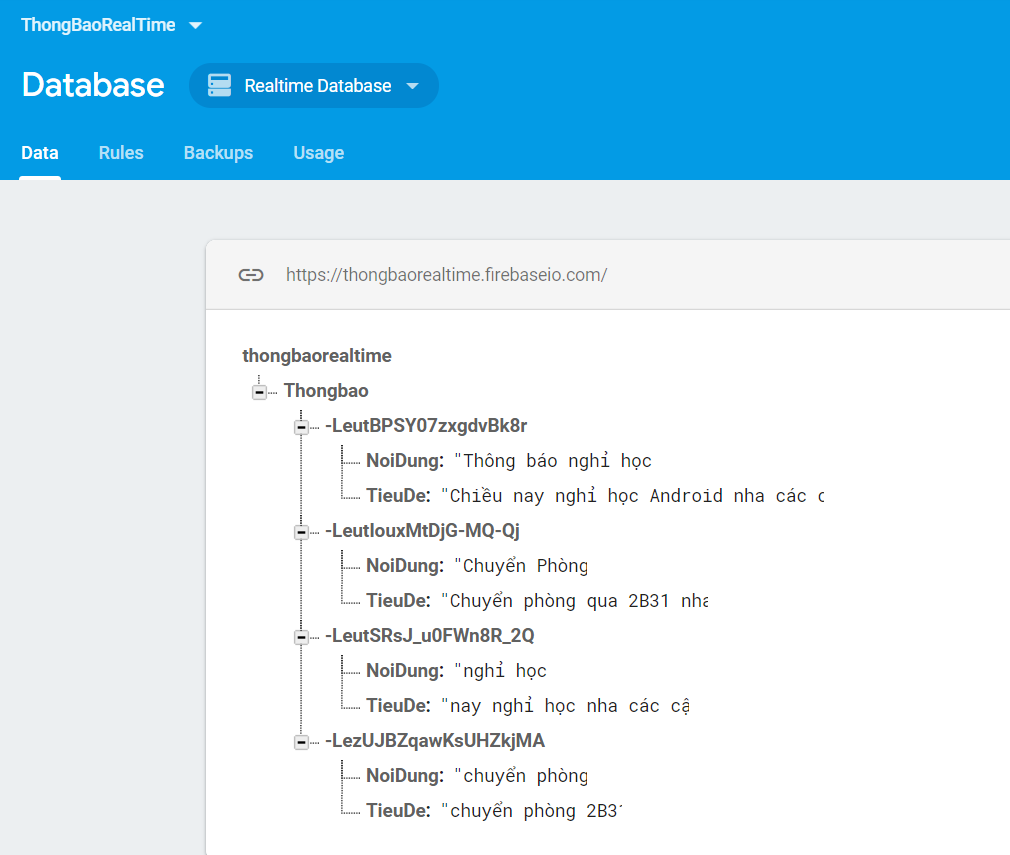


* Hoàn tất các bước trên bạn nhấn sync gradle là xong bạn có thể bắt đầu viết code handle việc nhận notifications.
* Để cài đặt Notification SDK vào ứng dụng bạn chỉ cần thêm dependency sau vào file build.gragle.



* Trong trường hợp dữ liệu không được ghi vào thì chúng ta có thể sửa lại RULES trong tab RULES để có thể ghi được mà không cần Authentication





* Bây giờ chúng ta tạo một Project mới trên android studio với MainActivity dạng Empty Activity. Khi đã cấu hình xong Firebase lên android studio ta cần kích hoạt firebase trong project bằng cách thêm gọi Firebase tại hàm onCreate trong MainActivity với devpro-firebase2 là tên app được tạo trên Firebase.

**Thêm dữ liệu.**

* Thêm dữ liệu String bằng setValue():
* Nhận dữ liệu sử dụng AddChild với Push Object



– value: android key Tiêu đề – value: Nội Dung Ở đây ta tạo một class là UpdateThongbao.class. và tạo 2 thuộc tính String là Tiêu Đề, và nội dungsau đó tạo Constructor cho 2 thuộc tính này. và một Constructor rỗng thông báo để nhận dữ liệu từ firebase về.

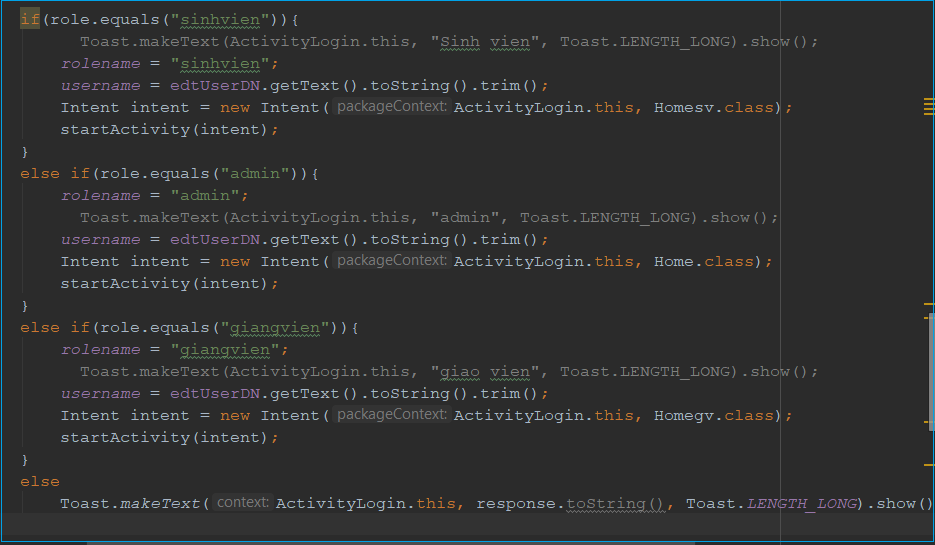
**Chức năng của màn hình Thông Báo**

* Phòng giáo vụ có thể gửi thông báo khẩn cấp tới tất cả các sinh viên. Với việc sử dụng fire base sẽ gửi thông báo Real Time tới sinh viên ,như Nghỉ học, chuyển phòng.

1. **Cài đặt và kiểm thử**
2. **Phần mềm sử dụng**

* Android Studio.
* Genymotion.

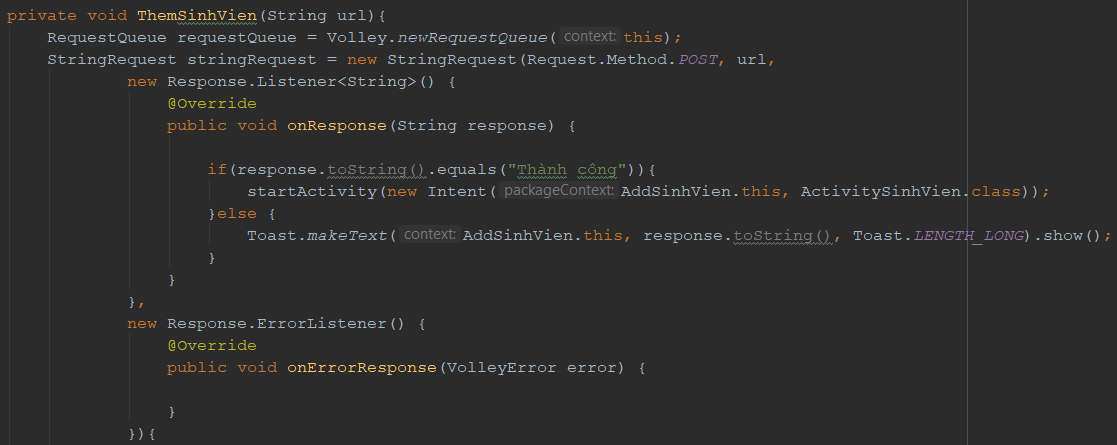
1. **Cài đặt, kiểm thử chương trình**
   1. **Code xử lý đăng nhập**

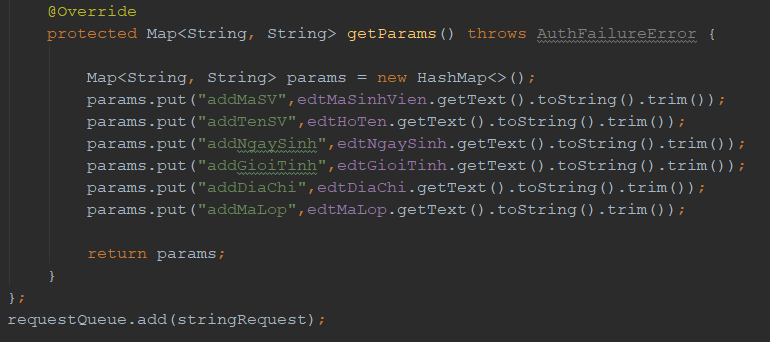
****

* 1. **Code xử lý Hiển thị, Thêm, Xóa, Sửa dữ liệu.**
  2. **Hiển thị**

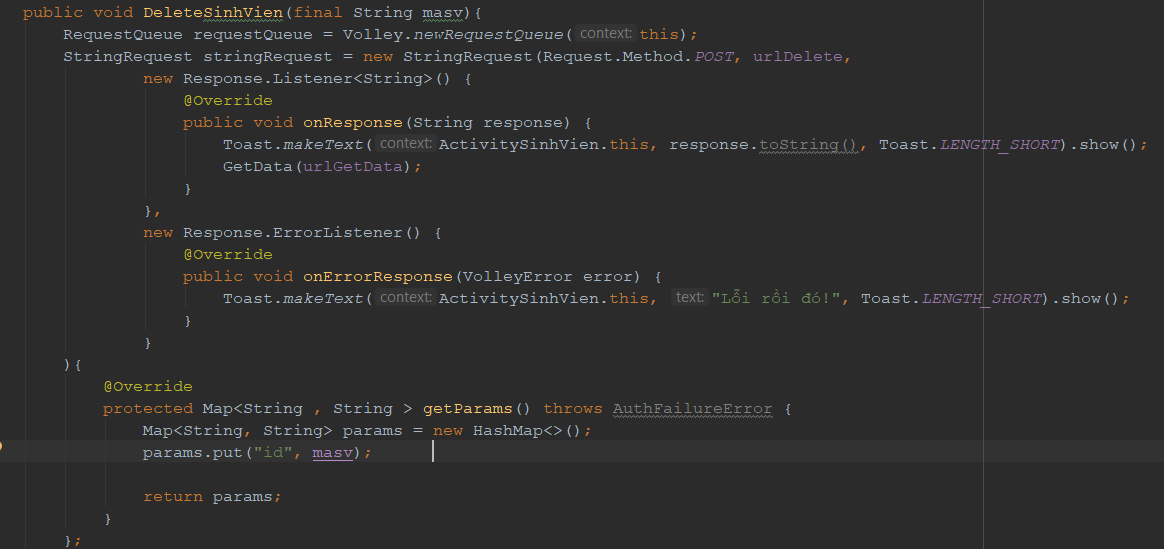
****

* 1. **Thêm**

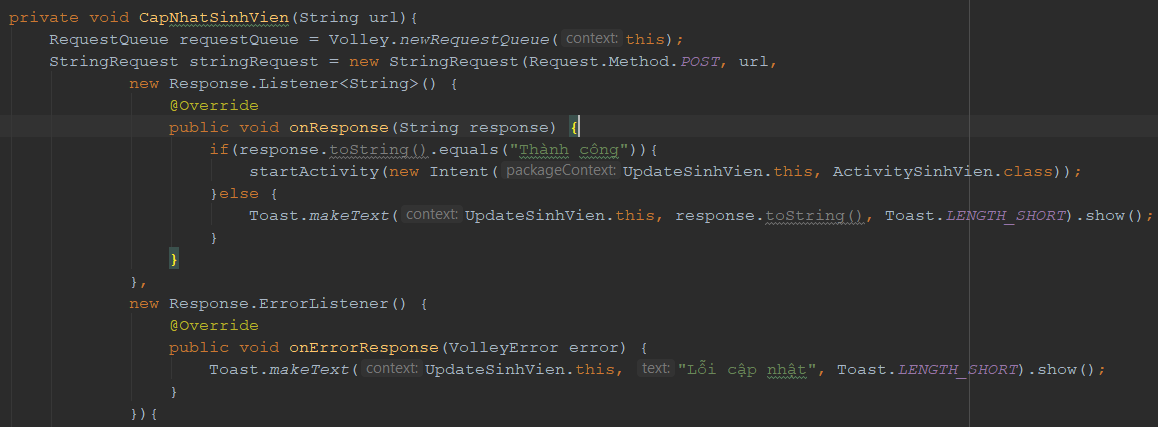
****

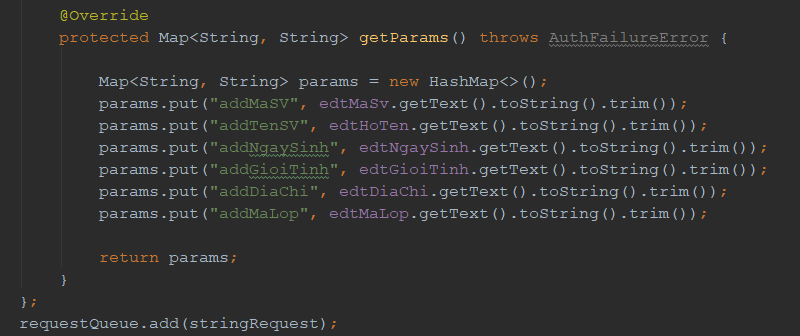
****

* 1. **Xóa**

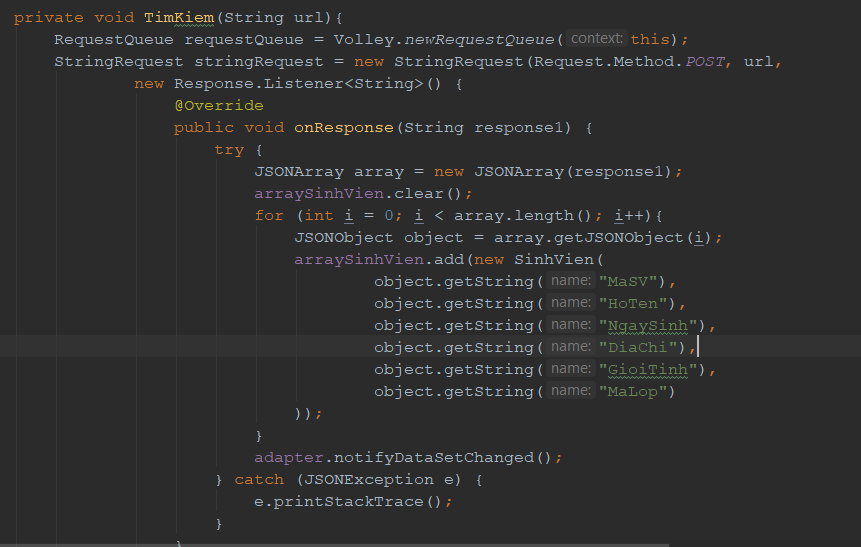
****

* 1. **Sửa**

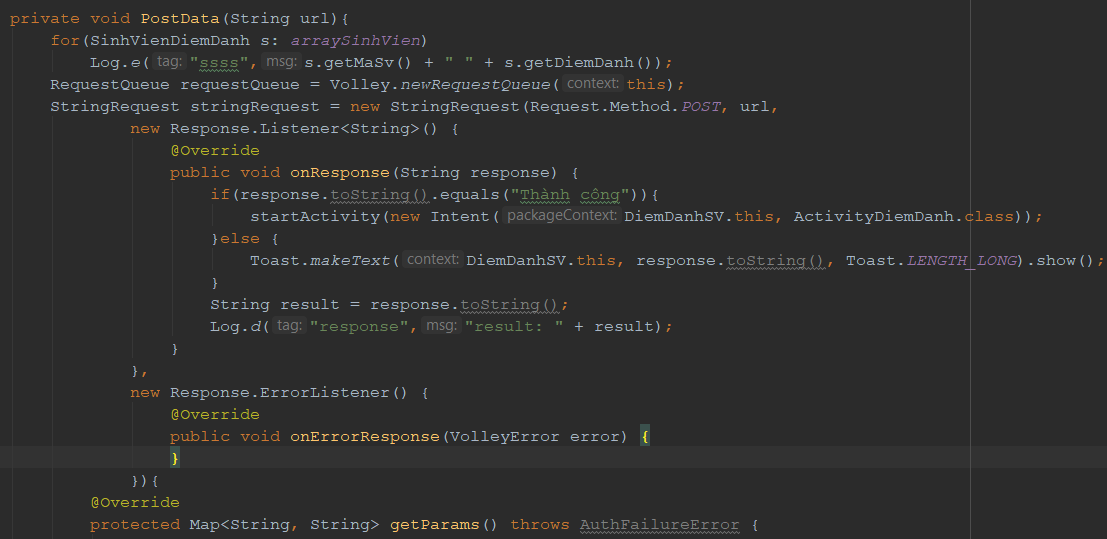
****

****

* 1. **Code tìm kiếm**

****

* 1. **Code điểm danh**

****

1. **Kết luận**
2. **Ưu điểm**

* Dễ dàng sử dụng quản lý.
* Người dùng dễ dàng kiểm soát.
* Dễ dàng tổ chức xử lý.
* Giao diện thân thiện với người sử dụng.
* Dữ liệu được cập nhật nhanh chóng.

1. **Nhược điểm**

* Vì sử dụng wifi để lấy dữ liệu và xử lý dữ liệu bằng webservice nên phải phụ thuộc vào tốc độ mạng.
* Dữ liệu gửi lên phải chính xác mới có thể thực hiện hành động và yêu cầu.

-----------------------------------------------HẾT-----------------------------------------------