|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **KHOA: CÔNG NGHỆ THÔNG TIN**  ---------------------------------------    **BÁO CÁO HỌC PHẦN**  **PHÁT TRIỂN ỨNG DỤNG TRÊN THIẾT BỊ DI ĐỘNG**      **XÂY DỰNG ỨNG DỤNG THEO DÕI CÁC HOẠT ĐỘNG THỂ CHẤT SINH VIÊN HAUI**     |  |  | | --- | --- | | GVHD: | ThS. Vũ Thị Dương | | Nhóm: | 1 | | Sinh viên: | Bùi Quốc Triệu  Đỗ Thế Nhuận  Nguyễn Văn An  Dương Bá Trung Kiên | | Lớp: | 20231IT6029002 | | Khoá: | 16 |           Hà Nội – Năm 2023 |

# DANH MỤC KÝ HIỆU, CÁC CHỮ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **STT** | **Ký hiệu, chữ viết tắt** | **Nội dung đầy đủ** |
| 1 | DVM | Dalvik Virtual Machine |
| 2 | JVM | Java Virtual Machine |
| 3 | TKB | Thời khoá biểu |

# DANH MỤC HÌNH ẢNH

[Hình 2‑1. Biểu đồ Use case 13](#_Toc154520258)

[Hình 2‑2. Biểu đồ lớp dữ liệu 15](#_Toc154520259)

[Hình 2‑3. Màn hình đăng nhập 20](#_Toc154520260)

[Hình 2‑4. Màn hình cảnh báo các vấn đề nguy hiểm 21](#_Toc154520261)

[Hình 2‑5. Màn hình báo cáo sức khoẻ và báo ốm 22](#_Toc154520262)

[Hình 2‑6. Màn hình rèn luyện 23](#_Toc154520263)

[Hình 2‑7. Màn hình xem lịch sử rèn luyện 24](#_Toc154520264)

[Hình 2‑8. Màn hình báo nghỉ 25](#_Toc154520265)

[Hình 2‑9. Màn hình nhập lý do báo nghỉ 26](#_Toc154520266)

[Hình 2‑10. Màn hình học phần thể chất 27](#_Toc154520267)

[Hình 2‑11. Màn hình các học phần thể chất 28](#_Toc154520268)

[Hình 2‑12. Màn hình kết quả thể chất 29](#_Toc154520269)

[Hình 2‑13. Màn hình chỉnh sửa thông tin cá nhân 30](#_Toc154520270)

[Hình 2‑14. Biểu đồ cơ sở dữ liệu mức vật lý 31](#_Toc154520271)

[Hình 2‑15. Màn hình báo cáo sức khoẻ và báo ốm 32](#_Toc154520272)

[Hình 2‑16. Màn hình theo dõi rèn luyện hàng ngày 43](#_Toc154520273)

[Hình 2‑17. Màn hình xem lịch sử rèn luyện 43](#_Toc154520274)

[Hình 2‑18. Màn hình báo nghỉ 44](#_Toc154520275)

[Hình 2‑19. Màn hình nhập lý do nghỉ 44](#_Toc154520276)

[Hình 2‑20. Màn hình học phần thể chất 63](#_Toc154520277)

[Hình 2‑21, Màn hình xem các học phần thể chất 63](#_Toc154520278)

[Hình 2‑22. Màn hình xem kết quả thể chất 70](#_Toc154520279)

[Hình 2‑23. Màn hình chỉnh sửa thông tin cá nhân 70](#_Toc154520280)

# DANH MỤC BẢNG BIỂU

[Bảng 1‑1. Yêu cầu phần cứng Android Studio với Windows 9](#_Toc154520256)

[Bảng 1‑2. Yêu cầu phần cứng Android Studio với Mac 10](#_Toc154520257)

# MỤC LỤC

[DANH MỤC KÝ HIỆU, CÁC CHỮ VIẾT TẮT 2](#_Toc154520223)

[DANH MỤC HÌNH ẢNH 3](#_Toc154520224)

[DANH MỤC BẢNG BIỂU 4](#_Toc154520225)

[MỤC LỤC 5](#_Toc154520226)

[PHẦN 1. MỞ ĐẦU 7](#_Toc154520227)

[1.1 Lý do chọn đề tài 7](#_Toc154520228)

[1.2 Tổng quan về lập trình ứng dụng di động 7](#_Toc154520229)

[1.2.1 Khái niệm 7](#_Toc154520230)

[1.2.2 Các phương pháp lập trình trên thiết bị di động 8](#_Toc154520231)

[1.3 Tổng quan về lập trình ứng dụng di động trên Android 8](#_Toc154520232)

[1.3.1 Khái niệm 8](#_Toc154520233)

[1.3.2 Giới thiệu về Android Studio và Android SDK 9](#_Toc154520234)

[1.4 Các kiến thức, kỹ năng cần thiết phục vụ cho đề tài 10](#_Toc154520235)

[PHẦN 2. KẾT QUẢ NGHIÊN CỨU 11](#_Toc154520236)

[2.1 Tìm hiểu yêu cầu bài toán 11](#_Toc154520237)

[2.2 Xây dựng bài toán 11](#_Toc154520238)

[2.2.1 Phân tích yêu cần người dùng 11](#_Toc154520239)

[2.2.1.1 Yêu cầu chức năng và phi chức năng 11](#_Toc154520240)

[2.2.1.2 Sơ đồ Use Case 13](#_Toc154520241)

[2.2.1.3 Biểu đồ lớp dữ liệu 15](#_Toc154520242)

[2.2.2 Thiết kế hệ thống 16](#_Toc154520243)

[2.2.2.1 Thống nhất quy tắc khi thiết kế giao diện 16](#_Toc154520244)

[2.2.2.2 Mô tả giao diện chính 19](#_Toc154520245)

[2.2.2.3 Thiết kế dữ liệu mức vật lý 31](#_Toc154520246)

[2.3 Thực hiện bài toán 31](#_Toc154520247)

[2.3.1 Nguyễn Văn An – Chức năng báo cáo sức khoẻ theo ngày và báo ốm 32](#_Toc154520248)

[2.3.2 Bùi Quốc Triệu – Chức năng theo dõi rèn luyện hàng ngày, xem lịch sử rèn luyện và báo nghỉ 42](#_Toc154520249)

[2.3.3 Dương Bá Trung Kiên – Chức năng xem các học phần thể chất 62](#_Toc154520250)

[2.3.4 Đỗ Thế Nhuận – Chức năng xem kết quả thể chất, chỉnh sửa thông tin cá nhân 69](#_Toc154520251)

[PHẦN 3. KIẾN THỨC LĨNH HỘI VÀ BÀI HỌC KINH NGHIỆM 84](#_Toc154520252)

[3.1 Nội dung đã thực hiện 84](#_Toc154520253)

[3.2 Hướng phát triển 84](#_Toc154520254)

[TÀI LIỆU THAM KHẢO 85](#_Toc154520255)

# MỞ ĐẦU

## Lý do chọn đề tài

Trong thời đại bùng nổ công nghệ thông tin hiện nay, những ứng dụng liên quan tới việc theo dõi các hoạt động thể chất, nhất là dành cho sinh viên ngày càng trở nên phổ biến và được nhiều bạn sinh viên quan tâm. Xuất phát từ nhu cầu quản lý các hoạt động thể chất, đặc biệt gắn với đời sống sinh viên và trường đại học, từ thực tiễn hiện nay trường Đại học Công nghiệp vẫn chưa có một ứng dụng theo dõi các hoạt động thể chất cho sinh viên trên ứng dụng di động, nhóm chúng em quyết định lựa chọn đề tài: **“Xây dựng ứng dụng theo dõi các hoạt động thể chất sinh viên HAUI”**.

Đề tài của chúng em bao gồm 3 phần như sau:

- Phần 1: Mở đầu: Mô tả một cách tổng quan về đề tài nghiên cứu, bao gồm ý tưởng, các kiến thức, kỹ năng liên quan được sử dụng trong đề tài để đạt được các mục tiêu hay chuẩn đầu ra của học phần.

- Phần 2: Kết quả nghiên cứu: Tìm hiểu yêu cầu của bài toán, phân tích yêu cầu người dùng, thiết kế hệ thống và thực hiện làm các chức năng giải quyết yêu cầu của bài toán đặt ra.

- Phần 3: Kiến thức lĩnh hội và bài học kinh nghiệm: Nêu lên những nội dung đã thực hiện được trong đề tài cũng như hướng phát triển đề tài trong tương lai

## Tổng quan về lập trình ứng dụng di động

### Khái niệm

Lập trình ứng dụng di động chính là việc sử dụng những ngôn ngữ lập trình để viết app phát triển những ứng dụng để gia tăng tới mức tốt nhất tiện ích cho thiết bị di động mà người dùng đang dùng. Trong đó, hai hệ điều hành được sử dụng phổ biến hiện nay là iOS và Android thì việc có thể lập trình ứng dụng mobile phù hợp sẽ đáp ứng tốt cho những yêu cầu, đòi hỏi thực tế của con người. Trong đó, những ứng dụng phổ biến như chat, game, từ điển, đọc truyện, tổng hợp thông tin chứng khoán, giá vàng, hay truy cập mạng xã hội, ứng dụng cho doanh nghiệp.

### Các phương pháp lập trình trên thiết bị di động

Hiện nay, có 2 phương pháp lập trình trên thiết bị di động phổ biến nhất đó là lập trình ứng dụng cho hệ điều hành Android và IOS.

Để viết ứng dụng Android, có 2 ngôn ngữ chính thường được sử dụng là Java và Kotlin. Java đang là ngôn ngữ lập trình ứng dụng Android được sử dụng phổ biến nhất hiện nay. Kotlin lại có chức năng chính là hỗ trợ khắc phục một số vấn đề gặp phải khi viết ứng dụng Android bằng ngôn ngữ Java.

Để viết ứng dụng IOS, có 2 ngôn ngữ chính thường được sử dụng là Swift và Objective-C. Ngôn ngữ lập trình Swift hiện đang được tin dùng nhất hiện nay. Mặc dù hiện nay, ngôn ngữ Swift ngày càng được sử dụng nhiều, Objective-C vẫn có tầm quan trọng và được tin dùng trong nhiều dự án.

Nhóm chúng em đã chọn phương pháp lập trình trên hệ điều hành Android và ngôn ngữ lập trình Java cho đề tài của mình bởi Android có số lượng người dùng lớn, đa dạng thiết bị từ nhiều nhà sản xuất khác nhau, linh hoạt và dễ tiếp cận và ngôn ngữ Java lại là một ngôn ngữ phổ biến và có cộng đồng hỗ trợ lớn mạnh.

## Tổng quan về lập trình ứng dụng di động trên Android

### Khái niệm

Android là một hệ điều hành mã nguồn mở đươc thiết kế dựa trên nền tảng Linux, dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng.

Giao diện người dùng của Android dựa trên nguyên tắc tương tác trực tiếp, sử dụng cảm ứng chạm đa điểm tương tự những động tác ngoài đời thực như vuốt, chạm, kéo dãn và thu lại để xử lý các đối tượng trên màn hình.

### Giới thiệu về Android Studio và Android SDK

Android Studio chạy trên Windows, Mac và Linux, nhằm thay thế cho Eclipse Android Development Tool (ADT) vốn được sử dụng làm IDE chính trong các năm trước đó.

Một số tính năng nổi bật:

- Bộ công cụ build ứng dụng dựa trên Gradle (thay vì Maven).

- Chức năng dò và sửa lỗi nhanh, hướng Android.

- Công cụ chỉnh sửa màn hình dạng kéo thả tiện lợi.

- Các wizard tích hợp nhằm giúp lập trình viên tạo ứng dụng từ mẫu có sẵn.

- Tích hợp Google Cloud Platform, dễ dàng tích hợp với Google Cloud Messaging và App Engine của Google.

Yêu cầu phần cứng của Android Studio:

- Đối với hệ điều hành Windows:

|  |  |  |
| --- | --- | --- |
| Yêu cầu | Tối thiểu | Đề xuất |
| Hệ điều hành | Microsoft Windows 8 64 bit | Phiên bản Windows 64 bit mới nhất |
| RAM | RAM 8 GB | RAM 16 GB trở lên |
| CPU | Kiến trúc CPU x86\_64; Intel Core thế hệ 2 trở lên hoặc CPU AMD có hỗ trợ Khung trình điều khiển ảo hoá của Windows. | Bộ xử lý Intel Core mới nhất |
| Dung lượng ổ đĩa | 8 GB (IDE, SDK Android và Trình mô phỏng) | Ổ thể rắn có dung lượng từ 16 GB trở lên |
| Độ phân giải màn hình | 1280 x 800 | 1920 x 1080 |

Bảng 1‑1. Yêu cầu phần cứng Android Studio với Windows

- Đối với hệ điều hành Mac:

|  |  |  |
| --- | --- | --- |
| Yêu cầu | Tối thiểu | Đề xuất |
| Hệ điều hành | MacOS 10.14 (Mojave) | Phiên bản MacOS mới nhất |
| RAM | RAM 8 GB | RAM 16 GB trở lên |
| CPU | Chip Apple M1 hoặc Intel Core thế hệ 2 trở lên có hỗ trợ Khung trình điều khiển ảo hoá. | Chip Apple Silicon mới nhất |
| Dung lượng ổ đĩa | 8 GB (IDE, SDK Android và Trình mô phỏng) | Ổ thể rắn có dung lượng từ 16 GB trở lên |
| Độ phân giải màn hình | 1280 x 800 | 1920 x 1080 |

Bảng 1‑2. Yêu cầu phần cứng Android Studio với Mac

Và đi kèm với Android Studio là Android SDK. Nếu Android Studio là trình soạn thảo code (IDE) thì Android SDK là bộ tổng hợp các công cụ để build app, các bản mẫu máy ảo Android (sử dụng để test app) cần thiết để làm ra một ứng dụng Android hoàn chỉnh.

## Các kiến thức, kỹ năng cần thiết phục vụ cho đề tài

Trước hết, cần có các kĩ năng xây dựng một ứng dụng Android hoàn chỉnh:

- Thiết kế giao diện trên Android

- Xử lý sự kiện trên giao diện ứng dụng

- Xây dựng các Intent

- Lưu trữ dữ liệu với file.

- Tương tác với cơ sở dữ liệu SQLite

- Cách sử dụng Fragment.

Không chỉ có vậy, ta cần có các kiến thức và kĩ năng trong việc viết báo cáo, phân tích thiết kế phần mềm.

Ngoài ra, những kỹ năng mềm khác như kỹ năng giao tiếp, kỹ năng làm việc nhóm là những kỹ năng không thể thiếu để giúp hoàn thành với kết quả tốt nhất.

# KẾT QUẢ NGHIÊN CỨU

## Tìm hiểu yêu cầu bài toán

Tên bài toán: **Xây dựng ứng dụng theo dõi các hoạt động thể chất Sinh viên HaUI**

Trong bối cảnh công nghệ phát triển nhanh chóng, việc ứng dụng công nghệ thông tin vào lĩnh vực giáo dục trở nên ngày càng phổ biến. Đặc biệt, với sự phổ cập của các thiết bị di động, việc xây dựng một ứng dụng theo dõi các hoạt động thể chất cho sinh viên trở thành một nhu cầu thiết yếu để tối ưu hóa quá trình theo dõi thể chất và quản lý thông tin cá nhân.

## Xây dựng bài toán

### Phân tích yêu cần người dùng

#### Yêu cầu chức năng và phi chức năng

Yêu cầu chức năng: ứng dụng cần phải thực hiện được các chức năng sau:

- Đăng nhập: Sinh viên có thể đăng nhập thông qua mã sinh viên và mật khẩu

- Nhập học: Điều hướng sang trang nhập học của trường Đại học Công nghiệp Hà Nội

- Đăng xuất: Đăng xuất khỏi tài khoản sinh viên

- Báo cáo sức khoẻ theo ngày: Báo cáo sức khoẻ (bình thường hoặc ốm) và bệnh (nếu ốm) theo ngày

- Xem lịch sử rèn luyện: Sinh viên có thể xem lịch sử rèn luyện của bản thân trong một khoảng thời gian bất kỳ

- Báo ốm: Sinh viên có thể báo ốm theo ngày dựa theo bệnh mà sinh viên nhập vào

- Báo nghỉ: Sinh viên có thể báo nghỉ theo môn học và theo ngày bất kỳ

- Theo dõi rèn luyện hằng ngày: Sinh viên có thể theo dõi tình trạng cơ thể (theo chỉ số BMI), việc uống nước và tập thể dục hàng ngày

- Xem các học phần thể chất: Sinh viên có thể xem các học phần thể chất của nhà trường

- Xem kết quả thể chất: Sinh viên có thể xem kết quả hoàn thành môn thể chất của bản thân

- Chỉnh sửa thông tin cá nhân: Sinh viên có thể chỉnh sửa chiều cao, cân nặng và bệnh nền của bản thân

Yêu cầu phi chức năng: ứng dụng cần đáp ứng được các yêu cầu phi chức năng sau:

- Độ tin cậy và hiệu suất: Đảm bảo ứng dụng hoạt động ổn định và chính xác, tốc độ nhanh.

- Giao diện: Cung cấp giao diện người dùng thân thiện, dễ sử dụng để sinh viên có thể theo dõi thông tin sức khỏe một cách thuận tiện.

- Tính khả dụng: Đảm bảo ứng dụng có sẵn khi cần thiết.

- Tuân thủ: Tuân thủ các quy định về phát triển phần mềm của nhà nước

- Chức năng: Cung cấp đầy đủ các chức năng theo yêu cầu với ít hoặc không có lỗi.

- Khả năng phát triển: Có khả năng mở rộng để tích hợp các tính năng mới và nâng cao trải nghiệm người dùng.

#### Sơ đồ Use Case

A diagram of a diagram

Description automatically generated

Hình 2‑1. Biểu đồ Use case

Mô tả tóm tắt các Use case:

- Đăng nhập: Use case này cho phép sinh viên nhập vào mã sinh viên và mật khẩu để đăng nhập vào tài khoản người dùng.

- Nhập học: Use case này cho phép sinh viên truy cập trang nhập học của hệ thống nhà trường.

- Đăng xuất: Use case này cho phép sinh viên đăng xuất khỏi ứng dụng.

- Báo cáo sức khoẻ theo ngày: Use case này cho phép sinh viên báo cáo sức khoẻ (bình thường hoặc ốm) theo ngày.

- Xem lịch sử rèn luyện: Use case này cho phép sinh viên xem lịch sử rèn luyện hàng ngày của mình trong một khoảng thời gian do sinh viên nhập vào.

- Báo ốm: Use case này cho phép sinh viên báo ốm cho các môn học trong ngày.

- Báo nghỉ: Use case này cho phép sinh viên báo nghỉ một ngày tuỳ chọn.

- Theo dõi rèn luyện hàng ngày: Use case này cho phép sinh viên theo dõi việc rèn luyện sức khoẻ hàng ngày của bản thân.

- Xem các học phần thể chất: Use case này cho phép sinh viên xem tất cả các học phần thể chất của nhà trường.

- Xem kết quả thể chất: Use case này cho phép sinh viên xem kết quả của các học phần thể chất và mức độ hoàn thành các môn thể chất (số tín đã và chưa hoàn thành) của mình.

- Chỉnh sửa thông tin cá nhân: Use case này cho phép sinh viên chỉnh sửa các thông tin cá nhân của mình.

#### Biểu đồ lớp dữ liệu

A diagram of a computer

Description automatically generated

Hình 2‑2. Biểu đồ lớp dữ liệu

Mô tả chức năng cho các bảng dữ liệu:

- Bảng students: lưu trữ thông tin các sinh viên

- Bảng results: lưu trữ thông tin các kết quả học tập theo lớp học phần của sinh viên

- Bảng onleaves: lưu trữ thông tin báo ốm, báo nghỉ của sinh viên

- Bảng classes: lưu trữ thông tin các lớp học phần của sinh viên

- Bảng courses: lưu trữ thông tin các học phần

- Bảng instructors: lưu trữ thông tin của các giáo viên

- Bảng trackers: lưu trữ thông tin theo dõi rèn luyện sức khoẻ hàng ngày của sinh viên

- Bảng healths: lưu trữ thông tin về sức khoẻ hàng ngày của sinh viên

### Thiết kế hệ thống

#### Thống nhất quy tắc khi thiết kế giao diện

Quy tắc đặt tên:

- Package: đặt tên chữ thường, không dấu, không viết cách

- Class: đặt tên theo kiểu PascalCase (viết hoa tất cả các chữ cái đầu của các từ)

- drawable: đặt tên theo kiểu snake\_case.

- Layout: đặt tên theo kiểu snake\_case, ví dụ:

+ activity\_<ACTIVITY\_NAME>.xml – cho activity

+ dialog\_<DIALOG\_NAME>.xml – cho dialog

+ list\_item\_<LIST\_NAME>.xml – cho list item trong ListView

+ fragment\_<FRAGMENT\_NAME>.xml - for fragment

+ layout\_<LAYOUT\_NAME>.xml – cho re-useable layout

+ merge\_<MERGE\_NAME>.xml – cho <merge> Tag only

+ widget\_<WIDGET\_NAME>.xml - cho custom view

+ appwidget\_<APPWIDGET\_NAME>.xml - cho app widget

- Android Widgets: đặt tên theo nguyên tắc camelCase, với kí tự viết tắt của Widget ở đầu như sau:

+ aclock - AnalogClock

+ actv - AutoCompleteTextView

+ btn - Button

+ cal - CalendarView

+ chb - CheckBox

+ chtv - CheckedTextView

+ chron - Chronometer

+ dp - DatePicker

+ edt - EditText

+ explv - ExpandableListView

+ fl - FrameLayout

+ gl - GridLayout

+ grv - GridView

+ hsv - HorizontalScrollView

+ imb - ImageButton

+ ims - ImageSwitcher

+ imv - ImageView

+ ll - LinearLayout

+ lsv - ListView

+ ctlr - MediaController

+ mactv - MultiAutoCompleteTextView

+ np - NumberPicker

+ pm - PopupMenu

+ pw - PopupWindow

+ pgb - ProgressBar

+ rb - RadioButton

+ rg - RadioGroup

+ rtb - RatingBar

+ rl - RelativeLayout

+ rv - RemoteViews

+ scv - ScrollView

+ schv - SearchView

+ skb - SeekBar

+ sap - ShareActionProvider

+ space - Space

+ spn - Spinner

+ stv - StackView

+ sw - Switch

+ tabh - TabHost

+ tabc - TabHost.TabSpec

+ tl - TableLayout

+ tr - TableRow

+ twg - TabWidget

+ tclock - TextClock

+ tsw - TextSwitcher

+ txv - TextView

+ tp - TimePicker

+ toast - Toast

+ tgb - ToggleButton

+ vdv - VideoView

+ vf - ViewFlipper

+ vsw - ViewSwitcher

Màu sắc: Dùng màu chủ đạo là màu xanh lá cây và màu trắng

Kích cỡ: Cỡ chữ là 16sp, các thành phần khác đặt kích cỡ sao cho phù hợp, dễ nhìn, dễ tương tác

Bố cục: Bố cục hài hoà, cân đối, dễ nhìn, tuân thủ các yêu cầu, nguyên tắc thiết kế

#### Mô tả giao diện chính

**Màn hình đăng nhập:**

A close-up of a card

Description automatically generated

Hình 2‑3. Màn hình đăng nhập

Màn hình đăng nhập bao gồm:

- 2 EditText: 1 EditText dùng để nhập mã sinh viên, 1 EditText dùng để nhập mật khẩu

- 1 CheckBox dùng để đánh dấu cho chương trình biết người dùng có muốn ghi nhớ lại thông tin đăng nhập hay không.

- 2 Button: 1 Button “Đăng nhập” dùng để đăng nhập vào hệ thống nếu mã sinh viên và mật khẩu nhập vào là đúng, 1 Button “Nhập học” sẽ điều hướng sang trang web nhập học của nhà trường nếu sinh viên chưa nhập học.

**Màn hình cảnh báo các vấn đề nguy hiểm:**

A screen shot of a sign

Description automatically generated

Hình 2‑4. Màn hình cảnh báo các vấn đề nguy hiểm

Màn hình cảnh báo các vấn đề nguy hiểm bao gồm:

- 3 TextView với cỡ chữ lớn, được làm nổi bật bằng màu sắc, kích cỡ, dùng để cảnh báo sinh viên về các vấn đề nguy hiểm đối với sức khoẻ cần tránh xa.

- 1 CheckBox “Xác nhận” thể hiện sự đồng ý, cam kết với cảnh báo ở trên

- 1 Button “Tiếp theo” dùng để truy cập vào hệ thống và chỉ có thể truy cập được khi sinh viên tích chọn vào CheckBox xác nhận.

**Màn hình báo cáo sức khoẻ và báo ốm:**

**A screenshot of a computer screen

Description automatically generated**

Hình 2‑5. Màn hình báo cáo sức khoẻ và báo ốm

Màn hình cài đặt bao gồm:

- 3 TextView chào người dùng, hỏi trạng thái và bệnh

- 1 Spinner để chọn tình trạng sức khỏe là bình thường hay bị ốm

- 1 EditText để nhập loại bệnh đang mắc phải

- 1 Button “Lưu” để lưu thông tin mình vừa chọn vào cơ sở dữ liệu

- 1 ListView để xem các môn học trong ngày, tiết học, giảng viên, tình trạng và bệnh, có thể thao tác trên ListView để báo ốm.

**Màn hình theo dõi rèn luyện hàng ngày:**

A screenshot of a computer screen

Description automatically generated

Hình 2‑6. Màn hình rèn luyện

Màn hình rèn luyện bao gồm:

- 1 TextView hiển thị chỉ số BMI dựa vào thông tin cá nhân.

- 1 TextView hiển thị tình trạng dựa vào chỉ số BMI tính được.

- 1 thanh tiến trình nhiệm vụ (dựa vào 2 CheckBox bên dưới để hiển thị tiến trình)

- 1 EditText để nhập thời gian tập thể dục, từ đó sẽ lưu thời gian tập thể dục này trong cơ sở dữ liệu và tính lượng nước cần uống.

- 2 CheckBox: 1 CheckBox “Uống đủ nước” và 1 CheckBox “Tập thể dục” để đánh dấu việc rèn luyện cá nhân của sinh viên. Bên dưới CheckBox “Uống đủ nước” là 1 TextView hiển thị lượng nước cần uống trong ngày (dựa trên thông tin cá nhân và thời gian hoạt động thể dục), bên dưới CheckBox “Tập thể dục” là 1 TextView lời khuyên về thời gian tập thể dục dựa trên thông tin cá nhân của từng sinh viên.

- 2 DatePicker chọn ngày bắt đầu và ngày kết thúc để xem lịch sử rèn luyện sức khoẻ.

- 1 Button “Xem lịch sử” để xem lịch sử rèn luyện, Button này sẽ gọi đến một màn hình xem lịch sử rèn luyện.

**Màn hình xem lịch sử rèn luyện:**

**A white rectangular frame with black text

Description automatically generated**

Hình 2‑7. Màn hình xem lịch sử rèn luyện

Màn hình xem lịch sử rèn luyện bao gồm:

- 1 TextView hiển thị khoảng thời gian mà người dùng đã chọn để xem lịch sử ở màn hình rèn luyện.

- 1 ListView hiển thị thông tin lịch sử rèn luyện của một ngày cụ thể bao gồm: ngày, BMI, tình trạng, bệnh, có uống đủ nước không, lượng nước cần uống trong ngày hôm đó, có tập thể dục không, thời gian tập thể dục.

- 1 nút có hình dạng mũi tên để quay về màn hình trước đó đã gọi nó (màn hình rèn luyện)

**Màn hình báo nghỉ:**

A screenshot of a cell phone

Description automatically generated

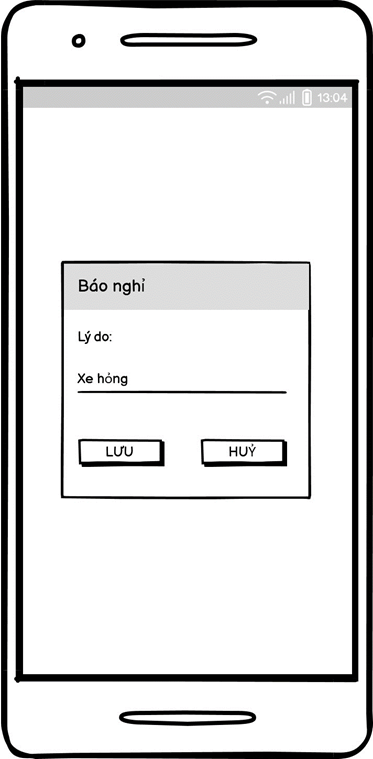
Hình 2‑8. Màn hình báo nghỉ

Màn hình báo nghỉ bao gồm:

- 1 DatePicker để chọn ngày báo nghỉ học.

- 1 ListView gồm các TextView để hiện thị dữ liệu thông tin như: tên môn học, tiết, tên giảng viên, tình trạng, lý do.

**Màn hình nhập lý do báo nghỉ:**

****

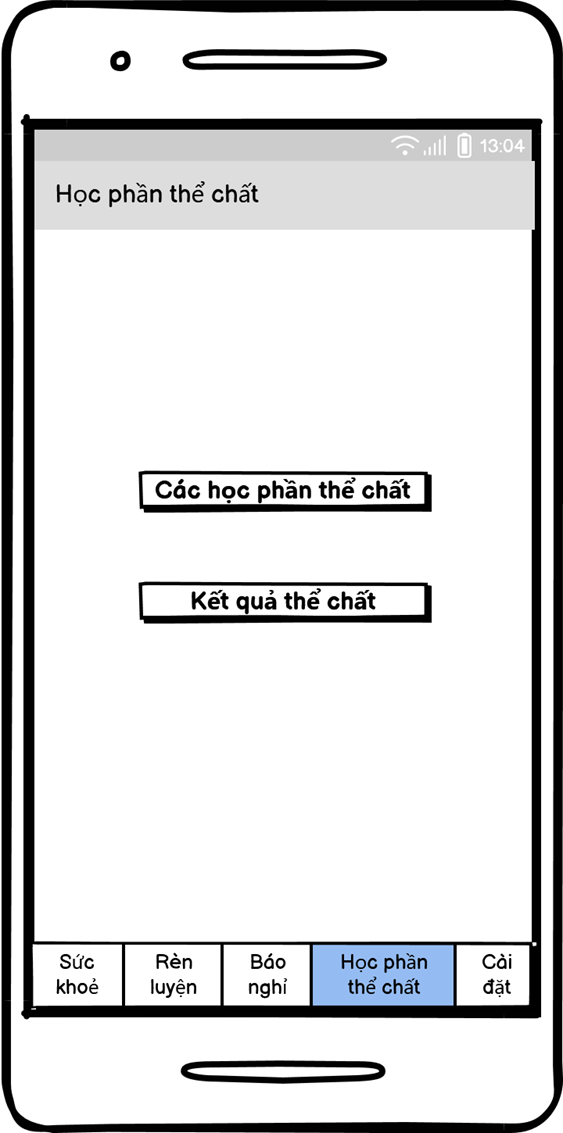
Hình 2‑9. Màn hình nhập lý do báo nghỉ

Màn hình nhập lý do báo nghỉ bao gồm:

- 1 EditText để nhập lý do báo nghỉ

- 2 Button là Button “Lưu” để khai báo lý do báo nghỉ và Button “Huỷ” để thoát khỏi màn hình này.

**Màn hình học phần thể chất:**

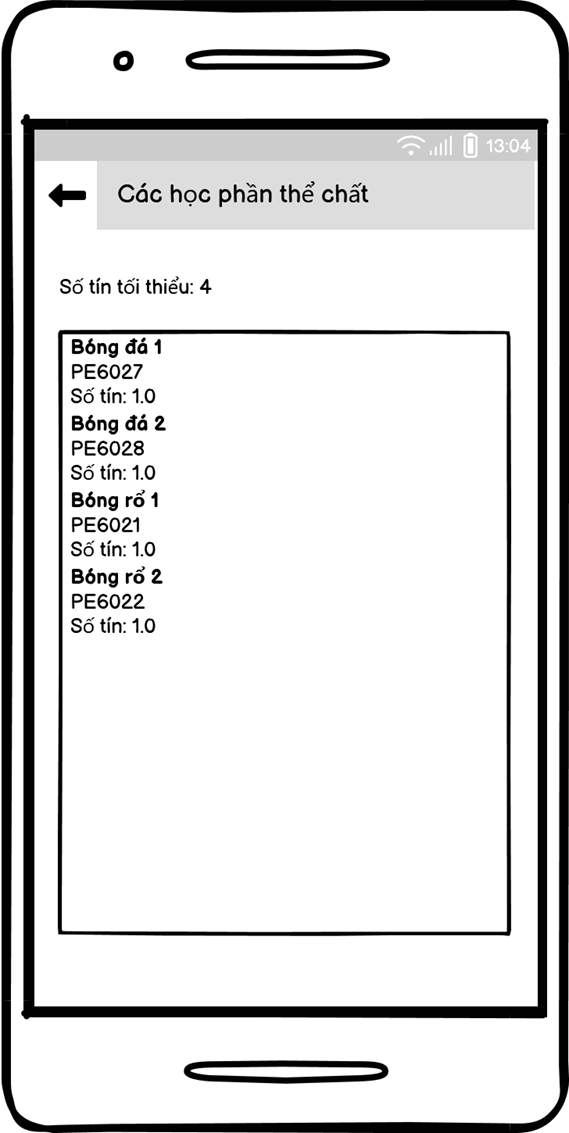


Hình 2‑10. Màn hình học phần thể chất

Màn hình bao gồm:

- 2 Button để chuyển hướng màn hình hiển thị nội dung cụ thể khi click vào button.

**Màn hình các học phần thể chất:**

****

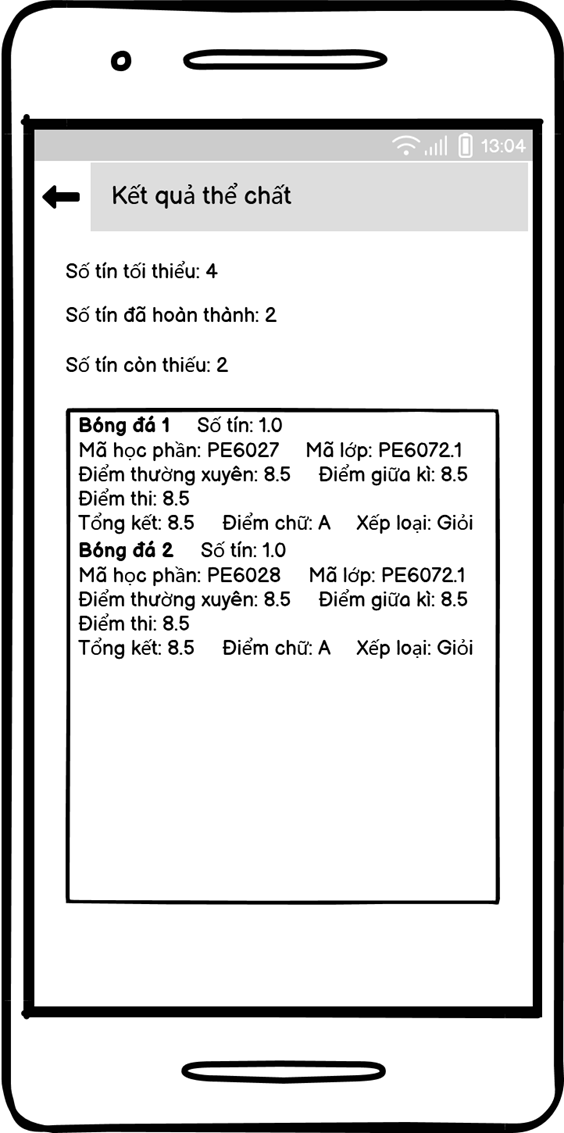
Hình 2‑11. Màn hình các học phần thể chất

Màn hình các học phần thể chất bao gồm:

- 1 TextView hiển thị số tín tối thiểu.

- 1 ListView chứa các thông tin bao gồm môn học, mã học phần và số tín chỉ của các học phần thể chất.

**Màn hình kết quả thể chất:**



Hình 2‑12. Màn hình kết quả thể chất

Màn hình bao gồm:

- 3 TextView hiển thị số tín tối thiểu, số tín đã hoàn thành và số tín còn thiếu

- 1 ListView hiển thị các nội dung bao gồm môn học, mã học phần, số tín chỉ, mã lớp, điểm thường xuyên, điểm giữa kì và điểm thi, điểm tổng kết, điểm chữ và xếp loại.

**Màn hình chỉnh sửa thông tin cá nhân:**

A close-up of a box

Description automatically generated

Hình 2‑13. Màn hình chỉnh sửa thông tin cá nhân

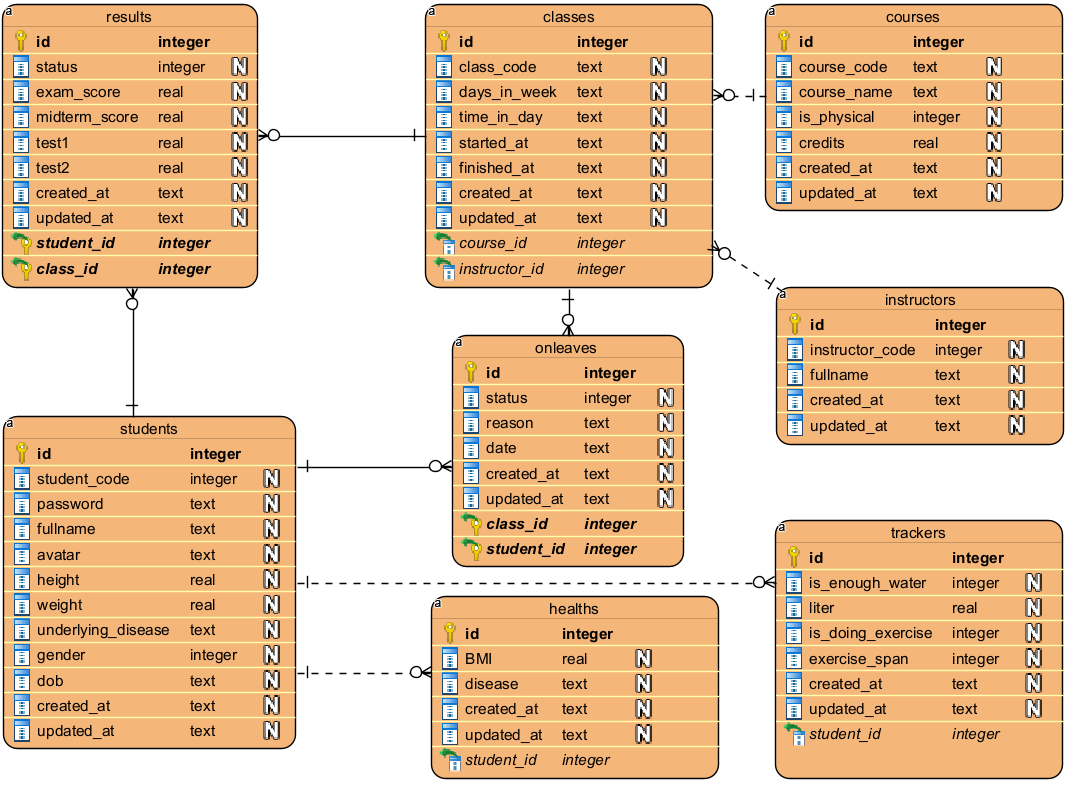
Màn hình cài đặt bao gồm:

- Các TextView để hiển thị các dữ liệu về thông tin của sinh viên như: mã sinh viên, tên, ngày sinh, giới tính, chiều cao, cân nặng bệnh nền.

- 3 EditText để cập nhật chỉnh sửa về chiều cao, cân nặng, bệnh nền và 1 Spinner để lựa chọn nhóm bệnh.

- 2 Button là Button Lưu để cập nhật lại thông tin và Button Đăng xuất để đăng xuất tài khoản của sinh viên khỏi ứng dụng.

#### Thiết kế dữ liệu mức vật lý



Hình 2‑14. Biểu đồ cơ sở dữ liệu mức vật lý

## Thực hiện bài toán

* Chụp màn hình đã thực hiện
* Nhiệm vụ của các thực hiện trên Activity/Fragment? Từ Activity/Fragment có gọi đến Activity nào không?
* Thiết kế giao diện (mã xml) (chỉ lấy xml của Fragment/Activity, không lấy những file xml của listview, menu,…)
* Một số mã nguồn, thuật toán chính:
  + Một số hàm (chỉ vài hàm chính) trong MyDatabaseHelper
  + Một số hàm (chỉ lấy vài hàm chính) trong file Fragment.java/Activity.java.
* Yêu cầu độ dài: Chỉ từ 5 – 10 trang/người
* Yêu cầu font: Courier New; font size 12

### Nguyễn Văn An – Chức năng báo cáo sức khoẻ theo ngày và báo ốm

**Màn hình thực hiện:**

**A screenshot of a phone

Description automatically generated**

Hình 2‑15. Màn hình báo cáo sức khoẻ và báo ốm

**Nhiệm vụ của Fragment:**

Fragment health hiển thị ra TKB, tình trạng sức khỏe của mỗi người dùng đã đăng nhập theo mã sinh viên và giúp cho người dùng báo cáo tình trạng sức khỏe và báo ốm.

Khi chọn tình trạng là “Bình thường” trên spinner và bấm nút “Lưu” thì hệ thông sẽ cập nhật tình trạng sức khỏe vào SQLite. Khi chọn tình trạng là “Ốm” thì màn hình sẽ hiển thị rõ editText để nhập loại bệnh mà người dùng mắc phải và bấm nút lưu thì hệ thống sẽ cập nhật tình trạng sức khỏe vào SQLite.

Người dùng có thể theo dõi TKB học tập của mình vào ngày đăng nhập gồm môn học, tiết học, giảng viên, tình trạng sức khỏe và tên bệnh (nếu tình trạng sức khỏe là “ốm”).

Khi người dùng nhấn vào 1 vị trí trên listView sau khi đã khai báo tình trạng sức khỏe ở trên thì listView sẽ cập nhật lại trạng thái là “Ốm” và sẽ hiển thị tên bệnh mà người dùng đã nhập.

Khi người dùng nhấn giữ vào 1 vị trí trên listView thì listView sẽ cập nhật thành trạng thái “Bình thường”.

**Thiết kế giao diện (mã xml):**

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="#DCEDC8"

tools:context=".HealthFragment">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent">

<androidx.appcompat.widget.Toolbar

android:id="@+id/toolbar4"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_alignParentTop="true"

android:layout\_marginTop="-3dp"

android:background="#4CAF50"

android:theme="?attr/actionBarTheme"

app:titleTextColor="#FFFFFF" />

<TextView

android:id="@+id/txtViewCH"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentStart="true"

android:layout\_alignParentTop="true"

android:layout\_marginStart="18dp"

android:layout\_marginTop="72dp"

android:text="Hôm nay bạn cảm thấy thế nào ?"

android:textSize="16sp" />

<Spinner

android:id="@+id/spnBenh"

android:layout\_width="195dp"

android:layout\_height="38dp"

android:layout\_alignParentStart="true"

android:layout\_alignParentTop="true"

android:layout\_marginStart="23dp"

android:layout\_marginTop="106dp" />

<TextView

android:id="@+id/txtViewBenh"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentStart="true"

android:layout\_alignParentTop="true"

android:layout\_marginStart="7dp"

android:layout\_marginTop="157dp"

android:text="Bệnh"

android:textSize="16sp" />

<EditText

android:id="@+id/edtBenh"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentTop="true"

android:layout\_marginTop="146dp"

android:layout\_marginRight="-259dp"

android:layout\_toLeftOf="@+id/txtViewBenh"

android:ems="10"

android:inputType="text"/>

<Button

android:id="@+id/btnLuu"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentStart="true"

android:layout\_alignParentTop="true"

android:layout\_marginStart="156dp"

android:layout\_marginTop="204dp"

android:background="#4CAF50"

android:textColor="#FFFFFF"

android:text="Lưu" />

<ListView

android:id="@+id/lstViewTKB"

android:layout\_width="417dp"

android:layout\_height="399dp"

android:layout\_alignParentStart="true"

android:layout\_alignParentTop="true"

android:layout\_alignParentBottom="true"

android:layout\_marginStart="0dp"

android:layout\_marginTop="280dp"

android:layout\_marginBottom="52dp" />

</RelativeLayout>

</androidx.constraintlayout.widget.ConstraintLayout>

**Một số mã nguồn, thuật toán chính:**

**-** Trong MyDatabaseHelper.java: hàm getLeaveReportByDate để hiển thị ra listView dữ liệu báo nghỉ của ngày đăng nhập, addSickReport để thêm dữ liệu báo ốm nếu đã có hoặc cập nhật dữ liệu báo ốm nếu đã có:

public ArrayList<LeaveReport> getLeaveReportsByDate(String date, long studentId){

SQLiteDatabase db = this.getReadableDatabase();

Cursor cursor = null;

String dates[] = date.split("/");

date = dates[2] + "-" + dates[1] + "-" + dates[0] ;

Calendar cal = Calendar.getInstance();

try {

cal.setTime(new SimpleDateFormat("yyyy-MM-dd").parse(date));

} catch (ParseException e) {

throw new RuntimeException(e);

}

int day\_of\_week = cal.get(Calendar.DAY\_OF\_WEEK);

if(day\_of\_week == 1){

day\_of\_week = 6;

}else{

day\_of\_week = day\_of\_week - 2;

}

final String myQuery = String.format("SELECT %s.%s, %s.%s, %s.%s, %s.%s, %s.%s, %s.%s, %s.%s, %s.%s " +

"FROM %s " +

"JOIN %s ON %s.%s = %s.%s " +

"JOIN %s ON %s.%s = %s.%s " +

"JOIN %s ON %s.%s = %s.%s " +

"LEFT JOIN %s ON (%s.%s = %s.%s AND %s.%s = %s.%s) " +

"WHERE %s.%s LIKE '%s' AND ('%s' BETWEEN %s.%s AND %s.%s) AND %s.%s = %s",

TABLE\_ONLEAVES, ONLEAVES\_ID, TABLE\_RESULTS, RESULTS\_CLASS\_ID, TABLE\_COURSES, COURSES\_COURSE\_NAME, TABLE\_INSTRUCTORS, INSTRUCTORS\_FULLNAME, TABLE\_CLASSES, CLASSES\_TIME\_IN\_DAY, TABLE\_ONLEAVES, ONLEAVES\_STATUS, TABLE\_ONLEAVES, ONLEAVES\_REASON, TABLE\_ONLEAVES, ONLEAVES\_DATE,

TABLE\_RESULTS,

TABLE\_CLASSES, TABLE\_RESULTS, RESULTS\_CLASS\_ID, TABLE\_CLASSES, CLASSES\_ID,

TABLE\_COURSES, TABLE\_CLASSES, CLASSES\_COURSE\_ID, TABLE\_COURSES, COURSES\_ID,

TABLE\_INSTRUCTORS, TABLE\_CLASSES, CLASSES\_INSTRUCTOR\_ID, TABLE\_INSTRUCTORS, INSTRUCTORS\_ID,

TABLE\_ONLEAVES, TABLE\_CLASSES, CLASSES\_ID, TABLE\_ONLEAVES, ONLEAVES\_CLASS\_ID, TABLE\_RESULTS, RESULTS\_STUDENT\_ID, TABLE\_ONLEAVES, ONLEAVES\_STUDENT\_ID,

TABLE\_CLASSES, CLASSES\_DAYS\_IN\_WEEK, "%" + day\_of\_week + "%", date, TABLE\_CLASSES, CLASSES\_STARTED\_AT, TABLE\_CLASSES, CLASSES\_FINISHED\_AT, TABLE\_RESULTS, RESULTS\_STUDENT\_ID, String.valueOf(studentId));

try {

cursor = db.rawQuery(myQuery, null);

} catch (Exception e) {

db.close();

return null;

}

if (cursor.getCount() == 0) {

cursor.close();

db.close();

return null;

} else {

cursor.moveToFirst();

ArrayList<LeaveReport> leaveReports = new ArrayList<LeaveReport>();

do{

LeaveReport leaveReport = new LeaveReport();

leaveReport.setClass\_id(cursor.getLong(1));

leaveReport.setCourse\_name(cursor.getString(2));

leaveReport.setInstructor\_name(cursor.getString(3));

leaveReport.setTime\_in\_day(cursor.getString(4));

leaveReports.add(leaveReport);

if(cursor.getString(7) != null){

if(cursor.getString(7).contains(date)){

leaveReport.setOnleave\_id(cursor.getLong(0));

leaveReport.setStatus(cursor.getLong(5));

leaveReport.setReason(cursor.getString(6));

}

}

} while (cursor.moveToNext());

cursor.close();

db.close();

return leaveReports;

}

}

//Thêm nếu chưa có, cập nhật nếu đã có

public boolean addSickReport(Long onleaveId, long classId, long studentId, String date, String reason) {

SQLiteDatabase db = this.getWritableDatabase();

Cursor cursor = null;

Date d;

try {

d = new SimpleDateFormat("dd/MM/yyyy").parse(date);

} catch (ParseException e) {

throw new RuntimeException(e);

}

date = new SimpleDateFormat("yyyy-MM-dd").format(d);

try {

cursor = db.query(TABLE\_ONLEAVES,

new String[]{ONLEAVES\_ID},

ONLEAVES\_ID + " = ?",

new String[]{String.valueOf(onleaveId)},

null, null, null);

} catch (Exception e) {

db.close();

return false;

}

if (cursor.getCount() == 0) {

ContentValues values = new ContentValues();

values.put(ONLEAVES\_STATUS, 1);

values.put(ONLEAVES\_REASON, reason);

values.put(ONLEAVES\_DATE, date);

values.put(ONLEAVES\_CLASS\_ID, classId);

values.put(ONLEAVES\_STUDENT\_ID, studentId);

db.insert(TABLE\_ONLEAVES, null, values);

cursor.close();

db.close();

return true;

} else{

cursor.moveToFirst();

ContentValues values = new ContentValues();

values.put(ONLEAVES\_STATUS, 1);

values.put(ONLEAVES\_REASON, reason);

db.update(TABLE\_ONLEAVES, values, ONLEAVES\_ID + " = ?", new String[]{String.valueOf(onleaveId)});

cursor.close();

db.close();

return true;

}

}

### Bùi Quốc Triệu – Chức năng theo dõi rèn luyện hàng ngày, xem lịch sử rèn luyện và báo nghỉ

**Màn hình thực hiện:**

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated  Hình 2‑16. Màn hình theo dõi rèn luyện hàng ngày | A screenshot of a phone  Description automatically generated  Hình 2‑17. Màn hình xem lịch sử rèn luyện |
| A screenshot of a phone  Description automatically generated  Hình 2‑18. Màn hình báo nghỉ | A screenshot of a computer  Description automatically generated  Hình 2‑19. Màn hình nhập lý do nghỉ |

**Nhiệm vụ thực hiện trên Fragment/Activity:**

Khi người dùng điều hướng tới màn hình theo dõi rèn luyện hàng ngày, hệ thống lấy dữ liệu rèn luyện hôm nay (số phút tập thể dục, uống đủ nước hay chưa, tập thể dục hay chưa) và tính chỉ số BMI như sau: BMI = Cân nặng (kg) / (Chiều cao (m))2. Sau đó, nó lưu chỉ số BMI vào cơ sở dữ liệu và hiện tình trạng của cơ thể như sau:

- Nếu BMI < 18.5: Cân nặng thấp (gầy)

- Nếu 18.5 ≤ BMI < 23: Bình thường

- Nếu 23 ≤ BMI < 25: Tiền béo phì

- Nếu 25 ≤ BMI < 30: Béo phì độ I

- Nếu > 30: Béo phì độ II

Nó cũng tính lượng nước tham khảo cho sinh viên và hiện thị lên màn hình: Số lít nước cần uống trong ngày = (Cân nặng (kg) + (Số phút tập thể dục) / 30 \* 12) \* 0.03. Nó cũng hiển thị thời gian tập thể dục tham khảo như sau:

- Nếu bệnh nền là “Không”, hiển thị:

"Thời gian tập thể dục tham khảo:

- 30 phút tập thể dục vừa phải hoặc 15 phút tập thể dục cường độ cao"

- Nếu bệnh nền là “Nhóm 1: Rối loạn chuyển hoá”, hiển thị:

"Thời gian tập thể dục tham khảo:

- 30 phút tập thể dục vừa phải

- Nên tập sau ăn từ 1 - 3 giờ vì đây là khoảng thời gian cơ thể trao đổi chất mạnh mẽ

- Đối với người bị rối loạn chuyển hoá ở mức quá cao, cần tuân theo tư vấn của bác sĩ trước khi lên kế hoạch tập luyện"

- Nếu bệnh nền là “Nhóm 2: Hô hấp”, hiển thị:

"Thời gian tập thể dục tham khảo

- Không tập thể dục quá sức, bắt đầu từ những bước nhỏ (ví dụ: 5 phút), tăng dần thời gian tập khi đã quen

- Chọn những bài tập phù hợp: những bài tập nhẹ nhàng như bơi lội, đi bộ, đạp xe, yoga, khí công, thái cực quyền

- Đối với người bị bệnh hô hấp nặng, bắt buộc phải tuân theo chỉ dẫn tập luyện của bác sĩ"

- Nếu bệnh nền là “Nhóm 3: Tim mạch”, hiển thị:

"Thời gian tập thể dục tham khảo:

- 30 phút tập thể dục vừa phải, ngắt quãng, không cố tập quá sức

- Chọn những bài tập phù hợp: những bài tập nhẹ nhàng như bơi lội, đi bộ, đạp xe, yoga, khí công, thái cực quyền

- Cần khởi động kỹ tối thiểu 15 phút để các hệ cơ - xương - khớp, hệ tuần hoàn và hô hấp thích nghi với nhịp độ vận động

- Đối với người bị bệnh tim mạch mức trung bình và cao, cần tuân theo chỉ dẫn của bác sĩ"

- Các bệnh nền còn lại, hiển thị:

"Thời gian tập thể dục tham khảo:

- Bệnh nền của bạn không có trong cơ sở dữ liệu đề xuất

- Hãy tập luyện tuân theo chỉ dẫn của bác sĩ"

Khi người dùng thay đổi thời gian tập thể dục trên màn hình, lượng nước uống sẽ được thay đổi theo, khi người dùng chọn các CheckBox “Uống đủ nước”, “Tập thể dục”, thanh tiến trình cũng sẽ cập nhật theo. Khi màn hình bị che khuất, hoặc kết thúc, dữ liệu sẽ tự được lưu vào cơ sở dữ liệu. Người dùng cũng có thể chọn ngày bắt đầu, ngày kết thúc trên màn hình và chọn nút “Xem lịch sử”, màn hình xem lịch sử rèn luyện sẽ được gọi. Màn hình này sẽ hiển thị khoảng thời gian người dùng đang xem lịch sử và một ListView chứa thông tin ngày, BMI, tình trạng, bệnh, có uống đủ nước không, lượng nước cần uống trong ngày hôm đấy, có tập thể dục không, thời gian tập thể dục trong ngày hôm đấy.

Khi người dùng điều hướng tới màn hình báo nghỉ, màn hình hiển thị 1 ListView chứa lịch học của ngày hôm đó bao gồm các thông tin: tên học phần, tiết, giảng viên, tình trạng, lý do (nếu báo nghỉ/báo ốm). Khi người dùng bấm giữ vào một hàng trên ListView, sẽ hiện ra một menu cho phép người dùng báo nghỉ hoặc đặt lại. Khi người dùng chọn báo nghỉ, màn hình nhập lý do nghỉ sẽ hiện ra. Người dùng nhập lý do nghỉ và bấm nút “Lưu” sẽ quay lại Fragment báo nghỉ và ListView sẽ cập nhật lại trạng thái là “Báo nghỉ” và lý do nghỉ mà sinh viên nhập vào. Khi người dùng chọn đặt lại, ListView sẽ cập nhật lại trạng thái là “Bình thường” và không hiển thị lý do nghỉ của môn đó.

**Thiết kế giao diện (mã xml):**

- Màn hình theo dõi rèn luyện hàng ngày:

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="#DCEDC8"

tools:context=".TodoFragment">

<TextView

android:id="@+id/txvBMITD"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="20dp"

android:layout\_marginTop="10dp"

android:text="BMI:"

android:textColor="#000000"

android:textSize="16sp"/>

<TextView

android:id="@+id/txvBMI"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="110dp"

android:layout\_marginTop="10dp"

android:text="0"

android:textSize="16sp"/>

<TextView

android:id="@+id/txvTinhTrangTD"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="20dp"

android:layout\_marginTop="10dp"

android:text="Tình trạng:"

android:textColor="#000000"

android:textSize="16sp"/>

<TextView

android:id="@+id/txvBMITT"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="110dp"

android:layout\_marginTop="10dp"

android:text="Bình thường"

android:textSize="16sp"/>

<androidx.appcompat.widget.Toolbar

android:id="@+id/toolbarTD"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:background="#4CAF50"

android:minHeight="?attr/actionBarSize"

android:theme="?attr/actionBarTheme"

app:title="Rèn luyện hàng ngày"

app:titleTextColor="#FFFFFF" />

<TextView

android:id="@+id/txvTienTrinh"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="20dp"

android:layout\_marginTop="10dp"

android:text="Tiến trình:"

android:textColor="#000000"

android:textSize="16sp"/>

<ProgressBar

android:id="@+id/pgbTienTrinh"

style="?android:attr/progressBarStyleHorizontal"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_marginStart="20dp"

android:layout\_marginTop="10dp"

android:layout\_marginEnd="20dp"

android:backgroundTint="#4CAF50"

android:max="2"

android:min="0"

android:progress="1"/>

<TextView

android:id="@+id/txvTGTTD"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="20dp"

android:layout\_marginTop="10dp"

android:text="Thời gian tập thể dục:"

android:textColor="#000000"

android:textSize="16sp"/>

<EditText

android:id="@+id/edtPhutTheDuc"

android:layout\_width="50dp"

android:layout\_height="48dp"

android:layout\_marginStart="10dp"

android:ems="10"

android:inputType="number"

android:text="0"

android:textSize="16sp"/>

<TextView

android:id="@+id/txvDVPhut"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="10dp"

android:layout\_marginTop="10dp"

android:text="phút"

android:textSize="16sp"/>

<TextView

android:id="@+id/txvNVHN"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="20dp"

android:layout\_marginTop="10dp"

android:text="Nhiệm vụ hàng ngày:"

android:textColor="#000000"

android:textSize="16sp"/>

<CheckBox

android:id="@+id/chbNuoc"

android:layout\_width="wrap\_content"

android:layout\_height="48dp"

android:layout\_marginStart="20dp"

android:text="Uống đủ nước"

android:textSize="16sp"/>

<TextView

android:id="@+id/txvNuoc"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="50dp"

android:text="2 lít"

android:textSize="16sp"/>

<CheckBox

android:id="@+id/chbTheDuc"

android:layout\_width="wrap\_content"

android:layout\_height="48dp"

android:layout\_marginStart="20dp"

android:text="Tập thể dục"

android:textSize="16sp"/>

<TextView

android:id="@+id/txvTheDuc"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_marginStart="50dp"

android:layout\_marginEnd="10dp"

android:text="Thời gian tham khảo"

android:textSize="16sp"/>

<androidx.constraintlayout.widget.ConstraintLayout

android:id="@+id/innerLayoutTodo"

android:layout\_width="0dp"

android:layout\_height="180dp"

android:layout\_marginStart="20dp"

android:layout\_marginTop="10dp"

android:layout\_marginEnd="20dp"

android:background="#C8E6C9">

<TextView

android:id="@+id/txvNBD"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="20dp"

android:layout\_marginTop="20dp"

android:text="Ngày bắt đầu:"

android:textColor="#000000"

android:textSize="16sp"/>

<TextView

android:id="@+id/txvNKT"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="20dp"

android:layout\_marginTop="20dp"

android:text="Ngày kết thúc:"

android:textColor="#000000"

android:textSize="16sp"/>

<TextView

android:id="@+id/txvBatDau"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="150dp"

android:layout\_marginTop="20dp"

android:labelFor="@id/imbBatDau"

android:textSize="16sp"/>

<ImageButton

android:id="@+id/imbBatDau"

android:layout\_width="48dp"

android:layout\_height="48dp"

android:layout\_marginTop="5dp"

android:layout\_marginEnd="20dp"

android:background="#4CAF50"/>

<ImageButton

android:id="@+id/imbKetThuc"

android:layout\_width="48dp"

android:layout\_height="48dp"

android:layout\_marginTop="5dp"

android:layout\_marginEnd="20dp"

android:background="#4CAF50"

android:src="@drawable/ic\_date\_picker"/>

<TextView

android:id="@+id/txvKetThuc"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="150dp"

android:layout\_marginTop="20dp"

android:labelFor="@id/imbKetThuc"

android:textSize="16sp"/>

<Button

android:id="@+id/btnLichSu"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_marginStart="80dp"

android:layout\_marginEnd="80dp"

android:layout\_marginBottom="20dp"

android:background="#4CAF50"

android:text="Xem lịch sử"

android:textColor="#FFFFFF"/>

</androidx.constraintlayout.widget.ConstraintLayout>

</androidx.constraintlayout.widget.ConstraintLayout>

- Màn hình báo nghỉ:

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="#DCEDC8"

tools:context=".LeaveReportFragment">

<androidx.appcompat.widget.Toolbar

android:id="@+id/toolbarLR"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:background="#4CAF50"

android:minHeight="?attr/actionBarSize"

android:theme="?attr/actionBarTheme"

app:title="Báo nghỉ"

app:titleTextColor="@color/white" />

<TextView

android:id="@+id/txvChonNgayBN"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="20dp"

android:layout\_marginTop="20dp"

android:text="Chọn ngày:"

android:textColor="#000000"

android:textSize="16sp"/>

<TextView

android:id="@+id/txvNgayNghi"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="150dp"

android:layout\_marginTop="20dp"

android:textSize="16sp"/>

<ImageButton

android:id="@+id/imbNgayNghi"

android:layout\_width="48dp"

android:layout\_height="48dp"

android:layout\_marginStart="300dp"

android:layout\_marginTop="10dp"

android:background="#4CAF50"

android:src="@drawable/ic\_date\_picker"/>

<ListView

android:id="@+id/lsvNgayNghi"

android:layout\_width="0dp"

android:layout\_height="0dp"

android:layout\_marginStart="20dp"

android:layout\_marginTop="10dp"

android:layout\_marginEnd="20dp"

android:layout\_marginBottom="20dp"/>

</androidx.constraintlayout.widget.ConstraintLayout>

**Một số mã nguồn, thuật toán chính:**

- Trong MyDatabaseHelper.java: hàm getTrackerInDateRange lấy dữ liệu rèn luyện trong một khoảng thời gian bất kỳ, hàm getLeaveReportsByDate lấy dữ liệu cho ListView trên màn hình báo nghỉ

public ArrayList<Tracker> getTrackerInDateRange(String start, String end, long studentId) {

SQLiteDatabase db = this.getReadableDatabase();

Cursor cursor = null;

try {

cursor = db.query(TABLE\_TRACKERS,

new String[]{TRACKERS\_ID, TRACKERS\_IS\_ENOUGH\_WATER, TRACKERS\_LITER,

TRACKERS\_IS\_DOING\_EXERCISE, TRACKERS\_EXERCISE\_SPAN, TRACKERS\_CREATED\_AT,

TRACKERS\_UPDATED\_AT, TRACKERS\_STUDENT\_ID},

TRACKERS\_STUDENT\_ID + " = ? AND (" + TRACKERS\_CREATED\_AT + " BETWEEN ? AND ?)",

new String[]{String.valueOf(studentId), start, end},

null, null, null);

} catch (Exception e) {

db.close();

return null;

}

if (cursor.getCount() == 0) {

cursor.close();

db.close();

return null;

} else {

cursor.moveToFirst();

ArrayList<Tracker> trackers = new ArrayList<Tracker>();

do {

Tracker tracker = new Tracker();

tracker.setId(cursor.getLong(0));

tracker.setIs\_enough\_water(cursor.getLong(1));

tracker.setLiter(cursor.getDouble(2));

tracker.setIs\_doing\_exercise(cursor.getLong(3));

tracker.setExercise\_span(cursor.getLong(4));

tracker.setCreated\_at(cursor.getString(5));

tracker.setUpdated\_at(cursor.getString(6));

tracker.setStudent\_id(cursor.getLong(7));

trackers.add(tracker);

} while (cursor.moveToNext());

cursor.close();

db.close();

return trackers;

}

}

public ArrayList<LeaveReport> getLeaveReportsByDate(String date, long studentId) {

SQLiteDatabase db = this.getReadableDatabase();

Cursor cursor = null;

String dates[] = date.split("/");

date = dates[2] + "-" + dates[1] + "-" + dates[0];

Calendar cal = Calendar.getInstance();

try {

cal.setTime(new SimpleDateFormat("yyyy-MM-dd").parse(date));

} catch (ParseException e) {

throw new RuntimeException(e);

}

int day\_of\_week = cal.get(Calendar.DAY\_OF\_WEEK);

if (day\_of\_week == 1) {

day\_of\_week = 6;

} else {

day\_of\_week = day\_of\_week - 2;

}

final String myQuery = String.format("SELECT %s.%s, %s.%s, %s.%s, %s.%s, %s.%s, %s.%s, %s.%s, %s.%s " +

"FROM %s " +

"JOIN %s ON %s.%s = %s.%s " +

"JOIN %s ON %s.%s = %s.%s " +

"JOIN %s ON %s.%s = %s.%s " +

"LEFT JOIN %s ON (%s.%s = %s.%s AND %s.%s = %s.%s) " +

"WHERE %s.%s LIKE '%s' AND ('%s' BETWEEN %s.%s AND %s.%s) AND %s.%s = %s",

TABLE\_ONLEAVES, ONLEAVES\_ID, TABLE\_RESULTS, RESULTS\_CLASS\_ID, TABLE\_COURSES, COURSES\_COURSE\_NAME, TABLE\_INSTRUCTORS, INSTRUCTORS\_FULLNAME, TABLE\_CLASSES, CLASSES\_TIME\_IN\_DAY, TABLE\_ONLEAVES, ONLEAVES\_STATUS, TABLE\_ONLEAVES, ONLEAVES\_REASON, TABLE\_ONLEAVES, ONLEAVES\_DATE,

TABLE\_RESULTS,

TABLE\_CLASSES, TABLE\_RESULTS, RESULTS\_CLASS\_ID, TABLE\_CLASSES, CLASSES\_ID,

TABLE\_COURSES, TABLE\_CLASSES, CLASSES\_COURSE\_ID, TABLE\_COURSES, COURSES\_ID,

TABLE\_INSTRUCTORS, TABLE\_CLASSES, CLASSES\_INSTRUCTOR\_ID, TABLE\_INSTRUCTORS, INSTRUCTORS\_ID,

TABLE\_ONLEAVES, TABLE\_CLASSES, CLASSES\_ID, TABLE\_ONLEAVES, ONLEAVES\_CLASS\_ID, TABLE\_RESULTS, RESULTS\_STUDENT\_ID, TABLE\_ONLEAVES, ONLEAVES\_STUDENT\_ID,

TABLE\_CLASSES, CLASSES\_DAYS\_IN\_WEEK, "%" + day\_of\_week + "%", date, TABLE\_CLASSES, CLASSES\_STARTED\_AT, TABLE\_CLASSES, CLASSES\_FINISHED\_AT, TABLE\_RESULTS, RESULTS\_STUDENT\_ID, String.valueOf(studentId));

try {

cursor = db.rawQuery(myQuery, null);

} catch (Exception e) {

db.close();

return null;

}

if (cursor.getCount() == 0) {

cursor.close();

db.close();

return null;

} else {

cursor.moveToFirst();

ArrayList<LeaveReport> leaveReports = new ArrayList<LeaveReport>();

do {

LeaveReport leaveReport = new LeaveReport();

leaveReport.setClass\_id(cursor.getLong(1));

leaveReport.setCourse\_name(cursor.getString(2));

leaveReport.setInstructor\_name(cursor.getString(3));

leaveReport.setTime\_in\_day(cursor.getString(4));

leaveReports.add(leaveReport);

if (cursor.getString(7) != null) {

if (cursor.getString(7).contains(date)) {

leaveReport.setOnleave\_id(cursor.getLong(0));

leaveReport.setStatus(cursor.getLong(5));

leaveReport.setReason(cursor.getString(6));

}

}

} while (cursor.moveToNext());

cursor.close();

db.close();

return leaveReports;

}

}

### Dương Bá Trung Kiên – Chức năng xem các học phần thể chất

**Màn hình thực hiện:**

|  |  |
| --- | --- |
| **A screenshot of a phone  Description automatically generated**  Hình 2‑20. Màn hình học phần thể chất | **A screenshot of a cell phone  Description automatically generated**  Hình 2‑21, Màn hình xem các học phần thể chất |

**Nhiệm của của Activity/Fragment:**

Màn hình học phần thể chất hiển thị 2 button “Các học phần thể chất” và “Kết quả thể chất” khi người dùng click vào Button nào thì màn hình sẽ gọi đến Ativity đó. Khi người dùng click vào button “Các học phần thể chất” sẽ mở ra màn hình Activity PhysicalCourseActivity để hiển thị thông tin các học phần thể chất và số tín tối thiểu phải học. Tên học phần, mã học phần, số tín được lấy từ trong cơ sở dữ liệu.

**Thiết kế giao diện (mã xml):**

**-** Màn hình học phần thể chất:

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="#DCEDC8"

android:orientation="vertical"

tools:context=".PhysicalStudyFragment">

<androidx.appcompat.widget.Toolbar

android:id="@+id/toolbar"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:background="#4CAF50"

android:theme="?attr/actionBarTheme"

app:title="Học phần thể chất"

app:titleTextColor="@color/white" />

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_gravity="center"

android:gravity="center"

android:orientation="vertical">

<Button

android:id="@+id/btnPhysicalCourses"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="36dp"

android:background="#4CAF50"

android:text="Các học phần thể chất"

android:textColor="#FFFFFF"

android:textSize="16sp" />

<Button

android:id="@+id/btnPhysicalResult"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="36dp"

android:background="#4CAF50"

android:text="Kết quả thể chất"

android:textColor="#FFFFFF"

android:textSize="16sp" />

</LinearLayout>

</LinearLayout>

- Màn hình xem các học phần thể chất:

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

tools:context=".PhysicalCoursesActivity">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:orientation="horizontal">

<androidx.appcompat.widget.Toolbar

android:id="@+id/toolbarBack"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:background="#4CAF50"

android:layout\_weight="1"

android:minHeight="?attr/actionBarSize"

android:theme="?attr/actionBarTheme"

app:titleTextColor="@color/white"

app:title="Các học phần thể chất" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

android:orientation="horizontal">

<TextView

android:id="@+id/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Số tín tối thiểu: "

android:textSize="16sp" />

<TextView

android:id="@+id/textView2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="4"

android:textSize="16sp" />

</LinearLayout>

<LinearLayout

android:backgroundTint="@android:color/white"

android:background="@android:color/black"

android:foregroundGravity="center"

android:padding="5dp"

android:elevation="5dp"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_margin="10dp"

android:orientation="vertical">

<ListView

android:id="@+id/lvPhysical"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" />

</LinearLayout>

</LinearLayout>

**Một số mã nguồn, thuật toán chính:**

**-** Trong MyDatabaseHelper.java: phương thức getPhysicalCourses lấy ra danh sách đối tượng Course là các học phần thể chất:

public ArrayList<Course> getPhysicalCourses() {

SQLiteDatabase db = this.getReadableDatabase();

Cursor cursor = null;

try {

cursor = db.query(TABLE\_COURSES,

new String[]{COURSES\_ID, COURSES\_COURSE\_CODE, COURSES\_COURSE\_NAME,

COURSES\_IS\_PHYSICAL, COURSES\_CREDITS, COURSES\_CREATED\_AT,

COURSES\_UPDATED\_AT},

COURSES\_IS\_PHYSICAL + " = 1",

new String[]{},

null, null, null);

} catch (Exception e) {

db.close();

return null;

}

if (cursor.getCount() == 0) {

cursor.close();

db.close();

return null;

} else {

cursor.moveToFirst();

ArrayList<Course> courses = new ArrayList<Course>();

do {

Course course = new Course();

course.setId(cursor.getLong(0));

course.setCourse\_code(cursor.getString(1));

course.setCourse\_name(cursor.getString(2));

course.setIs\_physical(cursor.getLong(3));

course.setCredits(cursor.getLong(4));

course.setCreated\_at(cursor.getString(5));

course.setUpdated\_at(cursor.getString(6));

courses.add(course);

} while (cursor.moveToNext());

cursor.close();

db.close();

return courses;

}

}

### Đỗ Thế Nhuận – Chức năng xem kết quả thể chất, chỉnh sửa thông tin cá nhân

**Màn hình thực hiện:**

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated  Hình 2‑22. Màn hình xem kết quả thể chất | A screenshot of a phone  Description automatically generated  Hình 2‑23. Màn hình chỉnh sửa thông tin cá nhân |

**Nhiệm vụ của Activity/Fragment:**

Khi người dùng click vào button “KẾT QUẢ THỂ CHẤT” sẽ mở ra màn hình Activity PhysicalResultActivity để hiển thị thông tin về kết quả thể chất của người dùng đã học, thông tin về số tín tối thiểu, số tín đã hoàn thành và số tín còn thiếu. Tên học phần, mã học phần, số tín, mã lớp, điểm thường xuyên, điểm giữa kì, điểm thi được lấy từ trong cơ sở dữ liệu SQLite.

Số tín tối thiểu mặc định là 4.

Số tín đã hoàn thành là tổng số tín học phần thể chất người dùng đã học.

Số tín còn thiếu là số tín mà người dùng cần phải hoàn thành.

Điểm tổng kết được tính bằng cách: tổng của 20% điểm thường xuyên cộng 20% điểm giữa kì cộng 50% điểm thi cuối kì.

Điểm tổng kết = 20%\*Điểm TX + 30%\*Điểm GK + 50%\*Điểm KTHP

Điểm chữ được tính dựa trên điểm tổng kết:

- Nếu nhỏ hơn 4 trả về điểm ‘F’

- Nếu từ 4 đến 4.6 trả về điểm ‘D’

- Nếu từ 4.7 đến 5.4 trả về điểm ‘D+’

- Nếu từ 5.5 đến 6.1 trả về điểm ‘C’

- Nếu từ 6.2 đến 6.9 trả về điểm ‘C+’

- Nếu từ 7.0 đến 7.6 trả về điểm ‘B’

- Nếu từ 7.7 đến 8.4 trả về điểm ‘B+’

- Lớn hơn hoặc bình 8.5 trả về điểm ‘A’

Thang điểm 4 được tính dựa trên điểm chữ:

- Nếu điểm chữ đạt ‘A’ trả về điểm 4

- Nếu điểm chữ đạt ‘B+’ trả về điểm 3.5

- Nếu điểm chữ đạt ‘B’ trả về điểm 3

- Nếu điểm chữ đạt ‘C+’ trả về điểm 2.5

- Nếu điểm chữ đạt ‘C’ trả về điểm 2

- Nếu điểm chữ đạt ‘D+’ trả về điểm 1.5

- Nếu điểm chữ đạt ‘D’ trả về điểm 1

- Nếu điểm chữ đạt ‘F’ trả về điểm 0

Xếp loại được tính dựa và thang điểm 4:

- Nếu thang điểm 4 nhỏ hơn 1 trả về loại “Kém”

- Nếu thang điểm 4 từ 1 đến 1.9 về loại “Yếu”

- Nếu thang điểm 4 từ 2 đến 2.4 về loại “Trung bình”

- Nếu thang điểm 4 từ 2.5 đến 3.1 về loại “Khá”

- Nếu thang điểm 4 từ 3.2 đến 3.5 về loại “Giỏi”

- Nếu thang điểm 4 từ 3.6 đến 4 về loại “Xuất sắc”

Fragment Settings (Màn hình chỉnh sửa thông tin cá nhân) hiển thị thông tin cá nhân của người dùng dưới dạng các TextView, EditText và Spinner. Khi truy cập vào Fragment Settings, thông tin cá nhân bao gồm: Mã sinh viên, tên, ngày sinh, giới tính, chiều cao, cân nặng, bệnh nền được hiển thị lên Fragment Settings.

Thông tin cá nhân của người dùng được lấy từ trong cơ sở dữ liệu SQLite

Khi người dùng Click vào nút “LƯU”, các thông tin mới được thay đổi trên màn hình sẽ được cập nhật và lưu vào cơ sở dữ liệu SQLite với dữ liệu tương ứng của người dùng hiện tại.

Khi người dùng Click vào nút “ĐĂNG XUẤT”, màn hình sẽ hiển thị một AlertDialog để thông báo hỏi người dùng có muốn đăng xuất hay không. Nếu người dùng chọn “Không”, Dialog sẽ tắt và người dùng vẫn ở trong ứng dụng với tài khoản đã đăng nhập trước đó. Nếu người dùng chọn “Có”, tài khoản của người dùng hiện tại sẽ được đăng xuất và sẽ chuyển đến màn hình Activity SignInActivity để đăng nhập tài khoản mới.

**Thiết kế giao diện (mã xml):**

**-** Màn hình PhysicalResultActivity:

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

tools:context=".PhysicalResultActivity">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal">

<androidx.appcompat.widget.Toolbar

android:id="@+id/toolbarBack"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:background="#4CAF50"

app:title="Kết quả thể chất"

app:titleTextColor="@color/white" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

android:orientation="horizontal">

<TextView

android:id="@+id/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Số tín tối thiểu: "/>

<TextView

android:id="@+id/textView2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

android:orientation="horizontal">

<TextView

android:id="@+id/textView1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Số tín đã hoàn thành: "/>

<TextView

android:id="@+id/txvCreditsComplete"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

android:orientation="horizontal">

<TextView

android:id="@+id/textView3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Số tín còn thiếu: "/>

<TextView

android:id="@+id/txvCreditsMissing"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_margin="10dp"

android:orientation="vertical">

<ListView

android:id="@+id/lvPhysicalResult"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" />

</LinearLayout>

</LinearLayout>

- Màn hình Settings:

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="#DCEDC8"

android:orientation="vertical">

<TextView

android:id="@+id/txvHeading"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

android:text="THÔNG TIN CÁ NHÂN"/>

<ImageView

android:id="@+id/imgAvatar"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:weightSum="4">

<TextView

android:id="@+id/txvMaSV"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:text="Mã sinh viên:" />

<TextView

android:id="@+id/txvId"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="3"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:weightSum="4">

<TextView

android:id="@+id/txvTen"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:text="Tên:"/>

<TextView

android:id="@+id/txvName"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="3"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:weightSum="4">

<TextView

android:id="@+id/txvNgaySinh"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"/>

<TextView

android:id="@+id/txvBirthday"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="3"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:weightSum="4">

<TextView

android:id="@+id/txvGioiTinh"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:text="Giới tính:"/>

<TextView

android:id="@+id/txvGender"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="3"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:weightSum="4">

<TextView

android:id="@+id/txvChieuCao"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:text="Chiều cao:" />

<EditText

android:id="@+id/edtHeight"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="3"

android:ems="10"

android:inputType="text"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:weightSum="4">

<TextView

android:id="@+id/txvWeight"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:text="Cân nặng:" />

<EditText

android:id="@+id/edtWeight"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="3"

android:ems="10"

android:inputType="text"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:weightSum="4">

<TextView

android:id="@+id/txvDisease"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:text="Bệnh nền:" />

<EditText

android:id="@+id/edtDisease"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="3"

android:inputType="text"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:weightSum="4">

<TextView

android:id="@+id/txvNull"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1" />

<Spinner

android:id="@+id/spnDisease"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_gravity="right"

android:layout\_weight="3"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_gravity="center"

android:gravity="center"

android:orientation="horizontal"

android:weightSum="2">

<Button

android:id="@+id/btnSave"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:background="#4CAF50"

android:text="Lưu"

android:textColor="#FFFFFF"/>

<Button

android:id="@+id/btnLogout"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

android:layout\_weight="1"

android:background="#4CAF50"

android:text="Đăng xuất"

android:textColor="#FFFFFF"/>

</LinearLayout>

</LinearLayout>

**Một số mã nguồn, thuật toán chính:**

**-** Trong MyDatabaseHelper.java: phương thức getPhysicalResults trong đối tượng MyDatabaseHelper để lấy ra danh sách thông tin về học phần và kết quả của học phần đó, updateStudent để cập nhật lại thông tin của người dùng hiện tại:

public ArrayList<PhysicalResult> getPhysicalResults(long studentId) {

SQLiteDatabase db = this.getReadableDatabase();

Cursor cursor = null;

String query = "SELECT courses.course\_code,courses.course\_name, courses.credits, " +

"classes.class\_code, results.exam\_score, results.midterm\_score, results.test1 " +"FROM students " +

"JOIN results ON students.id = results.student\_id " +

"JOIN classes ON results.class\_id = classes.id " +

"JOIN courses ON classes.course\_id = courses.id " +"WHERE students.id = "+ studentId +" AND courses.is\_physical = 1";

String rawQueryData = String.format("SELECT %s.%s, %s.%s, %s.%s, %s.%s, %s.%s, %s.%s, %s.%s " +"FROM %s " +

"JOIN %s ON %s.%s = %s.%s " +"JOIN %s ON %s.%s = %s.%s " +

"JOIN %s ON %s.%s = %s.%s " + WHERE %s.%s = " + "%s" + " AND %s.%s = %s", TABLE\_COURSES, COURSES\_COURSE\_CODE, TABLE\_COURSES, COURSES\_COURSE\_NAME, TABLE\_COURSES, COURSES\_CREDITS, TABLE\_CLASSES, CLASSES\_CLASS\_CODE, TABLE\_RESULTS, RESULTS\_EXAM\_SCORE, TABLE\_RESULTS, RESULTS\_MIDTERM\_SCORE, TABLE\_RESULTS, RESULTS\_TEST1, TABLE\_STUDENTS, TABLE\_RESULTS, TABLE\_STUDENTS, STUDENTS\_ID, TABLE\_RESULTS, RESULTS\_STUDENT\_ID, TABLE\_CLASSES, TABLE\_RESULTS, RESULTS\_CLASS\_ID, TABLE\_CLASSES, CLASSES\_ID, TABLE\_COURSES, TABLE\_CLASSES, CLASSES\_COURSE\_ID, TABLE\_COURSES, COURSES\_ID, TABLE\_STUDENTS, STUDENTS\_ID, String.valueOf(studentId), TABLE\_COURSES, COURSES\_IS\_PHYSICAL, "1");

try {

cursor = db.rawQuery(rawQueryData, null);

} catch (Exception e) {

db.close();

return (new ArrayList<PhysicalResult>());

}

if (cursor.getCount() == 0) {

cursor.close();

db.close();

return (new ArrayList<PhysicalResult>());

} else {

cursor.moveToFirst();

ArrayList<PhysicalResult> results = new ArrayList<PhysicalResult>();

do {

PhysicalResult result = new PhysicalResult();

result.setCourse\_code(cursor.getString(0));

result.setCourse\_name(cursor.getString(1));

result.setCredits(cursor.getLong(2));

result.setClass\_code(cursor.getString(3));

result.setExam\_score(cursor.getLong(4));

result.setMidterm\_score(cursor.getLong(5));

result.setTest1(cursor.getLong(6));

results.add(result);

} while (cursor.moveToNext());

cursor.close();

db.close();

return results;

}

}

//Cập nhật lại thông tin của student vào cơ sở dữ liệu

public boolean updateStudent(String id, Student student) {

SQLiteDatabase db = this.getWritableDatabase();

ContentValues st = new ContentValues();

st.put(STUDENTS\_HEIGHT, student.getHeight());

st.put(STUDENTS\_WEIGHT, student.getWeight());

st.put(STUDENTS\_UNDERLYING\_DISEASE, student.getUnderlying\_disease());

if (db.update(TABLE\_STUDENTS, st, STUDENTS\_ID + " = " + id, null) != 0) {

return true;}

return false;}

# KIẾN THỨC LĨNH HỘI VÀ BÀI HỌC KINH NGHIỆM

## Nội dung đã thực hiện

Trong đề tài này, chúng em đã thực hiện xây dựng được một ứng dụng theo dõi thể chất cho sinh viên HAUI đơn giản, bao gồm các chức năng chính như: đăng nhập, nhập học, đăng xuất, báo cáo sức khoẻ theo ngày, xem lịch sử rèn luyện, báo ốm, báo nghỉ, theo dõi rèn luyện hằng ngày, xem các học phần thể chất, xem kết quả thể chất, chỉnh sửa thông tin cá nhân. Thông qua đề tài, chúng em cũng được nắm vững được các kiến thức quan trọng khi lập trình một ứng dụng Android bằng ngôn ngữ Java, tăng thêm kinh nghiệm phân tích nghiệp vụ bài toán, làm việc nhóm, tìm kiếm và giải quyết các vấn đề gặp phải.

Tuy nhiên, trong khuôn khổ đề tài này, chúng em vẫn chưa thể tìm hiểu được hết những vấn đề và bài toán phát sinh trong đề tài, chưa thể kết nối ứng dụng với hệ thống nhà trường để tạo ra một hệ thống liên kết hoàn chỉnh. Ngoài ra, chúng em cũng chưa thể xây dựng một thể thống chạy trên cơ sở dữ liệu client – server để khiến nó mang tính thực tiễn cao hơn.

## Hướng phát triển

Đề tài có rất nhiều không gian để phát triển trong tương lai. Cải thiện giao diện thân thiện với người dùng hơn, chuyển cơ sở dữ liệu thành mô hình client – server sẽ là một hướng phát triển đơn giản nhưng hiệu quả. Kế tiếp, tích hợp thêm các tính năng nâng cao như xử lý báo ốm, báo nghỉ phía giáo viên và nhà trường, chức năng thông báo cho sinh viên khi đơn báo ốm, báo nghỉ được xử lý, nâng cao hiệu suất, tối ưu hoá hệ thống khi có nhiều người sử dụng đồng thời sẽ là một bước đi quan trọng giúp cho ứng dụng này phát triển mạnh mẽ hơn nữa.

# TÀI LIỆU THAM KHẢO

[1]. Website hướng dẫn lập trình ứng dụng di động. URL: [https://mona.media](https://mona.media/lap-trinh-ios-android). Lần truy cập gần nhất ngày: 24/11/2023.

[2]. Website hướng dẫn lập trình ứng dụng di động. URL: [https://teky.edu.vn](https://teky.edu.vn/blog/lap-trinh-ung-dung-di-dong). Lần truy cập gần nhất ngày: 24/11/2023.

[3]. Nguyễn Bá Nghiễn, Nguyễn Thái Cường, Vũ Thị Dương, Nguyễn Văn Tỉnh: Giáo trình phát triển ứng dụng cho thiết bị di động (Trường Đại học Công nghiệp Hà Nội). Nhà xuất bản Thống Kê, 2022.

[4]. Website giới thiệu lập trình Android. URL: [https://howkteam.vn](https://howkteam.vn/course/khoa-hoc-lap-trinh-android-co-ban/gioi-thieu-lap-trinh-android-va-cai-dat-moi-truong-1). Lần truy cập gần nhất: 24/11/2023.

[5]. Website hướng dẫn cài đặt Android Studio. URL: [https://developer.android.com](https://developer.android.com/studio/install?hl=vi). Lần truy cập gần nhất ngày: 24/11/2023.

[6]. Website hướng dẫn về cách tính BMI cho người Việt Nam, về thời gian luyện tập thể dục trong ngày cho người bình thường và người bị rối loạn chuyển hoá. URL: <https://www.vinmec.com>. Lần truy cập gần nhất ngày: 14/12/2023.

[7]. Website hướng dẫn công thức tính lượng nước uống trong ngày. URL: <https://www.nhathuocankhang.com>. Lần truy cập gần nhất ngày: 15/12/2023.

[8]. Website hướng dẫn tập thể dục cho người bị bệnh hô hấp. URL: <https://www.benhhen.vn>. Lần truy cập gần nhất ngày: 15/12/2023.

[9]. Website hướng dẫn tập thể dục cho người bị bệnh tim mạch. URL: <https://moh.gov.vn>. Lần truy cập gần nhất ngày: 15/12/2023.