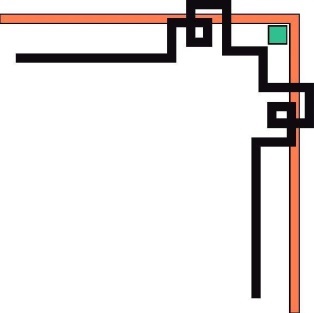
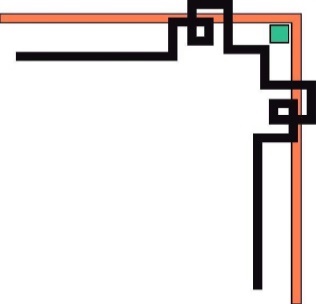
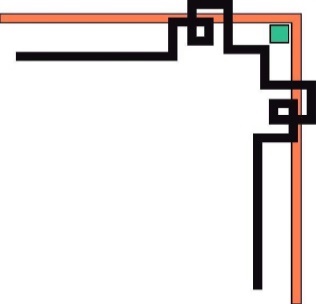
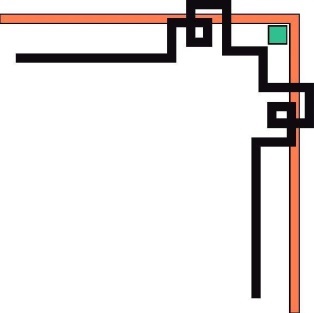
**BỘ GIÁO DỤC VÀ ĐÀO TẠO**



**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

🙠🟅🙢

Logo

Description automatically generated

**BÁO CÁO CUỐI KÌ**

**LẬP TRÌNH DI ĐỘNG**

**ĐỀ TÀI:**

**APP HỖ TRỢ TẬP GYM**

**GVDH: TRẦN VĂN ĐỊNH**

**SVTH:**

**Trần Kiện Khang 19110375**

**Lê Thị Minh Nguyệt 19110413**

**Nguyễn Văn Phú 19110430**

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM

*Thành phố Hồ Chí Minh – Tháng 6/2022*

KHOA CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI ĐỒ ÁN HỌC PHẦN

Giảng viên giảng dạy: Trần Văn Định

TÊN HỌC PHẦN: Lập trình Android

MÃ SỐ LỚP HP:

Tên đề tài: App hỗ trợ tập gym

Ngôn ngữ lập trình sử dụng: Java

Tên đăng nhập và Mật khẩu (password) (nếu có):

Tập dữ liệu thực nghiệm (nếu có):

Nhận xét của giảng viên: ………………………………………………………..

………………………………………………………..

………………………………………………………..

………………………………………………………..

………………………………………………………..

Điểm đánh giá: …………. (…………………)

Ngày……./……../2022

Giảng viên: Trần Văn Định

Ký tên

TP.HCM, tháng 5 năm 2022

**LỜI CẢM ƠN**

Lời đầu tiên, nhóm chúng em xin gửi lời cám ơn sâu sắc nhất đến thầy Trần Văn Định

Trong quá trình tìm hiểu và học tập môn Lập trình Android chúng em đã nhận được sự giảng dạy và hướng dẫn tận tình, tâm huyết từ thầy, giúp chúng em tích lũy thêm rất nhiều kiến thức. Từ những kiến thức thầy đã truyền đạt chúng em xin trình bày lại những gì mình đã làm được về đề tài app hỗ trợ tập gym

Tuy nhiên, kiến thức về môn học của chúng em vẫn còn những hạn chế nhất định và không tránh khỏi những thiếu sót trong quá trình hoàn thành bài báo cáo này. Chúng em mong nhận được lời góp ý, nhận xét đến từ thầy để đề tài báo cáo của chúng em được hoàn thiện hơn.

Chúc em kính chúc thầy sức khỏe, hạnh phúc và thành công hơn nữa trong sự nghiệp trồng người để tiếp tục dìu dắt nhiều thế hệ sinh viên đến những bến bờ tri thức.

Chúng em xin chân thành cảm ơn thầy!

Mục lục

[PHẦN 1: CƠ SỞ LÝ THUYẾT 5](#_Toc105235976)

[1. Giới thiệu về Android 5](#_Toc105235977)

[2. Intent & Manifest trong Lập trình Android 5](#_Toc105235978)

[2.1. Intent 5](#_Toc105235979)

[2.2. Manifest 6](#_Toc105235980)

[3. Activity 6](#_Toc105235981)

[4. Lưu trữ dữ liệu 8](#_Toc105235982)

[5. Broadcast Receiver 8](#_Toc105235983)

[6. Content Provider 9](#_Toc105235984)

[7. Service 10](#_Toc105235985)

[8. Notification 11](#_Toc105235986)

[PHẦN 2: GIỚI THIỆU CHUNG VỀ ĐỀ TÀI 12](#_Toc105235987)

[1. Hiện trạng 12](#_Toc105235988)

[2. Mục tiêu 12](#_Toc105235989)

[3. Đặc tả yêu cầu 12](#_Toc105235990)

[PHẦN 3: MÔ HÌNH HOÁ HỆ THỐNG 13](#_Toc105235991)

[PHẦN 4: GIAO DIỆN VÀ CHỨC NĂNG 14](#_Toc105235992)

[PHẦN 5: KẾT LUẬN 36](#_Toc105235993)

[1. Kết quả đạt được 36](#_Toc105235994)

[1.1. Kiến thức đã tìm hiểu được 36](#_Toc105235995)

[1.2. Chương trình đã làm được 36](#_Toc105235996)

[2. Ưu điểm 36](#_Toc105235997)

[2.1. Hạn chế 36](#_Toc105235998)

[3. Hướng phát triển 36](#_Toc105235999)

[TÀI LIỆU THAM KHẢO 37](#_Toc105236000)

# PHẦN 1: CƠ SỞ LÝ THUYẾT

## Giới thiệu về Android

Android là một hệ điều hành dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng. Ban đầu, Android được phát triển bởi Android Inc. với sự hỗ trợ tài chính từ Google và sau này được chính Google mua lại vào năm 2005.

Android ra mắt vào năm 2007 cùng với tuyên bố thành lập Liên minh thiết bị cầm tay mở: một hiệp hội gồm các công ty phần cứng, phần mềm, và viễn thông với mục tiêu đẩy mạnh các tiêu chuẩn mở cho các thiết bị di động. Chiếc điện thoại đầu tiên chạy Android được bán vào năm 2008.

Android có mã nguồn mở và Google phát hành mã nguồn theo Giấy phép Apache.

Hiện Android là một thương hiệu của Google. Có khả năng tùy biến rất cao và có thể chạy trên nhiều thiết bị, nhiều kiến trúc vi xử lý (ARM / x86).

Về kiến trúc của hệ điều hành Android, chúng ta có thể xem qua sơ đồ sau:

Graphical user interface

Description automatically generated

## Intent & Manifest trong Lập trình Android

### Intent

Intent (ý định) là những tin nhắn không đồng bộ cho phép các thành phần ứng dụng yêu cầu chức năng từ các thành phần Android khác. Intents cho phép bạn tương tác với các thành phần từ các ứng dụng như nhau cũng như với các thành phần đóng gói trong một ứng dụng khác.

Intent có thể sử dụng để:

+ Bắt đầu một Activity

+ Bắt đầu một Activity con.

+ Bắt đầu một dịch vụ (Service).

Android hỗ trợ 2 loại Intent là Intent tường minh (explicit) và Intent không tường minh (implicit).

Một đối tượng intent sẽ mang theo thông tin mà hệ thống android cần để quyết định thành phần nào sẽ đc start. Và những thông tin đó bao gồm :

* Component Name: Tên của component được start.
* Action: Là một chuỗi xác định hành động chung để thực hiện (chẳng hạn như xem hoặc chọn).
* Data: Sử dụng một đối tượng Uri tham chiếu tới dữ liệu sẽ đc thực hiện một hành động nào đó. Loại dữ liệu cung cấp thường được quyết định bởi action của intent.
* Category: Một chuỗi chứa thông tin bổ sung về loại thành phần sẽ xử lý intent. Để nhận đc các intent không tường minh thì bắt buộc ứng dụng phải khai báo category trong manifest, Và giá trị mạc định là CATEGORY\_DEFAULT.
* Extra: Các cặp key-value mang thông tin bổ sung cần thiết để hoàn thành hành động được yêu cầu.
* Flag: Cờ được định nghĩa trong lớp Intent có chức năng như siêu dữ liệu cho intent . Cờ có thể ra lệnh hệ thống Android về cách khởi chạy một (ví dụ, hoạt động sẽ thuộc về tác vụ nào ) và cách xử lý sau khi nó được khởi chạy

### Manifest

Mọi ứng dụng đều bắt buộc phải có một file AndroidManifest.xml ở thư mục gốc.

AndroidManifest cung cấp thông tin cơ bản của ứng dụng cho hệ điều hành Android ví dụ như:

* Tên java package của ứng dụng.
* Các thành phần của ứng dụng, như danh sách các thành phần của ứng dụng, như activity, services, broadcast receiver,…
* Tính chất của các thành phần trong ứng dụng (ví dụ, một app có nhiều activity thì activity nào sẽ được gọi đầu tiên khi vào app).
* Đánh giá các tiến trình chứa runtime của ứng dụng.
* Danh sách các quyền truy cập của ứng dụng cần được người dùng cho phép.
* Phiên bản Android tối thiểu.
* Các thư viện liên kết với ứng dụng Android.

## Activity

Activity trong Android là nơi diễn ra mọi hoạt động tương tác với người dùng, bởi vì tất cả các màn hình ứng dụng đều phải được “đính” trên một Activity.

Activity là một trong 4 thành phần quan trọng của ứng dụng Android gồm: Activity, Service, Content Provider, Broadcast receivers.

Việc xử lý logic tương ứng với từng trạng thái của Activity là việc rất quan trọng. Điều đó đảm bảo ứng dụng hoạt động ổn định và đáng tin cậy với người dùng. Bạn có thể tham khảo hình ảnh minh họa vòng đời Activity

Diagram

Description automatically generated

Các trạng thái vòng đời của một activity trong android:

+ onCreate(): Được gọi bởi OS khi activity được tạo(chỉ gọi duy nhất một lần). Tại đây, bạn có thể khởi tạo giao diện hoặc dữ liệu để dùng cho sau này. Ngoài ra, bạn để ý tham số savedInstanceState. Tham số này chính là trạng thái trước đó của Activity. Mục đích của nó là để bạn có thể khôi phục lại trạng thái của Activity

+ onStart(): Ngay trước khi activity được chạy và tương tác với người dùng thì hàm được gọi. Thông thường, tại hàm này chúng ta sẽ khởi tạo animation cho UI( nếu có), Audio, hay bất kì đối tượng nào cần thiết để activity có thể hiển thị cho người dùng

+ onResume(): Hàm này được gọi khi activity chuẩn bị chạy hoặc activity được kích hoạt trở lại từ trạng thái stop(Ví dụ: người dùng ấn phím HOME hoặc có cuộc gọi đến…). Tại hàm này, thường chúng ta sẽ khởi động lại animation, cập nhật UI, hay khởi động lại camera, Audio/Video playback… Nói chung là khỏi động lại tất cả những mà bạn đã release ở hàm onPause()

+ onPause(): Hàm này được gọi khi ứng dụng chuyển sang chạy nền( Tức là ứng dụng không hiển thị trên màn hình). Ngược với onResume(), tại hàm này chúng ta nên stop Animation, Audio/video playback, tạm dừng camera… Lý do đơn giản vì lúc này ứng dụng không hiển thị và tương tác với người dùng nữa.

+ onStop(): Hàm này được ngay sau onPause(), khi activity không còn tương tác với người dùng. Tại đây, tốt nhất là bạn nên tiến hành sao lưu dữ liệu để có thể khôi phục lại khi activity được kích hoạt lại.

+ onRestart(): Được gọi sau khi activity bị tạm dừng và giờ được kích hoạt trở lại

+ onDestroy(): Đây là hàm cuối cùng trong vòng đời của activity. Có 2 cách để hàm được gọi: một là bạn gọi hàm finish() để chủ động tắt một activity, hai là được gọi tự động bởi OS khi tài nguyên cạn kiệt. Tại hàm này, bạn nên giải phóng toàn bộ những task nào đang chạy nền, hoặc những đối tượng(object) toàn cục để tránh bị lỗi Memory Leak.

## Lưu trữ dữ liệu

Hệ điều hành Android có hỗ trợ kha khá các phương thức lưu trữ dữ liệu. Việc bạn chọn phương thức nào tuỳ thuộc vào nhu cầu và tính chất của nhu cầu đó, cũng như dung lượng cần lưu trữ là bao nhiêu.

Các phương thức nói trên bao gồm:

* **Shared Preferences:** Lưu trữ dữ liệu dạng **Key-Value**.
* **Internal Storage:** Lưu trữ dữ liệu vào vùng nhớ của thiết bị.
* **External Storage:**Lưu trữ dữ liệu vào vùng nhớ ngoài được chia sẻ.
* **SQLite Database:**Lưu trữ dữ liệu vào cơ sở dữ liệu được tích hợp trong Android. Cơ sở dữ liệu này là **SQLite**.
* **Network:**Lưu trữ dữ liệu trên server riêng của bạn, truy cập và lấy về qua API của server.

**SharedPreferences**là một Framework, nói đúng hơn là một tính năng giúp bạn lưu giữ liệu dưới dạng khoá-giá trị.

* Mỗi **SharedPreferences**đều đi kèm với một **Context**.
* Gọi hàm edit() để tạo ra một object của SharedPreferences.**Editor**()
* Thêm dữ liệu bằng các phương thức như **putString**/ **putBoolean**,…
* Thực thi chỉnh sửa bằng phương thức commit().

Ngoại trừ một số loại file trên External Storage, thì tất cả các cách lưu trữ này dành cho dữ liệu riêng tư của ứng dụng - dữ liệu không thể truy cập tự nhiên vào các ứng dụng khác. Nếu muốn chia sẻ file với các ứng dụng khác nên sử dụng FileProvider API.

Firebase là dịch vụ cơ sở dữ liệu hoạt động trên nền tảng đám mây – cloud. Kèm theo đó là hệ thống máy chủ cực kỳ mạnh mẽ của Google. Chức năng chính là giúp người dùng lập trình ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu.

## Broadcast Receiver

Android Broadcast Receiver là một thành phần nơi bạn có thể đăng ký sự kiện của hệ thống hay ứng dụng. Bạn sẽ nhận được thông báo về các sự kiện đã đăng ký trước đó. Việc phát tin broadcast có thể bắt nguồn từ hệ thống hoặc từ các ứng dụng

Ví dụ: một số broadcast từ hệ thống như thông báo pin yếu, bật tắt màn hình, kết nối hay ngắt kết thiết bị ngoại vi…

Để truyền thông tin, Broadcast receiver sử dụng Intent để đóng gói dữ liệu.

## Content Provider

Diagram

Description automatically generated

Content provider là một thành phần để quản lý truy cập dữ liệu, nó cung cấp các phương thức khác nhau để các ứng dụng có thể truy cập dữ liệu từ một ứng dụng khác bằng cách sử dụng ContentResolver. Content Provider có thể giúp cho một ứng dụng quản lý quyền truy cập đến dữ liệu được lưu bởi ứng dụng đó, hoặc các ứng dụng khác, và đó là một cách để ta có thể chia sẻ dữ liệu cho các ứng dụng khác nhau. Hình dưới đây biểu diễn cho việc cách content providers quản lý việc truy cập tới bộ nhớ

Content Provider điều phối việc truy cập tới bỗ lưu trữ dữ liệu thông qua các API và các component như hình dưới, nó bao gồm

* Chia sẻ dữ liệu từ ứng dụng của bán tới các ứng dụng khác
* Gửi dữ liệu sang widget
* Trả về một kết quả gợi ý khi search cho ứng dụng của bạn thông qua Seach Framework sử dụng [SearchRecentSuggestionsProvider](https://developer.android.com/reference/android/content/SearchRecentSuggestionsProvider.html" \t "_blank)
* Đồng bộ dữ liệu của ứng dụng với server bằng cách sử dụng [AbstractThreadedSyncAdapter](https://developer.android.com/reference/android/content/AbstractThreadedSyncAdapter.html" \t "_blank)
* Tải dữ liệu lên UI sử dụng [CursorLoader](https://developer.android.com/reference/android/content/CursorLoader.html" \t "_blank)

Content Provider hoạt động rất giống với một cơ sở dữ liệu, bạn có thể truy vấn, chỉnh sửa nội dung, cũng như là thêm xóa các nội dung sử dụng các phương thức: insert(), update(), delete(), query().

**Sử dụng Content Provider:**

Để sử dụng Content Provider ta làm theo các bước sau:

1. Xác định kiểu dữ liệu
2. Xác định Uniform Resource Identifier (URI)
3. Khai báo Content Provider trong manifest
4. Implement lớp ContentProvider và các phương thức được yêu cầu.

Các phương thức cần được Override trong lớp Content Provider:

* onCreate(): Phương thức này được gọi khi Provider được bắt đầu, nếu quá trình khởi tạo thành công trả về true, ngược lại là false
* query(): Phương thức nhận yêu cầu từ Client. Kết quả được trả về như một đối tượng Cursor.
* insert(): Phương thức chèn một dòng dữ liệu mới vào Content Provider.
* delete(): Phương thức xóa một dòng dữ liệu đã tồn tại.
* update(): Phương thức cập nhật một dòng dữ liệu nào đó đã tồn tại.
* getType(): Phương thức trả về kiểu MIME của dữ liệu tại các URI.

## Service

Service là một trong 4 thành phần của ứng dụng Android, có thể thực hiện các tác vụ cần nhiều thời gian và không hề có giao diện người dùng(UI). Từ Activity có thể khởi chạy một Service trong Android và sau đó thể tắt ứng dụng để chuyển sang ứng dụng khác mà Service vẫn tiếp tục công việc mà không bị kill.

Việc xử lý các tác vụ cần nhiều thời gian trên main UI sẽ làm cho ứng dụng bị treo, giảm trải nghiệm người dùng.Vì vậy, khi cần xử lý bất kì tác vụ cần thời gian thì Service trong Android là một trong số ứng viên có thể xem xét.

* Service không có giao diện người dùng nào và không thể communicate trực tiếp với activity.
* Một service có thể chạy dưới background vô thời hạn, ngay cả khi ứng dụng bị tắt theo cách thông thường như ấn phím Back, hay Home, hay tắt trong Recent History…
* Thông thường một service thực hiện một công việc đơn lẻ và tự dừng lại khi nhiệm vụ hoàn thành

Các loại service trong Android: Bound và Unbound

Cũng giống như Activity, Service trong android cũng có vòng đời từ lúc bắt đầu cho đến khi destroy.

Diagram

Description automatically generated

## Notification

Một notification là một thông điệp mà Android hiển thị bên ngoài giao diện ứng dụng của bạn để cung cấp cho người dùng lời nhắc, tin nhắn, hay bất kỳ thông tin gì từ ứng dụng của bạn. Người dùng có thể nhấn vào notification để mở ứng dụng, hoặc thực hiện một hành động trực tiếp trên notification như gửi tin nhắn.

# PHẦN 2: GIỚI THIỆU CHUNG VỀ ĐỀ TÀI

## Hiện trạng

Với cuộc sống ngày càng phát triển như hiện nay, ai cũng muốn mình sẽ ngày càng đẹp hơn ngày càng giỏi hơn để gây ấn tượng tốt với mọi người xung quanh. Chính vì thế mà nhu cầu tập gym của mọi người càng tang cao. Tập gym là một hình thức vận động thông qua sự hỗ trợ của máy móc vào các cơ nhằm giúp cơ thể khỏe mạnh. Giúp các bạn trẻ tự tin hơn về vóc dáng của mình. Khi đó nếu mà việc quản lý việc tập luyện của bản thân mình thông qua trí nhớ hay ghi chép bằng giấy mỗi ngày rất bất tiện và đôi khi mình lại bị quên và sẽ dẫn đến kết quả tập không được như ý.

Ví dụ như ngày hôm đó mình tập những gì, có thêm bài tập nào mới hay không, bài tập hôm sau có tiến bộ hơn hôm trước hay không, … Các việc đó mình phải cập nhật liên tục mà ghi nhớ mỗi ngày rất vất vả.

Nhận thấy những yêu cầu thiết yếu của quản lý việc tập gym của mỗi người thì chúng em đã làm ra một cái app giúp mỗi người có thể dễ dàng quản lý việc tập luyện của bản thân để sớm có được vóc dáng như ý. Tóm lại, app này ra đời nhằm phục vụ những nhu cầu cần thiết của mỗi người tập gym hiện nay.

## Mục tiêu

Xây dựng một app Hỗ trợ tập gym đơn giản, thân thiện, dễ sử dụng cho phép người dùng dễ dàng theo dõi quản lý việc tập luyện mỗi ngày của mình một cách thuận tiện và hợp lý. App được thiết kế với giao diện hài hoà dễ sử dụng.

## Đặc tả yêu cầu

App được xây dựng phục vụ người dùng quản lý việc tập luyện hằng ngày:

-Theo dõi bài tập

-Tạo các bài tập tùy chỉnh

-Tạo bình luận bài tập

-Thêm/ Sửa, xóa bài tập

-Phân tích cơ bản về dữ liệu tập luyện

-Thống kê biểu đồ

-Trực quan hóa dữ liệu tập luyện

-Khối lượng, thực tế, ước tính và theo dõi số liệu thống kê khác

-Hẹn giờ nghỉ ngơi

-Reset/ sao lưu dữ liệu

# PHẦN 3: MÔ HÌNH HOÁ HỆ THỐNG

Class Diagram

Diagram

Description automatically generated

# PHẦN 4: GIAO DIỆN VÀ CHỨC NĂNG

Giao diện chính của app

A picture containing shape

Description automatically generated

Ở phần bài tập có rất nhiều bài tập để ta lựa chọn quản lý

**Table

Description automatically generated**

Có thể tìm kiếm bài tập theo tên hoặc theo phần cơ thể cần tập

**Graphical user interface, application

Description automatically generated**

Có thể thêm bài tập mới

Graphical user interface, application

Description automatically generated

Khi đè vô mỗi bài tập sẽ có quyền Chỉnh sửa tên bài tập hoặc xoá bài tập

Graphical user interface, application

Description automatically generated

Giao diện chỉnh sửa bài tập

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

Description automatically generated

Giao diện xoá bài tập

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

Description automatically generated

Khi nhấn vào mỗi bài tập thì sẽ có giao diện để lưu lại cân nặng và số lần lặp lại của mỗi bài tập

Graphical user interface, application

Description automatically generated

Chúng ta có thể lưu nhiều lần tập trong một bài tập trong một ngày

Table

Description automatically generated

Khi lỡ tạo dư ta có thể xoá nó đi một cách dễ dàng

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

Description automatically generated

Ở cùng hang với tên bài tập sẽ có 4 cái icon tương ứng với 4 chức năng khác nhau

Graphical user interface

Description automatically generated with low confidence

Icon đầu tiên sẽ cho ta xem sơ lược lại bài tập này ở các ngày vừa qua mình đã tập như thế nào

Graphical user interface, application

Description automatically generated

Icon thứ 2 sẽ hiện ra biểu đồ cho thấy quá trình tập luyện của mình xem đang tiến bộ hay đi xuống

Chart, line chart

Description automatically generated

Ở icon tiếp theo hình cái đồng hồ thì sẽ là thời gian đếm ngược để mình chỉnh thời gian tập hoặc nghỉ ngơi cho hợp lý

Graphical user interface

Description automatically generated

Ở icon cuối cùng thì mình có thể them phần ghi chú như Đã xong

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

Description automatically generated

Tiếp đến với giao diện Nhật ký thì sẽ hiện ra toàn bộ các bài tập mà mình đã tập được qua các ngày

Graphical user interface, application

Description automatically generated

Khi nhấn vào các ngày thì sẽ hiện ra bảng tổng hợp về các bài tập

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

Description automatically generated

Còn khi nhấn vào tên các bài tập sẽ có tính toán chi tiết hơn về bài tập đó

Graphical user interface, application

Description automatically generated

Ở đây chúng ta cũng có thể xem lại các ghi chú

Graphical user interface, application

Description automatically generated

Còn icon đốm lửa sẽ hiện khi bài tập hôm sau tập với cường độ khó hơn hôm trước

Ở giao diện GymDiary sẽ thấy tổng hợp các bài tập trong ngày của mình

Graphical user interface, application

Description automatically generated

Ở phần giao diện Biểu đồ mình sẽ thấy được các biểu đồ hình tròn biểu diễn số lượng ngày tập mỗi năm, mình tập luyện tập trung vào phần nào nhiều nhất, tổng số bài tập mỗi lần tập

A picture containing chart

Description automatically generated

Chart

Description automatically generated

Ở phần giao diện của cá nhân thì có phần cài đặt

Shape

Description automatically generated with low confidence

Các chức năng của phần Cài đặt app

Graphical user interface, application

Description automatically generated

Mình có thể Export sang bộ nhớ điện thoại hoặc xoá hết dữ liệu trong app

Graphical user interface, text, application

Description automatically generated Graphical user interface, text, application, chat or text message

Description automatically generated

# PHẦN 5: KẾT LUẬN

## Kết quả đạt được

Sau một thời gian nghiên cứu và thực hiện đồ án, nhóm chúng em đã đạt được những kết quả như sau:

### Kiến thức đã tìm hiểu được

* Tìm hiểu được về các lý thuyết cần thiết trong Lập trình android
* Biết lưu trữ dữ liệu bằng nhiều cách như sqlite, firebase
* Biết các kiến thức xây dựng và phát triển một ứng dụng dành cho điện thoại
* Nắm rõ thế nào là lập trình hướng đối tượng trong java và áp dụng các tính chất cơ bản của nó vào xây dựng đồ án.

### Chương trình đã làm được

* Code và cài đặt được app hỗ trợ tập gym
* Thông qua việc thực hiện đồ án giúp chúng em hiểu biết thêm về lập trình di động và các kĩ thuật cũng như kĩ năng cơ bản của lập trình di động.

## Ưu điểm

* Giao diện dễ sử dụng
* Code chạy nhanh ít lỗi
* App sử dụng đúng mục đích yêu cầu đưa ra

## Hạn chế

* Còn một số chức năng vẫn chưa hoàn thiện
* Các rang buộc về dữ liệu vẫn chưa đầy đủ
* App còn quá đơn giản so với nhu cầu thực tế

## Hướng phát triển

* Tiếp tục nghiên cứu và hoàn thiện các chức năng còn thiếu và chưa hoàn chỉnh
* Tìm hiểu cách cài đặt chương trình để người dùng có thể sử dụng một cách tiện lợi nhất.
* Thiết kế cơ sở dữ liệu bảo mật hơn
* Có thêm chức năng để các người dùng có cùng sở thích tập gym dễ kết nối với nhau hơn
* Nghiên cứu tìm hiểu thực tế để phát triển các tính năng mới phù hợp với xu thế hiện nay như kết nối với smartwatch, them chức năng quản lý việc ăn uống, tính lượng calo…

# TÀI LIỆU THAM KHẢO

1. Trần Văn Định, *Bài giảng học phần Lập trình Android*, TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM.
2. Kteam, *Khoá học lập trình android cơ bản*, <https://howkteam.vn/course/khoa-hoc-lap-trinh-android-co-ban-3>
3. *Các hướng dẫn lập trình android*, <https://openplanning.net/11007/android>
4. *Documentation for app developers,* [*https://developer.android.com/docs*](https://developer.android.com/docs)