

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC CẦN THƠ  
KHOA CÔNG NGHỆ THÔNG TIN – TRUYỀN THÔNG**



**NIÊN LUẬN CÔNG NGHỆ THÔNG TIN**

**Đề tài:**

**XÂY DỰNG ỨNG DỤNG TIẾP NHẬN PHẢN ÁNH  
TRỰC TUYẾN CỦA NGƯỜI DÂN**

**Giảng viên hướng dẫn**

**TS. Trần Công Ân**

**Sinh viên thực hiện**

**Trần Phước Hậu – C1800007**

**Nguyễn Tấn Đạt – C1800005**

**Cần Thơ, 2021**

## LỜI CẢM ƠN

Sự thành công nào mà không gắn liền với những sự hỗ trợ, giúp đỡ dù ít hay nhiều, dù trực tiếp hay gián tiếp của người khác. Trong suốt thời gian từ khi bắt đầu học tập ở **Trường Đại Học Cần Thơ** đến nay, chúng em đã nhận được rất nhiều sự quan tâm, giúp đỡ của quý Thầy Cô và bạn bè.

Cho phép chúng em xin gửi lời cảm ơn chân thành đến quý Thầy Cô **Khoa Công Nghệ Thông Tin Và Truyền Thông - Trường Đại Học Cần Thơ**. Trong suốt thời gian qua, quý Thầy Cô đã tận tình chỉ dạy, truyền đạt những kiến thức cũng như kinh nghiệm quý báu cho chúng em. Trong học kỳ này, Khoa đã tạo điều kiện cho chúng em được làm đề tài niên luận cơ sở, đó được xem như thành quả mà chúng em có được khi học tập tại ngôi trường thân yêu này. Đồng thời chúng em xin được bày tỏ lòng biết ơn sâu sắc của thầy **Trần Công Án**, thầy là người đã hướng dẫn chúng em hoàn thành đề tài của mình.

Do thời gian có hạn cũng như kinh nghiệm, kiến thức còn thiếu nhiều nên đề tài báo cáo này sẽ không tránh khỏi những thiếu sót, hạn chế nhất định. Chúng em rất mong nhận được nhận xét góp ý quý báu của Thầy Cô để kiến thức của chúng em trong lĩnh vực này hoàn thiện tốt hơn.

Cuối lời, chúng em xin kính chúc quý Thầy Cô **Trường Đại Học Cần Thơ** luôn luôn mạnh khỏe để tiếp tục truyền đạt lại những kiến thức và kinh nghiệm cho thế hệ mai sau.

Chúng em xin chân thành cảm ơn !

## This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

TS. Trần Công Ân

## MỤC LỤC

LỜI CẢM ƠN .....	i
NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN HƯỚNG DẪN.....	ii
DANH MỤC HÌNH ẢNH.....	v
TÓM LƯỢC .....	vii
PHẦN GIỚI THIỆU.....	1
1. Đặt vấn đề .....	1
2. Mục tiêu đề tài .....	1
3. Đối tượng và phạm vi nghiên cứu .....	1
4. Phương pháp nghiên cứu .....	2
5. Kết quả đạt được .....	2
6. Bố cục niên luận.....	2
PHẦN NỘI DUNG .....	3
CHƯƠNG 1: ĐẶT TẢ YÊU CẦU.....	3
1.1 Khái niệm về phần mềm ứng dụng.....	3
1.2 Mô tả bài toán .....	3
CHƯƠNG 2: THIẾT KẾ GIẢI PHÁP .....	4
2.1 Giới thiệu về React Native .....	4
2.2 Giới thiệu về Android studio .....	6
2.3 Giới thiệu về Firebase.....	7
2.4 Thiết kế các chức năng của ứng dụng .....	10
2.5 Thiết kế cơ sở dữ liệu .....	11
CHƯƠNG 3: CÀI ĐẶT GIẢI PHÁP .....	12
3.1 Cài đặt môi trường lập trình Android.....	12
3.2 Thiết kế giao diện .....	30
CHƯƠNG 4: KIỂM THỬ VÀ KẾT QUẢ .....	36
4.1 Thiết bị kiểm thử .....	36
4.2 Mục tiêu kiểm thử.....	36

4.3 Kịch bản kiểm thử .....	36
4.4 Kết quả kiểm thử .....	37
PHẦN KẾT LUẬN .....	38
1. Kết quả đạt được .....	38
2. Hạn chế .....	38
3. Hướng phát triển .....	38
TÀI LIỆU THAM KHẢO .....	39

## DANH MỤC HÌNH ẢNH

Hình 1: React native .....	4
Hình 2: Sơ đồ giao diện người dùng .....	10
Hình 3: Sơ đồ cơ sở dữ liệu.....	11
Hình 4: Cài đặt Java JDK (1) .....	12
Hình 5: Cài đặt Java JDK (2) .....	13
Hình 6: Cài đặt Java JDK (3) .....	13
Hình 7: Cài đặt biến môi trường Java (1).....	14
Hình 8: Cài đặt biến môi trường Java (2).....	14
Hình 9: Cài đặt biến môi trường Java (3).....	15
Hình 10: Cài đặt biến môi trường Java (4).....	16
Hình 11: Cài đặt biến môi trường Java (5).....	16
Hình 12: Cài đặt biến môi trường Java (6).....	17
Hình 13: Cài đặt biến môi trường Java (7).....	18
Hình 14: Kiểm tra phiên bản Java đã được cài đặt .....	18
Hình 15: Tải Android Studio (1) .....	19
Hình 16: Tải Android Studio (2) .....	19
Hình 17: Tải Android Studio (3) .....	20
Hình 18: Cài đặt Android Studio (1).....	20
Hình 19: Cài đặt Android Studio (2).....	21
Hình 20: Cài đặt Android Studio (3).....	21
Hình 21: Cài đặt Android Studio (4).....	22
Hình 22: Cài đặt Android Studio (5).....	22
Hình 23: Cài đặt Android Studio (6).....	23
Hình 24: Cài đặt Android Studio (7).....	23
Hình 25: Cài đặt Android Studio (8).....	24
Hình 26: Cài đặt Android Studio (9).....	25
Hình 27: Cài đặt Android Studio (10).....	25

Hình 28: Cài đặt Android Studio (11).....	26
Hình 29: Cài đặt Android Studio (12).....	26
Hình 30: Cài đặt Android Studio (13).....	27
Hình 31: Cài đặt Android Studio (14).....	27
Hình 32: Cài đặt Android Studio (15).....	28
Hình 33: Kiểm tra sau khi cài đặt Android Studio (1) .....	28
Hình 34: Kiểm tra sau khi cài đặt Android Studio (2) .....	29
Hình 35: Kiểm tra sau khi cài đặt Android Studio (3) .....	29
Hình 36: Giao diện đăng nhập.....	30
Hình 37: Giao diện trang chủ của người quản trị.....	30
Hình 38: Giao diện xem các phản ánh đang chờ duyệt và đã duyệt .....	31
Hình 39: Giao diện duyệt phản ánh.....	31
Hình 40: Giao diện lịch sử trả lời của phản ánh.....	32
Hình 41: Giao diện cập nhật tiện ích và tình trạng giao thông .....	32
Hình 42: Giao diện trang chủ của người dùng .....	33
Hình 43: Giao diện bước 1 của chức năng phản ánh .....	33
Hình 44: Giao diện bước 2 của chức năng phản ánh .....	34
Hình 45: Giao diện bước 3 của chức năng phản ánh .....	34
Hình 46: Giao diện bước 4 của chức năng phản ánh .....	35
Hình 47: Giao diện bản đồ tiện ích .....	35

## TÓM LƯỢC

Ứng dụng “**Thông tin giao thông**” là một phần mềm được viết bằng ngôn ngữ Javascript, sử dụng thư viện React Native và cơ sở dữ liệu Firebase Cloud Firestore. Ứng dụng này tập trung phát triển các chức năng hỗ trợ tiếp nhận phản ánh, thông tin trực tuyến của người dân trên thiết bị di động một số vấn đề như: lấn chiếm lòng lề đường; các sự cố hạ tầng giao thông như tình trạng kẹt xe, đèn giao thông, đèn đường bị hỏng, cây xanh, thoát nước, đường bị hỏng,... Hệ thống góp phần giúp thông tin nhanh chóng đến các đơn vị quản lý để xử lý các sự cố trong thời gian ngắn nhất. Người dùng có thể đăng tải phản ánh bằng hình ảnh trực tiếp từ thiết bị di động, xem danh sách các phản ánh kèm nội dung và thời gian cập nhật. Ngoài ra, người dùng còn có thể thấy được thông tin phản hồi từ phía đơn vị quản lý. Trong trường hợp không thể kết nối mạng, phản ánh sẽ được lưu lại offline trên máy. Đến khi có kết nối mạng, người dùng dễ dàng gửi lại từ danh sách các phản ánh của mình.

Ứng dụng có giao diện đơn giản và dễ dàng sử dụng, các thao tác được thực hiện tuần tự nên người dùng sẽ dễ tiếp cận hơn. Tuy nhiên do kiến thức còn hạn chế nên một số chức năng chưa được tự động hoá mà người dùng vẫn phải thực hiện một cách thủ công. Đó cũng là hướng mà ứng dụng mong muốn sẽ phát triển trong tương lai.



## PHẦN GIỚI THIỆU

### 1. Đặt vấn đề

Từ những thập niên cuối thế kỷ XX đến nay, nhân loại đã phát minh và chứng kiến sự tiến triển thần kỳ của công nghệ thông tin. Những thành tựu của công nghệ thông tin đã góp phần rất quan trọng cho quá trình hình thành và phát triển nền kinh tế tri thức và xã hội thông tin mang tính chất toàn cầu. Công nghệ thông tin đóng vai trò quan trọng và không thể thiếu trong đời sống. Với sự phát triển với tốc độ nhanh chóng của công nghệ thông tin, cùng với sự phát triển của xã hội, công nghệ thông tin đóng vai trò ngày càng quan trọng. Việc áp dụng công nghệ thông tin vào các lĩnh vực xã hội như: quản lý, kinh tế, nghiên cứu,... đã cho ra đời những phần mềm ứng dụng để thay thế về cơ bản các công tác thủ công, giảm nhẹ tới mức tối thiểu việc sử dụng sức người trong công tác tiếp nhận và quản lý, tăng cường hiệu quả, tiết kiệm chi phí, thời gian và công sức lao động.

Các cơ quan, đơn vị quản lý cũng đang từng bước tin học hóa trong các công việc tiếp nhận và quản lý. Trong đó bao gồm việc tiếp nhận các thông tin của người dân về các vấn đề giao thông, xe cộ, đường xá, cây xanh.... Với việc lưu lượng tham gia giao thông ngày càng tăng, số lượng tuyến đường, đèn đường, cây xanh, tình trạng kẹt xe... cũng tăng theo thì công tác phát hiện và xử lý sự cố sẽ gặp nhiều khó khăn. Do đó, để giảm bớt những khó khăn chung, mang lại nhiều lợi ích cho việc phản ánh của người dân và giúp cho đơn vị quản lý để xử lý sự cố trong thời gian ngắn nhất. Vì vậy chúng em quyết định chọn đề tài ***“Xây dựng ứng dụng tiếp nhận phản ánh trực tuyến của người dân”***.

### 2. Mục tiêu đề tài

Xây dựng ứng dụng nhằm giúp cho việc phản ánh thông tin của người dân kịp thời, giúp cho đơn vị quản lý tiếp nhận và xử lý nhanh chóng hơn. Cập nhật, tìm kiếm, thống kê,... các thông tin một cách dễ dàng và chính xác.

Tự động hóa cho việc phản ánh của người dân, công tác tiếp nhận của đơn vị quản lý. Nhằm tạo nên tính chuyên nghiệp, tiết kiệm được công sức, thời gian và chi phí.

### 3. Đối tượng và phạm vi nghiên cứu

Trong giới hạn phạm vi của đề tài này chỉ xét đến một số khía cạnh nhỏ trong việc phản ánh thông tin của người dân và việc tiếp nhận thông tin của đơn vị quản lý một số vấn đề về các sự cố hạ tầng giao thông như tình trạng kẹt xe, đèn giao thông, đèn đường bị hỏng, cây xanh, thoát nước, đường bị hỏng,...

#### 4. Phương pháp nghiên cứu

Mục đích đề tài *“Xây dựng ứng dụng tiếp nhận phản ánh trực tuyến của người dân”* nhằm phục vụ tốt nhất cho công tác xử lý sự cố trong thời gian ngắn nhất. Vì vậy, để xây dựng được một ứng dụng đạt hiệu quả cao cần phải có phương pháp phù hợp và tìm hiểu thực tiễn như:

- Thu thập thông tin từ Thầy Cô và bạn bè.
- Thu thập thông tin trên mạng Internet.
- Nghiên cứu tài liệu và đề tài có liên quan.
- Tổng hợp thông tin, phân tích và xây dựng cơ sở dữ liệu.

#### 5. Kết quả đạt được

Xây dựng ứng dụng tiếp nhận phản ánh trực tuyến của người dân trên thiết bị di động. Ứng dụng này dễ sử dụng, tương thích nhiều thiết bị. Ứng dụng luôn cập nhật dữ liệu theo thời gian thực, chỉ cần bạn có Internet và một chiếc điện thoại thông minh thì bạn có thể phản ánh các thông tin về đường xá, cầu cống, ... và xem được tình trạng giao thông trên các tuyến đường.

#### 6. Bố cục niên luận

Phần giới thiệu

Giới thiệu tổng quát về đề tài đang thực hiện

Phần nội dung

Chương 1: Đặt tả yêu cầu

Chương 2: Thiết kế giải pháp

Chương 3: Cài đặt giải pháp

Chương 4: Kiểm thử và kết quả

Phần kết luận

Trình bày kết quả đạt được, ưu điểm, hạn chế và hướng phát triển

## PHẦN NỘI DUNG

### CHƯƠNG 1: ĐẶT TẢ YÊU CẦU

#### 1.1 Khái niệm về phần mềm ứng dụng

**Phần mềm ứng dụng** trên thiết bị di động còn được gọi tắt là ứng dụng di động, hoặc chỉ ứng dụng, (tiếng Anh: mobile application hoặc mobile app hoặc app) là phần mềm ứng dụng được thiết kế để chạy trên điện thoại thông minh, máy tính bảng và các thiết bị di động khác.

Ứng dụng di động ban đầu được cung cấp với mục đích thông tin tổng quát và các dịch vụ thông dụng trên mạng toàn cầu, bao gồm email, lịch, danh bạ, và thị trường chứng khoán và thông tin thời tiết. Tuy nhiên, nhu cầu chung của những người sử dụng thiết bị di động và khả năng phát triển của các nhà lập trình đã mở rộng thành các loại khác, chẳng hạn như trò chơi di động, tự động hóa nhà máy, GPS và các dịch vụ dựa trên địa điểm, định vị và ngân hàng, để theo dõi, mua vé và các ứng dụng y tế di động gần đây. Sự bùng nổ về số lượng và sự đa dạng của các ứng dụng đã tạo ra một tiềm năng và thị trường lớn.

#### 1.2 Mô tả bài toán

- Ứng dụng **“Thông tin giao thông”** là phần mềm được viết dựa trên ngôn ngữ Javascript tương tác trên cơ sở dữ liệu Firebase Cloud Firestore giúp cho việc phản ánh và tiếp nhận thông tin về tình trạng giao thông một cách tự động hóa, các công việc diễn ra được nhanh chóng, giúp tiết kiệm được thời gian, xử lý xự cố trong thời gian ngắn nhất.

## CHƯƠNG 2: THIẾT KẾ GIẢI PHÁP

### 2.1 Giới thiệu về React Native

#### 2.1.1 Khái niệm

React Native là một framework sử dụng mã nguồn mở để có thể xây dựng được các ứng dụng dùng Javascript do Facebook phát hành. Đây là một trong những framework sử dụng cấu hình thiết kế tương tự như React.

Hầu hết các giao diện và chức năng của nó đều được cấu thành từ rất nhiều thành phần con. React Native được sử dụng để phát triển cho rất nhiều các ứng dụng di động khác như: Android, iOS, Web, UWP.



Hình 1: React native

#### 2.1.2 Những khái niệm liên quan

##### - Native App:

Native App là các ứng dụng được phát triển và build trên chính những công cụ hỗ trợ mà nhà phát triển iOS và Android cung cấp cho người lập trình viên ở đây là XCode và Android.

Chúng được viết bằng ngôn ngữ mà nhà phát triển iOS và Android cung cấp ra iOS thì là Swift hiện tại, xưa là Objective C, còn Android thì hiện tại tồn tại song song là Java và Kotlin. Các ứng dụng Native được build dựa trên ngôn ngữ của chính hệ điều hành và sử dụng những tính năng sẵn có của hệ điều hành mà không phải thông qua bất cứ ứng dụng bên thứ ba hay engine nào khác để vận hành nên tốc độ là nhanh nhất.

##### - Hybrid App:

Hybrid App là sự kết hợp giữa ứng dụng Web và ứng dụng mobile. Tức là chúng vừa có thể cài đặt lên điện thoại người dùng giống như là những ứng dụng

Native bình thường và bạn có thể tìm thấy chúng trên chợ ứng dụng nhưng khó có thể nhận ra đâu là ứng dụng Native và đâu là ứng dụng Hybrid thường thì người dùng sẽ không để ý quá nhiều.

Đồng thời như ứng dụng Web chúng được xây dựng dựa trên HTML, CSS và JS. Nhược điểm của Hybrid App đó chính là vấn đề hiệu năng sẽ bị ảnh hưởng đáng kể cũng như không tương tác được hết những tài nguyên hệ thống phần mềm cũng như phần cứng.

- Vì vậy, mà Native App là sự lựa chọn hàng đầu của giới lập trình trong việc phát triển phần mềm di động bởi vì nó mang lại hiệu năng cao, trải nghiệm người dùng tốt.

### **2.1.3 Cách thức hoạt động**

React Native hoạt động chủ yếu nhờ vào sự tích hợp của 2 thread là Main Thread và JS Thread cho các ứng dụng mobile.

+ Main Thread sẽ đảm nhận các vai trò trong việc cập nhật các giao diện người dùng (UI), rồi xử lý các tương tác của người dùng.

+ JS Thread có nhiệm vụ thực thi và tiến hành xử lý code Javascript.

Đây là 2 luồng hoạt động hoàn toàn đối lập nhau trong React. Chính vì vậy, để hai Thread có thể tương tác được với nhau thì nó sẽ sử dụng một cầu nối Bridge. Chúng sẽ cho phép chúng có thể giao tiếp mà không quá phụ thuộc vào nhau cũng như các chuyển đổi từ thread này sang thread khác. Các dữ liệu từ hai Thread này đều sẽ được vận hành ngay khi tiếp nối các dữ liệu cho nhau.

### **2.1.4 Ưu điểm và nhược điểm**

- Ưu điểm:

+ Có khả năng tối ưu thời gian hiệu quả.

+ Có hiệu năng ổn định.

+ Có khả năng tiết kiệm chi phí.

+ React Native sở hữu đội ngũ phát triển ứng dụng không quá lớn.

+ Ứng dụng được xây dựng bởi React Native đều ổn định và đáng tin cậy.

+ Việc xây dựng các ứng dụng ít Native Code nhất cho nhiều hệ điều hành khác nhau.

- + Hầu hết, các trải nghiệm của người dùng đều sẽ tốt hơn nếu như được so sánh với các ứng dụng Hybrid.
- Nhược điểm:
  - + Nó yêu cầu Native code.
  - + Hiệu năng làm việc kém hơn so với Native App.
  - + Khả năng bảo mật của app chưa tốt do sử dụng JS.
  - + Khả năng quản lý bộ nhớ chưa tốt.
  - + Khả năng tùy biến chưa thực sự là quá tốt với một số module.

## **2.2 Giới thiệu về Android studio**

### **2.2.1 Khái niệm**

Android Studio là môi trường phát triển tích hợp (IDE) chính thức cho việc phát triển ứng dụng Android. Nơi mà các nhà phát triển viết code và lắp ráp các ứng dụng từ các gói. Thư viện phần mềm, công cụ lập trình tiện dụng, nhiều hơn để giúp bạn xây dựng, kiểm thử. Cũng như gỡ lỗi các ứng dụng Android.

Android Studio hỗ trợ một loạt các giả lập để xem trước ứng dụng, vì vậy ngay cả khi bạn không có thiết bị thử nghiệm, bạn vẫn có thể chắc chắn rằng mọi thứ đều hoạt động trơn tru. Bên cạnh đó, loạt công cụ như lời khuyên tối ưu hóa, đồ thị doanh số bán hàng, và số liệu lấy từ phân tích sẽ giúp các nhà phát triển quản lý ứng dụng đang bán của mình và tìm ra hướng đi cụ thể với từng thiết bị Android.

Android Studio hỗ trợ các hệ điều hành Windows, Mac OS X và Linux, và là IDE chính thức của Google để phát triển ứng dụng Android gốc để thay thế cho Android Development Tools (ADT) dựa trên Eclipse.

### **2.2.2 Các tính năng nổi bật**

Android studio có rất nhiều lợi ích cũng như tính năng mà người dùng có thể khai thác được và đặc biệt việc cài đặt android studio còn có thể giúp lập trình viên mô phỏng để có thể tiến hành sửa lỗi và nâng cấp sản phẩm của mình khi cần thiết. Trên trình soạn thảo mã và công cụ phát triển mạnh mẽ của IntelliJ, Android Studio cung cấp nhiều tính năng nâng cao hiệu suất của bạn khi xây dựng ứng dụng Android, chẳng hạn như:

- + Một hệ thống xây dựng Gradle linh hoạt.
- + Trình mô phỏng nhanh và tính năng phong phú.

- + Một môi trường hợp nhất nơi bạn có thể phát triển cho tất cả các thiết bị Android.
- + Instant Run để đẩy các thay đổi vào ứng dụng đang chạy của bạn mà không cần xây dựng một APK mới.
- + Tích hợp GitHub để giúp bạn xây dựng các tính năng ứng dụng phổ biến và nhập mã mẫu.
- + Các công cụ và khuôn khổ thử nghiệm mở rộng.
- + Lint công cụ để bắt hiệu suất, khả năng sử dụng, tương thích phiên bản, và các vấn đề khác
- + Hỗ trợ C ++ và NDK
- + Tích hợp hỗ trợ Google Cloud Platform , giúp dễ dàng tích hợp Google Cloud Messaging và App Engine
- + Hỗ trợ tích hợp sâu Firebase vào trong các ứng dụng chỉ sau một click chuột.

## **2.3 Giới thiệu về Firebase**

### **2.3.1 Khái niệm**

Firebase là một dịch vụ cơ sở dữ liệu thời gian thực hoạt động trên nền tảng đám mây được cung cấp bởi Google nhằm giúp các lập trình phát triển nhanh các ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu.

### **2.3.2 Các chức năng chính**

Với Google Firebase, bạn chỉ có thể tạo ra các ứng dụng chat như Yahoo Message của ngày xưa hoặc như Facebook Messenger của ngày nay trong thời gian cực ngắn như khoảng một ngày thậm chí là vài giờ bởi đơn giản là bạn chỉ cần lo phần client còn phần server và database đã có firebase lo. Firebase là sự kết hợp giữa nền tảng cloud với hệ thống máy chủ cực kì mạnh mẽ tới từ Google, để cung cấp cho chúng ta những API đơn giản, mạnh mẽ và đa nền tảng trong việc quản lý, sử dụng database. Cụ thể hơn Google Firebase cung cấp tới chúng ta những chức năng chính sau:

- Realtime Database (Cơ sở dữ liệu thời gian thực)

Firebase lưu trữ dữ liệu database dưới dạng JSON và thực hiện đồng bộ database tới tất cả các client theo thời gian thực. Cụ thể hơn là bạn có thể xây dựng được client đa nền tảng (cross-platform client) và tất cả các client này sẽ cùng sử dụng

chung 1 database đến từ Firebase và có thể tự động cập nhật mỗi khi dữ liệu trong database được thêm mới hoặc sửa đổi.

Ngoài ra Firebase còn cho phép bạn phân quyền một cách đơn giản bằng cú pháp tương tự như javascript.

- Firebase Cloud Firestore

Cloud Firestore là một cơ sở dữ liệu linh hoạt, có thể mở rộng để phát triển thiết bị di động, web và máy chủ từ Firebase và Google Cloud. Giống như Cơ sở dữ liệu thời gian thực của Firebase, nó giữ cho dữ liệu của bạn được đồng bộ hóa giữa các ứng dụng khách thông qua trình xử lý thời gian thực và cung cấp hỗ trợ ngoại tuyến cho thiết bị di động và web để bạn có thể tạo các ứng dụng đáp ứng hoạt động bất kể độ trễ mạng hoặc kết nối Internet. Cloud Firestore cũng cung cấp khả năng tích hợp liền mạch với các sản phẩm Firebase và Google Cloud khác, bao gồm cả Chức năng đám mây.

- Firebase Authentication (Hệ thống xác thực của Firebase)

Với Firebase bạn có thể dễ dàng tích hợp các công nghệ xác thực của Google, Facebook, Twitter, ... hoặc một hệ thống xác thực mà bạn tự mình tạo ra vào trong ứng dụng của bạn ở bất kỳ nền tảng nào như Android, iOS hoặc Web.

- Firebase Storage

Với firebase storage các lập trình viên có thể lưu trữ dữ liệu trực tiếp trên server của firebase như hình ảnh, video, tập tin một cách dễ dàng. Firebase còn bổ sung Google security để tải lên và tải về các ứng dụng firebase của bạn. Bạn có thể sử dụng nó để lưu trữ hình ảnh, âm thanh, video hoặc nội dung do người dùng tạo ra. Quả thật đây là một dịch vụ lưu trữ đối tượng mạnh mẽ, đơn giản và hiệu quả chi phí.

- Firebase Cloud Messaging

Hiện nay firebase còn cung cấp dịch vụ Cloud Messaging thay thế cho Google Cloud Messaging đã cũ kỹ. Một giải pháp đa nền tảng, đáng tin cậy cho phép bạn cung cấp các thông điệp miễn phí dạng tin nhắn đến người dùng. Sử dụng FCM, bạn có thể thông báo cho một ứng dụng client email mới hoặc dữ liệu khác có sẵn để đồng bộ hóa. Bạn có thể gửi tin nhắn thông báo cho lái xe reengagement sử dụng và duy trì. Đối với trường hợp sử dụng như tin nhắn tức thời, tin nhắn có thể chuyển một tải trọng lên đến 4KB để một ứng dụng client.

### 2.3.3 Một số lợi ích



Ở phía trên là các chức năng của google firebase, vậy các chức năng đó sẽ đem lại cho bạn những lợi ích gì, có lẽ một số bạn đã mừng tưng ra rồi nhưng cũng có bạn có lẽ vẫn còn mơ hồ vì vậy hãy đọc kĩ phần này sẽ biết câu trả lời chính xác nhất.

- Triển khai ứng dụng cực nhanh

Với Firebase bạn có thể giảm bớt rất nhiều thời gian cho việc viết các dòng code để quản lý và đồng bộ cơ sở dữ liệu, mọi việc sẽ diễn ra hoàn toàn tự động với các API của Firebase. Không chỉ có vậy Firebase còn hỗ trợ đã nền tảng nên bạn sẽ càng đỡ mất thời gian rất nhiều khi ứng dụng bạn muốn xây dựng là ứng dụng đa nền tảng.

Không chỉ nhanh chóng trong việc xây dựng database, Google Firebase còn giúp ta đơn giản hóa quá trình đăng kí và đăng nhập vào ứng dụng bằng các sử dụng hệ thống xác thực do chính Firebase cung cấp.

- Bảo mật

Firebase hoạt động dựa trên nền tảng cloud và thực hiện kết nối thông qua giao thức bảo mật SSL, chính vì vậy bạn sẽ bớt lo lắng rất nhiều về việc bảo mật của dữ liệu cũng như đường truyền giữa client và server. Không chỉ có vậy, việc cho phép phân quyền người dùng database bằng cú pháp javascript cũng nâng cao hơn nhiều độ bảo mật cho ứng dụng của bạn, bởi chỉ những user mà bạn cho phép mới có thể có quyền chỉnh sửa cơ sở dữ liệu.

- Tính linh hoạt và khả năng mở rộng

Sử dụng Firebase sẽ giúp bạn dễ dàng hơn rất nhiều mỗi khi cần nâng cấp hay mở rộng dịch vụ. Ngoài ra firebase còn cho phép bạn tự xây dựng server của riêng mình để bạn có thể thuận tiện hơn trong quá trình quản lý.

Việc Firebase sử dụng NoSQL, giúp cho database của bạn sẽ không bị bó buộc trong các bảng và các trường mà bạn có thể tùy ý xây dựng database theo cấu trúc của riêng bạn.

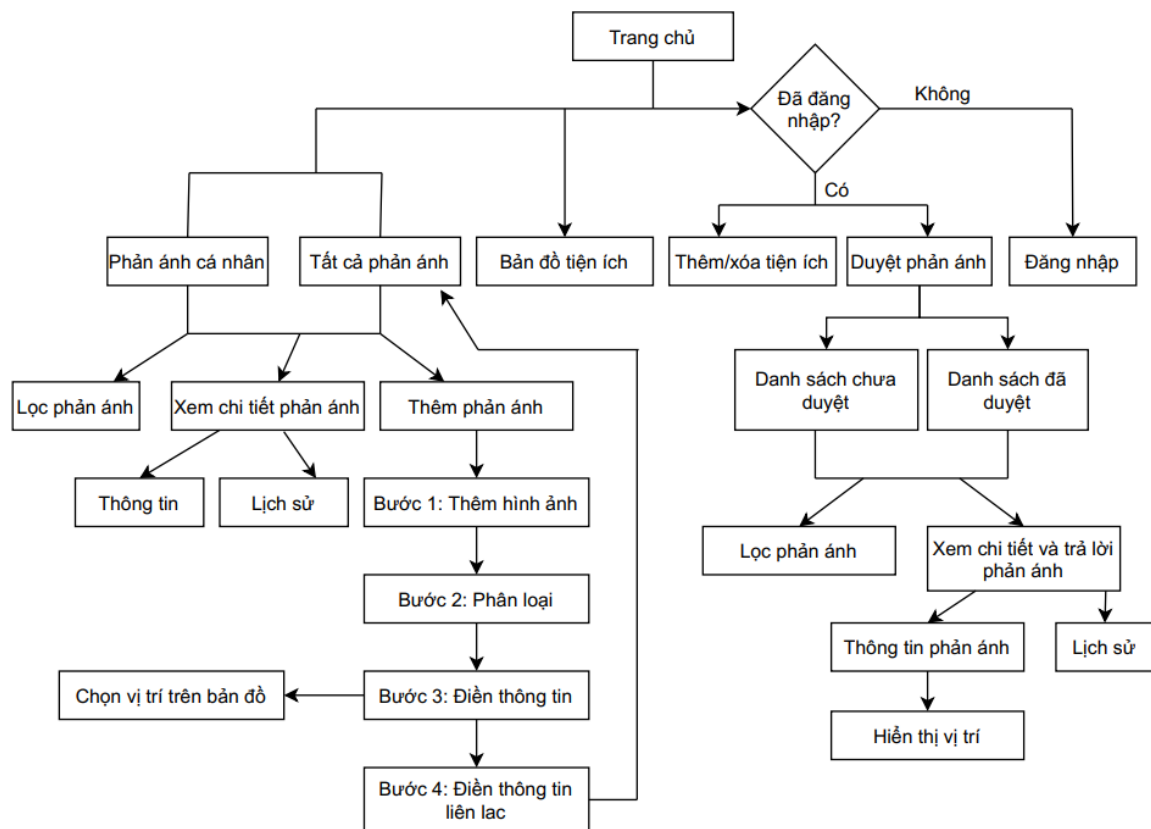
- Sự ổn định

Firebase hoạt động dựa trên nền tảng cloud đến từ Google vì vậy hầu như bạn không bao giờ phải lo lắng về việc sập server, tấn công mạng như DDOS, tốc độ kết nối lúc nhanh lúc chậm, ... nữa, bởi đơn giản là Firebase hoạt động trên hệ thống server của Google. Hơn nữa nhờ hoạt động trên nền tảng Cloud nên việc nâng cấp, bảo trì server cũng diễn ra rất đơn giản mà không cần phải dừng server để nâng cấp như truyền thống.

- Giá thành

Google Firebase có rất nhiều gói dịch vụ với các mức dung lượng lưu trữ cũng như băng thông khác nhau với mức giá dao động từ Free đến \$1500 đủ để đáp ứng được nhu cầu của tất cả các đối tượng. Chính vì vậy bạn có thể lựa chọn gói dịch vụ phù hợp nhất với nhu cầu của mình. Điều này giúp bạn tối ưu hóa được vốn đầu tư và vận hành của mình tùy theo số lượng người sử dụng. Ngoài ra bạn còn không mất chi phí để bảo trì, nâng cấp, khắc phục các sự cố bởi những điều này đã có Firebase lo.

## 2.4 Thiết kế các chức năng của ứng dụng



Hình 2: Sơ đồ giao diện người dùng

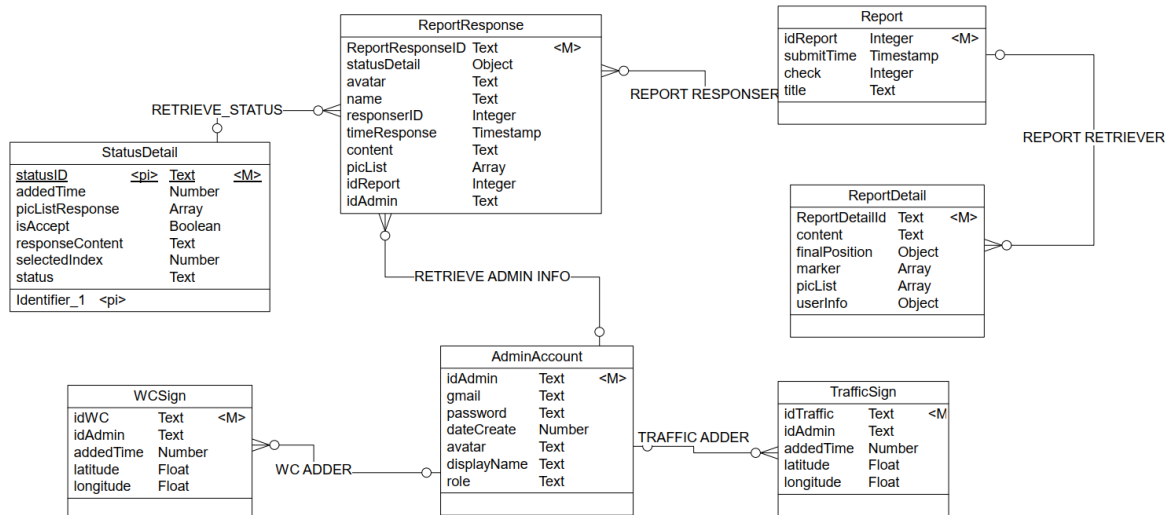
- Các chức năng của người quản trị:

- + Đăng nhập vào ứng dụng.
- + Xem, duyệt và trả lời các phản ánh.
- + Cập nhật các tiện ích và tình trạng giao thông.

- Các chức năng của người dùng:

- + Gửi phản ánh lên hệ thống.
- + Xem, tìm kiếm và lọc các phản ánh.
- + Xem các tiện ích và tình trạng giao thông trên bản đồ.

## 2.5 Thiết kế cơ sở dữ liệu



Hình 3: Sơ đồ cơ sở dữ liệu

- Bảng Report: Thông tin sơ lược của phản ánh
- Bảng ReportDetail: Thông tin chi tiết của phản ánh (hình ảnh, nội dung, ...)
- Bảng ReportResponse: Thông tin sơ lược về duyệt và trả lời phản ánh
- Bảng StatuDetail: Thông tin chi tiết từng giai đoạn duyệt phản ánh
- Bảng AdminAccount: Thông tin về tài khoản của người quản trị
- Bảng WCSign, TrafficSign: Thông tin các tiện ích và tình trạng giao thông trên bản đồ

## CHƯƠNG 3: CÀI ĐẶT GIẢI PHÁP

### 3.1 Cài đặt môi trường lập trình Android

#### 3.1.1 Cài đặt Java JDK

- Bước 1: Chọn phiên bản JDK phù hợp và tải về tại đây

<https://www.oracle.com/java/technologies/javase-jdk11-downloads.html>

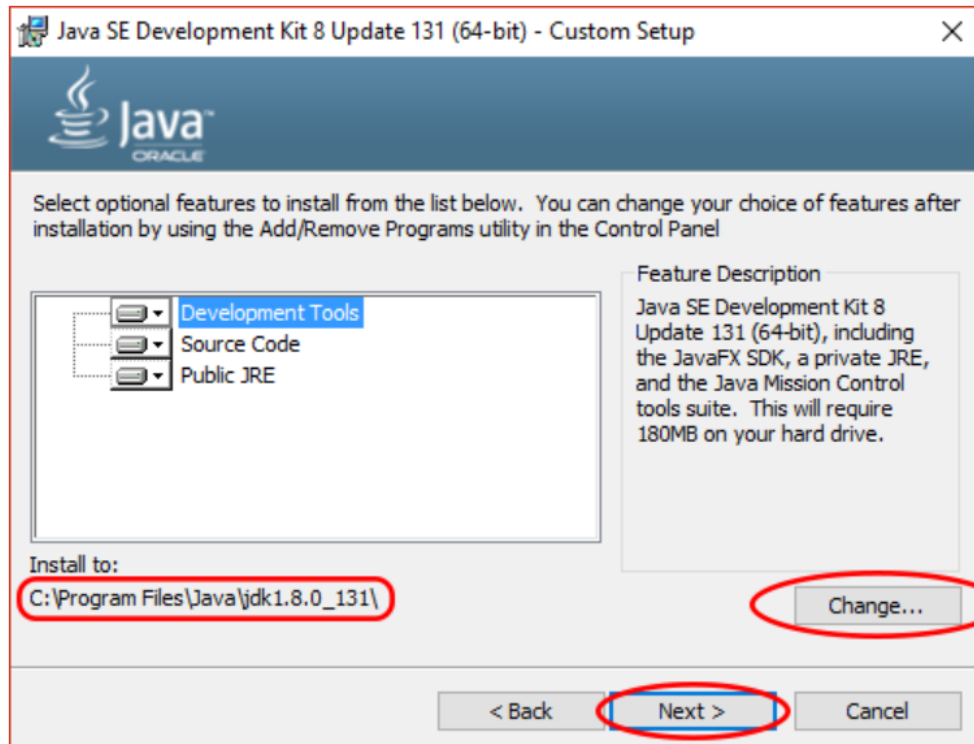
- Bước 2: Tiến hành cài đặt Java JDK

Để cài đặt, bạn click đúp chuột vào file .exe bắt đầu cài JDK.



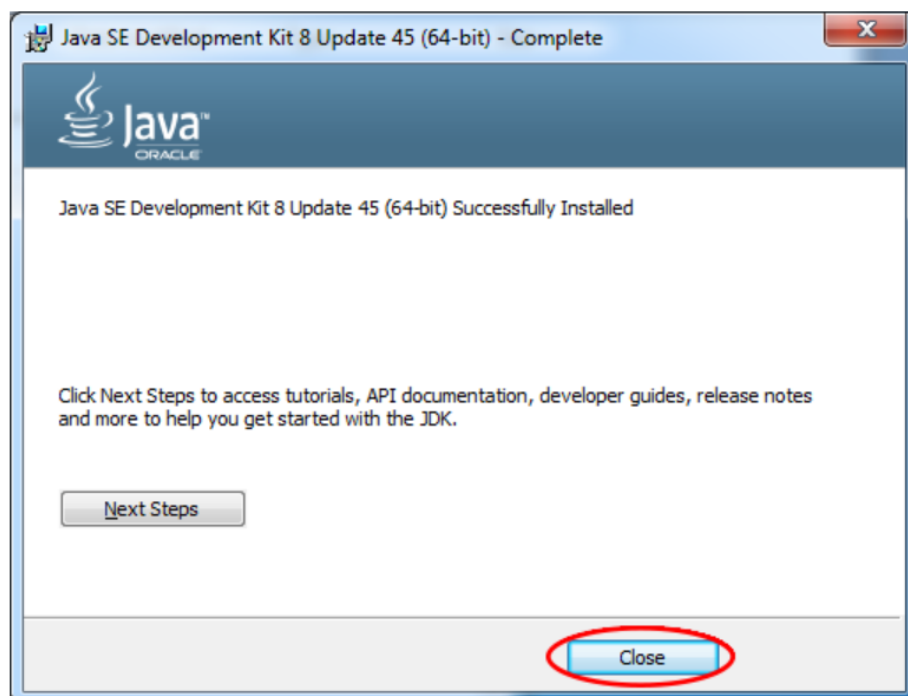
Hình 4: Cài đặt Java JDK (1)

Nhấn “Next” để tiếp tục.



Hình 5: Cài đặt Java JDK (2)

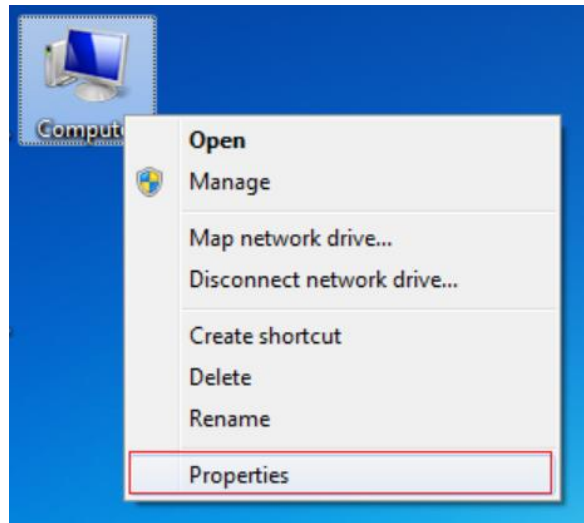
Bạn có thể thay đổi đường dẫn cài đặt JDK. Nhấn “Next >” để tiếp tục cài đặt mặc định.



Hình 6: Cài đặt Java JDK (3)

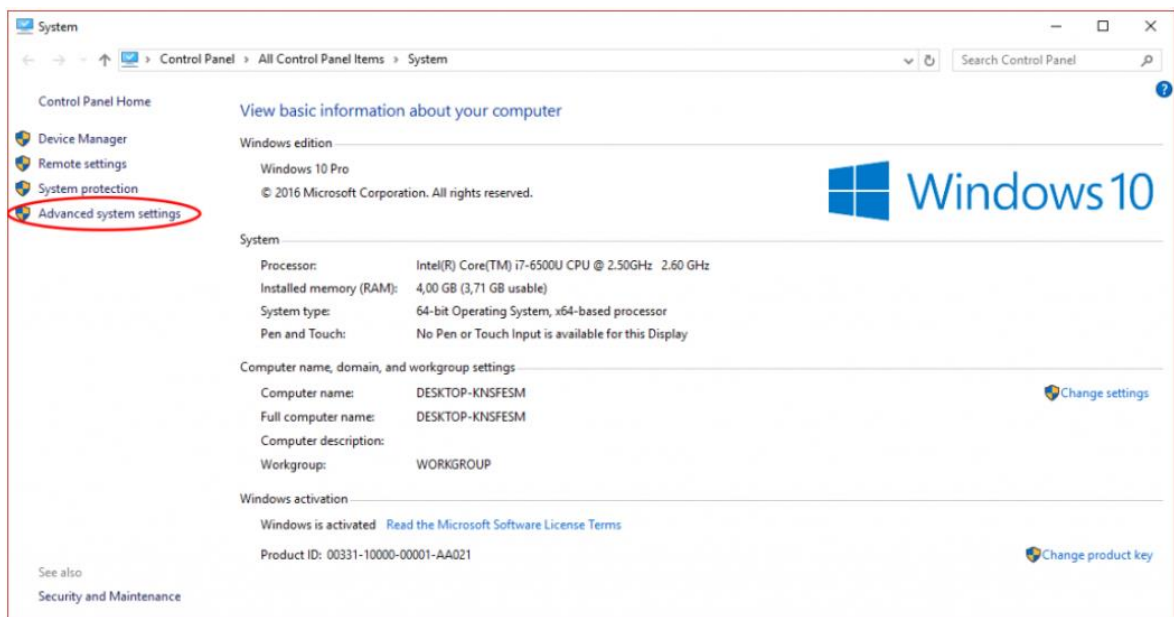
Nhấn “Close” để hoàn tất.

- Bước 3: Cài đặt biến môi trường Java



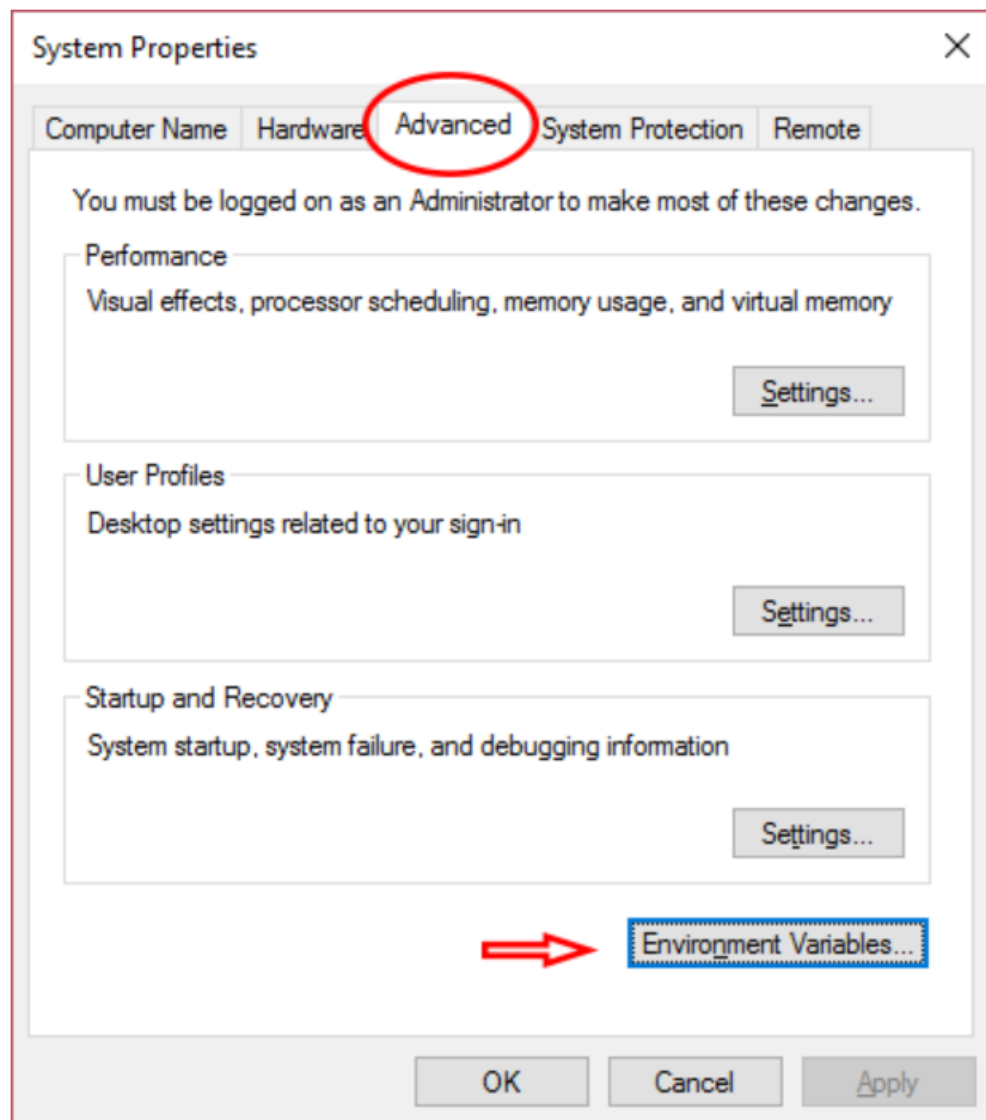
Hình 7: Cài đặt biến môi trường Java (1)

Nhấp chuột phải vào “My Computer” rồi chọn “Properties”.



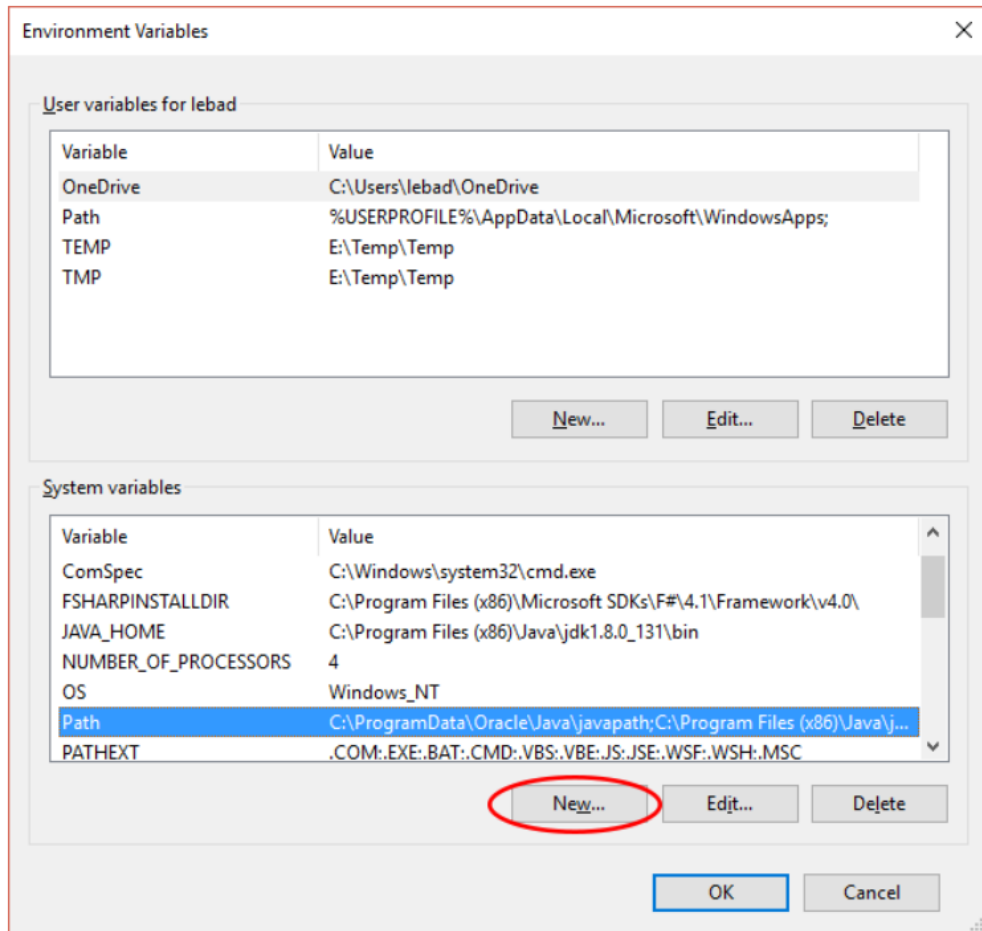
Hình 8: Cài đặt biến môi trường Java (2)

Chọn vào “Advanced system settings”.



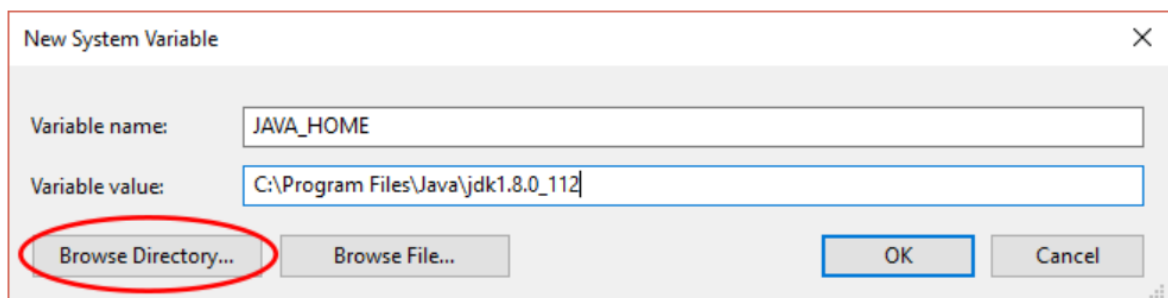
Hình 9: Cài đặt biến môi trường Java (3)

Chọn “Advanced”, tiếp tục chọn vào “Environment Variables...”



Hình 10: Cài đặt biến môi trường Java (4)

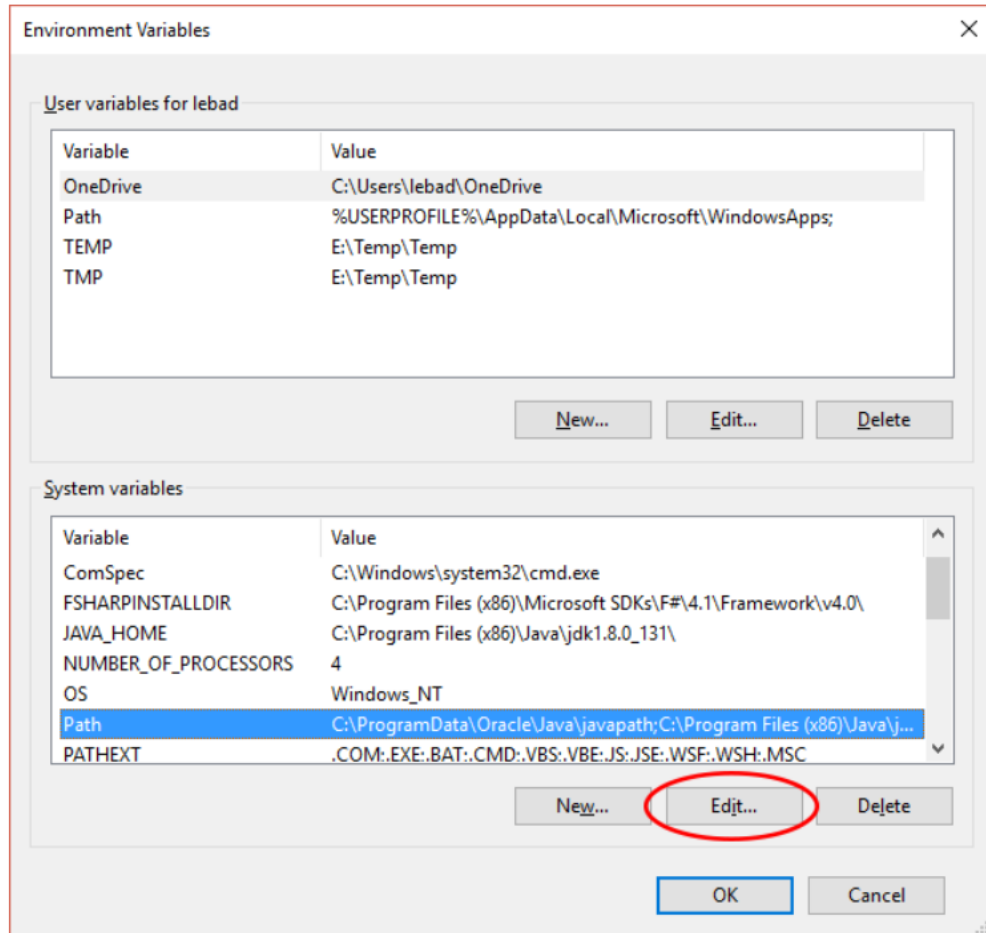
Chọn dòng Path, tiếp tục nhấn “New...”.



Hình 11: Cài đặt biến môi trường Java (5)

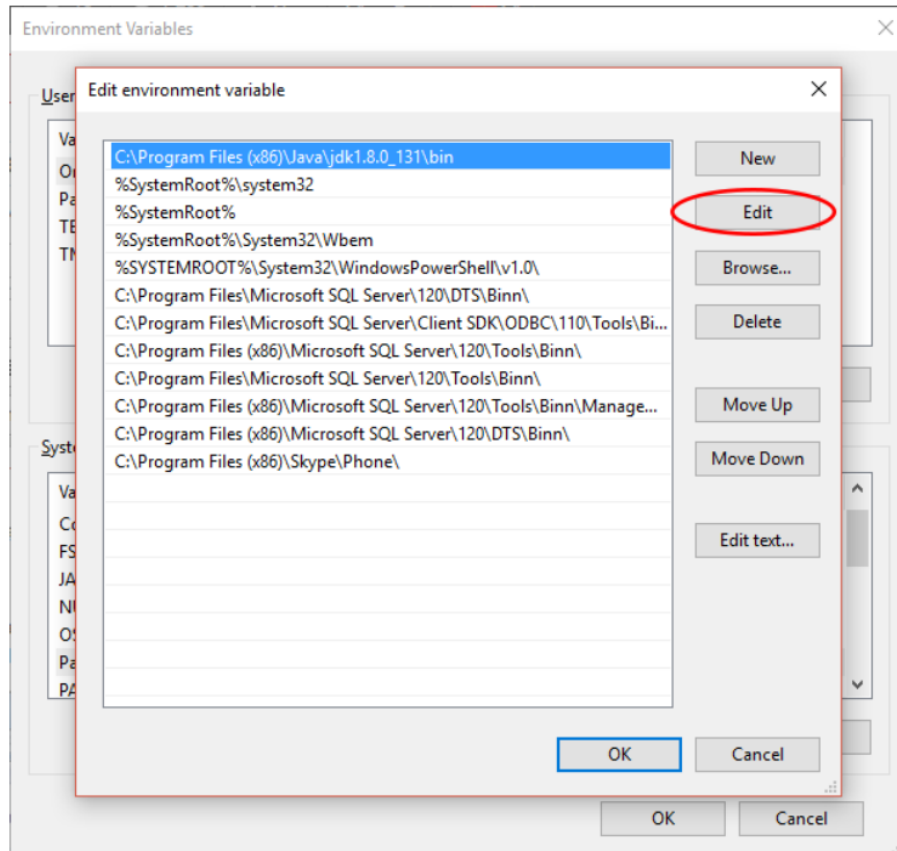
Trên dòng “Variable name” đặt tên là “JAVA\_HOME”, tiếp tục nhấn vào “Brows Directory...” để chọn đường dẫn tới java cho dòng “Variable value”, nhấn “OK” để hoàn tất.





Hình 12: Cài đặt biến môi trường Java (6)

Chọn vào dòng Path, nhấn vào “Edit...”.



Hình 13: Cài đặt biến môi trường Java (7)

Chọn vào dòng đầu rồi nhấn vào “Edit...”, chỉnh đường dẫn đến file bin của Java JDK, nhấn “OK” để hoàn tất.

- Bước 4: Kiểm tra

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\lebad>java -version
java version "1.8.0_131"
Java(TM) SE Runtime Environment (build 1.8.0_131-b11)
Java HotSpot(TM) Client VM (build 25.131-b11, mixed mode)

C:\Users\lebad>javac -version
javac 1.8.0_131

C:\Users\lebad>
```

Hình 14: Kiểm tra phiên bản Java đã được cài đặt

Mở cửa sổ “Command Prompt”, gõ các lệnh kiểm tra sau:

- java -version : dùng để kiểm tra phiên bản java chúng ta đã cài đặt.
- javac -version : dùng để kiểm tra phiên bản javac (nếu hiển thị phiên bản của javac ở đây thì bạn đã cài đặt thành công biến môi trường java).

### 3.1.2 Cài đặt Android Studio

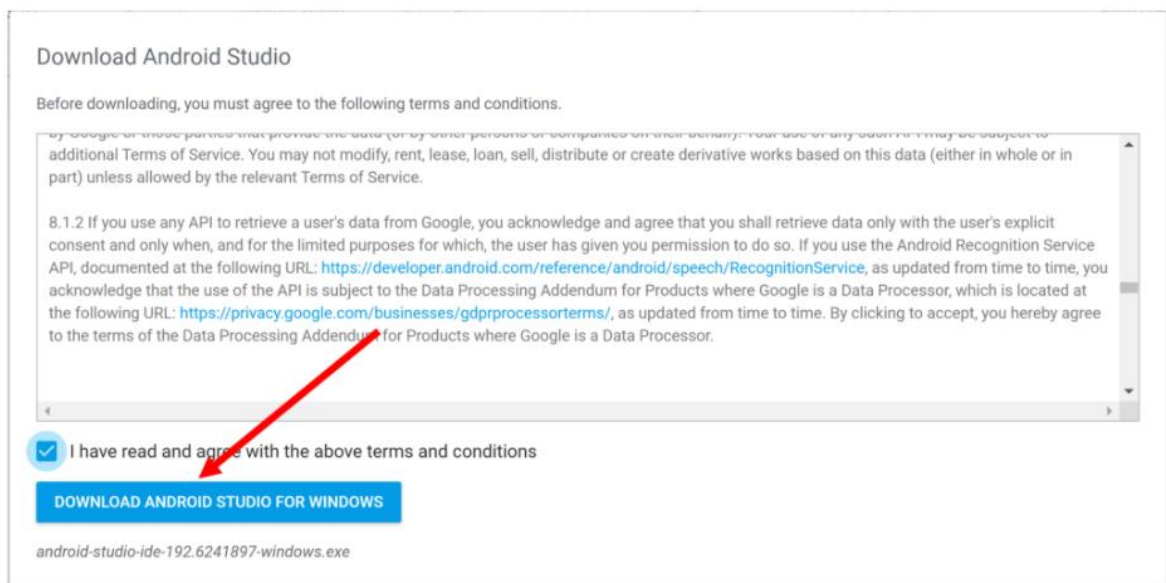
- Bước 1: Tải Android Studio tại đây

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



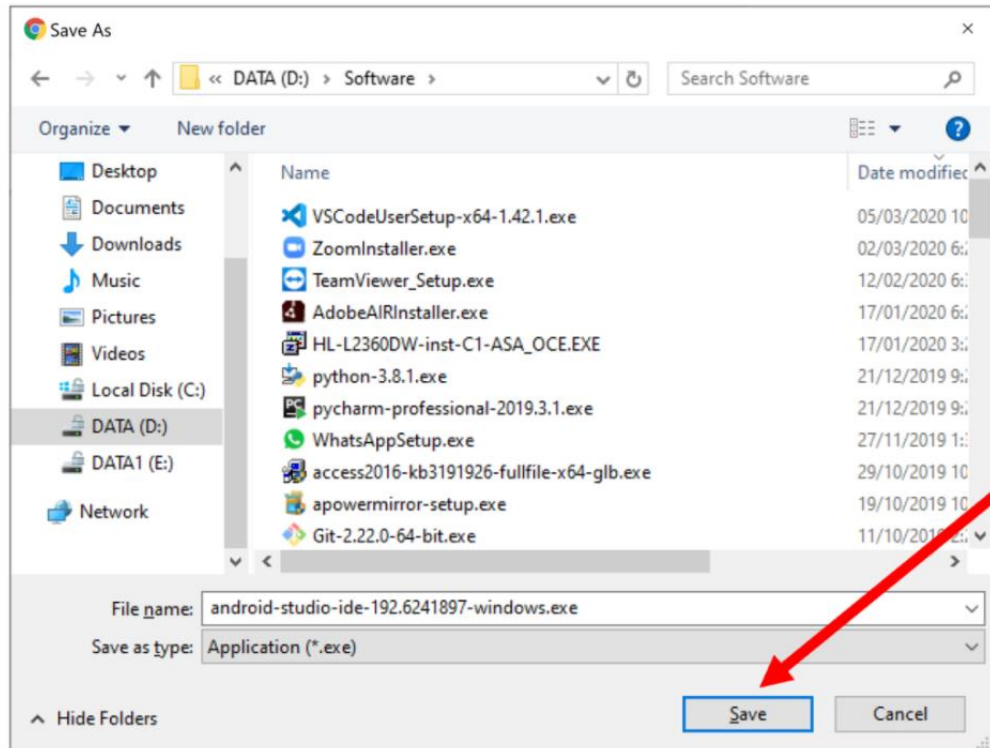
Hình 15: Tải Android Studio (1)

Nhấn vào “DOWNLOAD ANDROID STUDIO”.



Hình 16: Tải Android Studio (2)

Check vào “I have read and agree with the above terms and conditions”. Sau đó bấm “DOWNLOAD ANDROID STUDIO FOR WINDOWS”.

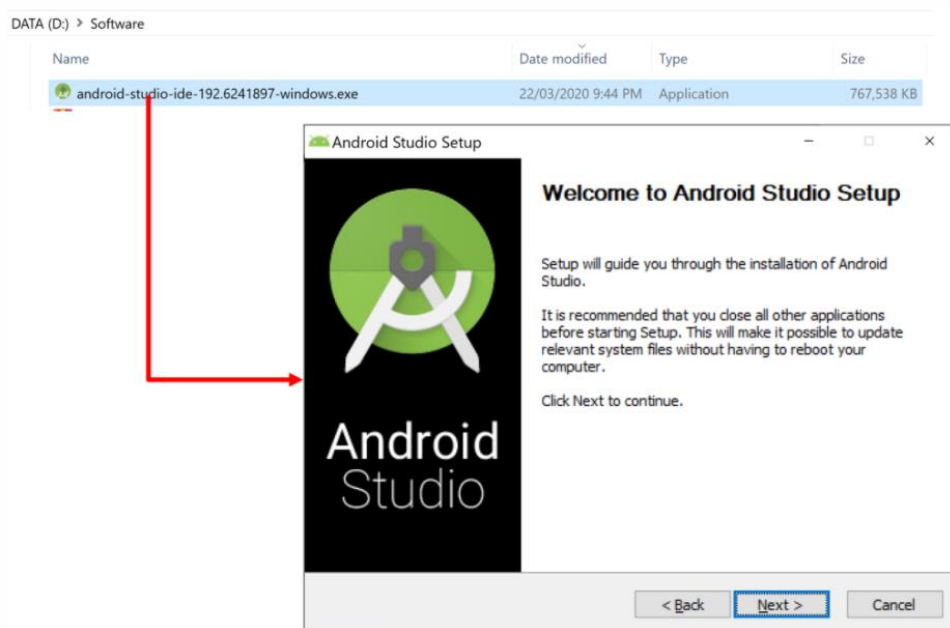


Hình 17: Tải Android Studio (3)

Bấm “Save” để tải về.

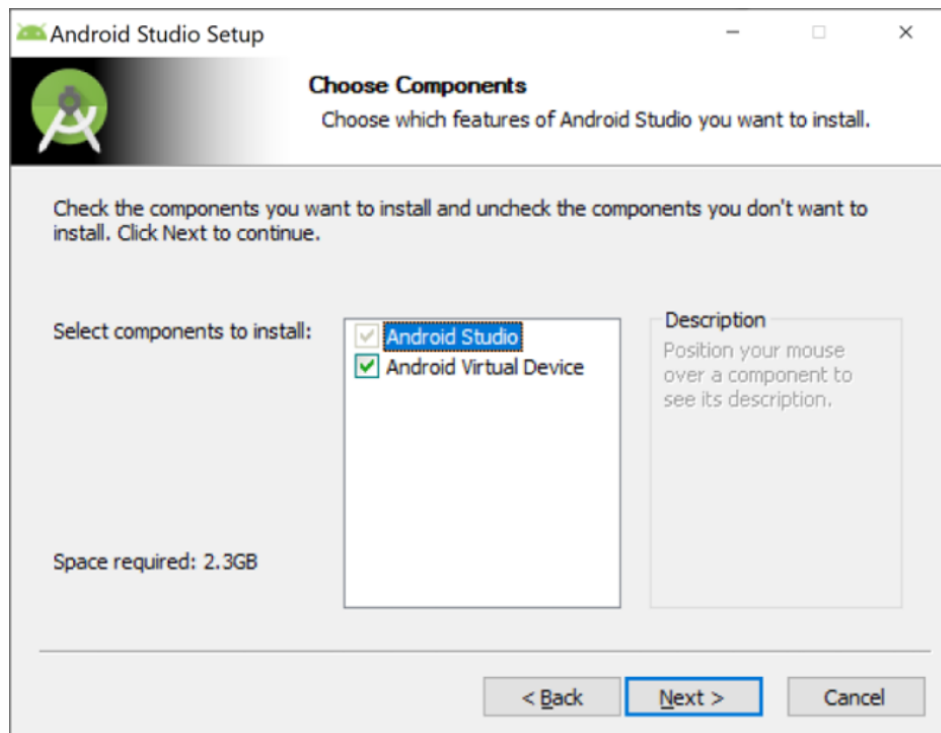
- Bước 2: Tiến hành cài đặt

Bây giờ double click vào file cài mới tải ở trên về để cài đặt.



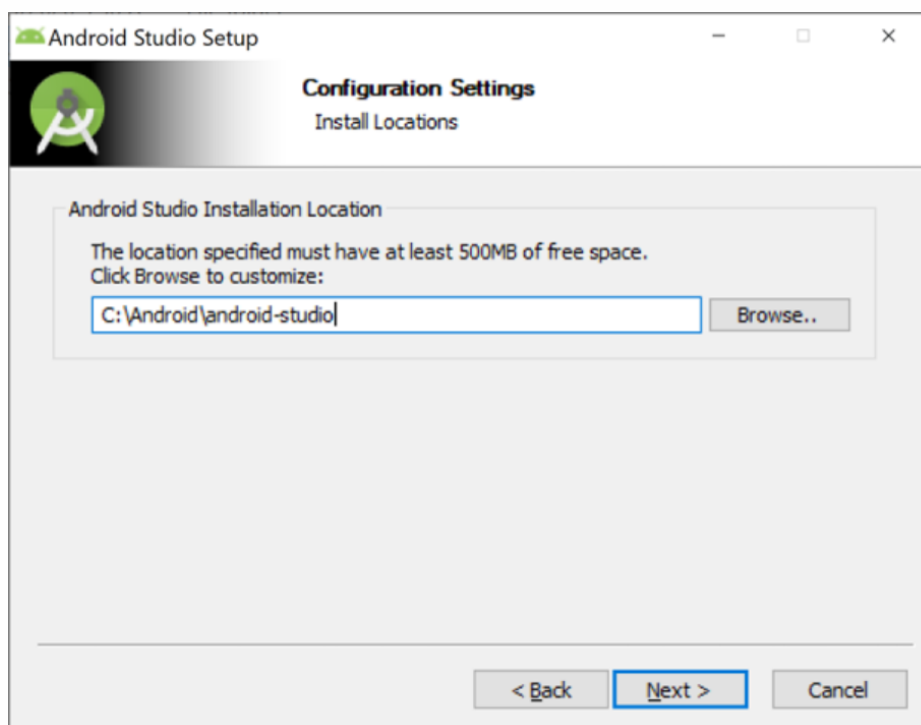
Hình 18: Cài đặt Android Studio (1)

Nhấn “Next >” để tiếp tục.



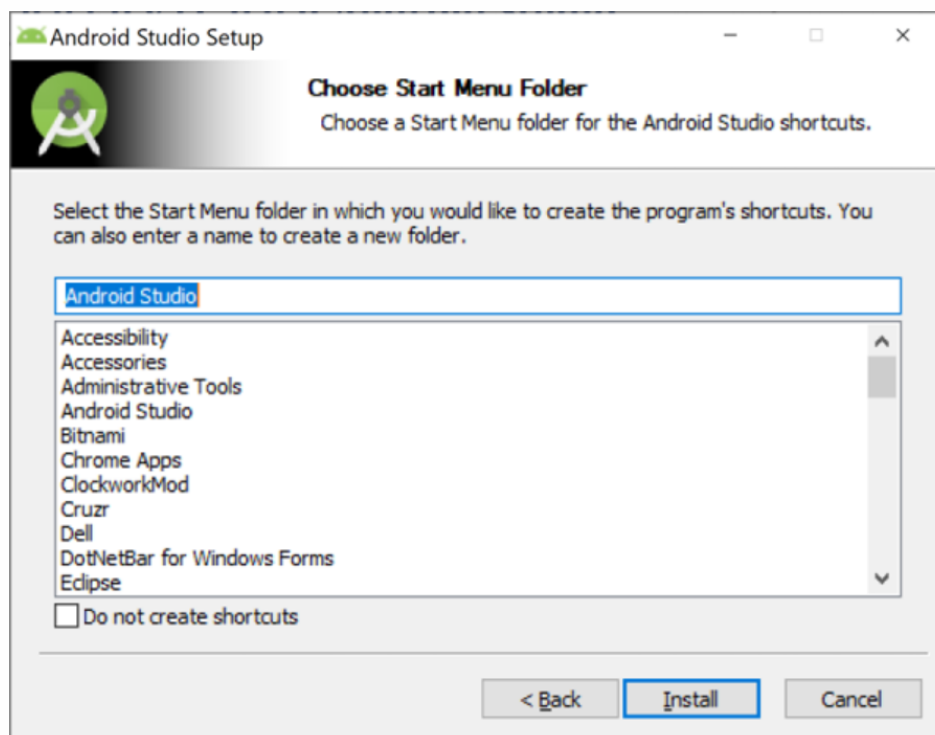
Hình 19: Cài đặt Android Studio (2)

Để mặc định rồi tiếp tục nhấn “Next >”.



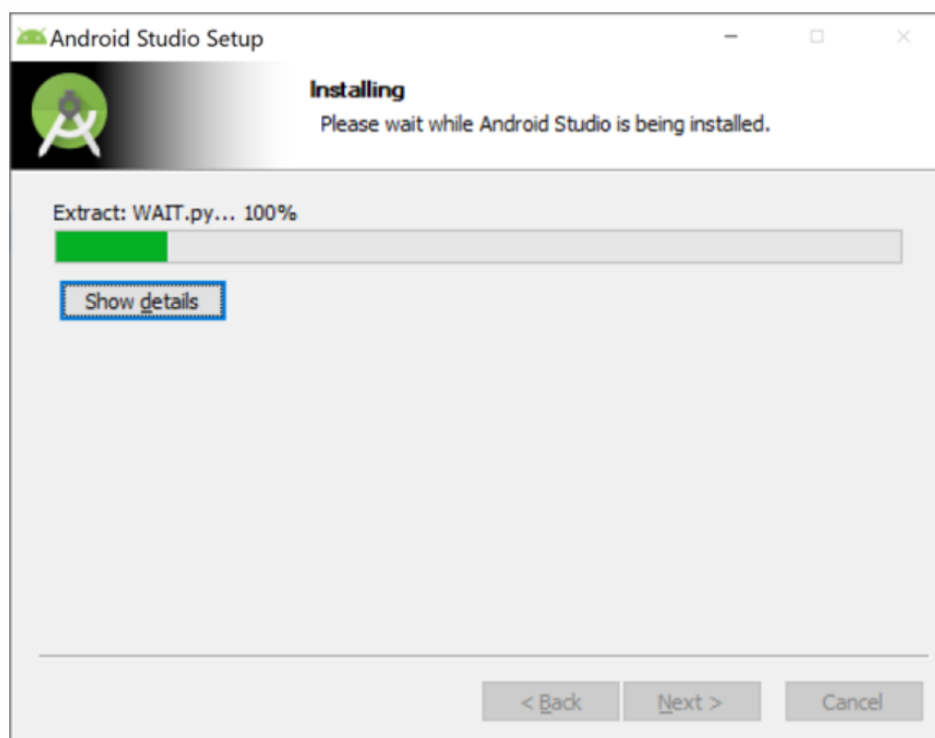
Hình 20: Cài đặt Android Studio (3)

Tiếp tục nhấn “Next >”.



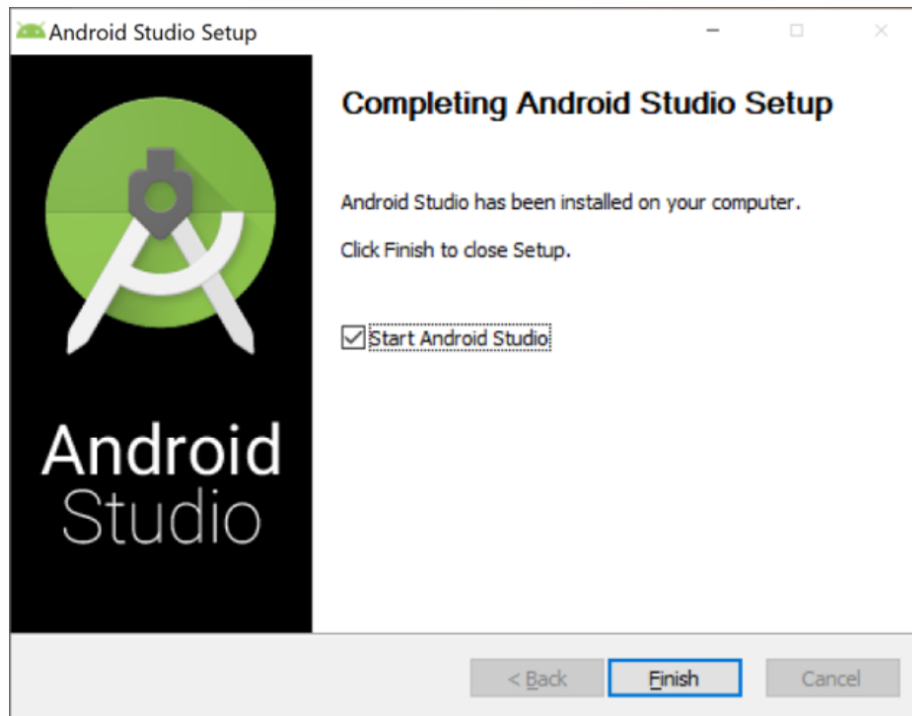
Hình 21: Cài đặt Android Studio (4)

Sau đó nhấn “Install” để bắt đầu cài đặt.



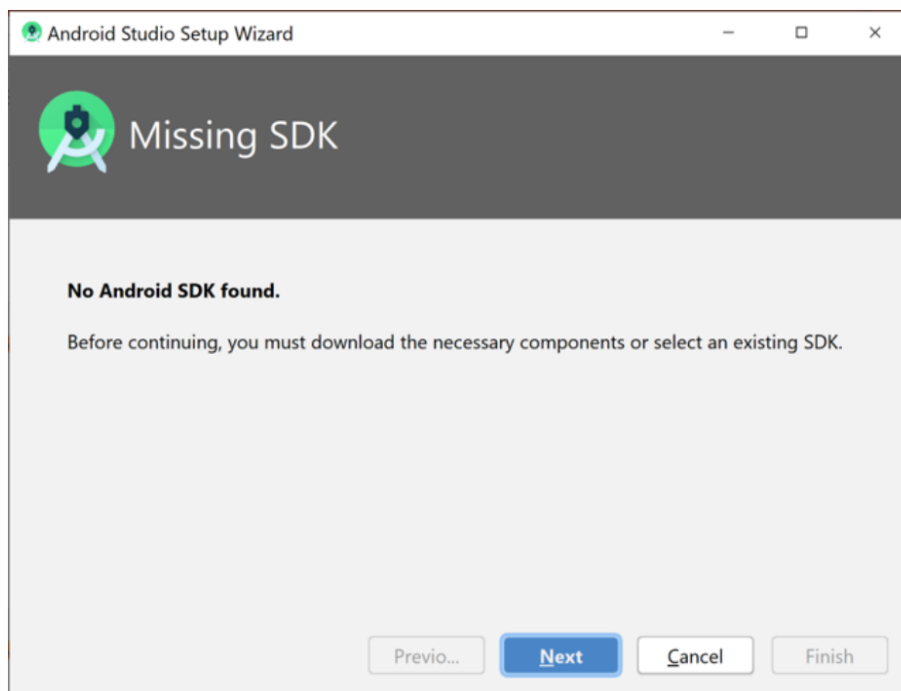
Hình 22: Cài đặt Android Studio (5)

Ngồi chờ xíu cho nó cài đặt, khi cài đặt xong sẽ có thông báo như dưới đây.



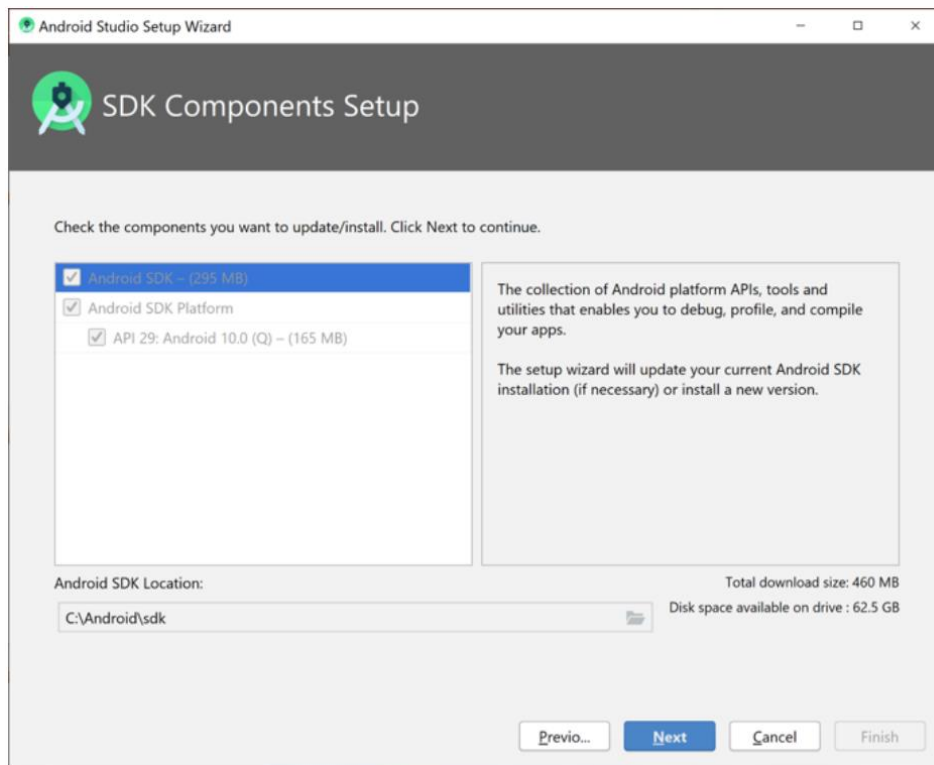
Hình 23: Cài đặt Android Studio (6)

Nếu như trước đó đã làm Android thì dĩ nhiên có SDK sẵn và phần mềm không yêu cầu gì thêm cả, nếu chưa bao giờ cài thì sẽ tiếp tục được yêu cầu cài SDK.



Hình 24: Cài đặt Android Studio (7)

Chương trình sẽ báo Missing SDK, ta tiếp tục nhấn “Next”.

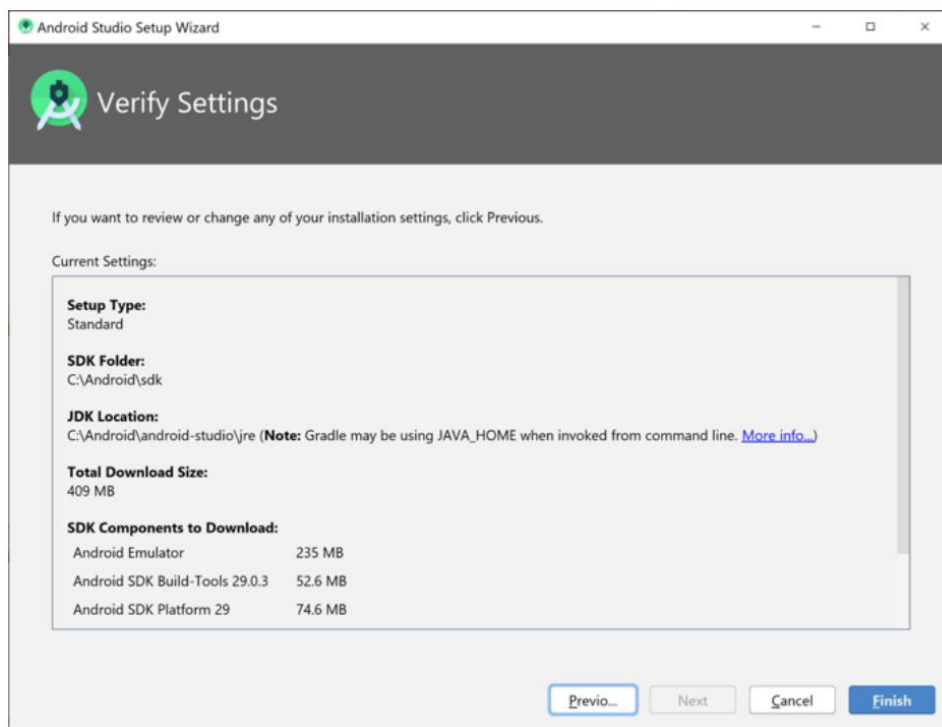


Hình 25: Cài đặt Android Studio (8)

Lưu ý Android SDK Location ta chọn đúng nơi mà ta đã tạo thư mục trước đó: C:\Android\sdk.

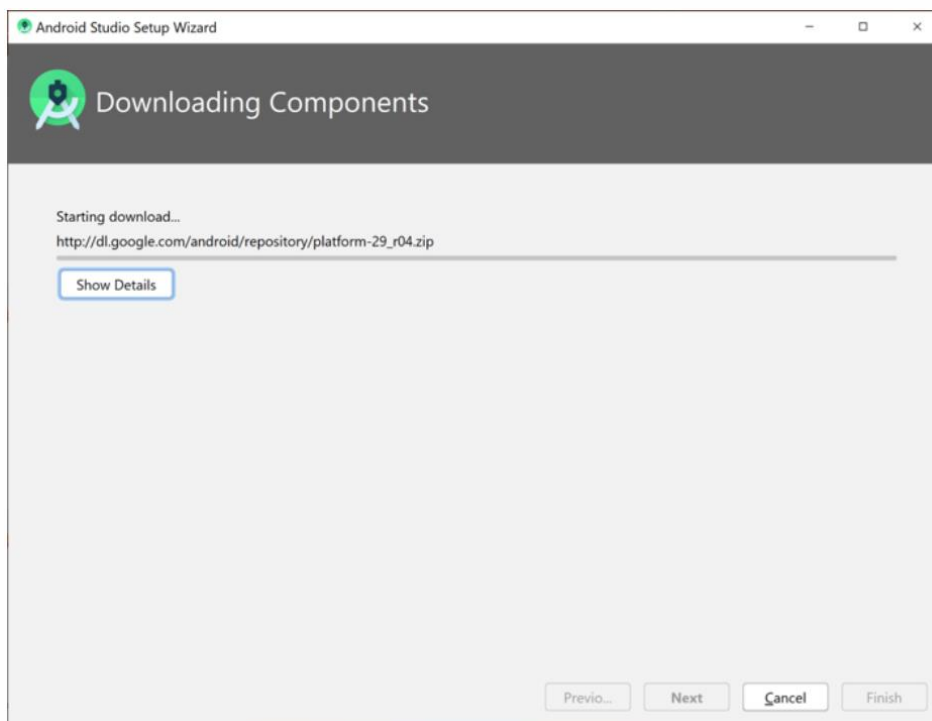
Sau đó nhấn “Next” để tiếp tục. Màn hình Verify Settings sẽ xuất hiện như dưới đây.





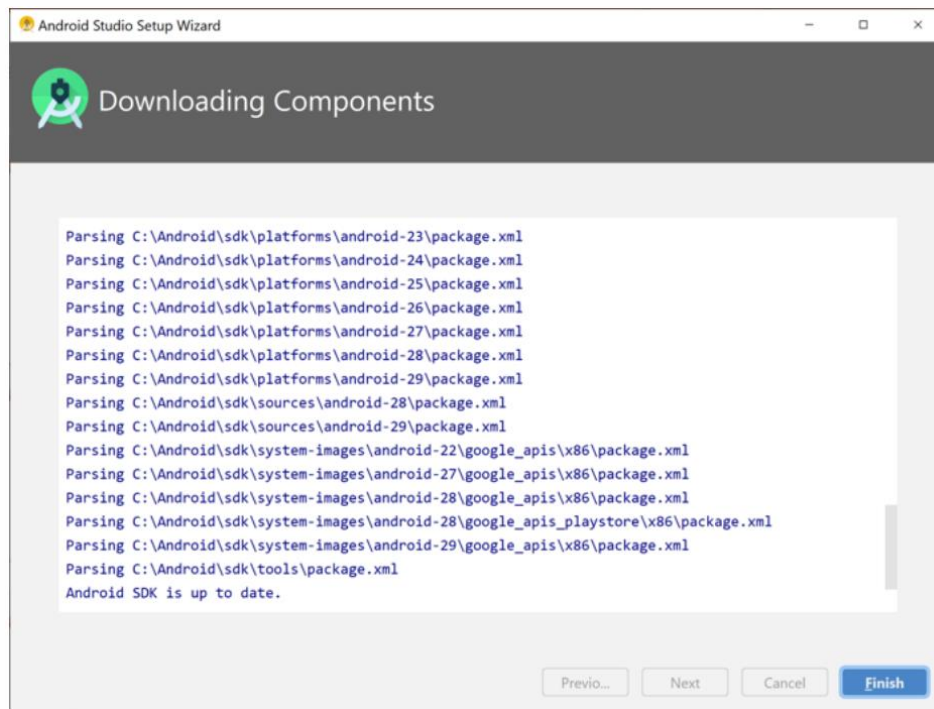
Hình 26: Cài đặt Android Studio (9)

Nhấn “Finish” để cài, màn hình Downloading Components sẽ hiển thị như dưới đây.



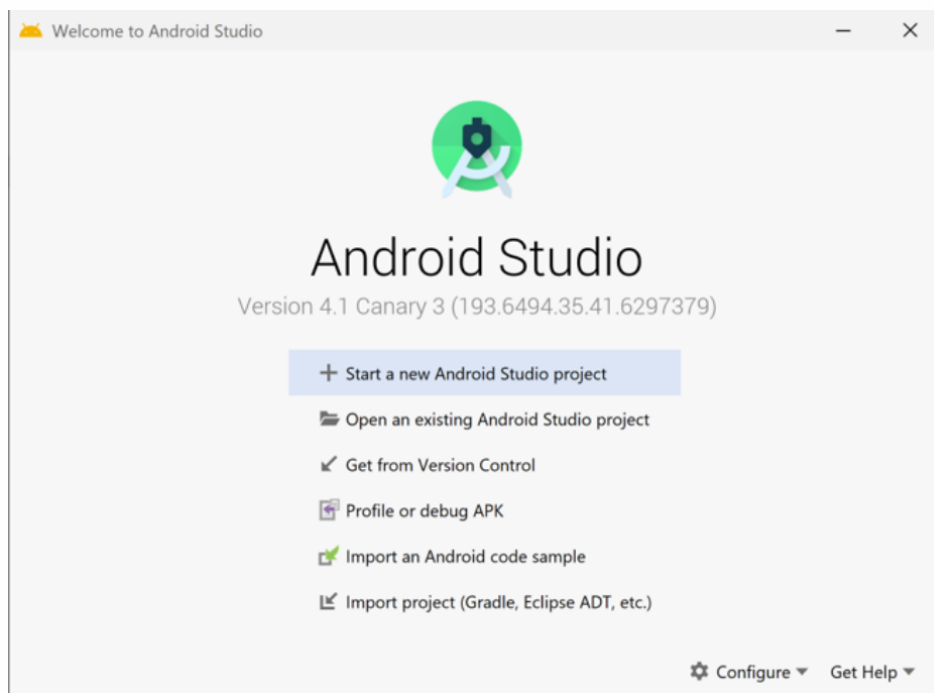
Hình 27: Cài đặt Android Studio (10)

Chờ cho tới khi nó báo hoàn tất.



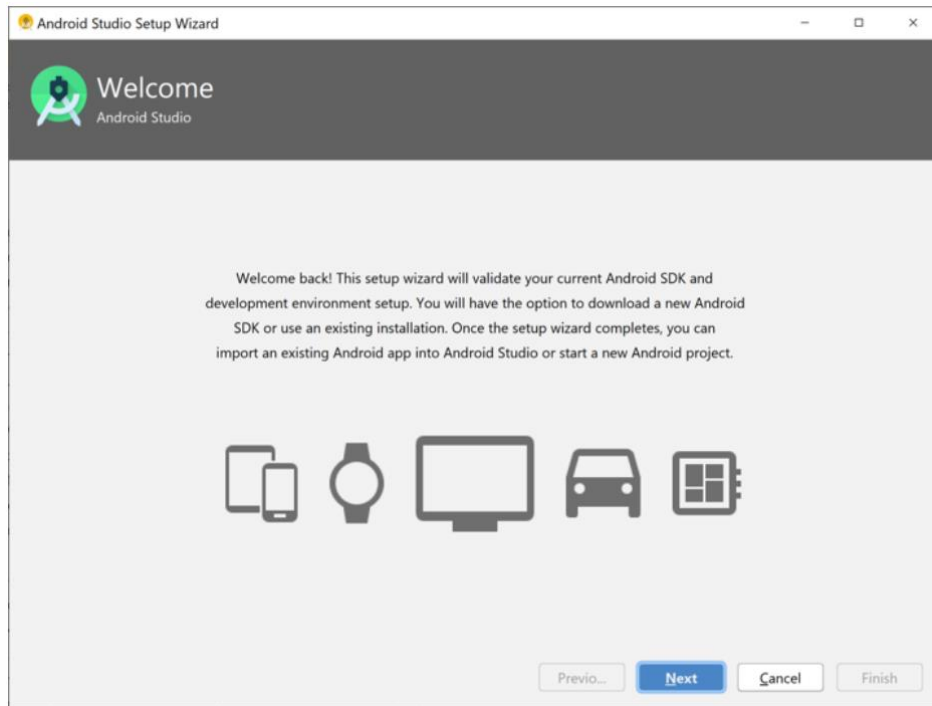
Hình 28: Cài đặt Android Studio (11)

Nhấn “Finish” để hoàn tất quá trình cài đặt SDK lúc này phần mềm Android Studio sẽ xuất hiện như dưới đây.



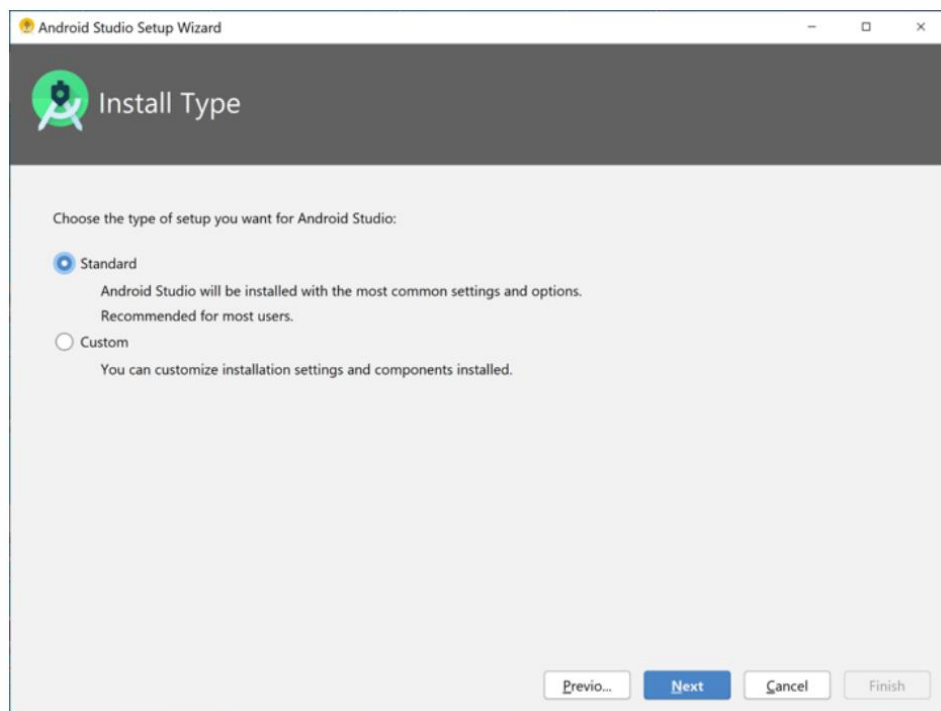
Hình 29: Cài đặt Android Studio (12)

Nếu Android Studio yêu cầu chọn một số Setup Wizard, ví dụ như xuất hiện các màn hình dưới đây.



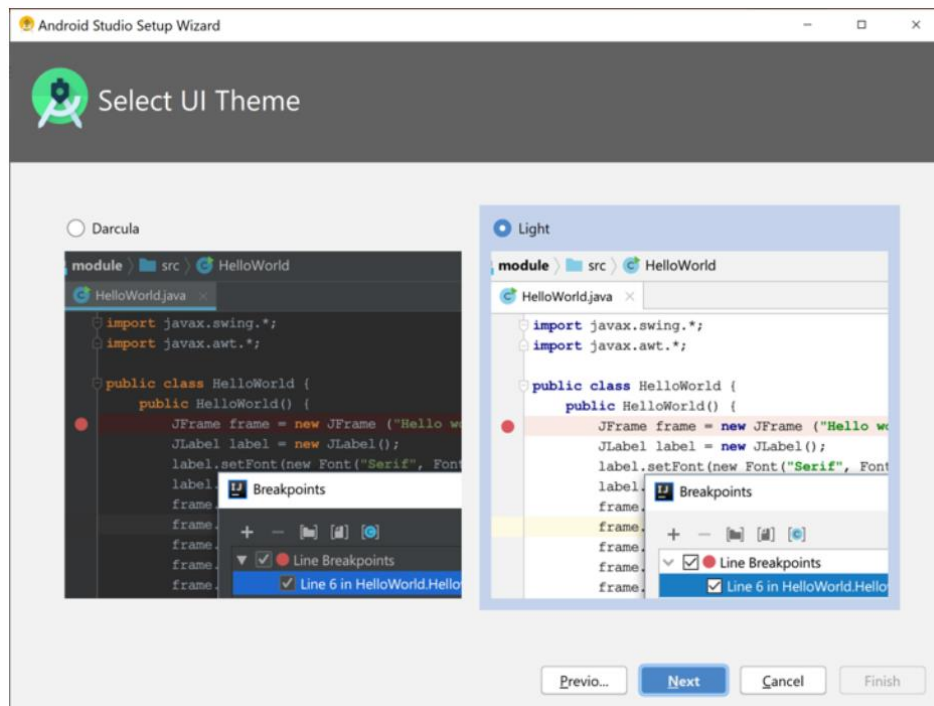
Hình 30: Cài đặt Android Studio (13)

Nhấn “Next”.



Hình 31: Cài đặt Android Studio (14)

Chọn “Standard” rồi nhấn “Next”.

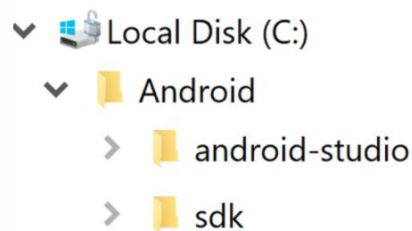


Hình 32: Cài đặt Android Studio (15)

Chọn giao diện là Light cho nó sáng, sau đó nhấn “Next” để hoàn tất, lúc này ra cái màn hình Android studio bình thường.

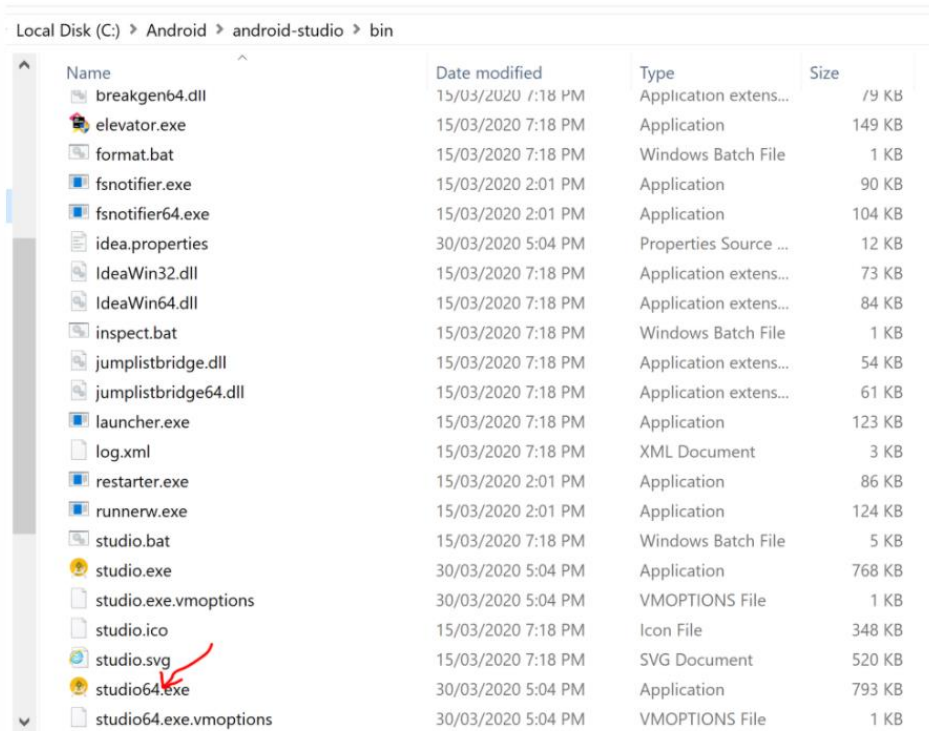
- Bước 3: Kiểm tra sau khi cài đặt hoàn tất

Dưới đây là một số hình ảnh khi cài đặt Android Studio hoàn tất.



Hình 33: Kiểm tra sau khi cài đặt Android Studio (1)

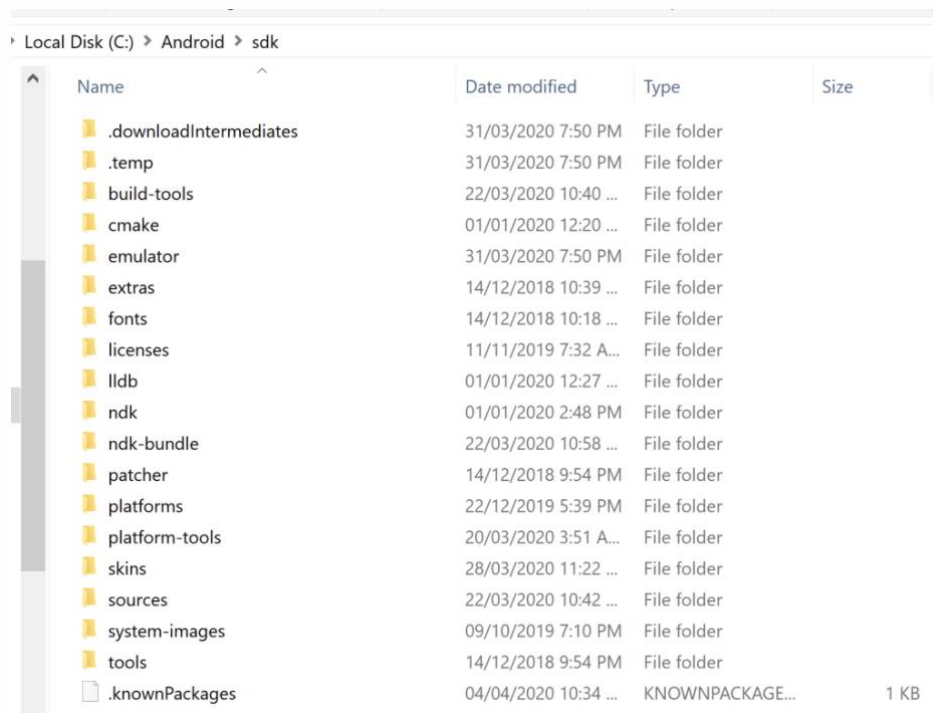
Dưới đây là hình chụp android-studio:



Name	Date modified	Type	Size
breakgen64.dll	15/03/2020 7:18 PM	Application extens...	79 KB
elevator.exe	15/03/2020 7:18 PM	Application	149 KB
format.bat	15/03/2020 7:18 PM	Windows Batch File	1 KB
fsnotifier.exe	15/03/2020 2:01 PM	Application	90 KB
fsnotifier64.exe	15/03/2020 2:01 PM	Application	104 KB
idea.properties	30/03/2020 5:04 PM	Properties Source ...	12 KB
IdeaWin32.dll	15/03/2020 7:18 PM	Application extens...	73 KB
IdeaWin64.dll	15/03/2020 7:18 PM	Application extens...	84 KB
inspect.bat	15/03/2020 7:18 PM	Windows Batch File	1 KB
jumplistbridge.dll	15/03/2020 7:18 PM	Application extens...	54 KB
jumplistbridge64.dll	15/03/2020 7:18 PM	Application extens...	61 KB
launcher.exe	15/03/2020 7:18 PM	Application	123 KB
log.xml	15/03/2020 7:18 PM	XML Document	3 KB
restarter.exe	15/03/2020 2:01 PM	Application	86 KB
runnerw.exe	15/03/2020 2:01 PM	Application	124 KB
studio.bat	15/03/2020 7:18 PM	Windows Batch File	5 KB
studio.exe	30/03/2020 5:04 PM	Application	768 KB
studio.exe.vmoptions	30/03/2020 5:04 PM	VMOPTIONS File	1 KB
studio.ico	15/03/2020 7:18 PM	Icon File	348 KB
studio.svg	15/03/2020 7:18 PM	SVG Document	520 KB
studio64.exe	30/03/2020 5:04 PM	Application	793 KB
studio64.exe.vmoptions	30/03/2020 5:04 PM	VMOPTIONS File	1 KB

Hình 34: Kiểm tra sau khi cài đặt Android Studio (2)

Ta có thể chạy công cụ lập trình có tên “studio64.exe”, hoặc đưa ra Desktop để chạy cho lẹ. Còn đây là SDK.



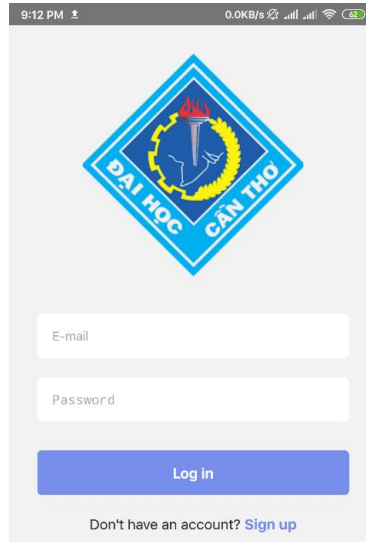
Name	Date modified	Type	Size
.downloadIntermediates	31/03/2020 7:50 PM	File folder	
.temp	31/03/2020 7:50 PM	File folder	
build-tools	22/03/2020 10:40 ...	File folder	
cmake	01/01/2020 12:20 ...	File folder	
emulator	31/03/2020 7:50 PM	File folder	
extras	14/12/2018 10:39 ...	File folder	
fonts	14/12/2018 10:18 ...	File folder	
licenses	11/11/2019 7:32 A...	File folder	
lldb	01/01/2020 12:27 ...	File folder	
ndk	01/01/2020 2:48 PM	File folder	
ndk-bundle	22/03/2020 10:58 ...	File folder	
patcher	14/12/2018 9:54 PM	File folder	
platforms	22/12/2019 5:39 PM	File folder	
platform-tools	20/03/2020 3:51 A...	File folder	
skins	28/03/2020 11:22 ...	File folder	
sources	22/03/2020 10:42 ...	File folder	
system-images	09/10/2019 7:10 PM	File folder	
tools	14/12/2018 9:54 PM	File folder	
.knownPackages	04/04/2020 10:34 ...	KNOWNPACKAGE...	1 KB

Hình 35: Kiểm tra sau khi cài đặt Android Studio (3)

## 3.2 Thiết kế giao diện

### 3.2.1 Giao diện của người quản trị

#### 3.2.1.1 Giao diện đăng nhập



Hình 36: Giao diện đăng nhập

- Giao diện đăng nhập của người quản trị, mỗi quản trị viên được cấp một tài khoản để đăng nhập vào quản trị hệ thống.

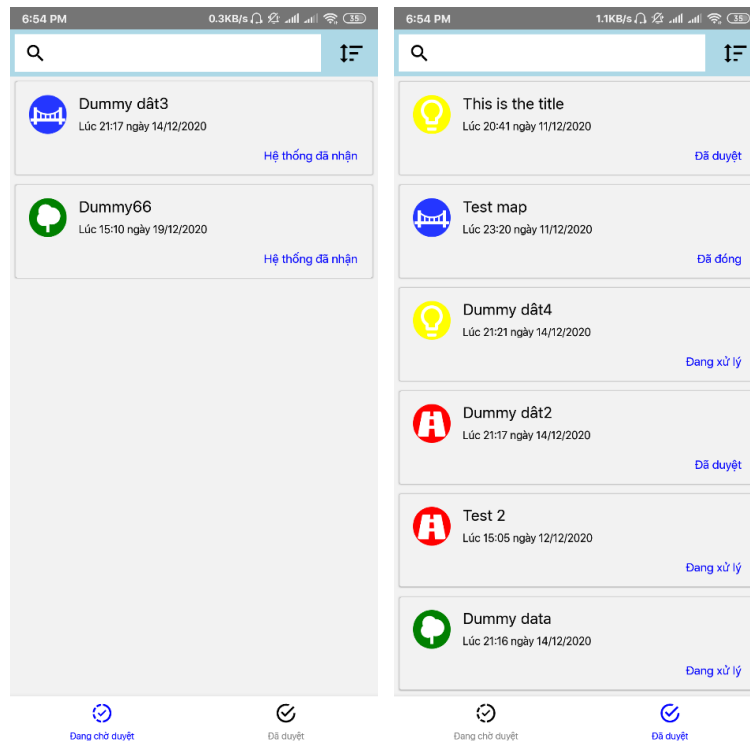
#### 3.2.1.2 Giao diện trang chủ



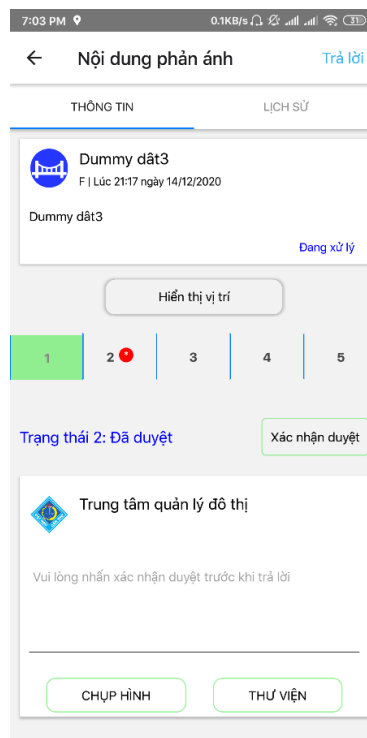
Hình 37: Giao diện trang chủ của người quản trị

- Giao diện trang chủ hiện các chức năng chính dành cho người quản trị.

#### 3.2.1.3 Giao diện duyệt phản ánh

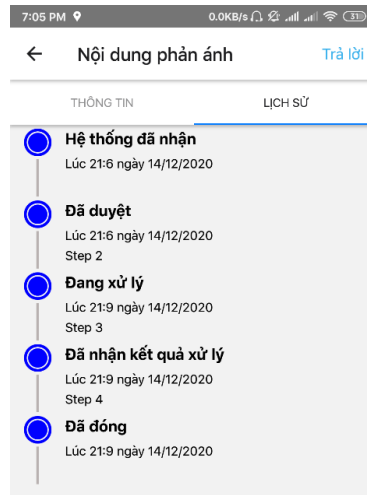


Hình 38: Giao diện xem các phản ánh đang chờ duyệt và đã duyệt



Hình 39: Giao diện duyệt phản ánh

- Chức năng duyệt phản ánh gồm có 5 bước để duyệt. Sau khi duyệt sẽ hiện thị nội dung trả lời của người quản trị ở trang lịch sử.



Hình 40: Giao diện lịch sử trả lời của phản ánh

### 3.2.1.4 Giao diện cập nhật tiện ích và tình trạng giao thông



Hình 41: Giao diện cập nhật tiện ích và tình trạng giao thông

- Giao diện này để người quản trị cập nhật các tiện ích ví dụ như nhà vệ sinh, bệnh viện, cây xăng, ....., cập nhật tình trạng giao thông trên các tuyến đường.

### 3.2.2 Giao diện của người dùng

#### 3.2.2.1 Giao diện trang chủ



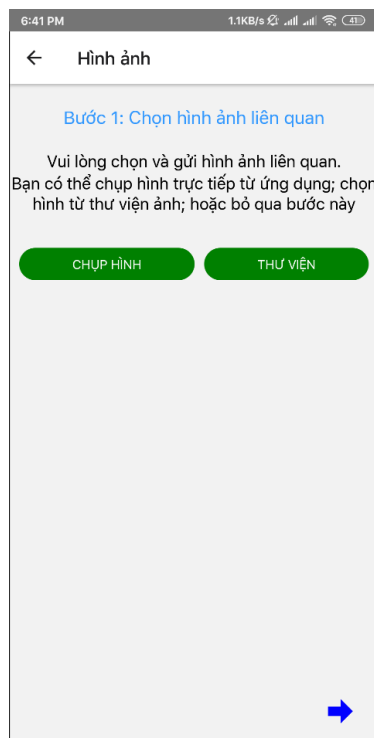


Hình 42: Giao diện trang chủ của người dùng

- Giao diện trang chủ của người dùng hiện các chức năng dành cho người dùng như: gửi phản ánh, xem bản đồ tiện ích, ....

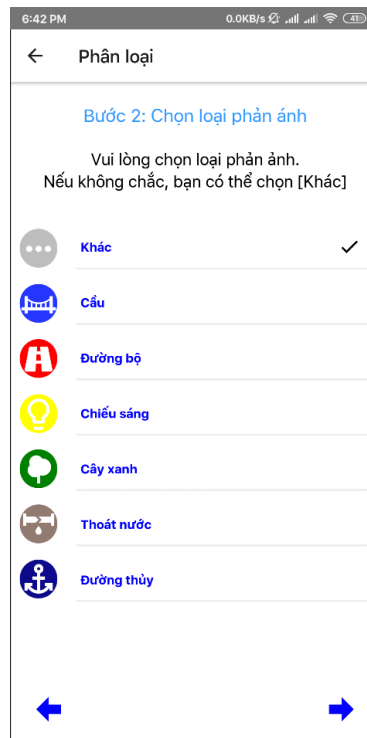
### 3.2.2.2 Giao diện phản ánh

Giao diện phản ánh để cho người dùng phản ánh thông tin trực tuyến về tình trạng đường xá, cầu cống, ... tới đơn vị quản lý xử lý. Gồm có các bước như sau:



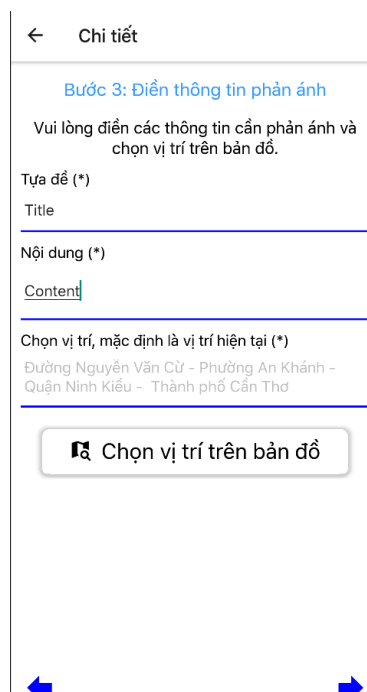
Hình 43: Giao diện bước 1 của chức năng phản ánh

- Bước 1: Chọn hình ảnh liên quan đến thông tin cần phản ánh



Hình 44: Giao diện bước 2 của chức năng phản ánh

- Bước 2: Chọn loại phản ánh ví dụ như phản ánh về cầu, đường bộ, chiếu sáng, cây xanh, thoát nước, đường thủy, ....



Hình 45: Giao diện bước 3 của chức năng phản ánh

- Bước 3: Điền thông tin cần phản ánh và chọn vị trí

←

Thông tin người đăng

Bước 4: Điền thông liên lạc

Vui lòng điền các thông tin cá nhân để chúng tôi liên lạc khi phản ánh của bạn được xử lý

Họ Tên (\*)

Số điện thoại (\*)

Email

☐ Ẩn danh

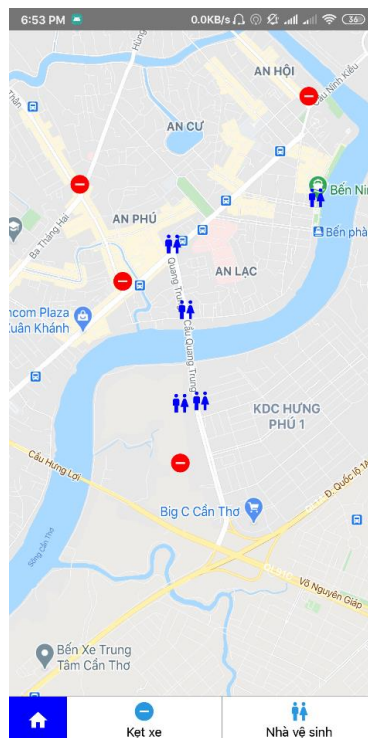
Đăng

←

Hình 46: Giao diện bước 4 của chức năng phản ánh

- Bước 4: Điền thông tin cá nhân để đơn vị quản lý liên lạc

### 3.2.2.3 Giao diện bản đồ tiện ích



Hình 47: Giao diện bản đồ tiện ích

## CHƯƠNG 4: KIỂM THỬ VÀ KẾT QUẢ

### 4.1 Thiết bị kiểm thử

Chúng tôi thực hiện kiểm thử trên các thiết bị:

- Dell 15-5455:
  - + Intel core i7-5500 2.4GHz.
  - + RAM 8GB.
  - + HDD 1TB.
  - + Windows 10 pro 64-bit.
- Xiaomi redmi note 5:
  - + Android 8.0.
  - + Snapdragon 636 8 nhân.
  - + IPS LCD 6'', Full HD.
- Máy ảo Google Pixel:
  - + Android 7.0.
  - + RAM: 4GB/ ROM 32GB.
  - + Màn hình: 5.5'', Quad HD.

### 4.2 Mục tiêu kiểm thử

- Kiểm tra tất cả màn hình có hiển thị đầy đủ các thể hiện đã tạo hay không.
- Kiểm tra các hình ảnh, các nút, các ô, ... có hiển thị đúng hay không.
- Kiểm tra các nút có hoạt động đúng như mục đích hay không.
- Kiểm tra có kết nối được với cơ sở dữ liệu Firebase hay không, nhanh hay chậm.
- Kiểm tra các tính năng có hoạt động đúng hay không.

### 4.3 Kịch bản kiểm thử

- Điều hướng đến tất cả các màn hình trong ứng dụng để kiểm tra giao diện có hiển thị đúng hay không.
- Thực hiện tìm kiếm, lọc phản ánh, xem bản đồ tiện ích, ... trên ứng dụng.

- Thực hiện tạo dữ liệu xem có kết nối được với internet hay không, có kết nối đúng tới cơ sở dữ liệu hay không.

#### **4.4 Kết quả kiểm thử**

- Tất cả các giao diện đều hiển thị rõ ràng trên môi trường kiểm thử.
- Các nút, các phím điều hướng thực hiện đúng chức năng đã tạo.
- Lấy được dữ liệu từ cơ sở dữ liệu Firebase về ứng dụng.

## PHẦN KẾT LUẬN

### 1. Kết quả đạt được

Quá trình nghiên cứu và thực hiện niên luận, chúng tôi đã đạt được kết quả sau:

- Ứng dụng “Thông tin giao thông” với các chức năng:
  - + Giúp người dùng gửi phản ánh trực tuyến đến đơn vị quản lý để xử lý trong thời gian ngắn nhất.
  - + Tìm kiếm, lọc và xem các phản ánh trực tuyến của người dân.
  - + Xem tình trạng giao thông
  - + Tìm kiếm các tiện ích như cây xăng, trường học, bệnh viện, nhà vệ sinh công cộng, ....

### 2. Hạn chế

Tuy nhiên ứng dụng còn một số hạn chế:

- Ứng dụng có độ chính xác chưa cao về vị trí các tiện ích trên bản đồ.
- Tốc độ xử lý cập nhật chưa được nhanh.
- Giao diện chưa được đẹp mắt.
- Chưa xây dựng và kiểm thử trên hệ điều hành iOS.

### 3. Hướng phát triển

Dựa trên các kết quả đạt được và những hạn chế cần khắc phục, hệ thống cần được phát triển theo hướng sau:

- Về giao diện, nghiên cứu và thiết kế giao diện và trải nghiệm người dùng mang đến cảm giác thích thú và tin tưởng khi sử dụng ứng dụng.
- Thuê một Server của Firebase để có thể đáp ứng được lượt truy xuất dữ liệu số lượng lớn tại một thời điểm.
- Triển khai ứng dụng trên 2 nền tảng Android và iOS.

## TÀI LIỆU THAM KHẢO

- 1.[https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n\\_m%E1%BB%81m\\_%E1%BB%A9ng\\_d%E1%BB%A5ng](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_%E1%BB%A9ng_d%E1%BB%A5ng)
- 2.<https://nordiccoder.com/blog/tong-quan-ve-react-native/>
- 3.<https://nordiccoder.com/blog/blog-tong-quan-ve-lap-trinh-android-lap-trinh-di-dong-android-la-gi/>
- 4.<https://blog.tinohost.com/firebase-la-gi/>
- 5.<https://firebase.google.com/>
- 6.<https://developer.android.com/>