**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**

**TRẦN NGUYỄN AN**

**<TÊN ĐỒ ÁN NGÀNH>**

**ĐỒ ÁN NGÀNH**

**NGÀNH KHOA HỌC MÁY TÍNH**

**TP. HỒ CHÍ MINH, 2023BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**

**TRẦN NGUYỄN AN**

**< TÊN ĐỒ ÁN NGÀNH >**

**Mã số sinh viên: 2051010004**

**ĐỒ ÁN NGÀNH**

**NGÀNH KHOA HỌC MÁY TÍNH**

**Giảng viên hướng dẫn: Tiến sĩ TRƯƠNG HOÀNG VINH**

**TP. HỒ CHÍ MINH, 2023**

LỜI CẢM ƠN

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

TÓM TẮT ĐỐ ÁN NGÀNH

Đề tài <tên đồ án> phát triển một phần mềm giúp người đi được nhận được cảnh báo về các đoạn đường đang bị kẹt xe. Sử dụng API của Google Map kết hợp với phương thức getCurrentPosition() để biết được người dùng đang ở vị trí nào và cảnh báo cho người đến sau qua thông báo của điện thoại. Sử dụng SQL Server để quản lý và truy xuất Database, sẽ tìm cách đưa Database triển khai online. Dùng ngôn ngữ C# tạo RESTful API để thực hiện các thao tác truy xuất dữ liệu từ SQL Server. Dùng ngôn ngữ Java trong lập trình ứng dụng di động.

Khi sử dụng người đi đường cần có tài khoản để có thể đăng lên các cảnh báo. Sẽ kiểm tra từ nhiều tài khoản tránh trường hợp có người muốn lợi dụng ngầm gây rồi cho người đi đường khác. Dựa trên tọa độ từ người đăng đầu tiên sẽ kiểm tra xung quanh xem có người khác đăng hay không. Sau đó sẽ dựa trên tọa độ đó để cảnh bảo cho những người đang ở gần.

ABSTRACT

**The topic "<project name>" involves the development of software aimed at providing real-time traffic congestion alerts to commuters. It utilizes the Google Maps API in conjunction with the getCurrentPosition() method to determine the user's current location and issues traffic alerts to those following behind via smartphone notifications. SQL Server is used for database management and retrieval, with plans to deploy the database online. A RESTful API is created using the C# programming language to facilitate data retrieval operations from SQL Server. Java is employed for mobile application development.**

**Users are required to have accounts to post alerts and prevent misuse by other users. Multiple accounts are checked to avoid situations where individuals may attempt to exploit the system to create false alerts. Based on the coordinates provided by the first user, the system checks the surrounding area for additional alerts from other users. Subsequently, it issues alerts to those who are in close proximity.**

**MỤC LỤC**

[DANH MỤC TỪ VIẾT TẮT 6](#_Toc51061474)

[DANH MỤC HÌNH VẼ 7](#_Toc51061475)

[DANH MỤC BẢNG 8](#_Toc51061476)

[MỞ ĐẦU 9](#_Toc51061477)

[Chương 1. TÊN CHƯƠNG 1 10](#_Toc51061478)

[1.1. Chủ đề cấp độ 2 10](#_Toc51061479)

[1.1.1. Chủ đề cấp độ 3 10](#_Toc51061480)

[1.1.2. Chủ đề cấp độ 3 10](#_Toc51061481)

[Chương 2. TÊN CHƯƠNG 2 11](#_Toc51061482)

[2.1. Chủ đề cấp độ 2 11](#_Toc51061483)

[2.1.1. Chủ đề cấp độ 3 11](#_Toc51061484)

[2.2. Chủ đề cấp độ 2 11](#_Toc51061485)

[2.2.1. Chủ đề cấp độ 3 11](#_Toc51061486)

[Chương 3. TÊN CHƯƠNG 3 12](#_Toc51061487)

[3.1. Chủ đề cấp độ 2 12](#_Toc51061488)

[3.1.1. Chủ đề cấp độ 3 12](#_Toc51061489)

[3.2. Chủ đề cấp độ 2 12](#_Toc51061490)

[TÀI LIỆU THAM KHẢO 13](#_Toc51061491)

[PHỤ LỤC 14](#_Toc51061492)

DANH MỤC TỪ VIẾT TẮT

DANH MỤC HÌNH VẼ

[Hình 1.1: Tên hình 1 10](#_Toc367742554)

DANH MỤC BẢNG

[Bảng 1.1: Tên bảng 1 10](#_Toc367742567)

[Bảng 2.1: Tên bảng 1 11](#_Toc367742568)

MỞ ĐẦU

# GIỚI THIỆU ĐỀ TÀI

## Giới thiệu

Trong thời đại hiện đại với tốc độ đô thị hóa gia tăng, vấn đề kẹt xe trở thành một thách thức lớn đối với cuộc sống hàng ngày của mọi người. Việc bị kẹt xe không chỉ gây ra sự bất tiện mà còn tác động tiêu cực đến môi trường, thời gian và tình trạng tâm lý của người tham gia giao thông. Vì vậy, để giải quyết vấn đề này, chúng tôi đã nghĩ đến việc phát triển một ứng dụng Android có chức năng cảnh báo các địa điểm kẹt xe gần khu vực của người sử dụng.

Một ứng dụng giúp người đi đường nhận cảnh báo về tình trạng kẹt xe trên các đoạn đường gần khu vực người sử dụng. Ứng dụng sử dụng API của Google Maps và các phương thức để xác định vị trí hiện tại của người dùng và cảnh báo họ về tình trạng giao thông qua thông báo trên điện thoại.

Để sử dụng ứng dụng này, người đi đường cần phải tạo một tài khoản để có thể đăng thông tin về tình trạng giao thông. Điều này giúp kiểm soát và tránh việc có người sử dụng ứng dụng với mục đích gây rối cho người đi đường khác. Ứng dụng sẽ kiểm tra thông tin từ nhiều tài khoản và dựa trên tọa độ của người đăng đầu tiên để cảnh báo cho những người ở gần tình trạng kẹt xe.

## Mục tiêu đề tài

### Thiết kế Giao Diện Người Dùng (UI/UX)

Trong giai đoạn này, mục tiêu là tạo ra một giao diện người dùng thân thiện, và dễ sử dụng. Điều này đảm bảo rằng người dùng có trải nghiệm tốt khi sử dụng ứng dụng.

### Phát triển Tính Năng Cơ Bản

Tạo ra các tính năng cơ bản như xác định vị trí hiện tại của người dùng, hiển thị thông tin về tình trạng giao thông, và cảnh báo kẹt xe. Đảm bảo tính năng hoạt động một cách chính xác và ổn định.

### Tích hợp Cơ Sở Dữ Liệu SQL Server

Tạo kết nối với cơ sở dữ liệu SQL Server để lưu trữ và quản lý thông tin về tình trạng giao thông. Bao gồm cả việc lưu trữ thông tin về tài khoản người dùng và thông tin đăng kẹt xe.

### Phát triển Tính Năng Đăng Nhận Cảnh Báo

Phát triển tính năng cho phép người dùng đăng thông tin về kẹt xe và nhận cảnh báo từ những người khác.

### Bảo Mật và Quản lý Tài Khoản

Bảo mật dữ liệu người dùng và quản lý tài khoản người dùng một cách an toàn.

## Lý do chọn đề tài

### Tính Thực Tiễn

Vấn đề kẹt xe là một thách thức phổ biến và đáng lo ngại trong cuộc sống đô thị ngày nay. Với tốc độ gia tăng của số lượng phương tiện và mật độ dân số tăng cao, ứng dụng như này có tiềm năng giúp cải thiện tình trạng giao thông.

### Tính Ứng Dụng Rộng Rãi

Ứng dụng cảnh báo kẹt xe có tiềm năng được sử dụng rộng rãi bởi mọi người tham gia giao thông, từ người lái xe hàng ngày đến những người cần di chuyển trong thành phố.

### Khả Năng Ứng Dụng Thực Tế

Có thể trở thành một dự án thực tế và hữu ích cho cộng đồng. Nó có tiềm năng để được triển khai và sử dụng hàng ngày.

### Khám Phá Kiến Thức Mới

Việc phát triển một ứng dụng Android đòi hỏi sự tìm hiểu và áp dụng nhiều kiến thức về lập trình, giao diện người dùng, và quản lý dữ liệu, giúp mở rộng kiến thức và kỹ năng của người thực hiện.

### Tính Đột Phá

Nếu thành công, đề tài này có tiềm năng trở thành một ứng dụng có tính đột phá trong việc giải quyết vấn đề kẹt xe và cải thiện giao thông đô thị.

### Chủ đề cấp độ 3

Nội dung

### Chủ đề cấp độ 3

#### Chủ đề cấp độ 4

Nội dung

Hình 1.1: Tên hình 1

Bảng 1.1: Tên bảng 1

# CƠ SỞ LÝ THUYẾT

## Quản Lý Dữ Liệu

Quản lý dữ liệu là quá trình tổ chức, lưu trữ, và quản lý thông tin trong một hệ thống hoặc tổ chức. Điều này bao gồm việc thu thập, lưu trữ, sắp xếp, xử lý, bảo mật và cung cấp dữ liệu một cách hiệu quả và an toàn.

### Thu thập dữ liệu

Quản lý dữ liệu bắt đầu bằng việc thu thập thông tin về tình trạng giao thông và kẹt xe được thực hiện thông qua dữ liệu từ người dùng ứng dụng.

### Lưu trữ dữ liệu

Dữ liệu thu thập cần được lưu trữ sao cho dễ dàng truy cập và sử dụng. Cơ sở dữ liệu có thể được sử dụng để lưu trữ thông tin về tình trạng giao thông, các điểm đánh dấu vị trí, và các thông tin khác liên quan đến đề tài.

### Xử lý dữ liệu

Dữ liệu thu thập thường cần được xử lý để trích xuất thông tin hữu ích và tạo ra báo cáo. Phân tích dữ liệu để xác định các điểm kẹt xe, tính toán thời gian ước tính để đi qua các tuyến đường, và hiển thị thông tin trên bản đồ.

### Bảo mật dữ liệu

Vì dữ liệu liên quan đến vị trí và giao thông có tính riêng tư và quan trọng, việc bảo mật dữ liệu là rất quan trọng.

### Quản lý tài khoản người dung

Thu thập và lưu trữ thông tin đăng nhập và quản lý quyền truy cập của người dùng.

### Quản lý dữ liệu trực tuyến

Quản lý và duy trì cơ sở dữ liệu trực tuyến, đảm bảo tính khả dụng và độ tin cậy của dữ liệu.

## Xây Dựng RESTful API (C#)

Việc xây dựng một RESTful API (Application Programming Interface) bằng ngôn ngữ C# là một phần quan trọng của quá trình phát triển ứng dụng.

### RESTful API là gì?

REST là viết tắt của "Representational State Transfer," là một kiến trúc dùng để thiết kế các dịch vụ web (web services) dựa trên giao thức HTTP. RESTful API là một loại API tuân theo các nguyên tắc và quy tắc của kiến trúc REST.

### Mục tiêu của RESTful API

Cung cấp một cách tiếp cận cho ứng dụng Android để truy xuất và tương tác với dữ liệu về tình trạng giao thông và thông tin kẹt xe. Cho phép người dùng đăng thông tin cảnh báo về kẹt xe thông qua giao diện API. Cung cấp các chức năng quản lý tài khoản và quyền truy cập cho người dùng ứng dụng di động.

### Ngôn ngữ lập trình C#

C# là một ngôn ngữ lập trình phát triển bởi Microsoft, được sử dụng phổ biến trong việc phát triển ứng dụng Windows và dịch vụ web. C# là một ngôn ngữ đa năng và mạnh mẽ, phù hợp cho việc xây dựng RESTful API.

### Các phương thức HTTP

RESTful API sử dụng các phương thức HTTP như GET, POST, PUT, DELETE để thực hiện các thao tác truy vấn và tương tác với dữ liệu.

* GET: Sử dụng để truy xuất thông tin từ API.
* POST: Sử dụng để tạo mới dữ liệu hoặc thực hiện các thao tác khác.
* PUT: Sử dụng để cập nhật thông tin hiện có.
* DELETE: Sử dụng để xóa thông tin.

### Định dạng dữ liệu

RESTful API trả về dữ liệu dưới dạng JSON (JavaScript Object Notation) để dễ dàng truyền tải và phân tích bởi ứng dụng di động.

### Xác thực và quản lý phiên làm việc

Để bảo mật API, cần xác thực người dùng và quản lý phiên làm việc (session) của họ bao gồm việc sử dụng mã thông báo (token) để đảm bảo tính riêng tư và bảo mật.

### Quản lý lỗi và xử lý ngoại lệ

RESTful API cần có khả năng xử lý lỗi và trả về các mã trạng thái HTTP thích hợp để báo cáo kết quả của các yêu cầu API, ví dụ: mã 200 OK, mã 404 Not Found, mã 500 Internal Server Error, v.v.

## An Toàn và Quản Lý Tài Khoản

### Bảo mật dữ liệu

Dữ liệu về tình trạng giao thông và thông tin kẹt xe là quan trọng và có giá trị, vì vậy cần phải được bảo mật.

### Xác thực người dùng

Để đảm bảo tính riêng tư và an toàn, ứng dụng cần phải xác thực người dùng khi họ đăng nhập. Sử dụng một cơ chế xác thực như tên người dùng và mật khẩu hoặc mã thông báo (token) dựa trên phiên làm việc.

### Bảo vệ khỏi tấn công

Cần phải có các biện pháp bảo vệ khỏi các loại tấn công như tấn công SQL Injection, tấn công Cross-Site Scripting (XSS), và tấn công Cross-Site Request Forgery (CSRF). Điều này đòi hỏi việc kiểm tra và xử lý đầu vào người dùng cẩn thận.

### Đăng ký và đăng nhập

Quản lý tài khoản bao gồm việc cho phép người dùng đăng ký tài khoản mới và đăng nhập vào tài khoản hiện có.

### Quên mật khẩu

Cần cung cấp một cách cho người dùng khôi phục mật khẩu nếu họ quên mật khẩu của mình. Thực hiện thông qua việc gửi mật khẩu mới được tạo thông qua một phương thức và gửi đến email đã đăng ký của người dùng.

### Quản lý thông tin cá nhân

Người dùng cần có khả năng quản lý thông tin cá nhân của họ, bao gồm việc cập nhật thông tin liên hệ và thay đổi mật khẩu.

### Quản lý quyền truy cập

Ứng dụng cần có khả năng quản lý quyền truy cập của người dùng. Xác định những tính năng và dữ liệu mà mỗi người dùng có quyền truy cập.

### Bảo mật mật khẩu

Mật khẩu của người dùng cần được bảo mật bằng cách lưu trữ chúng trong dạng được mã hóa và sử dụng các biện pháp bảo vệ như đặt điều kiện về độ dài và phức tạp của mật khẩu.

### Chủ đề cấp độ 3

#### Chủ đề cấp độ 4

Bảng 2.1: Tên bảng 1

## Chủ đề cấp độ 2

### Chủ đề cấp độ 3

# TÊN CHƯƠNG 3

## Chủ đề cấp độ 2

Nội dung …………………

Nội dung………………….

### Chủ đề cấp độ 3

#### Chủ đề cấp độ 4

## Chủ đề cấp độ 2

TÀI LIỆU THAM KHẢO

(Theo chuẩn IEEE – *bỏ dòng này khi viết báo cáo*)

PHỤ LỤC