

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**  
**KHOA KHOA HỌC VÀ KỸ THUẬT THÔNG TIN**

---



**PHÂN TÍCH THIẾT KẾ PHẦN MỀM**  
**TÊN ĐỀ TÀI: PHÂN TÍCH VÀ THIẾT KẾ HƯỚNG ĐỐI**  
**TƯỢNG VỚI UML 2.0**

**GVHD:** TS. Nguyễn Tấn Cầm

**Lớp:** IE108.O21.CNVN

**Sinh viên thực hiện:**

1. Nguyễn Cao Thắng MSSV: 22521329
2. Đỗ Đức Phú MSSV: 22521098

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

(Ký tên và ghi rõ họ tên)

## MỤC LỤC

<b>LỜI MỞ ĐẦU .....</b>	<b>5</b>
<b>LỜI CẢM ƠN .....</b>	<b>6</b>
<b>CHƯƠNG 1: TỔNG QUAN VỀ UML 2.0.....</b>	<b>7</b>
<b>CHƯƠNG 2: TỔNG QUAN VỀ PHÂN TÍCH VÀ THIẾT KẾ HƯỚNG ĐỐI TƯỢNG .....</b>	<b>10</b>
1. Phân tích hướng đối tượng.....	10
2. Thiết kế hướng đối tượng.....	11
<b>CHƯƠNG 3: HỆ THỐNG GIAO HÀNG NHANH.....</b>	<b>13</b>
1. Mô tả bài toán.....	13
2. Phân tích hướng đối tượng.....	14
2.1. Biểu đồ use case .....	14
2.2. Biểu đồ sequence.....	16
2.3. Biểu đồ class.....	19
2.4. Biểu đồ triển khai (Deployment diagram).....	20
3. Thiết kế hướng đối tượng.....	20
3.1. Thiết kế kiến trúc phần mềm.....	20
3.2. Mô tả môi trường.....	21
3.3. Thiết kế cơ sở dữ liệu .....	21
3.4. Thiết kế giao diện người dùng.....	25
<b>CHƯƠNG 4: KẾT LUẬN .....</b>	<b>26</b>
<b>TÀI LIỆU THAM KHẢO .....</b>	<b>27</b>

## **MỤC LỤC HÌNH ẢNH**

Hình 1: 14 loại biểu đồ UML 2.0.....	8
Hình 2: 4 loại quan hệ trong UML.....	9
Hình 3: Một số phần tử trong mô hình UML 2.0.....	9
Hình 4: Sơ đồ các bước phân tích hướng đối tượng .....	11
Hình 5: Sơ đồ các bước thiết kế hướng đối tượng .....	12
Hình 6: Minh họa ô tô chuyển phát nhanh không người lái của FedEx .....	13
Hình 7: Biểu đồ use case của hệ thống giao hàng.....	14
Hình 8: Biểu đồ tuần tự của tạo đơn hàng.....	16
Hình 9: Biểu đồ tuần tự của xác nhận giao hàng .....	17
Hình 10: Biểu đồ tuần tự cho thông báo tình trạng đơn hàng.....	18
Hình 11: Biểu đồ lớp cho hệ thống giao hàng .....	19
Hình 12: Biểu đồ triển khai hệ thống trên điện thoại di động.....	20
Hình 13: CSDL cho hệ thống giao hàng .....	24
Hình 14. Giao diện hệ thống giao hàng trên điện thoại di động .....	25

## **LỜI MỞ ĐẦU**

Đề tài "Phân tích và Thiết kế Hướng đối tượng bằng UML 2.0" là một chủ đề quan trọng trong lĩnh vực phát triển phần mềm. Trong đề tài này, chúng ta sẽ khám phá và áp dụng các khái niệm và công cụ của hướng đối tượng cùng với ngôn ngữ mô hình hóa thống nhất (UML) để phân tích và thiết kế các hệ thống phần mềm.

Chúng tôi chọn đề tài này vì nó phổ biến và cần thiết trong lĩnh vực phát triển phần mềm. Kiến thức sâu về phân tích và thiết kế hướng đối tượng cũng mở ra nhiều cơ hội nghề nghiệp trong nhiều lĩnh vực khác nhau, từ công nghệ thông tin đến phần mềm nhúng và thương mại điện tử.

## **LỜI CẢM ƠN**

Đầu tiên, nhóm em xin gửi lời cảm ơn sâu sắc đến giảng viên bộ môn – thầy Cầm đã truyền đạt những kiến thức bổ ích trong suốt thời gian học tập vừa qua. Trong thời gian tham gia lớp Phân tích thiết kế phần mềm, thầy đã dạy tận tình, chi tiết để chúng em có đủ kiến thức và vận dụng chúng vào đồ án này. Do chưa có nhiều kinh nghiệm làm đề tài cũng như những hạn chế về kiến thức, trong đồ án chắc chắn sẽ không tránh khỏi những thiếu sót. Rất mong nhận được sự nhận xét, ý kiến đóng góp từ phía thầy để đồ án chúng em được hoàn thiện hơn.

Lời cuối cùng, chúng em xin kính chúc thầy có nhiều sức khỏe, thành công và hạnh phúc.

Xin trân trọng cảm ơn!

## **CHƯƠNG 1: TỔNG QUAN VỀ UML 2.0**

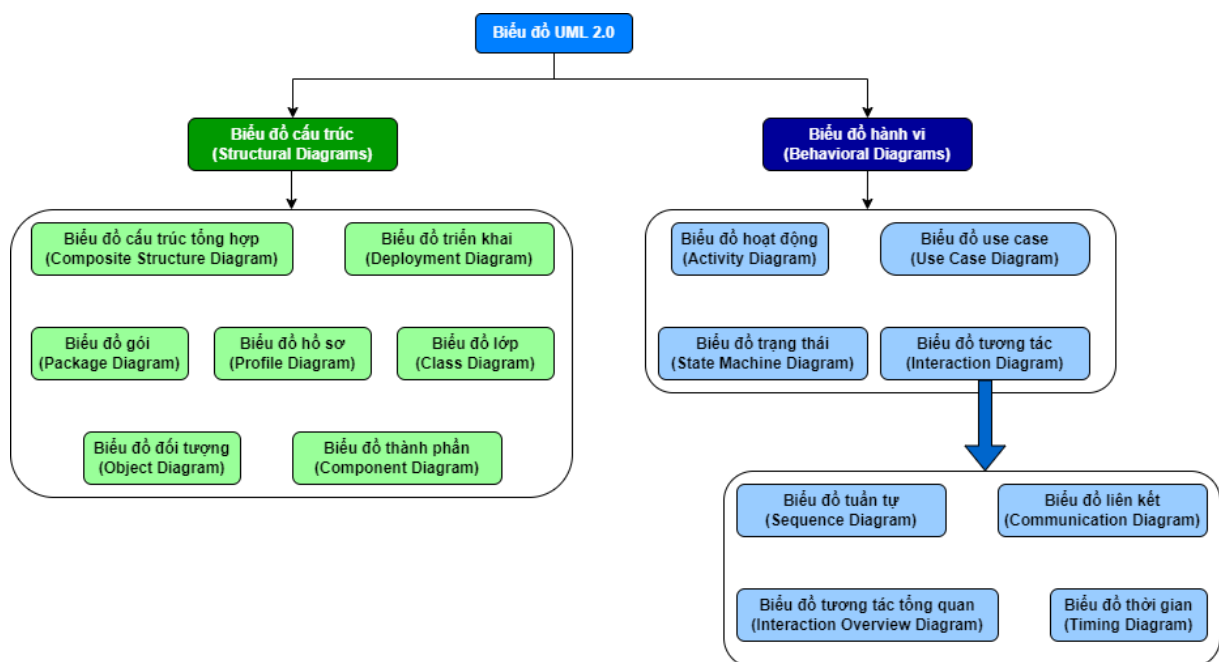
UML (Unified Modeling Language – Ngôn ngữ mô hình hóa thống nhất) là một công cụ dùng để xây dựng các mô hình cho hệ thống phần mềm. UML 2.0 là phiên bản cải tiến so với UML 1.x, được công bố vào tháng 7 năm 2005 [1].

UML 2.0 hỗ trợ 14 loại biểu đồ chính bao gồm:

- Biểu đồ cấu trúc tổng hợp (Composite Structure Diagram): Mô tả cấu trúc nội bộ của một lớp hoặc một thành phần, bao gồm các phần tử con và mối quan hệ giữa chúng
- Biểu đồ triển khai (Deployment Diagram): Mô tả cách các thành phần của hệ thống được triển khai trên các nút (node) vật lý hoặc logic
- Biểu đồ gói (Package Diagram): Mô tả cách các phần của hệ thống được tổ chức thành các gói và mối quan hệ giữa chúng
- Biểu đồ hồ sơ (Profile Diagram): Mô tả các nguyên tắc thiết kế, mẫu kiến trúc, hoặc các ngôn ngữ mô hình hóa đặc biệt cho một lĩnh vực cụ thể như ngành công nghiệp hoặc hệ thống cụ thể (y tế)
- Biểu đồ lớp (Class Diagram): Mô tả cấu trúc tĩnh của hệ thống bằng cách chỉ ra các lớp, thuộc tính, phương thức và mối quan hệ giữa chúng
- Biểu đồ đối tượng (Object Diagram): Mô tả một thể hiện cụ thể của các lớp và mối quan hệ giữa chúng tại một thời điểm cụ thể
- Biểu đồ thành phần (Component Diagram): Mô tả cấu trúc vật lý của hệ thống bằng cách chỉ ra các thành phần, gói và mối quan hệ giữa chúng
- Biểu đồ hoạt động (Activity Diagram): Mô tả các hoạt động, quy trình, hoặc luồng làm việc trong hệ thống, thường được sử dụng để mô tả quy trình kinh doanh hoặc logic của một chức năng
- Biểu đồ use case (Use Case Diagram): Biểu đồ này mô tả các tác nhân (người dùng hoặc hệ thống khác), các use case và mối quan hệ giữa chúng

## IE108.O21.CNVN – Phân tích thiết kế phần mềm

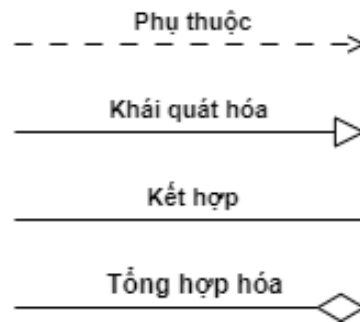
- Biểu đồ trạng thái (State Machine Diagram): Mô tả các trạng thái của một đối tượng hoặc một lớp và các sự kiện có thể chuyển đổi đối tượng hoặc lớp đó từ một trạng thái sang trạng thái khác
- Biểu đồ tuần tự (Sequence Diagram): Biểu đồ này mô tả các tương tác giữa các đối tượng theo thời gian, thường được sử dụng để mô tả các kịch bản sự kiện
- Biểu đồ liên kết (Communication Diagram): Tương tự như biểu đồ tuần tự, nhưng sử dụng các liên kết (links) thay vì các thực thể để biểu diễn tương tác
- Biểu đồ tương tác tổng quan (Interaction Overview Diagram): Trình bày các tương tác giữa các phần tử của hệ thống một cách tổng thể và rõ ràng hơn
- Biểu đồ thời gian (Timing Diagram): Mô tả sự xuất hiện của các sự kiện trong thời gian, đặc biệt là trong các hệ thống realtime hoặc nhúng



Hình 1: 14 loại biểu đồ UML 2.0

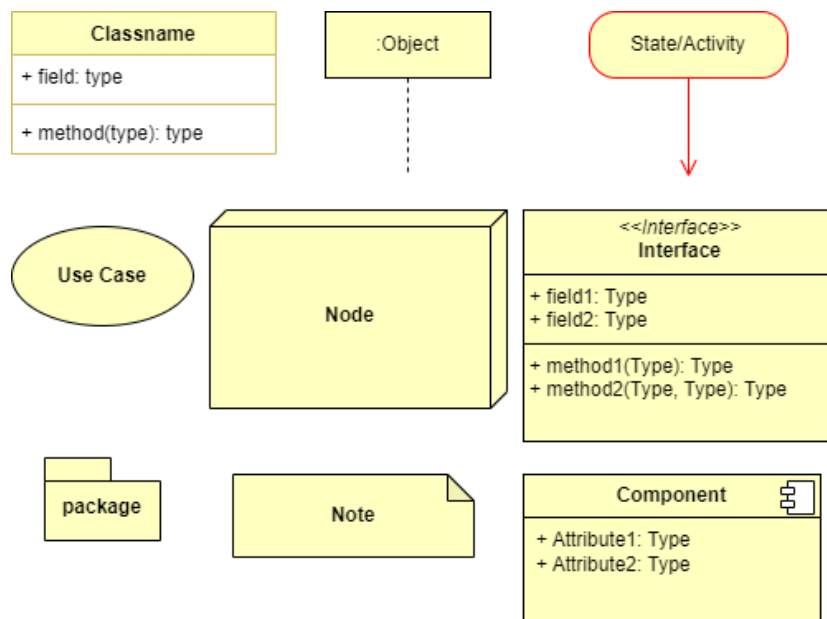
UML có bốn loại quan hệ gồm: phụ thuộc (dependency), khái quát hóa (generalization), kết hợp (association), tổng hợp hóa (aggregation)





Hình 2: 4 loại quan hệ trong UML

Một số phân tử phổ biến trong UML 2.0 bao gồm phân tử lớp, phân tử đối tượng, trạng thái/hành động, use case, nút, hành vi, gói, chú thích, thành phần,...



Hình 3: Một số phân tử trong mô hình UML 2.0

Năm giai đoạn phát triển phần mềm:

- Giai đoạn nghiên cứu sơ bộ
- Giai đoạn phân tích
- Giai đoạn thiết kế
- Giai đoạn xây dựng – lập trình
- Thử nghiệm

Trong phạm vi đồ án, nội dung sẽ tập trung vào giai đoạn phân tích và giai đoạn thiết kế phần mềm và cách áp dụng UML 2.0 vào hai giai đoạn này.

## CHƯƠNG 2: TỔNG QUAN VỀ PHÂN TÍCH VÀ THIẾT KẾ HƯỚNG ĐỐI TƯỢNG

### 1. Phân tích hướng đối tượng

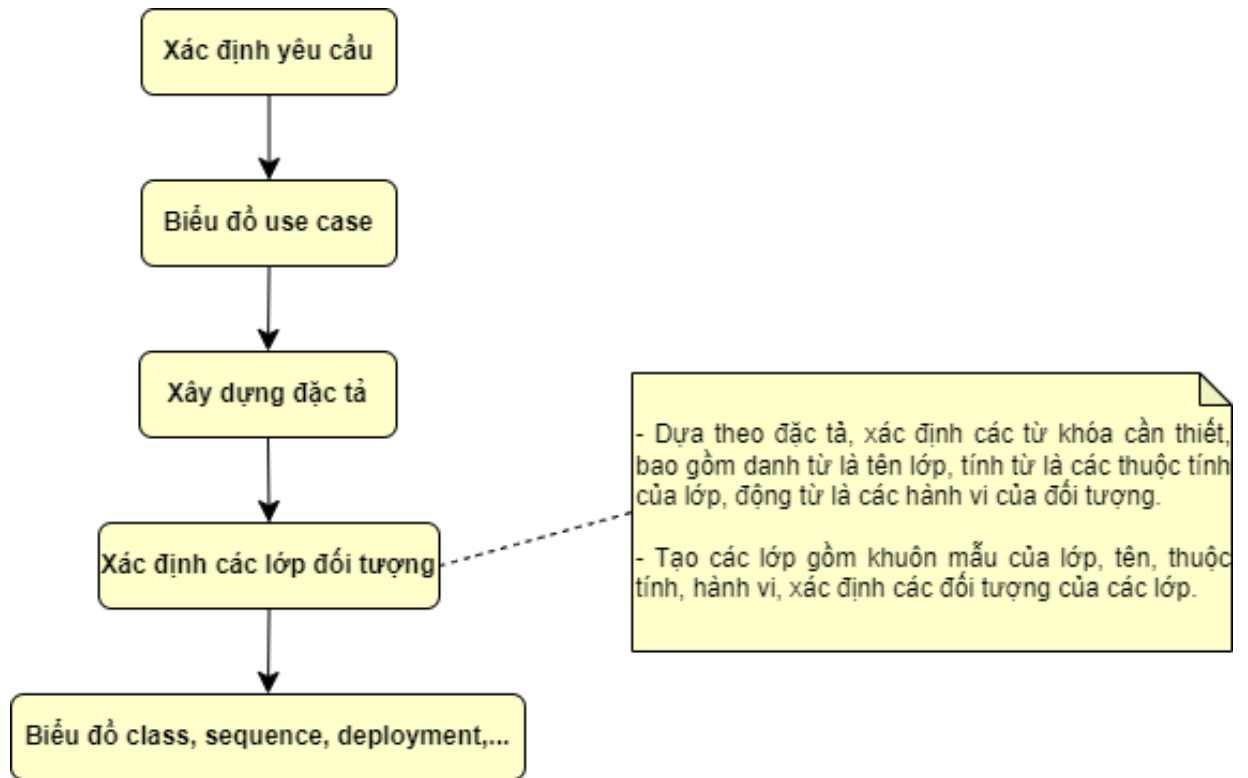
Phân tích hướng đối tượng là phân tích các yêu cầu của bài toán và mô hình hóa các yêu cầu dưới dạng đối tượng. Người phân tích cần xác định rõ các đối tượng, trừu tượng hóa các yêu cầu, so sánh với hệ thống đã có hoặc đang hoạt động trong thực tế để xây dựng một cấu trúc của hệ thống mới [2].

Ưu điểm của phân tích hướng đối tượng:

- Linh hoạt với các yêu cầu thay đổi của khách hàng, hệ thống phần mềm [3]
- Dễ bảo trì và mở rộng hệ thống trong tương lai [3]
- Khả năng tái sử dụng cho các yêu cầu khác, dự án khác [3]
- Hiểu rõ về mối quan hệ giữa các đối tượng trong mô tả bài toán [3]

Sau khi đã có mô tả bài toán và danh sách các yêu cầu, ta tiến hành phân tích hướng đối tượng theo các bước sau:

- Xác định yêu cầu chức năng của người dùng và các yêu cầu phi chức năng ràng buộc
- Mô hình hóa các yêu cầu theo biểu đồ use case
- Xây dựng đặc tả use case
- Xác định các lớp đối tượng
- Mô hình hóa các lớp đối tượng (biểu đồ class, sequence, activity,...)



Hình 4: Sơ đồ các bước phân tích hướng đối tượng

## 2. Thiết kế hướng đối tượng

Thiết kế hướng đối tượng là quá trình chuyển đổi các yêu cầu, đối tượng từ phân tích thành mô hình kiến trúc, cấu trúc cho hệ thống phần mềm, đảm bảo hệ thống dễ quản lý, bảo trì [2].

Thiết kế hướng đối tượng áp dụng nguyên tắc SOLID, trong đó:

- Single Responsibility Principle (SRP): Một lớp chỉ nên xử lý một chức năng duy nhất [3]
- Open-Closed Principle (OCP): Các lớp, module, chức năng nên mở cho việc mở rộng và đóng cho việc thay đổi [3]
- Liskov's Substitution Principle (LSP): Lớp dẫn xuất phải có khả năng thay thế hoàn toàn cho lớp cha của nó [3]
- Interface Segregation Principle (ISP): Chương trình không nên buộc phải cài đặt một interface mà nó không sử dụng [3]
- Dependency Inversion Principle (DIP): Các module cấp cao không nên phụ thuộc vào các module cấp thấp, cả hai nên phụ thuộc vào lớp trừu tượng [3]

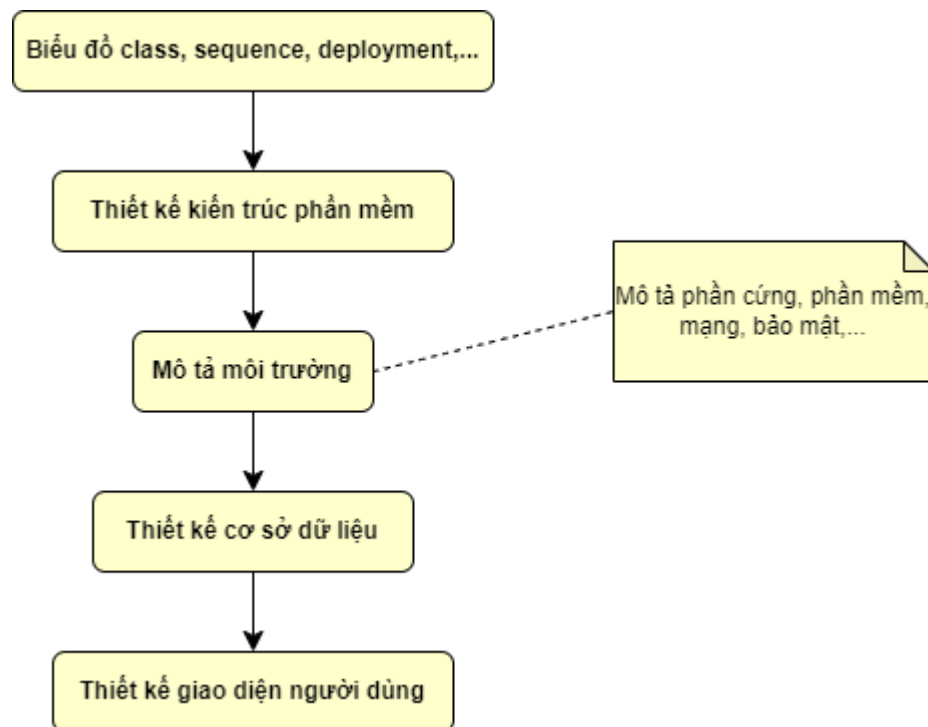
## IE108.O21.CNVN – Phân tích thiết kế phần mềm

Ưu điểm của thiết kế hướng đối tượng:

- Loại bỏ miền dữ liệu dùng chung thông qua cơ chế trao đổi thông tin giữa các đối tượng bằng thông báo [3]
- Mọi đối tượng được thiết kế là các thực thể độc lập, mọi thay đổi về trạng thái, bổ sung, sửa đổi các hoạt động chức năng của một đối tượng chỉ xảy ra bên trong của đối tượng đó, không ảnh hưởng đến đối tượng khác. Điều này đảm bảo hệ thống có tính dễ mở rộng và dễ thích nghi, đáp ứng được nhiều tính chất quan trọng của sản phẩm phần mềm [3]

Sau khi đã phân tích hướng đối tượng, ta chuyển sang giai đoạn thiết kế hướng đối tượng với các bước sau:

- Thiết kế kiến trúc phần mềm
- Mô tả phần mềm, phần cứng, mạng
- Thiết kế cơ sở dữ liệu
- Thiết kế giao diện người dùng



Hình 5: Sơ đồ các bước thiết kế hướng đối tượng

## CHƯƠNG 3: HỆ THỐNG GIAO HÀNG NHANH

### 1. Mô tả bài toán

Ứng dụng giao hàng nhanh là một phần mềm cho phép khách hàng đặt món ăn, mua sắm trực tuyến và nhận hàng hóa tại nhà một cách nhanh chóng và tiện lợi. Ứng dụng này kết nối khách hàng với các nhà hàng, cửa hàng và tài xế giao hàng để tạo điều kiện cho việc giao hàng diễn ra hiệu quả.

Mục tiêu là xây dựng hệ thống đáp ứng nhu cầu của:

- Khách hàng: tìm kiếm, đặt hàng, thanh toán, theo dõi đơn hàng, đánh giá dịch vụ.
- Nhà hàng/cửa hàng: tiếp nhận, xử lý đơn hàng, quản lý thông tin.
- Tài xế: nhận thông báo, di chuyển, cập nhật trạng thái đơn hàng.

Phân tích yêu cầu bao gồm chức năng (quản lý tài khoản, tìm kiếm, đặt hàng, thanh toán, theo dõi, đánh giá,...) và phi chức năng (hiệu suất, khả dụng, bảo mật, mở rộng, dễ sử dụng).



Hình 6: Minh họa ô tô chuyển phát nhanh không người lái của FedEx



## **IE108.O21.CNVN – Phân tích thiết kế phần mềm**

6. Quản trị viên, shipper cập nhật tình trạng đơn hàng cho khách hàng xem

7. Khách hàng nhận hàng thành công

### **- Alternative sequence**

1.1. Nếu thông tin đơn hàng không hợp lệ, quản trị viên gửi xác nhận hủy đơn hàng cho khách hàng, khách hàng đồng ý hủy hoặc cập nhật lại thông tin

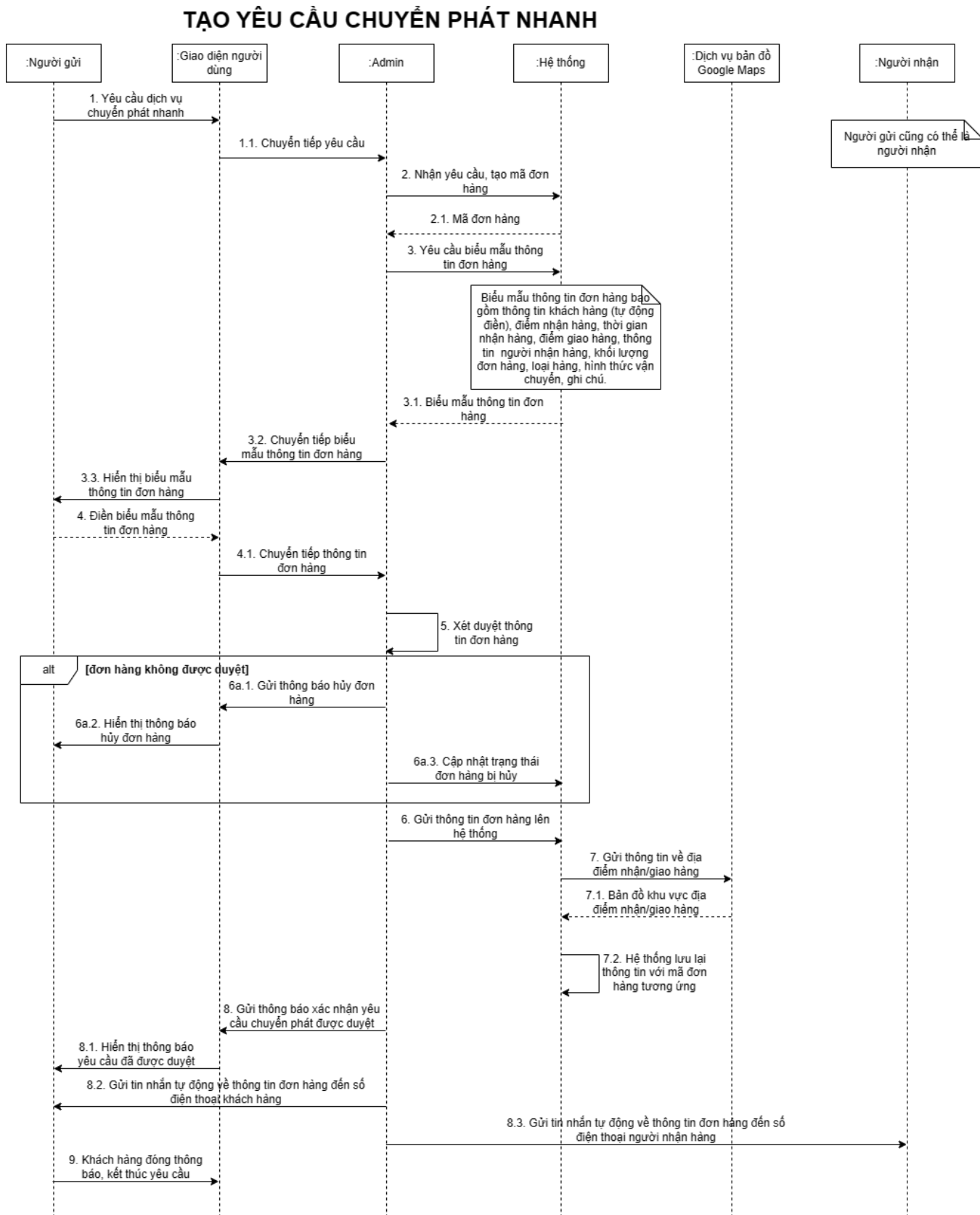
2.1. Nếu khách hàng lựa chọn phương tiện “Ô tô không người lái” thì quản trị viên sẽ liên hệ ngay với đối tác và gửi thông tin đơn hàng

4.1. Nếu thông tin đơn hàng không hợp lệ, đối tác gửi phản hồi cho quản trị viên xử lý

4.2. Quản trị viên gửi xác nhận hủy đơn hàng cho khách hàng, khách hàng đồng ý hủy hoặc cập nhật lại thông tin

7.1. Nếu khách hàng thanh toán tiền mặt thì quản trị viên cần chuyển khoản thanh toán cho đối tác

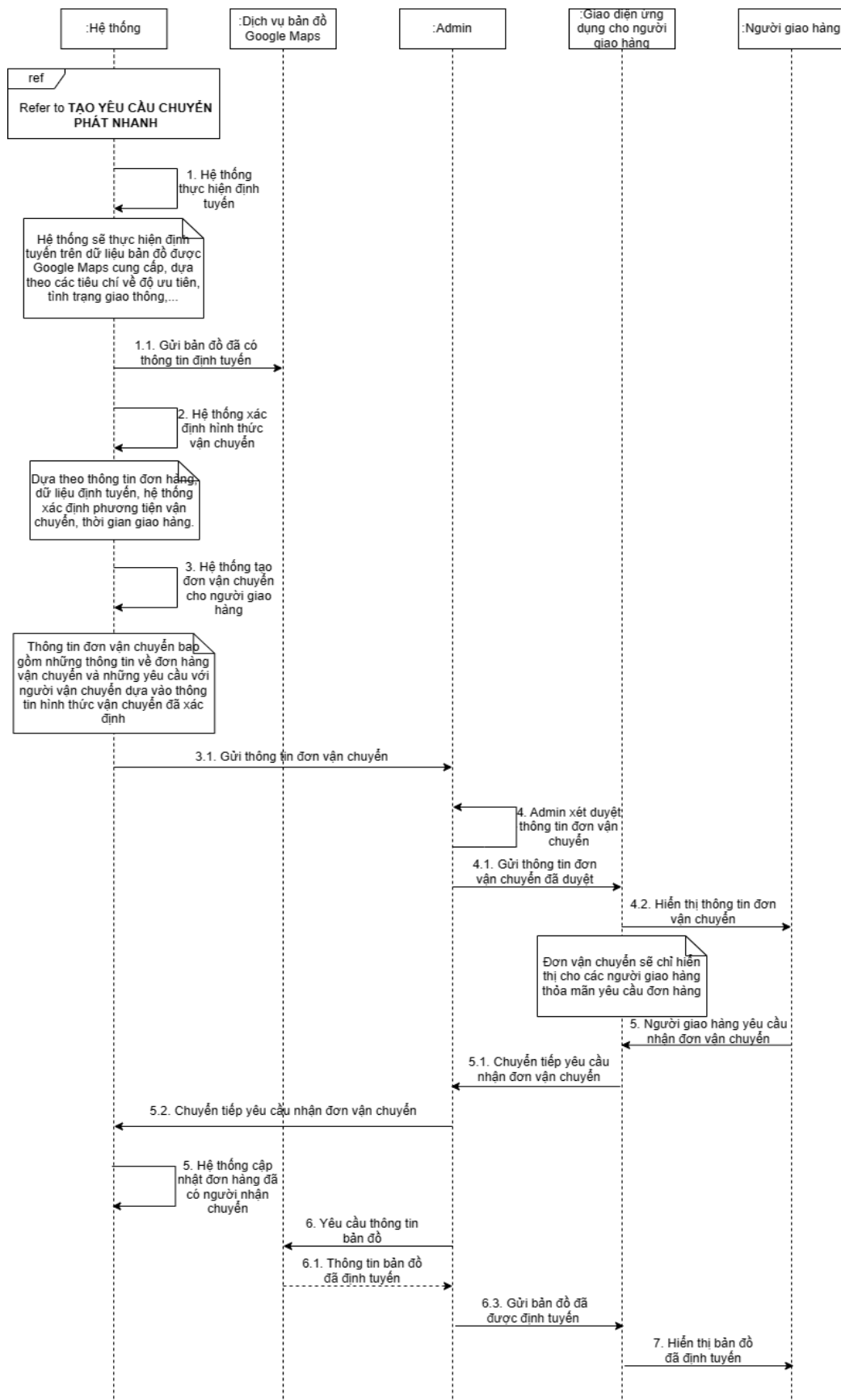
## 2.2. Biểu đồ sequence



Hình 8: Biểu đồ tuần tự của tạo đơn hàng

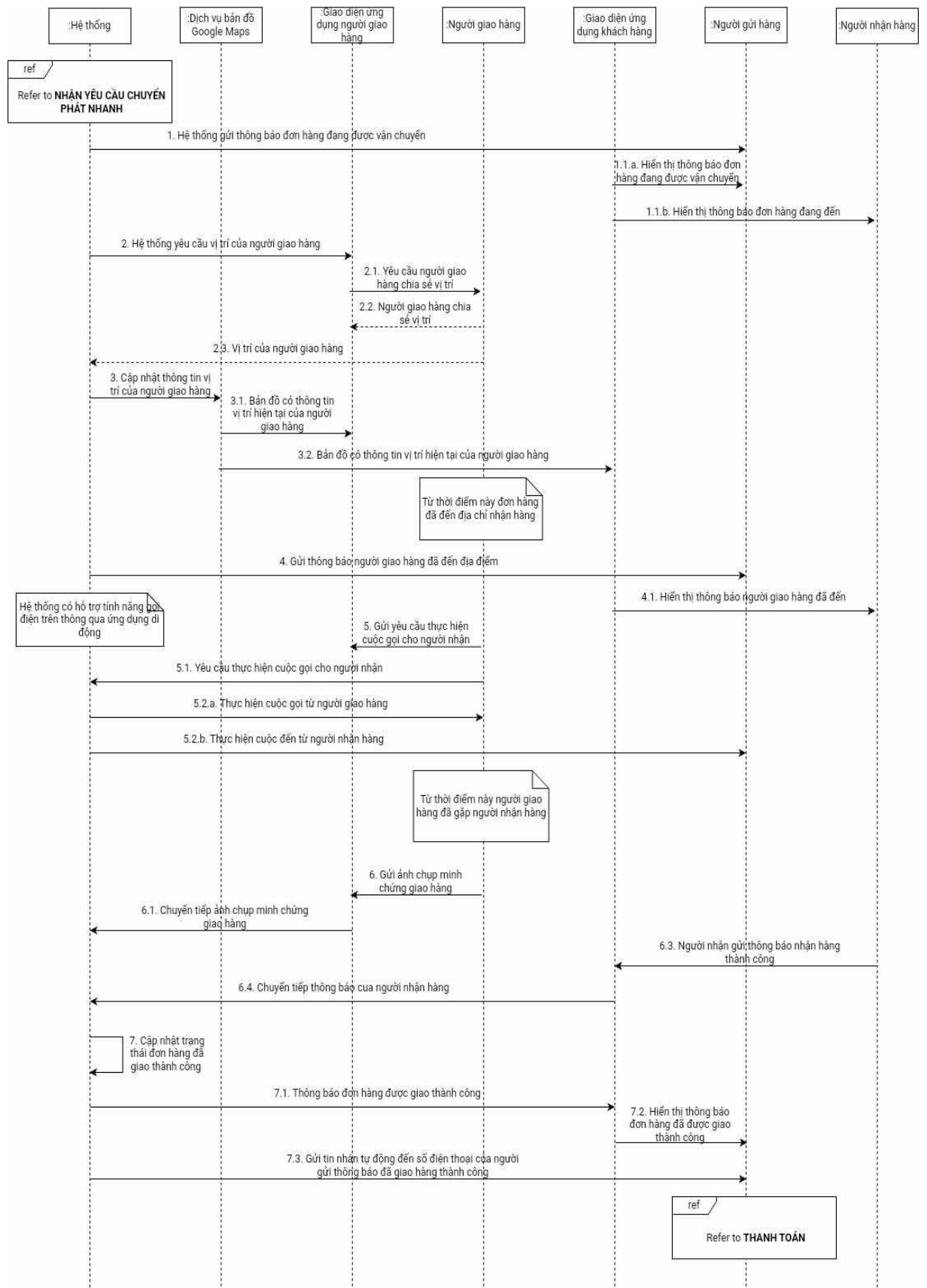


## NHẬN YÊU CẦU CHUYỂN PHÁT NHANH



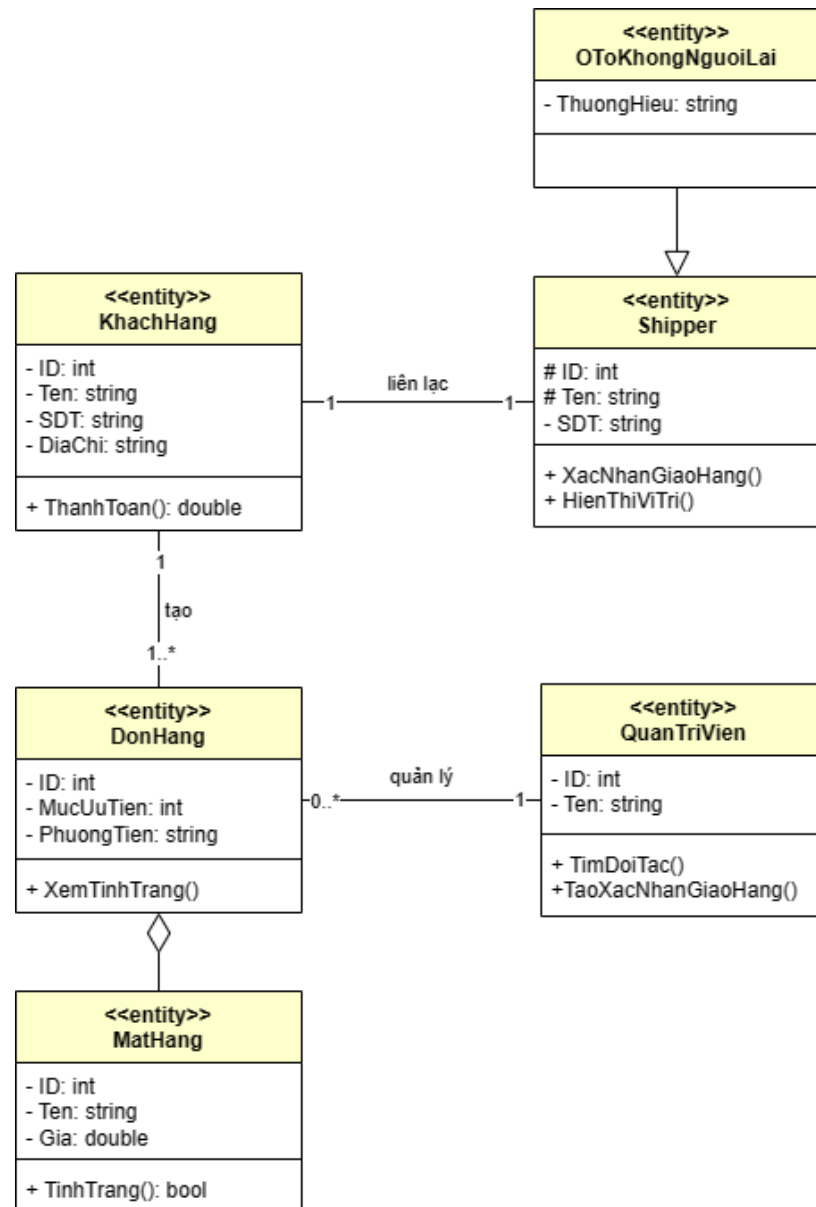
Hình 9: Biểu đồ tuần tự của xác nhận giao hàng

## IE108.O21.CNVN – Phân tích thiết kế phần mềm



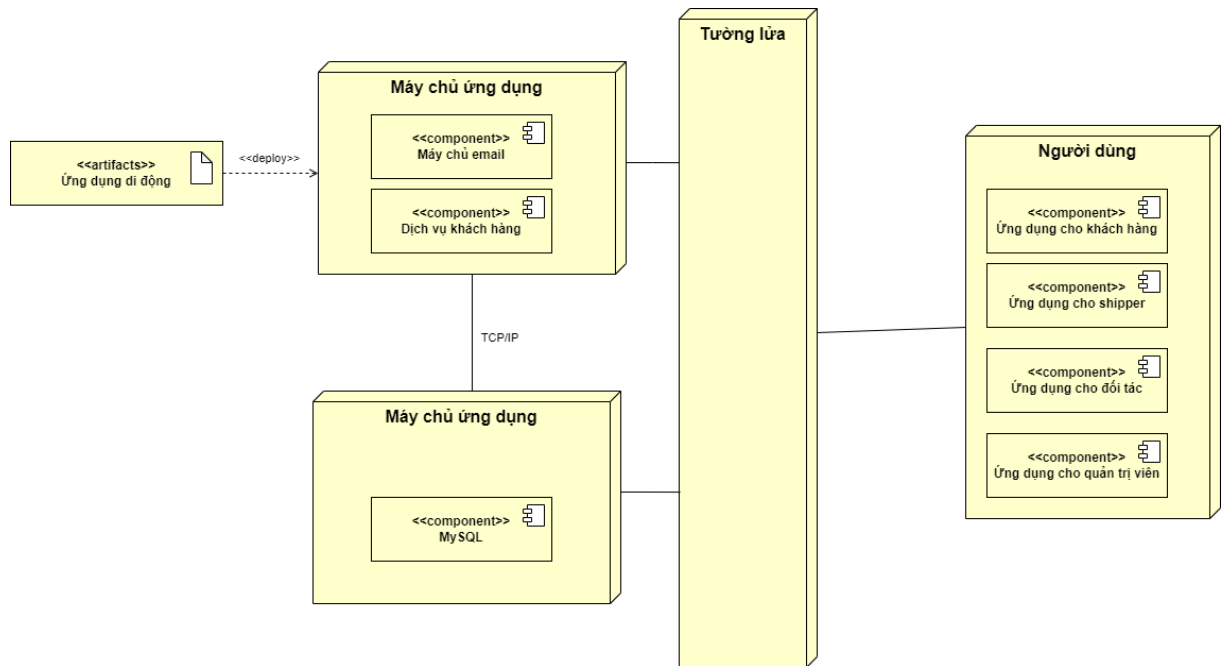
Hình 10: Biểu đồ tuần tự cho thông báo tình trạng đơn hàng

## 2.3. Biểu đồ class



Hình 11: Biểu đồ lớp cho hệ thống giao hàng

## 2.4. Biểu đồ triển khai (Deployment diagram)



Hình 12: Biểu đồ triển khai hệ thống trên điện thoại di động

## 3. Thiết kế hướng đối tượng

### 3.1. Thiết kế kiến trúc phần mềm

Kiến trúc phần mềm được sử dụng trong hệ thống giao hàng tự động: kiến trúc Hướng dịch vụ (SOA) và kiến trúc Vi dịch vụ (Microservices)

Chức năng cốt lõi của hệ thống giao hàng tự động:

- Quản lý quy trình giao hàng
- Điều khiển ô tô không người lái
- Hệ thống giám sát từ xa

Triển khai kiến trúc SOA:

- Phân tách các chức năng cốt lõi của hệ thống thành các dịch vụ độc lập như tạo đơn hàng, cập nhật tình trạng đơn hàng,...
- Mỗi dịch vụ được xây dựng giao diện thân thiện với người dùng và triển khai trên các máy chủ độc lập
- Các dịch vụ sẽ giao tiếp với nhau thông qua message hoặc giao thức HTTP

## **IE108.O21.CNVN – Phân tích thiết kế phần mềm**

- Quản trị viên sẽ giám sát các dịch vụ để theo dõi hiệu suất và khắc phục vấn đề một cách nhanh chóng

Triển khai kiến trúc Microservices:

- Phân tách các chức năng cốt lõi của hệ thống thành các dịch vụ nhỏ và cụ thể như thanh toán, hiển thị vị trí shipper,...
- Mỗi dịch vụ được xây dựng và triển khai riêng biệt trên các máy chủ độc lập
- Giảm thiểu phân bố ở các dịch vụ có nhu cầu thấp và tối đa phân bố ở các dịch vụ có nhu cầu cao giúp tăng khả năng xử lý lưu lượng truy cập
- Quản trị viên sẽ giám sát và quản lý hiệu suất của các dịch vụ microservices

### **3.2. Mô tả môi trường**

**Phần cứng:** điện thoại di động, máy chủ để lưu trữ máy chủ ứng dụng và cơ sở dữ liệu. Máy chủ ứng dụng bao gồm CPU 4-8 nhân, 8 GB RAM, 100 GB – 1 TB SSD. Máy chủ cơ sở dữ liệu bao gồm CPU 4-8 nhân, 8 GB RAM, 128 GB – 512 GB SSD. Các cảm biến như radar, lidar, camera, ultrasonic sensors, vận tốc giúp xe nhận biết môi trường xung quanh. Bộ điều khiển điện tử và cơ học giúp điều chỉnh hành vi của xe như tăng tốc, phanh và lái xe.

**Phần mềm:** Sử dụng các hệ điều hành như Windows Server, Linux, Android Automotive cho ô tô không người lái, hệ quản trị cơ sở dữ liệu MySQL, ngôn ngữ lập trình phát triển ứng dụng di động như Java (Android), Swift (iOS), ngôn ngữ lập trình phát triển ứng dụng máy chủ như Java, Node.js, Python, Ruby on Rails, API đối tác như RESTful API, phần mềm điều khiển ứng dụng trí tuệ nhân tạo, thị giác máy tính, hệ thống trợ lý ảo giúp giao tiếp với khách hàng, ứng dụng di động cho khách hàng, shipper, đối tác và quản trị viên.

**Mạng:** Kết nối internet bằng thông cao ít nhất 100 Mbps, mạng CAN (Controller Area Network) cho ô tô, công nghệ mạng 4G LTE, 5G, wifi, bluetooth.

### **3.3. Thiết kế cơ sở dữ liệu**

Cơ sở dữ liệu của hệ thống được tổ chức thành các bảng như sau:

- Bảng Khách hàng (Customers):
  - customer\_id (INT): Mã khách hàng (khóa chính)

## IE108.O21.CNVN – Phân tích thiết kế phần mềm

- name (VARCHAR(255)): Tên khách hàng
- gender (VARCHAR(10)): Giới tính (nam/nữ)
- photo (VARCHAR(255)): Ảnh đại diện của khách hàng
- phone (VARCHAR(20)): Số điện thoại được sử dụng làm tên đăng nhập
- email (VARCHAR(50)): Địa chỉ email
- identity\_number (VARCHAR(20)): Số CMND/CCCD
- address (VARCHAR(255)): Địa chỉ
- city (VARCHAR(50)): Thành phố
- state (VARCHAR(50)): Tỉnh/Thành phố
- country (VARCHAR(50)): Quốc gia
- postal\_code (VARCHAR(10)): Mã bưu điện
- **Bảng Thông tin tài khoản (Account\_Details):**
  - username (VARCHAR(20)): Số điện thoại được sử dụng làm tên đăng nhập (khóa ngoại tham chiếu customers.phone)
  - password (VARCHAR(255)): Mật khẩu đăng nhập (bắt buộc, được mã hóa)
  - account\_status (VARCHAR(50)): Trạng thái tài khoản (hoạt động, khóa, tạm ngưng) (mặc định: "hoạt động")
  - last\_login (DATETIME): Ngày đăng nhập gần nhất
- **Bảng Đơn hàng (Orders):**
  - order\_id (INT): Mã đơn hàng (khóa chính)
  - customer\_id (INT): Mã khách hàng (khóa ngoại tham chiếu customers.customer\_id)
  - order\_date (DATETIME): Ngày đặt hàng
  - delivery\_date (DATETIME): Ngày giao hàng dự kiến
  - status (VARCHAR(50)): Trạng thái đơn hàng (ví dụ: "Đã đặt hàng", "Đang xử lý", "Đang giao hàng", "Đã giao")
  - total\_amount (FLOAT (10,2)): Tổng số tiền thanh toán
- **Bảng Sản phẩm (Products):**
  - product\_id (INT): Mã sản phẩm (khóa chính)
  - product\_name (VARCHAR(255)): Tên sản phẩm
  - product\_description (TEXT): Mô tả sản phẩm
  - product\_price (FLOAT): Giá bán sản phẩm
  - weight (FLOAT): Trọng lượng sản phẩm (kg)
  - dimensions (VARCHAR(50)): Kích thước sản phẩm (Dài x Rộng x Cao)
- **Bảng Chi tiết đơn hàng (Order\_Details):**
  - order\_detail\_id (INT): Mã chi tiết đơn hàng (khóa chính)
  - order\_id (INT): Mã đơn hàng (khóa ngoại tham chiếu orders.order\_id)
  - product\_id (INT): Mã sản phẩm (khóa ngoại tham chiếu products.product\_id)
  - quantity (INT): Số lượng sản phẩm
  - unit\_price (FLOAT): Giá bán sản phẩm trong đơn hàng
- **Bảng Nhân viên giao hàng (Delivery\_Agents):**
  - agent\_id (INT): Mã nhân viên (khóa chính)
  - name (VARCHAR(255)): Tên nhân viên

## IE108.O21.CNVN – Phân tích thiết kế phần mềm

- photo (VARCHAR(255)): Ảnh đại diện của nhân viên
- gender (VARCHAR(10)): Giới tính (nam/nữ)
- phone (VARCHAR(20)): Số điện thoại
- email (VARCHAR(50)): Địa chỉ email
- date\_of\_birth (DATE): Ngày sinh
- identity\_number (VARCHAR(20)): Số CMND/CCCD
- driving\_license (VARCHAR(20)): Số bằng lái xe (nếu có)
- vehicle\_type (VARCHAR(50)): Loại phương tiện giao hàng (xe máy, ô tô, xe tải)
- vehicle\_license\_plate (VARCHAR(20)): Biển số xe
- delivery\_rating (FLOAT): Xếp hạng giao hàng (trung bình từ đánh giá của khách hàng)
- address (VARCHAR(255)): Địa chỉ
- city (VARCHAR(50)): Thành phố
- state (VARCHAR(50)): Tỉnh/Thành phố
- country (VARCHAR(50)): Quốc gia
- postal\_code (VARCHAR(10)): Mã bưu điện
- Bảng Giao hàng (Deliveries):
  - delivery\_id (INT): Mã giao hàng (khóa chính)
  - order\_id (INT): Mã đơn hàng (khóa ngoại tham chiếu orders.order\_id)
  - agent\_id (INT): Mã nhân viên giao hàng (khóa ngoại tham chiếu delivery\_agents.agent\_id)
  - delivery\_date (DATETIME): Ngày giao hàng thực tế
  - delivery\_status (VARCHAR(50)): Trạng thái giao hàng (ví dụ: "Đang giao", "Đã giao hàng", "Giao hàng thất bại")
- Bảng Bản đồ khu vực giao hàng (Delivery\_Zones):
  - zone\_id (INT): Mã khu vực (khóa chính)
  - zone\_name (VARCHAR(50)): Tên khu vực
  - city (VARCHAR(50)): Thành phố
  - state (VARCHAR(50)): Tỉnh/Thành phố
  - country (VARCHAR(50)): Quốc gia
  - postal\_code\_from (VARCHAR(10)): Mã bưu điện bắt đầu
  - postal\_code\_to (VARCHAR(10)): Mã bưu điện kết thúc
- Bảng Chi phí vận chuyển (Shipping\_Costs):
  - shipping\_cost\_id (INT): Mã chi phí vận chuyển (khóa chính)
  - delivery\_zone\_id (INT): Mã khu vực giao hàng (khóa ngoại tham chiếu delivery\_zones.zone\_id)
  - order\_weight (FLOAT): Cân nặng đơn hàng (kg)
  - shipping\_fee (FLOAT): Phí vận chuyển
- Bảng Phương thức thanh toán (Payment\_Methods):
  - payment\_method\_id (INT): Mã phương thức thanh toán (khóa chính)
  - payment\_method\_name (VARCHAR(50)): Tên phương thức thanh toán (ví dụ: Tiền mặt, Thanh toán qua thẻ, Chuyển khoản ngân hàng)
- Bảng Lịch sử giao hàng (Delivery\_History):
  - history\_id (INT): Mã lịch sử (khóa chính)

## IE108.O21.CNVN – Phân tích thiết kế phần mềm

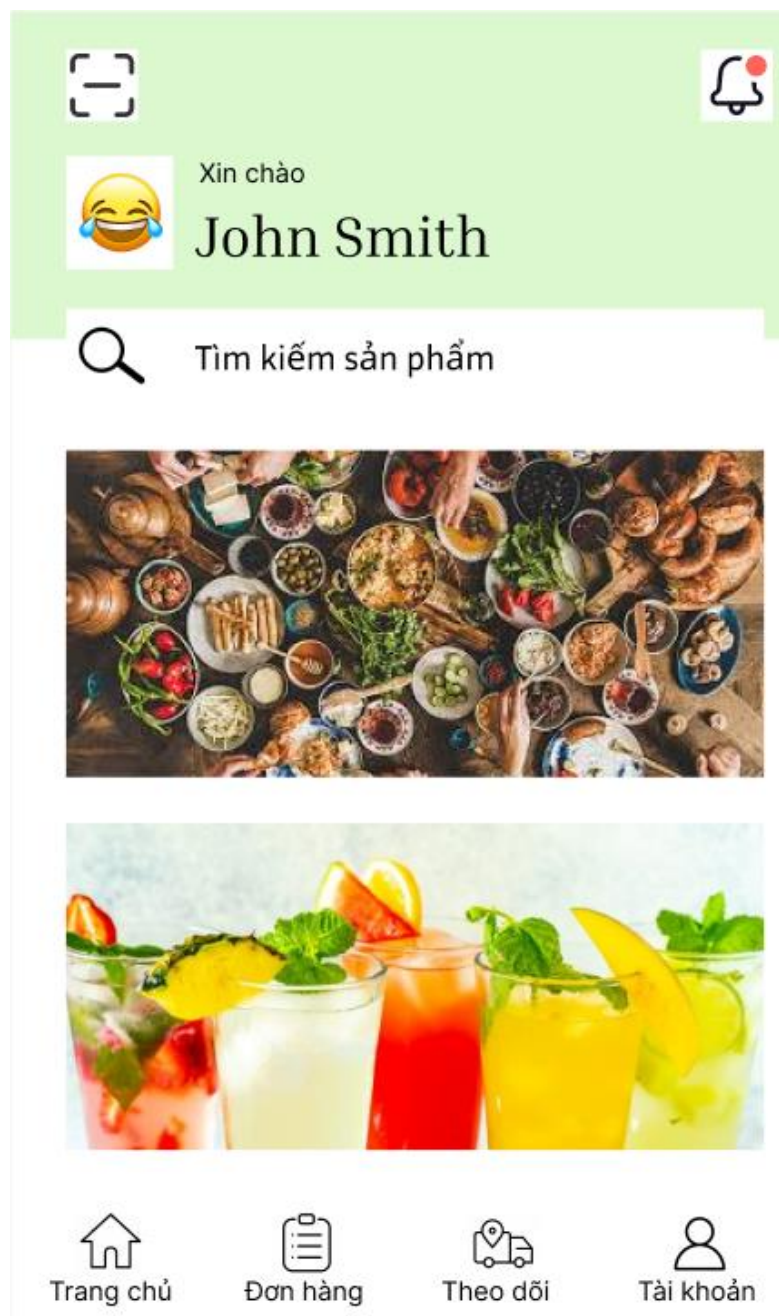
- order\_id (INT): Mã đơn hàng (khóa ngoại tham chiếu orders.order\_id)
- delivery\_status (VARCHAR(50)): Trạng thái giao hàng trước đây
- updated\_date (DATETIME): Ngày cập nhật trạng thái
- Bảng Đánh giá khách hàng (Customer\_Reviews):
  - review\_id (INT): Mã đánh giá (khóa chính)
  - order\_id (INT): Mã đơn hàng (khóa ngoại tham chiếu orders.order\_id)
  - customer\_rating (INT): Đánh giá của khách hàng (1-5 sao)
  - customer\_comment (VARCHAR(500)): Bình luận của khách hàng
- Bảng Quản lý tài khoản quản trị viên (Admins):
  - admin\_id (INT): Mã quản trị viên (khóa chính)
  - name (VARCHAR(255)): Tên quản trị viên
  - email (VARCHAR(50)): Địa chỉ email
  - password (VARCHAR(255)): Mật khẩu
  - role (VARCHAR(50)): Quyền hạn quản trị viên (ví dụ: "Quản trị viên hệ thống", "Quản trị viên bán hàng")



Hình 13: CSDL cho hệ thống giao hàng



3.4. Thiết kế giao diện người dùng



Hình 14. Giao diện hệ thống giao hàng trên điện thoại di động

## **CHƯƠNG 4: KẾT LUẬN**

Trong đồ án này, nhóm em đã làm rõ các bước phân tích, thiết kế hướng đối tượng bằng UML 2.0 và áp dụng chúng vào bài toán hệ thống giao hàng nhanh. Từ việc phân tích các yêu cầu chức năng, phi chức năng của hệ thống để tạo ra các đối tượng và biểu diễn các đối tượng này qua các mô hình trực quan giúp người đọc hiểu rõ hơn về hệ thống. Tổng quan, đồ án đã đạt được mục tiêu của việc phân tích, thiết kế một hệ thống giao hàng tự động và mở ra tiềm năng phát triển và ứng dụng rộng rãi trong thực tế

## **TÀI LIỆU THAM KHẢO**

- [1] iViettech Education, “Cơ bản về phân tích và thiết kế hướng đối tượng và UML”, iViettech. Truy cập: 8 Tháng Năm 2024. [Online]. Available at: <https://iviettech.vn/blog/302-oaduml.html>
- [2] Dao A., “Phân tích và thiết kế hướng đối tượng (OOAD)”, Học Java. Truy cập: 8 Tháng Năm 2024. [Online]. Available at: <https://hocjava.com/oad/>
- [3] “[PDF]Phân Tích Thiết Kế Hướng Đối Tượng - Đh Công Nghệ Thông Tin - Nguyễn Đình Loan Phương - Bài Giảng Phân Tích Thiết Kế Hướng Đối Tượng Uit.Pdf”. Truy cập: 8 Tháng Năm 2024. [Online]. Available at: <https://cuuduongthancong.com/pvf/5869138/phan-tich-thiet-ke-huong-doi-tuong/nguyen-dinh-loan-phuong/bai-giang-phan-tich-thiet-ke-huong-doi-tuong-uit.pdf?src=afile&action=hover>