**Link: https://github.com/PeachhJelly/CNPM**

**Chủ đề 2:xây dựng ứng dụng backend 3 lớp**

**Problem Alignment**

* Thời gian giao đồ ăn truyền thống bị chậm và biến động vì kẹt xe, giờ cao điểm, làm giảm trải nghiệm khách hàng,
* Khó dự đoán và cân bằng nguồn lực shipper theo mùa vụ → phát sinh chi phí nhàn rỗi hoặc thiếu hụt đột xuất.
* Thiếu khả năng giao nhanh các mặt hàng cần gấp (ví dụ: thực phẩm khẩn, vật dụng y tế nhẹ) trong khoảng thời gian rất ngắn.
* Khách hàng ở khu vực ngoại ô / vùng khó tiếp cận ít được phục vụ hoặc phải chờ lâu do hạn chế shipper
* Doanh nghiệp muốn mở rộng khả năng tiếp cận và tăng tần suất tương tác với khách hàng

**High-Level Approach**

* Xây dựng ứng dụng web và di động cho phép khách hàng đặt và thanh toán món ăn trực tuyến.
* Tự động hóa toàn bộ quy trình giao hàng bằng drone thông minh, không cần tài xế trung gian.
* Tích hợp hệ thống định vị GPS để tính toán đường bay, giám sát drone theo thời gian thực và xác định vị trí giao hàng chính xác.
* Kết nối với cổng thanh toán điện tử (Momo, ZaloPay, Visa,...) để hỗ trợ thanh toán trả trước.
* Phát triển trung tâm điều phối drone có khả năng phân công, giám sát và điều hướng tự động.

**Narrative**

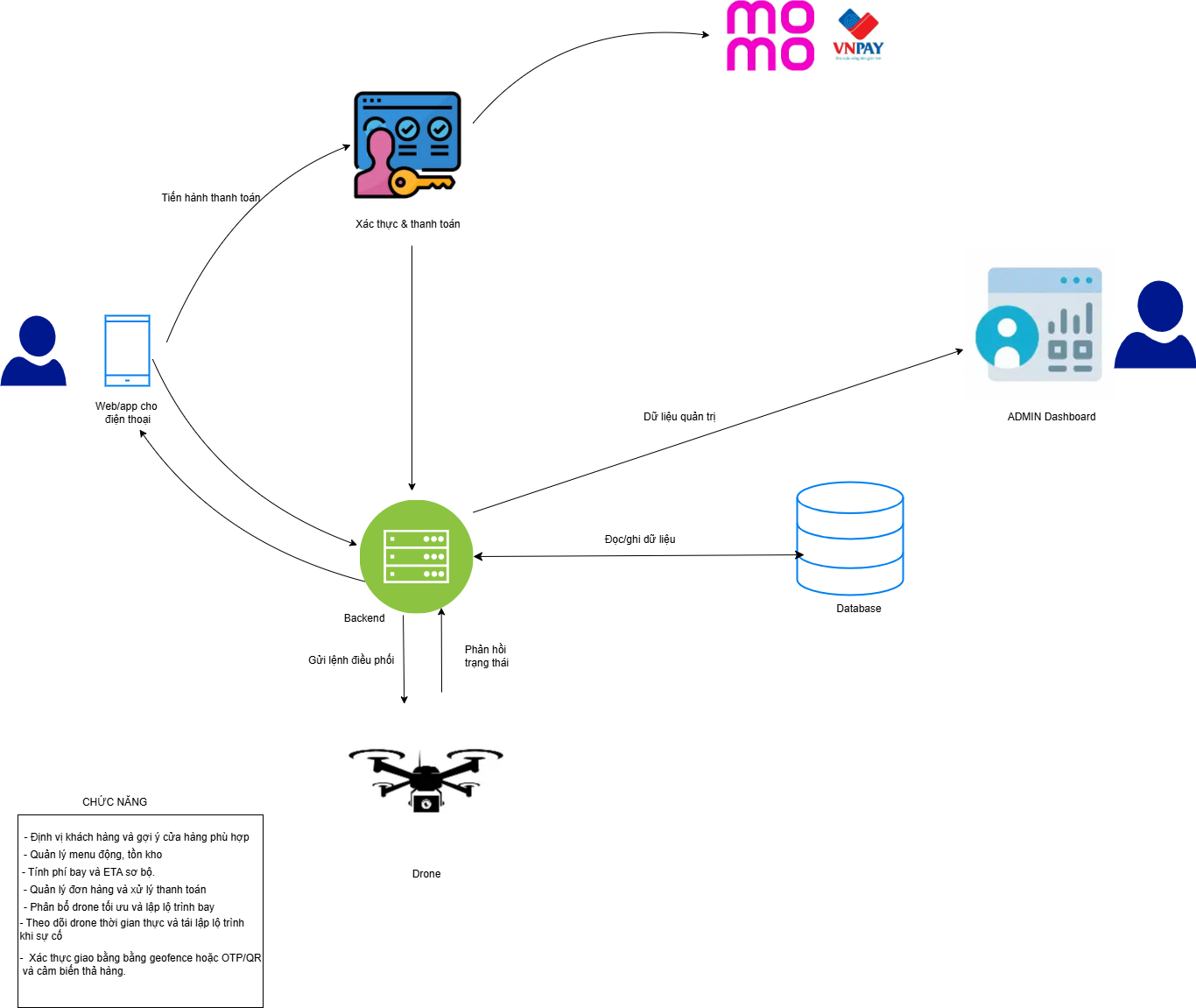
* Người dùng mở ứng dụng FastFood Delivery Drone, đăng nhập hoặc đăng ký tài khoản.
* Hệ thống xác định vị trí hiện tại và hiển thị danh sách cửa hàng gần nhất.
* Người dùng tìm kiếm món ăn, thêm vào giỏ hàng và tiến hành thanh toán trực tuyến.
* Sau khi thanh toán thành công, hệ thống gửi đơn đến cửa hàng tương ứng.
* Cửa hàng xác nhận đơn, chế biến món và bàn giao cho drone tại khu vực drone pad.
* Drone tự động nhận hàng, bay theo lộ trình tối ưu đến vị trí của khách.
* Khách hàng theo dõi chuyến bay theo thời gian thực, nhận thông báo khi drone sắp đến.
* Khách quét mã QR hoặc nhập OTP để mở khoang nhận hàng.
* Sau khi giao hàng hoàn tất, drone quay về trạm sạc và hệ thống cập nhật trạng thái đơn hàng.
* Người dùng có thể đánh giá đơn hàng và trải nghiệm giao hàng.

**Goals**

* Rút ngắn đáng kể thời gian giao bằng tuyến bay trực tiếp của drone so với giao thông đường bộ truyền thống.
* Đảm bảo chất lượng món (độ nóng, độ tươi, tính nguyên vẹn) bằng cách giảm thời gian lưu chuyển và rung lắc.
* Tự động hoá toàn bộ chu trình sau thanh toán: lựa chọn drone phù hợp, lập kế hoạch bay (flight plan), giám sát và cập nhật ETA theo thời gian thực.
* Cung cấp trải nghiệm tracking trực quan: bản đồ trực tiếp, trạng thái chuyến bay, cảnh báo khi drone sắp đến hoặc có điều chỉnh lộ trình.

A diagram of a delivery service

AI-generated content may be incorrect.



Mobile App: Dành cho khách hàng (đặt hàng, theo dõi đơn, nhận thông báo).

Web Admin: Dành cho quản trị viên (theo dõi, điều phối drone, báo cáo).

Backend Server:

* API Gateway quản lý giao tiếp giữa các microservice.
* Các service chính:
  + User Service: Quản lý thông tin người dùng.
  + Product/Restaurant Service: Quản lý thực đơn và cửa hàng.
  + Order Service: Quản lý đơn hàng và trạng thái.
  + Payment Service: Quản lý thanh toán (qua ví điện tử hoặc thẻ).
  + Drone Control Service: Kết nối và điều phối hệ thống drone.
* CSDL: PostgreSQL (dữ liệu giao dịch), MongoDB (dữ liệu drone và vị trí).

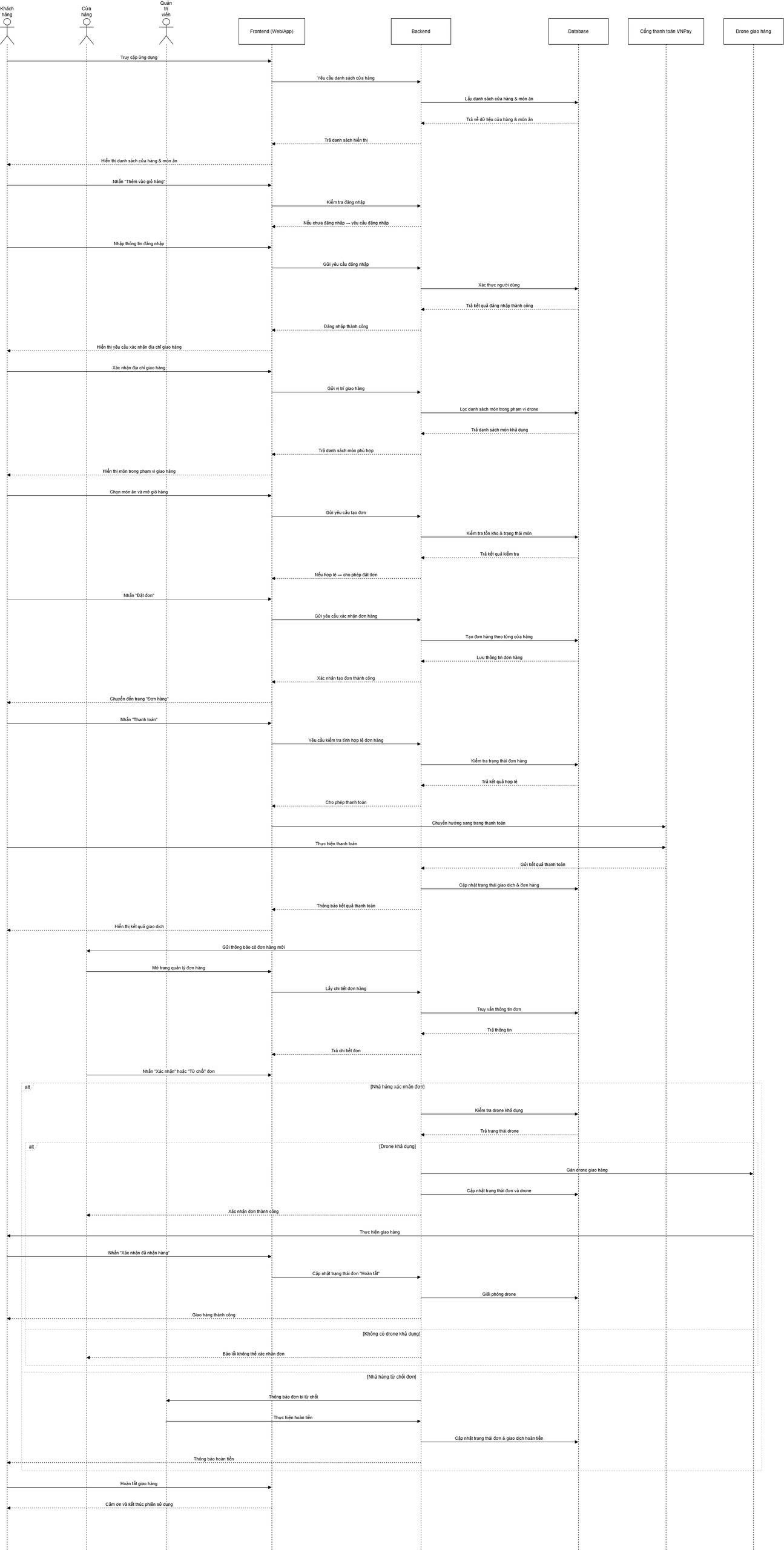
Drone Delivery System:

* Drone Control Station: Điều phối, giám sát và giao tiếp với drone.
* Drone: Thiết bị bay thực hiện giao hàng.

**Key Features**

* + **Location & Store Discovery**: Định vị GPS khách hàng, gợi ý cửa hàng trong hành lang bay an toàn, lọc theo thời tiết/tải trọng/khoảng cách.
  + **Menu & Cart Management**: Hiển thị menu động, kiểm tra tồn kho, tính toán trọng lượng & gợi ý tách đơn nếu vượt tải.
  + **Pricing & ETA Estimation**: Tính phí bay theo quãng đường/năng lượng, hiển thị tiết kiệm CO₂, cung cấp ETA sơ bộ.
  + **Order & Payment Processing**: Quản lý trạng thái đơn hàng, tích hợp nhiều cổng thanh toán, đảm bảo tính nhất quán giao dịch.
  + **Drone Dispatch & Flight Planning**: Chọn drone tối ưu, sinh lộ trình bay chi tiết, kiểm tra thời tiết & vùng cấm bay.
  + **Real-time Tracking & Re-planning**: Cập nhật telemetry & ETA động, hiển thị drone trên bản đồ, tái lập lộ trình hoặc fallback khi sự cố.
  + **Delivery Confirmation**: Xác thực giao bằng geofence hoặc OTP/QR, xác nhận thả hàng bằng cảm biến.





**Key Logic**

**Order & Menu Retrieval**

* Ứng dụng định vị GPS khách hàng, tải danh sách cửa hàng nằm trong hành lang bay an toàn.
* Menu và thông tin kho (inventory, trọng lượng khả dụng) được tải ngay sau khi khách chọn cửa hàng.
* Toàn bộ dữ liệu menu được hiển thị cục bộ trên frontend trong phiên đặt hàng, hạn chế request liên tục.

### ****Cart & Payload Validation****

* Khi khách thêm món vào giỏ, hệ thống tự động tính trọng lượng và thể tích.
* Nếu vượt khả năng tải drone, ứng dụng gợi ý tách đơn hoặc điều chỉnh.
* ETA sơ bộ được hiển thị dựa trên khoảng cách bay + thời gian chuẩn bị bếp + thời gian nạp drone.

### ****Payment & Order Confirmation****

* Order Service tạo đơn hàng với trạng thái **CREATED**.
* Payment gọi sang cổng thanh toán (MoMo/VNPay/thẻ) → nhận callback → xác nhận **PAID**.
* Sau khi thanh toán thành công, đơn chuyển trạng thái **PAID**, phát sự kiện **ORDER\_PAID** cho hệ thống dispatch

### ****Drone Dispatch & Flight Planning****

* chọn drone khả dụng (đủ pin, vị trí gần cửa hàng, không bảo trì).
* Flight Plan Service sinh lộ trình (waypoints: TAKEOFF → CRUISE → DROP → RETURN).
* Kế hoạch bay kiểm tra chéo với thời tiết, vùng cấm bay, giới hạn tải.
* Đơn cập nhật trạng thái **READY\_TO\_DISPATCH**

### ****Launch & Real-time Tracking****

* Khi món đã sẵn sàng, drone nhận hàng và cất cánh → trạng thái **LAUNCHED**.
* Drone phát telemetry định kỳ (lat/lon/alt/battery).
* Tracking Service tính ETA động và đẩy về app qua WebSocket.

### ****Delivery & Confirmation****

* Khi drone tới vùng giao, hệ thống xác thực:
  + Geofence match (GPS trong vùng thả) hoặc
  + OTP/QR do khách xác nhận.
* Drone thả hàng, cảm biến xác nhận gói đã được thả.
* Order Service chuyển trạng thái **DELIVERED**.

**Refund Handling**

* Nếu cửa hàng từ chối đơn → hệ thống tự động gửi yêu cầu hoàn tiền về Payment Gateway và cập nhật trạng thái.

BACKEND DESIGN – Drone Food Delivery System

**Sub-system**

* **Web/Mobile App Backend**
  + REST API cho ứng dụng web/mobile.
  + Xử lý logic đặt hàng, quản lý menu, giỏ hàng, thanh toán, trạng thái đơn hàng.
* **Drone Dispatch & Flight Control Service**
  + Quản lý drone: lựa chọn drone khả dụng, lập kế hoạch bay, cập nhật vị trí và ETA thời gian thực.
  + Kiểm tra điều kiện thời tiết, vùng cấm bay, payload tối đa.
* **Order Service**
* Quản lý toàn bộ vòng đời đơn hàng từ khi tạo, xác nhận, thanh toán đến hoàn tất giao hàng.
* **Store Service**
  + - Quản lý cửa hàng, menu, giá món, và khu vực phục vụ.
* **Notification service**
  + Gửi push notification cho khách hàng khi trạng thái đơn thay đổi.
* **Payment Service**
  + Kết nối và xử lý giao dịch với cổng thanh toán (Momo, ZaloPay, Visa,...).

## **App Backend**

* **Framework & Ngôn ngữ:** Spring Boot 3 (Java 21)
* **API Design:** RESTful API
* **Authentication & Authorization:** Spring Security + JWT
* **Caching:** Redis (LRU cho dữ liệu menu, store list, cart tạm thời)
* **Containerization:** Docker để deploy dịch vụ độc lập

## **Drone Dispatch & Flight Planning**

* **Input:** đơn đã thanh toán, vị trí cửa hàng, payload, drone khả dụng.
* **Output:** flight plan + ETA.
* **Logic:**
  1. Chọn drone dựa trên vị trí và pin.
  2. Tính ETA = thời gian bay trực tiếp + chuẩn bị bếp + nạp drone.
  3. Tối ưu tuyến bay thẳng nếu khả thi, tránh vùng cấm và gió mạnh.
  4. Cập nhật ETA thời gian thực khi drone đang bay.

## **Data**

* **Database:** MySQL Server (quan hệ)

DFD mức 0

**A white circle with black text

AI-generated content may be incorrect.**

DFD mức 1

**A black and white diagram

AI-generated content may be incorrect.**

State diagram

Order

A screenshot of a chat

AI-generated content may be incorrect.

Delivery

A black background with white rectangles

AI-generated content may be incorrect.

Sổ cái

A screenshot of a cell phone

AI-generated content may be incorrect.

Payment

A black background with white rectangles and black text

AI-generated content may be incorrect.

Drone

A black background with white text

AI-generated content may be incorrect.

A diagram of a network

AI-generated content may be incorrect.

|  |  |  |
| --- | --- | --- |
| A white square with black text and a black arrow  AI-generated content may be incorrect. | | |
| **Use Case Number:** | UC1 | |
| **Use Case Name:** | Thêm hàng vào giỏ hàng | |
| **Actor (s):** | Khách hàng | |
| **Maturity:** | Focused | |
| **Summary:** | Khách hàng chọn món ăn từ ứng dụng FastFood Delivery Drone, thêm vào giỏ hàng | |
| **Basic Course of Events:** luồng nghiệp vụ chính (đăng kí học phần) | **Actor Action** | **System Response** |
| 1. Khách hàng **truy cập vào trang chủ của ứng dụng. A1** |  |
|  | 2. Hệ thống lấy vị trí người dùng và hiển thị danh sách món ăn của nhà hàng nằm trong phạm vi giao |
| 3. Người dùng chọn sản phẩm muốn thêm. A2,E1,E2 |  |
|  | 4.Hệ thống lấy thông tin xác thực và câp nhật giỏ hàng |
| 5. Kết thúc |  |
| **Alternative Paths** | A1. Khách hàng chưa đăng nhập   1. Hệ thống hiển thị danh sách toàn bộ sản phẩm 2. Quay lại bước 3 trong luồng chính   A2. Người dùng nhấn thêm vào giỏ hàng khi chưa đăng nhập   1. Hệ thống yêu cầu người dùng đăng nhập 2. Người dùng đăng nhập và cập nhật tọa độ vào hệ thống 3. Quay lại bước 2 trong luồng chính   A3. Người dùng nhập sản phẩm muốn tìm   1. Người dùng nhập sản phẩm muốn thêm và bấm tìm kiếm 2. Hệ thống hiển thị kết quả tìm kiếm 3. Quay lại bước 3 trong luồng chính | |
| **Exception Paths:** luồng vi phạm (kiểm tra điều kiện) | E1. Số lượng món ăn không đủ, ,hoặc món ăn đã inactive  Hệ thống thông báo món ăn không hợp lệ và quay lại bước 3 của luông chính  E2. Món ăn thuộc nhà hàng nằm ngoài phạm vi giao  Hệ thống thông báo món ăn không nằm trong phạm vi giao quay lại bước 3 của luồng chính | |
| **Assumptions:** | Khách hàng ở vị trí nằm trong phạm vi giao của ít nhất 1 cửa hàng | |
| **Preconditions:** | Khách hàng đã có tài khoản với quyền khách hàng và đã đăng nhập | |
| **Post Conditions:** | Giỏ hàng cập nhật thành công | |
| **Activity Diagram:**  **A diagram of a flowchart  AI-generated content may be incorrect.** | | |
| **Sequence:A white sheet of paper with black and yellow lines  AI-generated content may be incorrect.** | | |
| **Class Diagram:**  **A diagram of a company  AI-generated content may be incorrect.** | | |

|  |  |  |
| --- | --- | --- |
| A diagram of a diagram  AI-generated content may be incorrect. | | |
| **Use Case Number:** | UC2 | |
| **Use Case Name:** | Đặt hàng | |
| **Actor (s):** | Khách hàng | |
| **Maturity:** | Focused | |
| **Summary:** | Khách hàng cập nhật giỏ hàng thành đơn hàng | |
| **Basic Course of Events:** luồng nghiệp vụ chính (đăng kí học phần) | **Actor Action** | **System Response** |
| 1. Khách hàng đã **thêm hàng vào giỏ** |  |
| 2.Khách hàng chọn giỏ hàng |  |
|  | 3. Hệ thống hiển thị trang giỏ hàng với các sản phẩm đã thêm |
| 4.Khách hàng chọn đặt hàng, A1,A2,E1 |  |
|  | 5. Hệ thống tạo các đơn hàng theo nhà hàng, cập nhật trạng thái và lưu. E2,E3 |
| . | 6.Hệ thống thông báo đặt hàng thành công. |
| **7. Kết thúc sử dụng tại đây** |  |
|  | A1. Sửa/Xóa số lượng hàng trong giỏ   1. Khách hàng sửa số lượng sản phẩm trong giỏ hàng. 2. Hệ thống cập nhật lại theo số lượng đơn hàng được sửa/xóa 3. Quay lại bước 3 của luồng chính | |
| A2. Người dùng chọn tiếp tục mua sắm  Quay lại bước 1 của luồng chính | |
| **Exception Paths:** luồng vi phạm (kiểm tra điều kiện) | E1. Số lượng sản phẩm không hơp lệ hoặc sản phẩm không còn được bán   1. Hệ thống thông báo sản phẩm lỗi hoặc hết, yêu cầu khách hàng xóa sản phẩm lỗi hoặc giảm số lượng sản phẩm 2. Quay lại bước 3 của luồng chính   E2. Đơn hàng của nhà hàng đang nằm ngoài phạm vi giao   1. Hệ thống báo đơn hàng của nhà hàng nằm ngoài phạm vi giao và không thể đặt đơn 2. Quay lại bước 3 của luồng chính   E3. Giỏ hàng rỗng   1. Hệ thống thông báo giỏ hàng rỗng, quay lại bước 3 của luồng chính | |
| **Assumptions:** | Khối lượng, kích thước đơn hàng hợp lệ | |
| **Preconditions:** | Đã thực hiện chức năng thêm hàng vào giỏ | |
| **Post Conditions:** | Tạo thành các đơn hàng theo từng cửa hàng | |
| **Activity Diagram:**  A diagram of a work flow  AI-generated content may be incorrect. | | |
| **Sequence diagram:**  **A white paper with black and yellow lines  AI-generated content may be incorrect.** | | |
| **Class Diagram: A screenshot of a diagram  AI-generated content may be incorrect.** | | |

|  |  |  |
| --- | --- | --- |
| A diagram of a diagram  AI-generated content may be incorrect. | | |
| **Use Case Number:** | UC3 | |
| **Use Case Name:** | Thanh toán | |
| **Actor (s):** | *Khách hàng* | |
| **Maturity:** | Focused | |
| **Summary:** | Khách hàng nhận được thông báo thanh toán thành công hoặc thất bại, hệ thống cập nhật thành công trạng thái của đơn hàng và trạng thái giao dịch | |
| **Basic Course of Events:** luồng nghiệp vụ chính (đăng kí học phần) | Actor Action | System Respond |
| **1.Khách hàng đã đặt hàng và vào trang đơn hàng** |  |
|  | 2 .Hệ thống hiển thị danh sách đơn hàng |
| 3 .Khách hàng chọn đơn hàng muốn thanh toán |  |
|  | 4 .Hệ thống hiển thị chi tiết đơn hàng.E1 |
| 5.Người dùng nhấn nút thanh toán đơn hàng |  |
|  | 6. Hệ thống điều hướng người dùng tới bên thanh toán thứ 3 |
| **7. Người dùng thực hiện thanh toán với bên thứ 3 và gửi kết quả về hệ thống** |  |
|  | 8. Hệ thống nhận kết quả thanh toán và cập nhật trạng thái đơn hàng và trạng thái thanh toán |
|  | 9. Hệ thống thông báo kết quả thanh toán cho người dùng |
| **Exception Paths:** luồng vi phạm (kiểm tra điều kiện) | E1. Đơn hàng không thể thanh toán  Đơn hàng đã bị nhà hàng cancel hoặc đã thanh toán thì không thể thực hiện thanh toán, hệ thống sẽ báo đơn hàng không hợp lệ và quay lại bước 2 của luồng chính | |
| **Preconditions:** | * Có đơn hàng | |
| **Post Conditions:** | * Cập nhật trạng thái giao dịch , trạng thái đơn hàng và thông báo kết quả thanh toán cho người dùng | |
| **Activity Diagram:**  A diagram of a flowchart  AI-generated content may be incorrect. | | |
| **Sequence:**  **A screenshot of a computer  AI-generated content may be incorrect.** | | |
| **Class diagram:**  **A screenshot of a diagram  AI-generated content may be incorrect.** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| A diagram of a diagram  AI-generated content may be incorrect. | | | |
| **Use Case Number:** | UC4 | | |
| **Use Case Name:** | Quản lý đơn hàng | | |
| **Actor (s):** | *Admin, Cửa hàng, Khách hàng* | | |
| **Maturity:** | Focused | | |
| **Summary:** | cập nhật thông tin người dùng trong hệ thống | | |
| **Basic Course of Events:** luồng nghiệp vụ chính (đăng kí học phần) | **Actor Action** | | **System Response** |
| 1. Khách hàng đăng nhập và vào trang đơn hàng,A1,A2 | |  |
|  | | 2. Hệ thống hiển thị danh sách đơn hàng của khách hàng |
| 3. Khách hàng chọn xem chi tiết đơn hàng.A3 | |  |
|  | | 4. Hệ thống hiển thị chi tiết đơn hàng |
| 5.Kết thúc sử dụng tại đây | |  |
|  | | |
| **Alternative Paths** | A1. Người đăng nhập là admin | | |
| **Actor Action** | **System Response** | |
| 1.Admin đăng nhập và truy cập quản lý đơn hàng |  | |
|  | 2. Hệ thống hiển thị danh sách các đơn hàng | |
| 3. Admin cập nhật trạng thái hoàn tiền cho đơn.A4 |  | |
|  | 4. Hệ thống cập nhật trạng thái đơn và thông báo, quay lại bước 5 của luồng chính. E1 | |
| A2. Người đăng nhập là cửa hàng   1. Cửa hàng vào trang đơn hàng 2. Hệ thống hiển thị đơn hàng của cửa hàng 3. Cửa hàng nhấn vào xem chi tiết đơn 4. Hệ thống hiển thị chi tiết đơn 5. Kết thúc sử dụng tại đây. A5   A3. Khách hàng chọn hủy đơn   1. Hệ thống cập nhật trạng thái và thông báo, kết thúc dử dụng tại đây.E2   A4.Admin lọc đơn theo nhà hàng   1. Hệ thống lọc đơn theo nhà hàng admin đã chọn, kết thúc sử dụng tại đây   A5 Nhà hàng xác nhận hoặc không nhận đơn   1. Nếu nhà hàng không xác nhận đơn thì gửi yêu cầu hoàn tiền cập nhật trạng thái đơn hàng và kết thúc sử dụng tại đây 2. Nếu nhà hàng xác nhận đơn thì kiểm tra xem đơn hàng có hợp lệ để xác nhận không nếu hợp lệ thì cập nhật đơn và kết thúc sử dụng nếu không hệ thống thông báo không thể chấp nhận đơn và hiển thị lại danh sách đơn hàng | | |
| **Exception Paths:** luồng vi phạm (kiểm tra điều kiện) | E1. Đơn không đủ điều kiện hoàn tiền  Hệ thống thông báo đơn không đủ điều kiện để hoàn tiền và hiển thị lại danh sách các đơn hàng  E2. Đơn hàng không đủ điều kiện hủy  Hệ thống thông báo không đủ điều kiện hủy quay lại hiển thị danh sách đơn hàng | | |
| **Assumptions:** |  | | |
| **Preconditions:** | * Tài khoản có quyền thực hiện chức năng theo đúng phân quyền | | |
| **Post Conditions:** | * Hệ thống cập nhật trạng thái đơn hàng | | |
| **Activity Diagram:A diagram of a flowchart  AI-generated content may be incorrect.** | | | |
| **Sequence Diagram:**  **A white paper with black and yellow lines  AI-generated content may be incorrect.** | | | |
| **Class Diagram:**  **A screenshot of a computer flowchart  AI-generated content may be incorrect.** | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| A diagram of a diagram  AI-generated content may be incorrect. | | | |
| **Use Case Number:** | UC5 | | |
| **Use Case Name:** | Quản lý vận đơn | | |
| **Actor (s):** | *Cửa hàng, Khách hàng* | | |
| **Maturity:** | Focused | | |
| **Summary:** | Xem và cập nhật trạng thái vận đơn | | |
| **Basic Course of Events:** luồng nghiệp vụ chính (đăng kí học phần) | **Actor Action** | | **System Response** |
| 1. Nhà hàng đăng nhập và chấp nhận đơn | |  |
|  | | 2. Hệ thống tạo kế hoạch vận chuyển và cập nhật trạng thái đơn. E1,E2 |
| 3. Người dùng chọn xem chi tiết vận chuyển | |  |
|  | | 4. Hệ thống hiển thị trạng thái vận chuyển |
| 5.Kết thúc sử dụng tại đây.A1 | |  |
| **Alternative Paths** | A1. Người đăng nhập là khách hàng | | |
| **Actor Action** | **System Response** | |
| 1. Khách hàng chọn đơn và nhấn xác nhận đã nhận đơn |  | |
|  | 2. Hệ thống giải phóng drone và cập nhật trạng thái đơn hàng.E3 | |
| **Exception Paths:** luồng vi phạm (kiểm tra điều kiện) | E1. Đơn không đủ điều kiện để chấp nhận  Hệ thống thông báo đơn không đủ điều kiện để chấp nhận, kết thúc sử dụng tại đây  E2. Không có drone khả dụng  Hệ thống thông báo không có drone khả dụng và kết thúc sử dụng  E3. Đơn hàng khách hàng xác nhận không phải là đơn đang được giao tới  Hệ thống báo đơn hàng xác nhận không đúng và yêu cầu xác nhận lại | | |
| **Assumptions:** |  | | |
| **Preconditions:** | * Tài khoản có quyền thực hiện chức năng theo đúng phân quyền | | |
| **Post Conditions:** | * Trạng thái vận đơn được cập nhật | | |
| **Activity Diagram:A diagram of a flowchart  AI-generated content may be incorrect.** | | | |
| **Sequence Diagram:**  **A white paper with yellow lines  AI-generated content may be incorrect.** | | | |
| **Class diagram:**  **A screenshot of a computer flowchart  AI-generated content may be incorrect.** | | | |

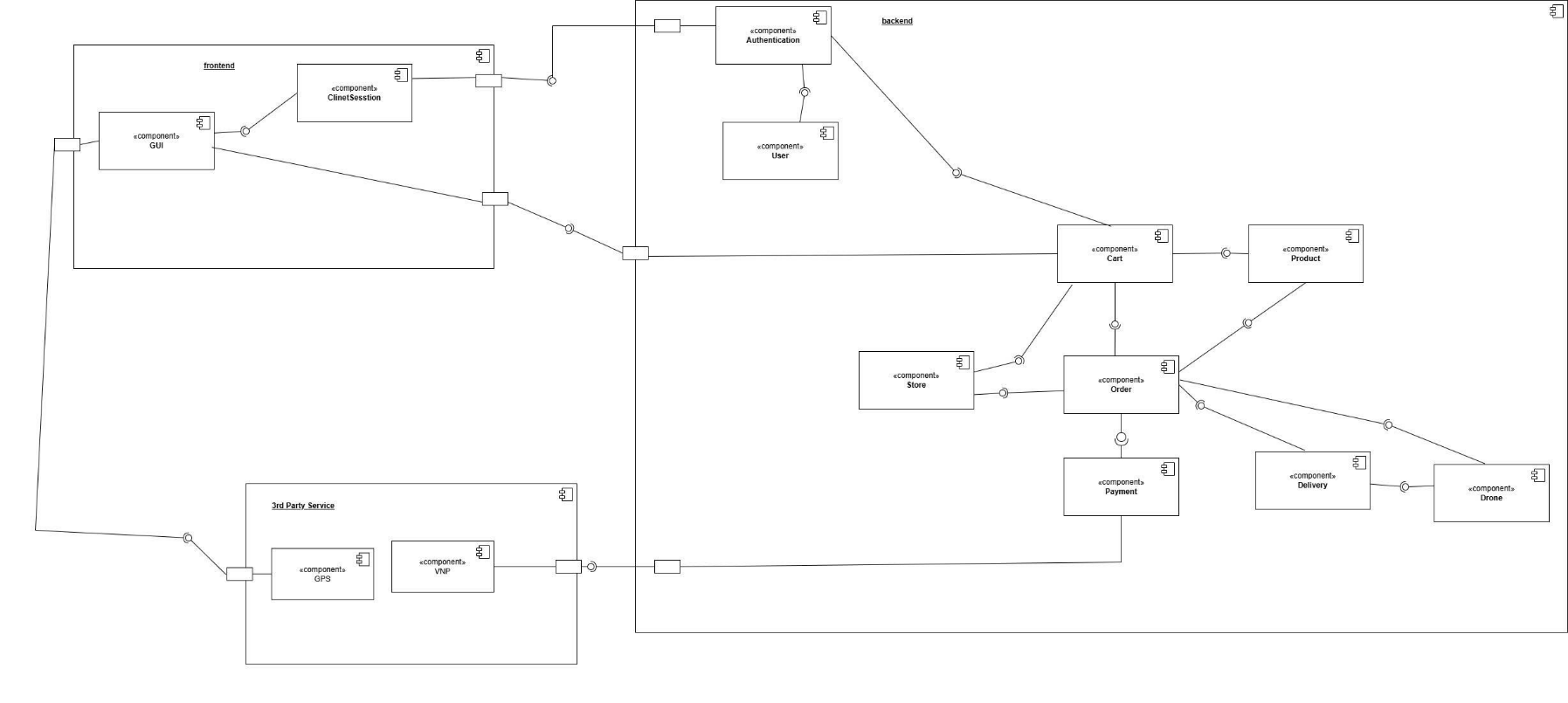
|  |  |  |
| --- | --- | --- |
| A diagram of a diagram  AI-generated content may be incorrect. | | |
| **Use Case Number:** | UC6 | |
| **Use Case Name:** | Cập nhật sản phẩm | |
| **Actor (s):** | *Cửa hàng, admin, khách hàng* | |
| **Maturity:** | Focused | |
| **Summary:** | Sản phẩm được cập nhật theo nhu cầu của admin hoặc Cửa hàng | |
| **Basic Course of Events:** luồng nghiệp vụ chính (đăng kí học phần) | Actor Action | System Respond |
| **1.Người dung đăng nhập là cửa hàng.A1,A2** |  |
|  | 2 .Hệ thống hiển thị trang chào mừng |
| 3 Người dùng chọn quản lý sản phẩm |  |
|  | 4 .Hệ thống hiển thị danh sách món ăn cửa hàng đang quản lý |
| 5 .Người dùng chọn sản phẩm muốn chỉnh sửa thông tin |  |
|  | 6. Hệ thống hiên thị chi tiêts sản phẩm vửa chọn |
|  | **7. Người dùng nhập thông tin muốn chỉnh sửa. E1** |  |
|  |  | 8.Hệ thống xác thực và ghi nhận lại thông tin |
| **Alternative Paths** | A1. Người đăng nhập là admin   1. Admin chọn sản phẩm muốn cập nhật 2. Hiển thị trạng thái của sản phẩm 3. Admin chỉ được cập nhật trạng thái sản phẩm 4. Quay lại bước 8 của luồng chính | |
| **Exception Paths:** luồng vi phạm (kiểm tra điều kiện) | E1. Thông tin nhập không hợp lệ   1. Hệ thống thông báo nhập không hợp lệ 2. Quay lại bước 6 trong luồng chính | |
| **Preconditions:** | * Là admin hoặc nhà hàng | |
| **Post Conditions:** | * Sản phẩm được cập nhật | |
| **Activity Diagram:**  A diagram of a flowchart  AI-generated content may be incorrect. | | |
| **A white sheet of paper with text  AI-generated content may be incorrect.** | | |
| **A screenshot of a computer program  AI-generated content may be incorrect.** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| A diagram with text and arrows  AI-generated content may be incorrect. | | | |
| **Use Case Number:** | UC7 | | |
| **Use Case Name:** | Quản lý Sổ Cái và Thanh Toán Cửa Hàng | | |
| **Actor (s):** | *Admin, Cửa hàng* | | |
| **Maturity:** | Focused | | |
| **Summary:** | Admin và cửa hàng quản lý sổ cái giao dịch và xử lý thanh toán cho cửa hàng. | | |
| **Basic Course of Events:** luồng nghiệp vụ chính (đăng kí học phần) | **Actor Action** | | **System Response** |
| 1. Actor đăng nhập vào hệ thống | |  |
|  | | 2. Hệ thống hiển thị trang chào mừng với các chức năng chính |
| 3. Actor chọn quản lý sổ cái | |  |
|  | | 4. Hệ thống hiển thị các chức năng: xem số tiền chưa thanh toán, lịch sử thanh toán |
| 5. Actor chọn xem số tiền chưa thanh toán của cửa hàng | |  |
|  | | 6. Hệ thống tính toán và hiển thị tổng số tiền chưa thanh toán |
| 7. Admin tạo lô thanh toán cho cửa hàng A1,E1 | |  |
|  | | 8. Hệ thống tạo PayoutBatch và chuyển trạng thái các giao dịch thành "PENDING\_PAYOUT" |
| **9. Admin xác nhận đã thanh toán và nhập mã giao dịch** | |  |
|  | | 10. Hệ thống cập nhật trạng thái PayoutBatch thành "PAID" và ghi nhận thông tin thanh toán |
| **11. Kết thúc hành động** | |  |
|  | | |
| **Alternative Paths** | A1. Xem lịch sử thanh toán | | |
| **Actor Action** | **System Response** | |
| **1. Actor chọn xem lịch sử thanh toán của cửa hàng** |  | |
|  | **2. Hệ thống hiển thị danh sách các PayoutBatch (batchId, amount, createdDate, paidDate, status)** | |
|  | **3 .Quay lại bước 4 luồng chính** | |
|  | | |
| **Exception Paths:** luồng vi phạm (kiểm tra điều kiện) | **E1. Không có tiền để tạo lô thanh toán**  Hệ thống thông báo "Không có giao dịch nào chưa thanh toán cho cửa hàng này"  Quay lại bước 4 của luồng chính | | |
| **Assumptions:** | * Chỉ các giao dịch có trạng thái UNPAID mới được tính vào số tiền chưa thanh toán * Một PayoutBatch chỉ có thể được đánh dấu là PAID một lần * Admin phải nhập mã giao dịch khi xác nhận thanh toán | | |
| **Preconditions:** | Actor phải đăng nhập với quyền Admin hoặc Store  Có ít nhất một giao dịch đã được ghi nhận trong Ledger | | |
| **Post Conditions:** | Số tiền chưa thanh toán được cập nhật  PayoutBatch được tạo và theo dõi  Cửa hàng nhận được thông tin thanh toán | | |
| **Activity Diagram:**  A diagram of a flowchart  AI-generated content may be incorrect. | | | |
| **A white sheet of paper with yellow lines  AI-generated content may be incorrect.** | | | |
| **A screenshot of a computer  AI-generated content may be incorrect.** | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| A black and white background with white dots  AI-generated content may be incorrect. | | | |
| **Use Case Number:** | UC4 | | |
| **Use Case Name:** | Quản lý cửa hàng | | |
| **Actor (s):** | [*Student*](file:///D:\Download\Zalo%20Received%20Files\DomainModel.doc)*,* [*Advisor*](file:///D:\Download\Zalo%20Received%20Files\DomainModel.doc) | | |
| **Maturity:** | Focused | | |
| **Summary:** |  | | |
| **Basic Course of Events:** | **Actor Action** | | **System Response** |
| 1. Người dùng đăng nhập | |  |
|  | | 2. Hệ thống kiểm tra phân quyền. |
| 3. Chủ cửa hàng xem thông tin hiện tại. | |  |
| 4. Chỉnh sửa thông tin cửa hàng. **E1** | |  |
| **Alternative Paths:** | A1. Tạo cửa hàng mới | | |
|  | **Actor Action** | **System Response** | |
|  | **A1.1** Điền form đăng kí. E2 |  | |
|  |  | **A1.2** Hệ thống tạo store mới với trạng thái chờ duyệt | |
|  |  |  | |
|  | **A2.** Admin duyệt cửa hàng | | |
|  | **Actor Action** | **System Response** | |
|  |  | A2.1 Hệ thống hiển thị thông tin cửa hàng các cửa hàng | |
|  | A2.2 Chọn cửa hàng |  | |
|  | A2.3 Cập nhật trạng thái cửa hàng |  | |
|  |  | | |
| **Exception Paths:** | **E1.**  Thông tin nhập vào bị lỗi, quay lại bước 4 | | |
|  | **E2.** Thông tin nhập vào bị lỗi, quay lại bước A1.1 | | |
| **Extension Points:** | None | | |
| **Triggers:** | The actor wishes to generate student report. | | |
| **Assumptions:** | [*Student preferences*](file:///D:\Download\Zalo%20Received%20Files\Glossary.doc) have been entered into the system. | | |
| **Preconditions:** | None | | |
| **Post Conditions:** | A student report may be generated. | | |
| **Reference: Business Rules:** | None | | |
| **Reference: Risks** | None | | |
| **Author(s):** | XXXXXX | | |
| **Date:** | 11-14-07 | | |
| **Activity Diagram:**  A screenshot of a computer  AI-generated content may be incorrect. | | | |
| **Sequence diagram:**  **A black background with white and yellow lines  AI-generated content may be incorrect.** | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| A white oval with black text  AI-generated content may be incorrect. | | | |
| **Use Case Number:** | UC5 | | |
| **Use Case Name:** | Quản lý drone | | |
| **Actor (s):** | Admin | | |
| **Maturity:** | Focused | | |
| **Summary:** |  | | |
| **Basic Course of Events:** | **Actor Action** | | **System Response** |
| 1. Admin mở trang Quản lý Drone | |  |
|  | | 2. Hiển thị danh sách Drone |
| 3. Chọn chức năng A1. | |  |
|  | |  |
| **Alternative Paths:** | A1. Thêm drone mới | | |
|  | **Actor Action** | **System Response** | |
|  |  | **A1.1** Hiển thị form thêm mới | |
|  | **A1.2** Nhập thông tin. E1 |  | |
|  |  |  | |
|  |  | **A1.3** Tạo Drone mới trong Database | |
|  |  | **A1.4** Thông báo: Thêm thành công! | |
|  | **A2.** Xem chi tiết drone | | |
|  | **Actor Action** | **System Response** | |
|  | **A2.1** Chọn xem chi tiết drone |  | |
|  |  | A2.2 Hiển thị thông tin | |
|  | **A3.** Chỉnh sửa thông tin drone | | |
|  | **Actor Action** | **System Response** | |
|  | **A3.1** Mở form chỉnh sửa |  | |
|  | **A3.2** Cập nhật thông tin. **E2** |  | |
|  |  | **A3.3** Cập nhật Drone vào Database. **E2** | |
|  |  | **A3.4** Thông báo: Cập nhật thành công! | |
|  |  | | |
| **Exception Paths:** | **E1.**  Thông tin nhập vào bị lỗi, quay lại bước A1.2 | | |
|  | **E2.** Thông tin nhập vào bị lỗi, quay lại bước A3.2 | | |
| **Extension Points:** | None | | |
| **Triggers:** | The actor wishes to generate student report. | | |
| **Assumptions:** | [*Student preferences*](file:///D:\Download\Zalo%20Received%20Files\Glossary.doc) have been entered into the system. | | |
| **Preconditions:** | None | | |
| **Post Conditions:** | A student report may be generated. | | |
| **Reference: Business Rules:** | None | | |
| **Reference: Risks** | None | | |
| **Author(s):** | XXXXXX | | |
| **Date:** | 11-14-07 | | |
| **Activity Diagram:**  A screenshot of a computer  AI-generated content may be incorrect. | | | |
| **Sequence diagram:**  **A black screen with white and yellow lines  AI-generated content may be incorrect.** | | | |

**Component diagram**

****

**Deployment diagram**

**A white rectangular object with black text

AI-generated content may be incorrect.**

**1. Client Device**

* **Hệ điều hành:** Windows 10/11 → phổ biến, dễ cài đặt, tương thích hầu hết trình duyệt.
* **Trình duyệt:** Chrome 119+, Edge 120+, Firefox 120+ → hỗ trợ tốt HTTP/2, WebSocket, và ES6.
* **Cấu hình tối thiểu:** RAM 4GB, CPU 2 nhân → đảm bảo chạy web app ổn định.
* **Frontend:** React 18.2 → cập nhật giao diện theo thời gian thực khi theo dõi drone.
* **Vai trò:** Gửi yêu cầu đặt món, thanh toán VNPay và xem trạng thái giao hàng.

**2. Application Server (Spring Boot – Java 21)**

* **Nền tảng:** Java 21 LTS (OpenJDK)
  + Hỗ trợ Virtual Threads (Project Loom) → tăng hiệu năng xử lý đồng thời.
  + Cải thiện bảo mật và hiệu suất JVM.
* **Framework:** Spring Boot 3.3 + Tomcat 10.1
  + Quản lý REST API, xác thực người dùng, gọi dịch vụ VNPay & Google Maps.
* **Cấu hình phần cứng:**
  + CPU: Intel Xeon 4 cores @ 2.6GHz
  + RAM: 8GB DDR4
  + Ổ đĩa: SSD 100GB (Windows Server 2022)
* **Vai trò:**
  + Xử lý nghiệp vụ: tạo đơn hàng, tính phí, theo dõi trạng thái drone.
  + Giao tiếp với VNPay (thanh toán), Google Maps (tính quãng đường).
  + Kết nối và ghi dữ liệu xuống MySQL Database Server.
* **Cấu hình JVM:**
  + -Xms2g -Xmx6g → tối ưu vùng nhớ Heap, giảm thời gian GC.

**3. Database Server (MySQL 8.0.35 trên Windows Server)**

* **Phần cứng:**
  + CPU: Intel Xeon 8 cores @ 3.0GHz
  + RAM: 16GB DDR4
  + Ổ đĩa: 500GB NVMe SSD (RAID 10) → tốc độ cao, độ an toàn dữ liệu cao.
* **Hệ điều hành:** Windows Server 2022 Datacenter.
* **Cấu hình cơ bản:**
  + innodb\_buffer\_pool\_size = 8G (cache dữ liệu thường xuyên truy cập).
  + max\_connections = 200.
  + character\_set = utf8mb4 → hỗ trợ emoji và tiếng Việt đầy đủ.
* **Chức năng:**
  + Lưu trữ thông tin người dùng, cửa hàng, đơn hàng, drone, lịch sử giao hàng.
  + Thực hiện backup toàn bộ mỗi đêm lúc 2 giờ sáng.
* **Kết nối:** qua port **3306**, chỉ cho phép Application Server truy cập (firewall IP whitelist).

**4. Load Balancer (NGINX trên Windows)**

* **Chức năng:**
  + Phân phối lưu lượng truy cập tới nhiều App Server Spring Boot.
  + SSL termination – giải mã HTTPS tại đây để giảm tải cho server phía sau.
* **Cấu hình:**
  + Thuật toán: **Round Robin** → chia đều lượng request.
  + Giao thức: **HTTPS (TLS 1.3)** + **HTTP/2** → tăng tốc độ tải trang.
  + Nén dữ liệu: **gzip + brotli** → giảm băng thông ~70%.
  + **Keep-Alive:** 60s → tái sử dụng kết nối TCP, giảm độ trễ.
* **Port:**
  + 443 (HTTPS) cho client.
  + 8080 (HTTP nội bộ) cho Application Server.
* **Hệ điều hành:** Windows Server 2022, băng thông mạng 1 Gbps.

**5. VNPay Payment Gateway**

* **Dịch vụ thanh toán nội địa Việt Nam** → hỗ trợ hầu hết ngân hàng, QR Code, mobile banking.
* **API version:** 2.1.0 (VNPay Sandbox).
* **Cấu hình tích hợp:**
* vnPay:
* tmn-code: "8VWZT7KJ"
* hash-secret: "ANAIDKYHLTU5HVKRMUM4NVRMLCN0SJU1"
* payment-url: "https://sandbox.vnpayment.vn/paymentv2/vpcpay.html"
* return-url: "http://localhost:5173/checkout/payment-callback"
* **Lý do chọn VNPay:**
  + Phổ biến tại Việt Nam, dễ tích hợp REST API.
  + Không cần lưu thông tin thẻ → đảm bảo an toàn thanh toán.
  + Hỗ trợ **retry 3 lần (exponential backoff)** khi timeout hoặc lỗi kết nối.
* **Timeout:** 30s → đủ thời gian xử lý xác thực ngân hàng.
* **Quy trình:**
  + App Server gửi yêu cầu tạo thanh toán tới VNPay.
  + VNPay trả về URL redirect.
  + Client mở giao diện VNPay, xác nhận giao dịch.
  + Sau khi thanh toán thành công, VNPay gọi callback về /payment-callback.

**6. Google Maps Platform**

* **APIs sử dụng:** Directions API, Distance Matrix API, Geocoding API.
* **Mục đích:**
  + Tính quãng đường bay giữa cửa hàng và khách hàng.
  + Hiển thị tuyến đường drone trên bản đồ real-time.
* **Timeout:** 15s, retry 3 lần (1s → 2s → 4s).
* **Cache Redis:** TTL 10 phút để giảm chi phí và tăng tốc độ phản hồi.
* **Bảo mật:** API key giới hạn theo domain và IP, tự động rotation mỗi 90 ngày.

**7. Network & Protocols**

* **Băng thông:**
  + App Server: 1 Gbps.
  + Database Server: 10 Gbps → đảm bảo tốc độ truy vấn khi có nhiều kết nối.
* **Giao thức:**
  + HTTPS (TLS 1.3, AES-256-GCM) → mã hóa toàn bộ dữ liệu.
  + HTTP/2 → hỗ trợ multiplexing, giảm latency.
  + WebSocket → truyền dữ liệu vị trí drone real-time tới client.

**8. Luồng hoạt động chính**

1. Người dùng truy cập website trên Windows → gửi yêu cầu đặt món qua **HTTPS**.
2. **Nginx Load Balancer** tiếp nhận và phân phối request đến **Application Server (Spring Boot – Java 21)**.
3. Backend xử lý đơn hàng, gọi **VNPay API** để thanh toán, lưu dữ liệu vào **MySQL Database**.
4. Sau khi thanh toán thành công, backend gọi **Google Maps API** để tính toán lộ trình bay.
5. Drone di chuyển, dữ liệu vị trí được cập nhật real-time về client qua **WebSocket**.
6. Toàn bộ quá trình được ghi log và lưu lịch sử giao hàng trong **Database Server**.

**ERD**

**A black background with white squares

AI-generated content may be incorrect.**