

CHƯƠNG I: TỔNG QUAN DỰ ÁN

1.1. Website tối ưu hóa quy trình giao hàng cho chuỗi cửa hàng Bách Hóa Xanh

1.1.1. Mục đích sử dụng

Website được xây dựng nhằm hỗ trợ Bách Hóa Xanh trong việc tối ưu hóa toàn bộ quy trình giao hàng nội bộ, từ kho trung tâm hoặc nhà cung cấp đến các cửa hàng trên toàn địa bàn thành phố Hồ Chí Minh.

Cụ thể, hệ thống sẽ giúp:

- Tự động lập kế hoạch giao hàng, giảm trùng tuyến, giảm thời gian giao và tiết kiệm chi phí vận chuyển
- Theo dõi trạng thái đơn hàng và lộ trình giao hàng theo thời gian thực.
- Phân tích hiệu suất giao hàng, giúp cải thiện quy trình và tăng tính chính xác.

Website hướng đến mục tiêu giao hàng đúng giờ, đúng hàng, đúng nơi, từ đó nâng cao năng suất vận hành và chất lượng phục vụ tại các cửa hàng.

1.1.2. Đối tượng áp dụng

- Điều phối viên giao hàng tại kho trung tâm: Lập lịch giao hàng, điều xe, sắp xếp tài xế.
- Tài xế giao hàng: Xem lộ trình, cập nhật trạng thái đơn hàng (qua mobile app hoặc giao diện web).
- Nhân viên nhận hàng tại cửa hàng: Theo dõi thời gian dự kiến giao, kiểm hàng.
- Bộ phận vận hành và giám sát: Giám sát tiến độ, phân tích hiệu suất giao nhận theo ngày/tháng.

1.1. 3. Các hệ thống tương tự hiện nay

Tại Việt Nam

Tên hệ thống	Mô tả	Liên quan
Giao Hàng Nhanh (GHN)	Nền tảng logistics thương mại điện tử	Mạnh về tracking, chưa dành riêng cho nội bộ chuỗi bán lẻ

Abivin vRoute	Hệ thống tối ưu hóa lộ trình giao hàng bằng AI	Dành cho logistics lớn như Vinamilk, TH True Milk
---------------	--	---

Trên thế giới

Tên hệ thống	Mô tả	Điểm nổi bật
Bringg (Mỹ)	Hệ thống điều phối giao hàng theo thời gian thực	Sử dụng bởi Walmart, McDonald's
Onfleet (Mỹ)	Giao diện tối ưu lộ trình, app tài xế, báo cáo giao hàng	Rất mạnh về bản đồ và UX

1.1.4. Khác biệt của hệ thống đề xuất

- Thiết kế chuyên biệt cho mô hình phân phối nội bộ nhiều điểm bán như Bách Hóa Xanh.
- Tối ưu hóa tuyến đường theo mặt hàng, vị trí, khung giờ hoạt động cửa hàng.
- Tích hợp với hệ thống ERP/kho nội bộ và quản lý hàng hóa (tồn kho, ưu tiên đơn hàng).

1.2. Input và Output của Hệ thống

- Dữ liệu đầu vào:
 - Thông tin Kho: Vị trí (địa chỉ hoặc tọa độ). # Hiện tại, đang sử dụng kinh độ, vĩ độ
 - Thông tin Khách hàng/Điểm giao hàng: Danh sách các điểm cần giao, vị trí (địa chỉ hoặc tọa độ), khối lượng/số lượng hàng hóa yêu cầu tại mỗi điểm.
 - Thông tin Đội xe: Số lượng xe, loại xe, tải trọng tối đa (theo khối lượng hoặc thể tích) của từng loại xe/từng xe cụ thể. Ngoài ra, người dùng có thể cung cấp thêm thông tin liên quan đến việc thuê xe (outsourcing từ bên ngoài) trong trường hợp số lượng xe có sẵn không đủ.
 - Thông tin Thời gian:

- Thời gian di chuyển ước tính giữa kho và các điểm giao hàng, và giữa các điểm giao hàng với nhau (có thể là ma trận khoảng cách/thời gian hoặc dựa trên API bản đồ).
- Thời gian bốc dỡ hàng hóa trung bình tại mỗi điểm giao (có thể cố định hoặc thay đổi theo khối lượng hàng).
- Ràng buộc:
 - Thời gian làm việc tối đa cho một xe/một chuyến (ví dụ: 8 tiếng).
 - (Tùy chọn) Cửa sổ thời gian giao hàng tại một số điểm (Time Windows - ví dụ: chỉ giao hàng từ 9h-11h sáng).
- Kết quả đầu ra:
 - Lịch trình tối ưu cho từng xe: Đây là output chính của dự án
 - Xe được chỉ định (ví dụ: Xe 01, Xe 02,...).
 - Thứ tự các điểm sẽ ghé thăm (bao gồm điểm xuất phát là kho và điểm kết thúc là quay về kho). Ví dụ: Kho -> Khách hàng A -> Khách hàng C -> Kho.
 - Thời gian dự kiến đến và rời khỏi mỗi điểm.
 - Tổng quãng đường dự kiến của tuyến.
 - Tổng thời gian dự kiến của tuyến.
 - Tổng tải trọng hàng hóa trên xe cho tuyến đó.
 - Thông tin tổng hợp: Tổng quãng đường toàn đội xe, tổng thời gian, số xe được sử dụng và tổng chi phí vận chuyển của đội xe.

1.3. Hướng phát triển trong tương lai

- **Tích hợp dữ liệu giao thông thời gian thực** để tính toán **thời gian di chuyển chính xác** hơn và đưa ra cảnh báo/đề xuất điều chỉnh. Trong phạm vi cuối kỳ, nhóm chưa tích hợp dữ liệu theo thời gian thực, chủ yếu dựa vào data trước đó để dự đoán thời gian giao hàng.
- **Phát triển ứng dụng di động cho tài xế (công nghệ V2X - từ khoá để tìm hình ảnh)**
- **Mở rộng xử lý các ràng buộc phức tạp hơn:** Tức là đáp ứng yêu cầu khắt khe hơn từ khách hàng, ví dụ: Đơn vị nhận hàng chỉ nhận hàng vào đúng thời điểm cụ thể.

- **Hỗ trợ bài toán tối ưu hóa đa kho:** Người sử dụng có thể add thêm nhiều kho hoặc nhà phân phối với một khả năng cung ứng hữu hạn, kèm theo 1 số các thông tin khác (vị trí, thời gian bốc hàng lên xe, thời gian hoạt động,).

CHƯƠNG II: TIẾN ĐỘ DỰ ÁN

2.1. Front end (Giao diện người dùng, Map,), cần làm gì ở từng bước này

2.1.1. Đã hoàn thành:

- **Tích hợp bản đồ** (sử dụng Google Maps API):
- Hiện thị được rõ và tương tác được với 50 vị trí cửa hàng Bách Hóa Xanh tại TP.HCM trên map.
- **Kết nối API Backend tạm thời:**
- Hiện thị được tuyến đường ngắn nhất (chỉ dùng thuật toán đơn giản để tìm đường đi ngắn nhất và giả sử tìm đường đi ngắn nhất từ kho đến 1 cửa hàng gần nhất và chưa chịu bất kì ràng buộc nào, chưa kết nối API backend với mô hình GNN mà nhóm đang làm) từ kho đến 1 cửa hàng nhờ vào việc gửi yêu cầu tính toán tuyến đường đến thư viện Openrouteservice.

2.1.2. Công việc đang thực hiện / cần hoàn thiện:

- **Giao diện trực quan hóa kết quả:**
- Hiện thị rõ thứ tự các điểm ghé.
- Cập nhật UI theo kết quả GNN (chi phí, quãng đường).
- **Tích hợp bản đồ nâng cao:**
- Vẽ tuyến đường và hiện thị chi tiết theo lộ trình thực tế qua các cửa hàng bách hóa xanh.
- Cập nhật và hiện thị thời gian/tổng quãng đường/chi phí dự kiến tương ứng tuyến đường của xe.

2.1.3. Kế hoạch tiếp theo & định hướng phát triển cuối kỳ:

- Tích hợp hoàn chỉnh với mô hình GNN, kết nối API chính thức để nhận dữ liệu từ mô hình GNN.
- Tạo ra một trang web để cho người dùng có thể nhập Input (thông tin xe, loại hàng, cửa hàng cần đến, ...) và hiện thị được kết quả của mô hình GNN lên map bằng cách gửi yêu cầu API

- Thêm các tính năng phụ trợ như là xuất báo cáo tuyến đường PDF/Excel, thống kê chi phí vận hành theo tuyến.

2.2 Back end

Input: Map và xe

Xử lý input: Thông tin map được xử lý bởi GNN

- Thông tin của map: Đặc điểm và liên kết của từng node

Xây dựng tuyến đường:

- Vòng lặp: Xe từ xuất phát từ depot thông qua “điểm hấp dẫn” để lựa chọn node tiếp theo. Điểm hấp dẫn dựa trên tích của emb current và emb candidate điều chỉnh bởi các yếu tố động như thời gian, khoảng cách và tải trọng còn lại. Node có điểm hấp dẫn cao nhất được chọn. Xe vượt quá quy định tái vòng lặp.

Lựa chọn tối ưu:

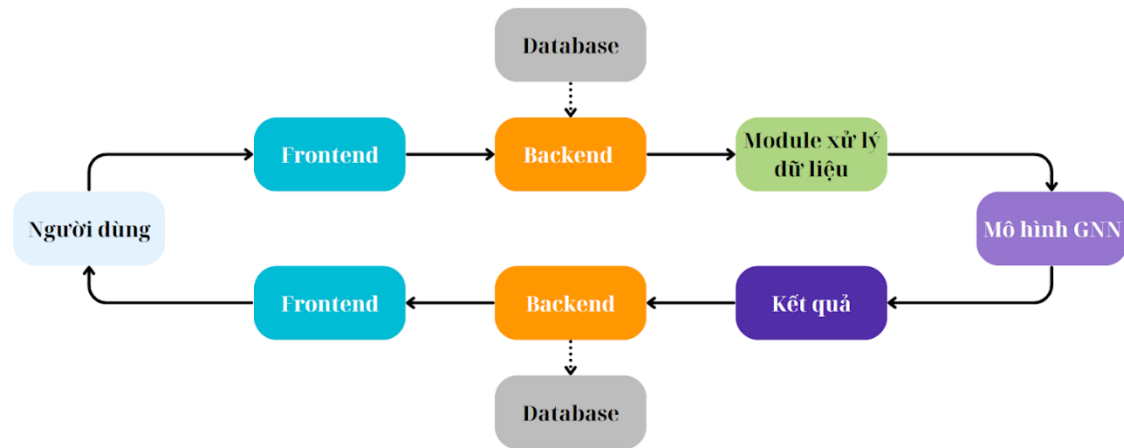
- Lựa chọn giữa các loại xe: Vòng lặp được thực hiện với tất cả loại xe, các loại xe thông qua đánh giá bởi siêu tham số với điểm số cao nhất sẽ được chọn.

- Siêu tham số dựa trên mức độ quan trọng của các tiêu chí sau: số lượng node mới được phục vụ, hiệu quả về chi phí, hiệu quả về tải trọng và hiệu quả về thời gian.

Output: Tối ưu chi phí (Nhiên liệu + lương + phạt)

2.3 Kiến trúc hệ thống.





1. Người dùng nhập thông tin các điểm giao hàng, kho, xe,... trên Frontend.
2. Frontend gửi yêu cầu đến Backend.
3. Backend nhận yêu cầu, gọi Module Xử lý Dữ liệu để chuyển đổi input thành dạng đồ thị.
4. Gửi đồ thị này đến Module Machine Learning. (GNN, GAT)
5. Module Machine Learning tải Model đã huấn luyện (F) và thực hiện dự đoán (inference) để tìm ra các tuyến đường tối ưu.
6. Kết quả (các routes) được trả về Backend.
7. Backend (có thể lưu vào Database) và gửi kết quả về Frontend.
8. Frontend nhận kết quả và hiển thị các tuyến đường trên bản đồ cho người dùng xem.