

Báo Cáo Đề Cương Dự Án: Website Công Thức Nấu Ăn Với Hệ Thống Gợi Ý KNN

Tuần 1: Giới Thiệu Dự Án

- **Mục tiêu:**
Xây dựng một website công thức nấu ăn cho phép người dùng tìm kiếm, xem chi tiết công thức và nhận gợi ý các công thức phù hợp dựa trên sở thích hoặc nguyên liệu sẵn có, sử dụng thuật toán KNN (K-Nearest Neighbors).
- **Phạm vi dự án:**
 - Tạo một nền tảng web thân thiện với người dùng.
 - Tích hợp hệ thống gợi ý dựa trên dinh dưỡng người dùng đưa ra
 - Đảm bảo giao diện đơn giản, dễ sử dụng và hiệu quả.
- **Kế hoạch sơ bộ:**
 - Đưa ra phần mềm giới thiệu món ăn và công thức món ăn
 - Xác định công nghệ sử dụng: ngôn ngữ lập trình (Python cho backend, HTML/CSS/JavaScript, React cho frontend), framework (Flask/Django), cơ sở dữ liệu (SQLite hoặc MongoDB).
 - Lên ý tưởng về các tính năng chính: tìm kiếm công thức, hiển thị chi tiết, gợi ý công thức liên quan.
- **Kết quả mong đợi:**
 - Tổng quan sơ bộ về dự án

Tuần 2: Nguồn Dữ liệu và Tiền Xử Lí Dữ liệu

- **Mục tiêu:**
Thu thập và chuẩn bị dữ liệu cần thiết để xây dựng hệ thống gợi ý dựa trên KNN.
- **Nguồn dữ liệu:**
 - Tìm kiếm tập dữ liệu công thức nấu ăn từ các nguồn mở (Kaggle, Food.com)
- **Tiền xử lí dữ liệu:**
 - Làm sạch dữ liệu: loại bỏ giá trị trùng lặp, chuẩn hóa định dạng (ví dụ: chuyển "1 cup" thành đơn vị thống nhất).

- **Kết quả mong đợi:**

- Tập dữ liệu đã được làm sạch và chuyển đổi.
- Tài liệu mô tả cấu trúc dữ liệu và cách tiền xử lí.

Tuần 3: Tìm Hiểu và Áp Dụng Thuật Toán KNN

- **Mục tiêu:**

Hiểu cách hoạt động của KNN và áp dụng nó vào hệ thống gợi ý công thức nấu ăn.

- **Nghiên cứu thuật toán KNN:**

- Tìm hiểu cách KNN tính khoảng cách (Euclidean, Manhattan) giữa các điểm dữ liệu.
- Xác định số lượng hàng xóm gần nhất (k) tối ưu cho gợi ý.
- Đánh giá hiệu quả của mô hình qua các chỉ số như độ chính xác hoặc độ phù hợp của gợi ý.

- **Áp dụng vào dự án:**

- Sử dụng Python (thư viện scikit-learn) để triển khai KNN.
- Huấn luyện mô hình trên tập dữ liệu đã tiền xử lí: ví dụ, gợi ý công thức dựa trên nguyên liệu người dùng nhập hoặc công thức họ đã xem trước đó.
- Kiểm tra kết quả: chạy thử với các kịch bản (người dùng nhập dinh dưỡng → gợi ý món ăn).

- **Kết quả mong đợi:**

- Mô hình KNN cơ bản hoạt động được.
- Báo cáo phân tích ưu/nhược điểm của KNN trong bài toán gợi ý công thức.

Tuần 4: Code Frontend Cho Dự Án

- **Mục tiêu:**

Xây dựng giao diện người dùng (frontend) cho website, kết nối với backend đã phát triển.

- **Thiết kế giao diện:**

- Trang chủ: hiển thị danh sách công thức nổi bật, ô tìm kiếm.

- Trang chi tiết công thức: nguyên liệu, hướng dẫn, gợi ý công thức liên quan (dùng KNN).
- Form nhập liệu: người dùng nhập nguyên liệu hoặc sở thích để nhận gợi ý.
- **Công nghệ sử dụng:**
 - HTML/CSS: tạo bố cục và giao diện đẹp mắt.
 - JavaScript: xử lý tương tác (tìm kiếm, hiển thị gợi ý dynamically).
 - Kết nối với backend qua API (RESTful API sử dụng Flask/Django).
- **Triển khai:**
 - Code các thành phần giao diện chính.
 - Tích hợp hệ thống gợi ý KNN từ backend vào frontend (gọi API để lấy danh sách công thức gợi ý).
 - Kiểm tra tính tương thích trên các thiết bị (desktop, mobile).
- **Kết quả mong đợi:**
 - Giao diện website hoàn chỉnh, hoạt động mượt mà.
 - Tính năng gợi ý hiển thị chính xác trên frontend dựa trên kết quả từ KNN.

Tuần 5: Phát Triển Backend

- **Mục tiêu:**
Xây dựng backend để xử lý logic và cung cấp dữ liệu cho frontend.
- **Công việc:**
 - Thiết lập server: dùng Flask/Django để tạo API.
 - Quản lý cơ sở dữ liệu: lưu trữ công thức (SQLite/MongoDB).
 - Tích hợp KNN: xử lý yêu cầu gợi ý từ frontend (nhập nguyên liệu → trả về danh sách công thức).
 - API endpoints:
 - /search: tìm kiếm công thức.
 - /recommend: gợi ý công thức dựa trên KNN.
 - /recipe/:id: lấy chi tiết công thức.

- **Kết quả mong đợi:**

- Backend hoạt động, kết nối thành công với frontend, hệ thống gợi ý hoàn chỉnh.

Tuần 6: Hoàn Thiện Frontend

- **Mục tiêu:**

Tối ưu và hoàn thiện giao diện người dùng.

- **Công việc:**

- Cải thiện giao diện: thêm hiệu ứng (hover, animation), tối ưu responsive (mobile, desktop).
- Hoàn thiện tương tác: xử lý form tìm kiếm, hiển thị gợi ý động từ API.
- Kiểm tra và sửa lỗi: đảm bảo không có lỗi hiển thị hoặc tương tác.

- **Kết quả mong đợi:**

- Frontend hoàn chỉnh, sẵn sàng kết nối với backend, trải nghiệm người dùng mượt mà.