Dự tính thời gian

[Tuần 1: Lập kế hoạch & Thiết kế hệ thống 1](#_Toc7880)

[1.1 Xác định yêu cầu dự án 1](#_Toc11260)

[1.2 Chọn công nghệ 1](#_Toc26250)

[1.3 Lập kế hoạch & Thiết kế kiến trúc hệ thống 2](#_Toc10446)

[Tuần 2: Thiết kế Database & Cấu hình môi trường 2](#_Toc744)

[2.1 Thiết kế Database 2](#_Toc28120)

[2.2 Cấu hình môi trường 2](#_Toc28032)

[Tuần 3: Xây dựng chức năng thu thập dữ liệu 2](#_Toc26768)

[3.1 Viết Scraper để thu thập dữ liệu 3](#_Toc5879)

[3.2 Lưu dữ liệu vào Database 3](#_Toc17384)

[3.3 Tự động hóa Scraping 3](#_Toc4055)

[Tuần 4: Xây dựng API Backend 3](#_Toc567)

[4.1 Tạo API REST 3](#_Toc32061)

[4.2 Bảo mật API 3](#_Toc22436)

[Tuần 5: Xây dựng Frontend 3](#_Toc24630)

[5.1 Thiết kế giao diện 3](#_Toc20640)

[5.2 Trang chi tiết sản phẩm 4](#_Toc11445)

[5.3 Các trang phụ trợ 5](#_Toc28014)

[5.4 Kết nối API với Frontend 5](#_Toc21895)

[Tuần 6: Hoàn thiện & Cải thiện hiệu suất 5](#_Toc25305)

[6.1 Tối ưu hiệu suất API 5](#_Toc23741)

[6.2 Cải thiện giao diện 6](#_Toc29046)

[Tuần 7: Kiểm thử & Fix bug 6](#_Toc13391)

[7.1 Kiểm thử dữ liệu 6](#_Toc4139)

[7.2 Kiểm tra bảo mật 6](#_Toc22801)

[Tuần 8: Triển khai & Viết tài liệu 6](#_Toc19419)

[8.1 Triển khai lên VPS 6](#_Toc8762)

[8.2 Viết tài liệu 6](#_Toc3416)

[Tuần 9-10: Báo cáo & Đánh giá 6](#_Toc20986)

[9.1 Chạy thử nghiệm thực tế 6](#_Toc16403)

[10.1 Đánh giá & Hoàn thiện 7](#_Toc6139)

[Tóm tắt nhiệm vụ theo tuần 7](#_Toc26401)

## ****Tuần 1: Lập kế hoạch & Thiết kế hệ thống****

### ****1.1 Xác định yêu cầu dự án****

Mục tiêu: Xây dựng một web thu thập dữ liệu giá sản phẩm sữa từ các trang web bán lẻ, giúp người dùng so sánh giá.

- Chia thành 4 loại sữa sau: trẻ em, người lớn, người già và các loại khác

- Cần phải biết được phương thức làm sao khi thực hiện kĩ thuật lấy thông tin giá từ nhiều nguồn khác nhau.

- Chỉ so sánh các loại sữa có trên web hiển thị.

Chức năng chính:

Thu thập dữ liệu từ nhiều nguồn.

Lưu trữ lịch sử giá sản phẩm.

Hiển thị dữ liệu trên web.

API để truy vấn dữ liệu.

### ****1.2 Chọn công nghệ****

**Backend**: Django (hoặc FastAPI/Flask nếu muốn nhẹ hơn).

**Database**: PostgreSQL.

**Frontend**:

Tùy chọn: React,HTML/CSS.

UI framework: Bootstrap.

**Scraping**: Scrapy hoặc Requests + BeautifulSoup.

**Triển khai**: Phát triển sau

**Quản lý code**: GitLab (thiết lập repository, quy tắc commit).

### ****1.3 Lập kế hoạch & Thiết kế kiến trúc hệ thống****

Phác thảo **kiến trúc hệ thống**:

Backend xử lý logic, cung cấp API.

Cơ sở dữ liệu PostgreSQL lưu thông tin sản phẩm.

Frontend hiển thị dữ liệu từ API.

Vẽ sơ đồ **luồng dữ liệu** giữa các thành phần.

## ****Tuần 2: Thiết kế Database & Cấu hình môi trường****

### ****2.1 Thiết kế Database****

Tạo sơ đồ ERD với các bảng chính:

products (id, name, price, store, url, created\_at).

sources (id, name, base\_url).

price\_history (id, product\_id, price, date).

Viết script SQL để tạo bảng trong PostgreSQL.

### ****2.2 Cấu hình môi trường****

Cài đặt Python và các thư viện (Django/FastAPI, psycopg2, Scrapy…).

Thiết lập kết nối giữa backend và PostgreSQL.

Tạo file .env để lưu thông tin cấu hình database.

## ****Tuần 3: Xây dựng chức năng thu thập dữ liệu****

### ****3.1 Viết Scraper để thu thập dữ liệu****

Dùng **Scrapy hoặc Requests + BeautifulSoup** để lấy dữ liệu từ các trang web.

Trích xuất thông tin sản phẩm:

Tên sản phẩm.

Giá.

URL sản phẩm.

Cửa hàng bán.

### ****3.2 Lưu dữ liệu vào Database****

Viết pipeline để lưu dữ liệu vào PostgreSQL.

Kiểm tra dữ liệu tránh trùng lặp.

### ****3.3 Tự động hóa Scraping****

Dùng cron hoặc Celery để chạy scraper theo lịch (hằng ngày, mỗi giờ…).

Lưu lịch sử giá vào bảng price\_history.

## ****Tuần 4: Xây dựng API Backend****

### ****4.1 Tạo API REST****

Xây dựng API với Django Rest Framework hoặc FastAPI:

**GET /products**: Lấy danh sách sản phẩm.

**GET /products/{id}**: Lấy chi tiết sản phẩm.

**GET /products/{id}/price-history**: Lấy lịch sử giá của sản phẩm.

### ****4.2 Bảo mật API****

Chỉ cho phép truy vấn từ frontend bằng **CORS**.

Thêm xác thực nếu cần (JWT token).

## ****Tuần 5: Xây dựng Frontend****

### ****5.1 Thiết kế giao diện****

Dùng **Tailwind CSS / Bootstrap** để tạo UI:

| **Thành phần** | **Mô tả** | **Công nghệ** | **Thời gian ước tính** |
| --- | --- | --- | --- |
| **Header** | Chứa logo, thanh tìm kiếm, bộ lọc | Tailwind/Bootstrap | **4 giờ** |
| **Thanh tìm kiếm** | Cho phép nhập tên sản phẩm để tìm kiếm | JavaScript | **2 giờ** |
| **Bộ lọc sản phẩm** | Lọc theo giá, cửa hàng, loại sữa | JavaScript + API | **5 giờ** |
| **Danh sách sản phẩm** | Dạng lưới (grid) hiển thị thông tin sản phẩm | Tailwind/Bootstrap | **5 giờ** |
| **Card sản phẩm** | Hiển thị hình ảnh, tên, giá, nút xem chi tiết | Tailwind/Bootstrap | **4 giờ** |
| **Phân trang** | Chia nhỏ danh sách sản phẩm nếu có quá nhiều | JavaScript + API | **3 giờ** |
| **Tổng thời gian** | - | - | **23 giờ (~3 ngày)** |

#### ****5.1.1 Bố cục****

**Header (Thanh tìm kiếm & bộ lọc)**

**Thanh tìm kiếm**: Cho phép nhập tên sản phẩm để tìm kiếm nhanh.

**Bộ lọc sản phẩm**:

**Lọc theo giá** (khoảng giá từ thấp đến cao).

**Lọc theo cửa hàng** (VD: VinMart, Bách Hóa Xanh, Co.opmart).

**Lọc theo loại sữa** (Sữa tươi, sữa bột, sữa đặc…).

**Danh sách sản phẩm**

**Card sản phẩm** (hiển thị dạng lưới hoặc danh sách):

Hình ảnh sản phẩm.

Tên sản phẩm.

Giá hiện tại.

Cửa hàng.

Nút **"Xem chi tiết"** (dẫn đến trang chi tiết sản phẩm).

**Phân trang (Pagination)**

Chia nhỏ danh sách sản phẩm nếu có quá nhiều dữ liệu.

### ****5.2 Trang chi tiết sản phẩm****

**Mục đích:** Hiển thị thông tin chi tiết về sản phẩm và lịch sử giá.

#### ****5.2.1 Bố cục****

| **Thành phần** | **Mô tả** | **Công nghệ** | **Thời gian ước tính** |
| --- | --- | --- | --- |
| **Thông tin sản phẩm** | **Hình ảnh, tên, giá, cửa hàng, link mua** | **Tailwind/Bootstrap** | **5 giờ** |
| **Biểu đồ lịch sử giá** | **Hiển thị biến động giá theo thời gian** | **Chart.js / Recharts** | **6 giờ** |
| **Danh sách lịch sử giá** | **Dạng bảng hiển thị giá theo từng ngày** | **Tailwind/Bootstrap** | **3 giờ** |
| **Tổng thời gian** | **-** | **-** | **14 giờ (~2 ngày)** |

**Thông tin sản phẩm (Top Section)**

* **Hình ảnh sản phẩm**.
* **Tên sản phẩm**.
* **Giá hiện tại** (bôi đậm, có màu nổi bật).
* **Cửa hàng bán** + Link đến trang web mua hàng.

**Biểu đồ lịch sử giá (Graph)**

Hiển thị biểu đồ thay đổi giá theo thời gian (dùng Chart.js hoặc Recharts).

Trục X: Ngày cập nhật.

Trục Y: Giá sản phẩm.

**Danh sách lịch sử giá (Bảng)**

Hiển thị giá từng ngày theo bảng.

### ****5.3 Các trang phụ trợ****

| **Thành phần** | **Mô tả** | **Công nghệ** | **Thời gian ước tính** |
| --- | --- | --- | --- |
| **Trang “Giới thiệu”** | **Mô tả web, hướng dẫn sử dụng** | **Tailwind/Bootstrap** | **3 giờ** |
| **Trang “Liên hệ”** | **Form nhập tên, email, nội dung** | **Tailwind/Bootstrap + Backend xử lý** | **5 giờ** |
| **Tổng thời gian** | **-** | **-** | **8 giờ (~1 ngày)** |

#### ****Trang “Liên hệ”****

Form liên hệ với các trường:

Họ và tên.

Email.

Nội dung góp ý.

#### ****Trang “Giới thiệu”****

Mô tả mục đích của website.

Hướng dẫn sử dụng.

### ****5.4 Kết nối API với Frontend****

Dùng **Fetch API hoặc Axios** để lấy dữ liệu từ backend.

Hiển thị dữ liệu lên UI.

### ****Tối ưu & Kiểm thử giao diện****

| **Thành phần** | **Mô tả** | **Công nghệ** | **Thời gian ước tính** |
| --- | --- | --- | --- |
| **Responsive UI** | **Đảm bảo giao diện đẹp trên điện thoại & tablet** | **Tailwind/Bootstrap** | **5 giờ** |
| **Hiệu suất tải trang** | **Giảm tải CSS, tối ưu hình ảnh** | **Tailwind/Bootstrap** | **4 giờ** |
| **Kiểm thử UI** | **Test trên trình duyệt khác nhau (Chrome, Firefox…)** | **Manual Testing** | **6 giờ** |
| **Tổng thời gian** | **-** | **-** | **15 giờ (~2 ngày)** |

### ****Tổng ước tính thời gian hoàn thành UI****

| **Phần** | **Thời gian ước tính** |
| --- | --- |
| **Trang chủ** | **23 giờ (~3 ngày)** |
| **Trang chi tiết sản phẩm** | **14 giờ (~2 ngày)** |
| **Trang phụ trợ** | **8 giờ (~1 ngày)** |
| **Tối ưu & Kiểm thử** | **15 giờ (~2 ngày)** |

## ****Tuần 6: Hoàn thiện & Cải thiện hiệu suất****

### ****6.1 Tối ưu hiệu suất API****

Cải thiện truy vấn PostgreSQL bằng **indexing, caching**.

Nén dữ liệu JSON trả về từ API.

### ****6.2 Cải thiện giao diện****

Tối ưu UI cho mobile.

Cải thiện tốc độ tải trang.

## ****Tuần 7: Kiểm thử & Fix bug****

### ****7.1 Kiểm thử dữ liệu****

Đảm bảo scraper lấy đúng dữ liệu.

Kiểm tra tránh dữ liệu trùng lặp.

### ****7.2 Kiểm tra bảo mật****

Chống **SQL Injection, XSS, CORS**.

Kiểm thử API với Postman.

## ****Tuần 8: Triển khai & Viết tài liệu****

### ****8.1 Triển khai lên VPS****

Chọn nền tảng deploy (AWS, DigitalOcean, Railway…).

Cấu hình server (Nginx, Gunicorn).

Tạo domain hoặc dùng IP server.

### ****8.2 Viết tài liệu****

Hướng dẫn cài đặt & sử dụng.

Hướng dẫn quản lý dữ liệu & API.

## ****Tuần 9-10: Báo cáo & Đánh giá****

### ****9.1 Chạy thử nghiệm thực tế****

Xem hiệu suất khi có nhiều dữ liệu.

Cải tiến nếu cần.

### ****10.1 Đánh giá & Hoàn thiện****

Tổng hợp feedback.

## ****Tóm tắt nhiệm vụ theo tuần****

| **Tuần** | **Công việc** |
| --- | --- |
| **1** | Lập kế hoạch, chọn công nghệ, tạo GitLab |
| **2** | Thiết kế database, cài đặt môi trường |
| **3** | Viết scraper, thu thập dữ liệu tự động |
| **4** | Xây dựng API backend |
| **5** | Phát triển frontend, hiển thị dữ liệu |
| **6** | Tối ưu hiệu suất API & giao diện |
| **7** | Kiểm thử & fix bug |
| **8** | Triển khai, viết tài liệu |
| **9-10** | Chạy thử nghiệm, báo cáo, đánh giá |

Với kế hoạch này, bạn có **lộ trình chi tiết, rõ ràng**, giúp hoàn thành dự án một cách hiệu quả nhất khi làm một mình.