**NOTE:** *Sau đây là các công nghệ phổ biến được sử dụng để phát triển website , tuy nhiên đây chỉ mới mang tính liệt kê và có mang tính khách quan trong quá trình có thể thay đổi do một số ngôn ngữ chưa được tiếp cận hoặc hiểu sâu nên trong quá trình làm sẻ ưu tiên làm cách dễ tiếp cận nhất và sẻ dần cải thiện sau này:*

**DoctorHub** là một nền tảng liên quan đến dịch vụ y tế, có các chức năng chính như:

* **Trang chủ**
* **Đặt lịch khám**
* **Tư vấn sức khỏe**
* **Dịch vụ y tế**
* **Liên hệ**
* **Blog**
* **Bài test**

**Các thành phần cần thiết cho hệ thống:**

Để xây dựng một nền tảng tương tự, hệ thống cần có:

1. **Giao diện người dùng (Frontend)**
   * Hiển thị thông tin dịch vụ, đặt lịch khám, blog, tư vấn sức khỏe...
   * Tương tác với người dùng qua biểu mẫu và chatbot.(đang tìm hiểu)
2. **Hệ thống quản lý đặt lịch khám**
   * Quản lý lịch hẹn của bệnh nhân với bác sĩ.
   * Tích hợp nhắc lịch qua email hoặc SMS.
3. **Tư vấn sức khỏe trực tuyến**
   * Hỗ trợ tư vấn qua chatbot hoặc video call.(zoom,,…)
   * Lưu trữ lịch sử tư vấn để theo dõi tình trạng sức khỏe.
4. **Hệ thống quản lý bài viết (Blog, Bài Test)**
   * Hệ thống để quản lý nội dung.(các bài post sức khỏe,video,..)
   * Bài kiểm tra sức khỏe trực tuyến.(test nhanh)
5. **Quản lý dịch vụ y tế**
   * Danh mục các dịch vụ y tế có sẵn.
   * Đặt lịch hoặc đăng ký sử dụng dịch vụ.
   * Mua một số loại thuốc phổ biến
6. **Hệ thống quản trị (Admin Dashboard)**
   * Quản lý người dùng (bệnh nhân, bác sĩ).
   * Quản lý nội dung, bài viết, dịch vụ.

**Đề xuất công nghệ sử dụng:**

**Frontend (Giao diện người dùng)**

* **Lựa chọn 1 trong số:**
* **React.js hoặc Next.js**: Hiệu suất cao, tối ưu SEO.
* Html,Css,…
* **Tailwind CSS hoặc Bootstrap**: Tạo giao diện đẹp và dễ tùy chỉnh.

**Backend (Xử lý dữ liệu và logic nghiệp vụ)**

* **Lựa chọn 1 trong số:**
* **Node.js với Express.js** hoặc **Django với Python**: Xử lý API nhanh chóng.
* **GraphQL hoặc REST API**: Giao tiếp giữa frontend và backend.
* **Json-server, Javascript**

**Cơ sở dữ liệu**

* **Lựa chọn 1 trong số:**
* **PostgreSQL hoặc MongoDB**: Lưu trữ thông tin bệnh nhân, lịch hẹn.
* **Redis**: Cache dữ liệu giúp tăng tốc độ phản hồi.

**Realtime Communication**

* **Lựa chọn 1 trong số:**
* **WebSocket với Socket.io**: Hỗ trợ tư vấn trực tuyến.
* **Twilio hoặc Zoom API**: Hỗ trợ gọi video giữa bác sĩ và bệnh nhân.

**Bảo mật**

* **Lựa chọn 1 trong số:**
* **JWT (JSON Web Token)**: Xác thực người dùng.
* **OAuth 2.0**: Đăng nhập bằng Google/Facebook.
* **Mã hóa dữ liệu (AES, SSL/TLS)**: Đảm bảo an toàn thông tin y tế.

**Triển khai & Lưu trữ**

* **Lựa chọn 1 trong số:**
* **AWS / Google Cloud / Azure**: Đám mây, đảm bảo tính sẵn sàng cao.
* **Docker + Kubernetes**: Tự động mở rộng hệ thống khi cần thiết.
* **Github**
* **Vercel**

**Tối ưu hóa dữ liệu hình ảnh/video**

* **Lựa chọn 1 trong số:**
* **Lazyload cho hình ảnh**
* **Sử dụng các chuẩn webp,jpeg,svg,mp4,webM,…**
* **Sử dụng các thư viện built sẵn như lazySizes**
* **Giảm dung lượng ảnh ,video:**
* [Squoosh](https://squoosh.app/) (JPEG, PNG, WebP, AVIF)
* [TinyPNG](https://tinypng.com/) (JPG, PNG, WebP)
* **Google PageSpeed Insights** (pagespeed.web.dev)
* **GTmetrix** ([gtmetrix.com](https://gtmetrix.com/))
* **Lighthouse (Chrome DevTools)**