

CÔNG NGHỆ PHẦN MỀM

PHÂN TÍCH, ĐẶC TẢ, THIẾT KẾ VÀ PHÁT TRIỂN ỨNG DỤNG QUẢN LÝ BÁN SÁCH

Thành viên nhóm	Mã sinh viên
Chu Mạnh Hiếu	20213684
Bùi văn Giang	20213898
Hồ Thị Quỳnh Như	20213829

Các tools hỗ trợ

1. Firebase

Giới thiệu:

Firebase là một nền tảng phát triển ứng dụng của Google, cung cấp một loạt các công cụ và dịch vụ giúp các nhà phát triển tạo, quản lý và triển khai các ứng dụng di động và web. Firebase hỗ trợ backend-as-a-service (BaaS), giúp các nhà phát triển tập trung vào việc xây dựng giao diện mà không phải lo lắng về phần backend.

Cách dùng:

- **Realtime Database:** Firebase cung cấp cơ sở dữ liệu thời gian thực, nơi dữ liệu được đồng bộ hóa ngay lập tức giữa các client. Nhà phát triển có thể lưu trữ và nhận dữ liệu dễ dàng.
- **Firestore:** Một cơ sở dữ liệu đám mây khác của Firebase, cung cấp khả năng truy vấn mạnh mẽ, tích hợp tốt với các công cụ khác của Google.
- **Authentication:** Firebase hỗ trợ các phương pháp xác thực đơn giản như email/password, Google, Facebook, Twitter, v.v.
- **Firebase Hosting:** Cung cấp dịch vụ hosting cho các ứng dụng web với khả năng phân phối nội dung nhanh chóng qua CDN.

- **Cloud Messaging:** Hỗ trợ gửi thông báo đẩy (push notification) tới người dùng.
- **Crashlytics:** Công cụ giám sát sự cố của ứng dụng để phát hiện và sửa lỗi nhanh chóng.

Ưu điểm:

- **Tích hợp toàn diện:** Firebase cung cấp đầy đủ các công cụ từ cơ sở dữ liệu, xác thực, hosting, phân tích dữ liệu cho đến thông báo đẩy.
- **Thời gian thực:** Dữ liệu được đồng bộ hóa giữa client và server ngay lập tức, hỗ trợ các ứng dụng có tính tương tác cao.
- **Dễ sử dụng:** Firebase cung cấp SDK dễ cài đặt và tài liệu phong phú, giúp các nhà phát triển dễ dàng triển khai các dịch vụ.
- **Tích hợp mạnh mẽ với Google:** Firebase tích hợp tốt với các dịch vụ của Google như Analytics, Cloud Functions.

Nhược điểm:

- **Giới hạn tùy biến:** Firebase có một số giới hạn trong khả năng tùy biến và kiểm soát so với khi tự xây dựng backend riêng.
- **Chi phí:** Khi ứng dụng mở rộng và yêu cầu nhiều tính năng phức tạp, chi phí sử dụng Firebase có thể tăng đáng kể.
- **Khả năng di chuyển:** Khi đã sử dụng Firebase, việc di chuyển sang nền tảng khác có thể gặp khó khăn do sự phụ thuộc sâu vào hệ sinh thái của Firebase.

2. ChatGPT

Giới thiệu:

ChatGPT là một mô hình ngôn ngữ lớn phát triển bởi OpenAI, được huấn luyện trên một lượng lớn dữ liệu văn bản để hiểu và tạo ra ngôn ngữ tự nhiên. ChatGPT có thể thực hiện các tác vụ liên quan đến ngôn ngữ như trả lời câu hỏi, tạo nội dung, phân tích và tư vấn dựa trên các chủ đề khác nhau.

Cách dùng:

- **Giao tiếp:** ChatGPT có thể được sử dụng để trò chuyện với người dùng trong các ứng dụng chatbot, trợ lý ảo, hỗ trợ khách hàng tự động.
- **Sáng tạo nội dung:** ChatGPT có thể giúp viết nội dung, mô tả sản phẩm, tạo kịch bản, v.v.
- **Phân tích:** ChatGPT có thể phân tích văn bản, tóm tắt tài liệu, trả lời các câu hỏi liên quan đến nội dung kỹ thuật hoặc sáng tạo.
- **Tích hợp API:** OpenAI cung cấp API để các nhà phát triển tích hợp ChatGPT vào các ứng dụng của họ, từ đó hỗ trợ các nhiệm vụ tự động hóa ngôn ngữ.

Ưu điểm:

- **Hiểu ngôn ngữ tự nhiên:** ChatGPT có khả năng hiểu và phản hồi rất tự nhiên, gần gũi với con người.
- **Ứng dụng đa dạng:** ChatGPT có thể sử dụng cho nhiều mục đích như dịch vụ khách hàng, giáo dục, sáng tạo nội dung.
- **Dễ tích hợp:** Với API đơn giản, ChatGPT có thể dễ dàng tích hợp vào các ứng dụng khác để thực hiện các nhiệm vụ tự động hóa.
- **Luôn cập nhật:** Mô hình liên tục được cải tiến, giúp đáp ứng ngày càng tốt hơn các nhu cầu ngôn ngữ của người dùng.

Nhược điểm:

- **Hiểu biết có giới hạn:** ChatGPT không có khả năng truy cập thông tin thời gian thực (nếu không được kết nối với cơ sở dữ liệu hoặc API trực tiếp), và dữ liệu được huấn luyện có thể cũ.
- **Chi phí:** Sử dụng API ChatGPT có thể tốn kém đối với các ứng dụng yêu cầu xử lý ngôn ngữ ở quy mô lớn.
- **Hạn chế trong ngữ cảnh phức tạp:** Mặc dù ChatGPT hoạt động tốt trong các cuộc hội thoại ngắn hoặc đơn giản, nhưng trong các ngữ cảnh phức tạp và dài, mô hình có thể bị mất ngữ cảnh hoặc phản hồi không chính xác.

3. GitHub

Giới thiệu:

GitHub là một nền tảng quản lý mã nguồn phân tán dựa trên Git, cho phép các nhà phát triển lưu trữ, chia sẻ và quản lý mã nguồn một cách hiệu quả. GitHub hỗ trợ cộng tác, theo dõi thay đổi, và phân nhánh trong quá trình phát triển phần mềm.

Cách dùng:

- **Quản lý mã nguồn:** Lưu trữ và quản lý mã nguồn cho các dự án phần mềm, hỗ trợ theo dõi lịch sử thay đổi và hợp tác giữa các thành viên trong nhóm.
- **Phân nhánh và hợp nhất:** Sử dụng **branch** để phát triển các tính năng mới, sửa lỗi hoặc làm việc song song mà không ảnh hưởng đến mã nguồn chính.
- **Pull Request:** Yêu cầu kiểm tra và hợp nhất mã giữa các nhánh, giúp cải thiện chất lượng mã và đảm bảo sự ổn định của dự án.
- **Issues và Projects:** Sử dụng hệ thống Issues để theo dõi lỗi, yêu cầu tính năng, hoặc tổ chức công việc với Project Boards.

Ưu điểm:

- **Cộng tác mạnh mẽ:** GitHub giúp nhiều lập trình viên làm việc đồng thời trên cùng một dự án, theo dõi và kiểm tra mã của nhau thông qua các pull request.
- **Quản lý phiên bản tốt:** Lịch sử thay đổi mã nguồn được theo dõi chặt chẽ, giúp kiểm soát phiên bản và cho phép quay lại phiên bản trước nếu cần.
- **Tích hợp CI/CD:** GitHub dễ dàng tích hợp với các công cụ CI/CD như GitHub Actions, Jenkins, Travis CI để tự động hóa việc kiểm tra và triển khai mã nguồn.
- **Hệ sinh thái mở:** GitHub hỗ trợ lưu trữ các dự án mã nguồn mở và có một cộng đồng lớn giúp phát triển dự án nhanh hơn.

Nhược điểm:

- **Khả năng sử dụng hạn chế khi không có mạng:** Mặc dù Git có thể hoạt động cục bộ, các tính năng cộng tác của GitHub yêu cầu kết nối internet.
- **Độ phức tạp khi làm việc với dự án lớn:** Khi các dự án lớn, việc quản lý mã nguồn và kiểm tra pull request có thể trở nên phức tạp và tốn thời gian hơn.
- **Phụ thuộc vào Git:** Các nhà phát triển cần phải hiểu về Git để làm việc hiệu quả với GitHub, điều này có thể gây khó khăn cho những người mới.