**Tổng quan về APIs (Application Programming Interface)**

**Định nghĩa API (Application Programming Interface)**

API là một **bộ các quy tắc** cho phép các ứng dụng khác nhau giao tiếp và trao đổi dữ liệu với nhau. Nó cung cấp một cách tiêu chuẩn để một phần mềm có thể gửi yêu cầu và nhận phản hồi từ phần mềm khác mà không cần biết chi tiết bên trong của hệ thống đó.

**Mục đích và vai trò của API trong hệ thống phần mềm**

API giúp kết nối các ứng dụng, cho phép chúng tương tác một cách linh hoạt và hiệu quả. Nó giúp phát triển phần mềm nhanh hơn bằng cách tận dụng các dịch vụ có sẵn thay vì phải xây dựng lại từ đầu. Ngoài ra, API còn giúp hệ thống mở rộng dễ dàng và hỗ trợ tích hợp với nhiều nền tảng khác nhau.

**Ví dụ:**

Hãy tưởng tượng bạn đang ở một nhà hàng và muốn gọi món ăn. Bạn là khách hàng, nhà bếp là hệ thống cung cấp món ăn, còn người bồi bàn là API. Bạn không cần biết nhà bếp chuẩn bị món như thế nào, bạn chỉ cần yêu cầu món ăn từ thực đơn thông qua bồi bàn (API). Bồi bàn sẽ chuyển yêu cầu của bạn đến nhà bếp, nơi đầu bếp chuẩn bị món ăn. Khi món ăn sẵn sàng, bồi bàn sẽ mang nó đến cho bạn.

Tương tự, trong thế giới công nghệ, API hoạt động như bồi bàn, giúp truyền tải yêu cầu từ người dùng đến hệ thống xử lý dữ liệu, rồi mang kết quả về cho người dùng mà không cần họ biết hệ thống hoạt động ra sao.

A diagram of a waiter

AI-generated content may be incorrect.

Một ví dụ Minh họa khả năng mở rộng của API

Để hiểu rõ cách API giúp cung cấp dữ liệu và dịch vụ một cách linh hoạt, hãy xem xét hai phương thức đặt vé máy bay.

**Phương thức thứ nhất:** Người dùng truy cập trực tiếp vào trang web của một hãng hàng không cụ thể để đặt vé. Tại đây, họ nhập thông tin chuyến bay mong muốn, và hệ thống của hãng hàng không sẽ truy vấn cơ sở dữ liệu nội bộ để cung cấp các tùy chọn phù hợp. Quá trình này chỉ giới hạn trong phạm vi dữ liệu của hãng hàng không đó.

A blue diagram with text and arrows

AI-generated content may be incorrect.

**Phương thức thứ hai**: Người dùng sử dụng một nền tảng đặt vé trung gian, chẳng hạn như một trang web dịch vụ du lịch. Thay vì chỉ lấy dữ liệu từ một hãng hàng không duy nhất, nền tảng này có thể truy vấn thông tin từ nhiều hãng hàng không khác nhau thông qua các API mà các hãng cung cấp. Khi người dùng nhập yêu cầu đặt vé, hệ thống sẽ sử dụng API để gửi yêu cầu đến nhiều hãng hàng không, thu thập dữ liệu về giá cả, chỗ ngồi và hành trình, sau đó tổng hợp và hiển thị trên một giao diện duy nhất.

A diagram of a travel service

AI-generated content may be incorrect.

Nhờ API, hệ thống đặt vé trung gian có thể dễ dàng mở rộng để tích hợp nhiều hãng hàng không mà không cần xây dựng và duy trì cơ sở dữ liệu riêng. Điều này không chỉ giúp người dùng tiếp cận nhiều tùy chọn hơn mà còn cho phép các doanh nghiệp tối ưu hóa dịch vụ của mình mà không cần can thiệp vào hệ thống nội bộ của từng hãng hàng không.

**Phân loại API**

* **API mở (Open API / Public API)**: API mở là API công khai, có thể được sử dụng bởi bất kỳ ai. Thường yêu cầu xác thực hoặc ủy quyền cơ bản và có thể miễn phí hoặc tính phí. API này cho phép bên thứ ba tích hợp dịch vụ vào ứng dụng của họ mà không cần quyền truy cập đặc biệt. Mặc dù công khai, API mở thường có giới hạn truy vấn hoặc quyền truy cập để bảo vệ tài nguyên.
* **API đối tác (Partner API)**: API đối tác chỉ dành cho các đối tác kinh doanh và yêu cầu quyền truy cập hoặc giấy phép cụ thể. API này giúp các tổ chức chia sẻ dữ liệu và dịch vụ với nhau một cách có kiểm soát. So với API mở, API đối tác yêu cầu xác thực nghiêm ngặt hơn và thường đi kèm với các điều khoản hợp tác cụ thể.
* **API nội bộ (Internal API / Private API)**: API nội bộ được sử dụng trong phạm vi tổ chức, không dành cho bên thứ ba, giúp kết nối và đồng bộ hệ thống nội bộ. API này giúp doanh nghiệp tối ưu hóa quy trình làm việc, quản lý dữ liệu nội bộ và tự động hóa các tác vụ mà không cần công khai tài liệu API.
* **API tổng hợp (Composite API)**: API tổng hợp kết hợp nhiều API thành một để xử lý yêu cầu phức tạp trong một lần gọi. API này giúp giảm độ trễ, tối ưu hóa hiệu suất và cải thiện trải nghiệm người dùng bằng cách kết hợp nhiều nguồn dữ liệu hoặc thực hiện nhiều thao tác chỉ trong một lần yêu cầu.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Phân loại API | Phạm vi truy cập | Đối tượng sử dụng | Xác thực & Uỷ quyền | Mục đích |
| API mở (Public API) | Công khai | Bất kỳ ai | Cơ bản, có thể miễn phí hoặc trả phí | Cho phép bên thứ ba tích hợp dịch vụ |
| API đối tác (Partner API) | Hạn chế | Đối tác kinh doanh | Nghiêm ngặt, cần cấp quyền | Hỗ trợ hợp tác giữa các tổ chức |
| API nội bộ (Private API) | Chỉ nội bộ | Nhân viên, hệ thống nội bộ | Không cần cấp quyền bên ngoài | Quản lý dữ liệu và tối ưu hóa quy trình nội bộ |
| API tổng hợp (Composite API) | Phụ thuộc vào API thành | Bên thứ ba hoặc nội bộ | Tùy vào API thành phần | Tối ưu hóa hiệu suất khi xử lý nhiều yêu cầu cùng |

**Các loại Web Service API**

Web service là dịch vụ hoạt động trên Internet, cho phép các hệ thống trao đổi dữ liệu với nhau thông qua World Wide Web. Có bốn loại Web Service API chính:

1. **SOAP (Simple Object Access Protocol)**  
   SOAP là một giao thức sử dụng XML để truyền dữ liệu. Chức năng chính của SOAP là xác định cấu trúc của thông điệp và phương thức giao tiếp giữa các ứng dụng. SOAP tuân theo các tiêu chuẩn nghiêm ngặt và thường được sử dụng trong các hệ thống yêu cầu độ bảo mật cao.
2. **REST (Representational State Transfer)**  
   REST không phải là một giao thức mà là một tập hợp các nguyên tắc kiến trúc cho việc thiết kế API. REST có các đặc điểm như:
   * Giao diện đơn giản và dễ sử dụng.
   * Truy cập tài nguyên qua các phương thức HTTP phổ biến (GET, POST, PUT, DELETE).
   * Hỗ trợ định dạng dữ liệu linh hoạt như JSON, XML.

**So sánh SOAP và REST**

|  |  |  |
| --- | --- | --- |
| Tiêu chí | SOAP | REST |
| Quy tắc | Tuân theo các tiêu chuẩn nghiêm ngặt, có bảo mật tiên tiến | Ít ràng buộc hơn, linh hoạt cho nhà phát triển |
| Cách hoạt động | Điều khiển dựa trên hàm (function-based) | Điều khiển dựa trên dữ liệu (data-driven) |
| Băng thông | Yêu cầu nhiều băng thông | Tiêu tốn băng thông tối thiểu |
| Định dạng dữ liệu | Chỉ hỗ trợ XML | Hỗ trợ nhiều định dạng như JSON, XML, HTML |

1. **XML-RPC (eXtensible Markup Language - Remote Procedure Call)**  
   XML-RPC là giao thức gọi thủ tục từ xa (RPC) sử dụng XML để truyền dữ liệu. So với SOAP, XML-RPC đơn giản hơn và tiêu tốn ít băng thông hơn, giúp cải thiện hiệu suất giao tiếp giữa các hệ thống.
2. **JSON-RPC (JavaScript Object Notation - Remote Procedure Call)**  
   JSON-RPC hoạt động tương tự XML-RPC nhưng thay vì sử dụng XML, nó sử dụng JSON để truyền dữ liệu. Điều này giúp JSON-RPC nhẹ hơn, nhanh hơn và phù hợp hơn với các ứng dụng web hiện đại.

|  |  |  |
| --- | --- | --- |
| Tiêu chí | JSON-RPC | XML-RPC |
| Loại dữ liệu hỗ trợ | Chỉ hỗ trợ text và numbers | Hỗ trợ nhiều loại dữ liệu như text, numbers, images, charts, v.v. |
| Mục tiêu chính | Tập trung chủ yếu vào dữ liệu (Data) | Tập trung vào tài liệu (Document) |
| Bảo mật | Bảo mật thấp hơn do JSON ít có cơ chế kiểm soát dữ liệu | Bảo mật tốt hơn do XML có thể tích hợp với các cơ chế xác thực nâng cao |
| Hiệu suất | Nhanh hơn do JSON nhẹ hơn | Chậm hơn do XML có cấu trúc phức tạp hơn |
| Băng thông | Tiêu tốn ít băng thông do dữ liệu nhỏ gọn | Yêu cầu nhiều băng thông hơn do cú pháp XML dài |
| Ứng dụng phổ biến | API web hiện đại, đặc biệt là các ứng dụng JavaScript | Các hệ thống yêu cầu xử lý tài liệu hoặc tích hợp với SOAP |

**So sánh bốn loại Web Service API**

A close-up of a list

AI-generated content may be incorrect.

Tác giả: Nguyễn Hữu Lộc- PKT VNPRO