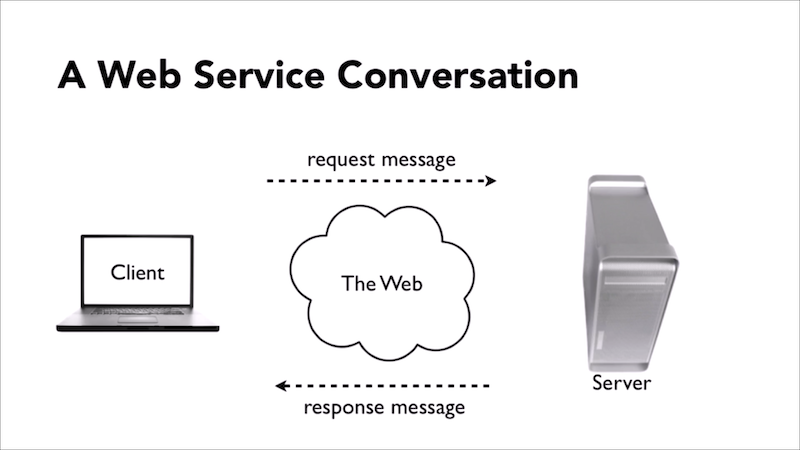
**TRUYỀN THÔNG VỚI WEB SERVICE – FIREBASE**

1. **TRUYỀN THÔNG VỚI WEB SERVICE:**
2. **Web Service là gì?**

* Web Service là tập hợp các giao thức và tiêu chuẩn mở được sử dụng để trao đổi dữ liệu giữa các ứng dụng hoặc giữa các hệ thống.
* Nó là phương thức giao tiếp giữa hai thiết bị qua mạng hoặc giữa các thành phần ứng dụng để giao tiếp.
* Các ứng dụng phần mềm viết bằng các ngôn ngữ khác nhau và chạy trên các nền tảng khác nhau có thể sử dụng Web Service để trao đổi dữ liệu qua mạng máy tính.
* Web Service hoạt động độc lập không phụ thuộc vào bất kỳ ngôn ngữ nào, cho phép các ứng dụng Java, .NET, PHP và các ngôn ngữ khác giao tiếp với nhau thông qua Web Service.



1. **Các loại Web Service phổ biến:**

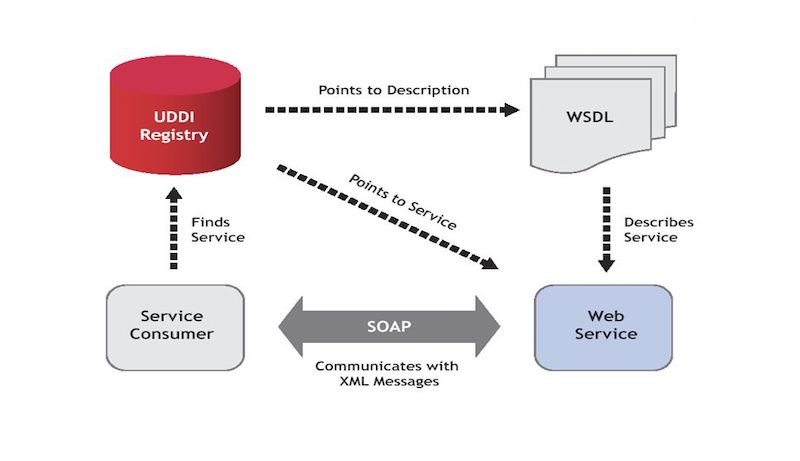
Có một số loại Web Service phổ biến bao gồm:

* XML-RPC (Gọi thủ tục từ xa) là giao thức XML cơ bản nhất để trao đổi dữ liệu giữa nhiều loại thiết bị khác nhau. Nó sử dụng HTTP để truyền dữ liệu và truyền thông tin khác một cách nhanh chóng và dễ dàng từ máy khách đến máy chủ.
* UDDI (Mô tả chung, Khám phá và Tích hợp) là một tiêu chuẩn dựa trên XML để mô tả chi tiết, xuất bản và khám phá các Web Service. Về cơ bản, đây là cơ quan đăng ký internet cho các doanh nghiệp trên khắp thế giới. Mục tiêu là hợp lý hóa các giao dịch kỹ thuật số và thương mại điện tử giữa các hệ thống của công ty.
* SOAP là một giao thức Web Service dựa trên XML để trao đổi dữ liệu và tài liệu qua HTTP hoặc SMTP (Giao thức truyền thư đơn giản). Nó cho phép các quy trình độc lập hoạt động trên các hệ thống khác nhau giao tiếp bằng XML.
* REST cung cấp khả năng giao tiếp và kết nối giữa các thiết bị và internet cho các tác vụ dựa trên API. Hầu hết các dịch vụ RESTful sử dụng HTTP làm giao thức hỗ trợ.

1. **Những thành phần của Web Service:**

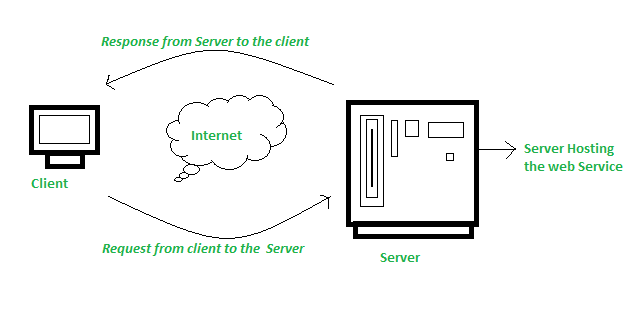
Để tạo thành một web service hoàn chỉnh cần 3 thành phần chính gồm: Simple Object Access Protocol, Web Services Description Language và Universal Description, Discovery & Integration. Các thành phần này cụ thể như sau:

* SOAP (Simple Object Access Protocol): Là giao thức dựa trên XML, giúp các ứng dụng có thể trao đổi với nhau thông qua giao thức HTTP hoặc HTTPs. SOAP chính là giao thức truy cập đối tượng đơn giản.
* UDDI (Universal Description, Discovery and Integration): Là tiêu chuẩn dựa trên XML để mô tả, xuất bản và tìm kiếm các dịch vụ web.
* WSDL (Web Services Description Language): Là ngôn ngữ dựa trên XML để mô tả các dịch vụ web và cách truy cập chúng.



1. **Web Service hoạt động như thế nào?**

Sơ đồ dưới đây mô tả một phiên bản rất đơn giản về cách một Web Service hoạt động. Máy khách sẽ yêu cầu gửi một chuỗi các cuộc gọi dịch vụ web đến một máy chủ lưu trữ dịch vụ web thực tế.



Các cuộc gọi thủ tục từ xa (Remote procedure calls) là những gì được sử dụng để thực hiện các yêu cầu này. Các cuộc gọi đến các phương thức được lưu trữ bởi Web Service có liên quan được gọi là Các cuộc gọi thủ tục từ xa (RPC). Ví dụ: Flipkart cung cấp dịch vụ web hiển thị giá cho các mặt hàng được cung cấp trên Flipkart.com. Giao diện người dùng hoặc lớp trình bày có thể được viết bằng .Net hoặc Java, nhưng Web Service có thể được giao tiếp bằng một trong hai ngôn ngữ lập trình.

Dữ liệu được trao đổi giữa máy khách và máy chủ, là XML, và là phần quan trọng nhất của thiết kế Web Service. XML (Ngôn ngữ đánh dấu mở rộng) là một ngôn ngữ trung gian đơn giản được hiểu bởi nhiều ngôn ngữ lập trình khác nhau. Nó là một bản sao của HTML. Khi các chương trình giao tiếp với nhau, chúng sử dụng XML. Điều này tạo ra một nền tảng chung cho các ứng dụng được viết bằng các ngôn ngữ lập trình khác nhau để giao tiếp với nhau.

Để truyền dữ liệu XML giữa các ứng dụng, các dịch vụ web sử dụng SOAP (Giao thức truy cập đối tượng đơn giản). Dữ liệu được gửi bằng HTTP tiêu chuẩn. Thông báo SOAP là dữ liệu được gửi từ Web Service đến ứng dụng. Một tài liệu XML là tất cả những gì chứa trong một thông báo SOAP.

1. **Ưu điểm và hạn chế của Web Service:**
2. *Ưu điểm:*

- Là công cụ trung gian hỗ trợ việc giao tiếp, liên lạc giữa các bộ phận diễn ra thuận lợi và dễ dàng hơn.

- Giúp các ứng dụng trên các nền tảng có thể trao đổi thông tin dữ liệu với nhau.

- Web service giúp giải quyết tất cả các vấn để về tương tác, mỗi ứng dụng sẽ được web service cung cấp một cách thức để liên hệ, trao đổi và kết nối các dữ liệu lại với nhau.

- Web service cũng cho phép các ứng dụng giao tiếp cũng như chia sẻ thông tin, các dịch vụ được chia sẻ.

1. *Hạn chế:*

- Vấn đề bảo mật của web service vẫn chưa được khắc phục triệt để, người sử dụng cần có những biện pháp phòng tránh những tình huống xấu nhất.

- Sử dụng web service, lập trình viên bắt buộc phải ghi nhớ rất nhiều những tiêu chuẩn mở. Đây vừa là ưu điểm vừa là hạn chế của web service khi sử dụng giao thức và mở chuẩn đơn giản.

- Sử dụng web service bắt buộc người dùng phải thường xuyên cập nhật và nâng cấp hệ thống liên tục. Nếu bạn vô tình quên không cập nhật, hệ thống của bạn sẽ thường xuyên gặp lỗi, mất nhiều thời gian chỉnh sửa.

1. **Lợi ích của Web Service:**

* Tăng tương tác giữa các ứng dụng:Web service cho phép các ứng dụng có thể liên hệ và trao đổi thông tin dữ liệu với nhau, ngay cả khi những ứng dụng này không được phát triển trên cùng một nền tảng, cùng một ngôn ngữ.
* Giao thức chuẩn hóa: Như đã nói trong phần định nghĩa web service là gì phía trên, web service giao tiếp bằng cách sử dụng các giao thức chuẩn hóa. Tất cả các lớp của web service đều tách biệt trong stack giao thức của web service. Điều này sẽ giúp doanh nghiệp có nhiều cơ hội lựa chọn, tối ưu chi phí cũng như gia tăng chất lượng.
* Tối ưu hoá chi phí truyền thông: Sử dụng web service có thể triển khai các dịch vụ web với đường truyền khiêm tốn. Điều này sẽ giúp doanh nghiệp tiết kiệm được không ít chi phí so với việc sử dụng các giải pháp khác như EDI hay B2B
* Hiển thị các chức năng hiện có trên mạng: Web service là một đơn vị mã có thể gọi từ xa thông qua giao thức HTTP. Web service sẽ được kích hoạt bằng cách sử dụng những yêu cầu HTTP. Từ đó, các web service sẽ cho phép bạn hiển thị các chức năng của mã hiện có qua mạng. Khi nó được hiển thị trên mạng, ứng dụng khác có thể sử dụng chức năng của chương trình.

1. **So sánh Web Service và API:**

Hầu hết các Web Service đều cung cấp một API, với tập hợp các lệnh và chức năng của nó, được sử dụng để truy xuất dữ liệu. Đây là một ví dụ: Twitter cung cấp API cho phép nhà phát triển truy cập các tweet từ máy chủ, sau đó thu thập dữ liệu ở định dạng JSON. Nhưng: Tất cả các Web Service đều có thể là API, nhưng không phải tất cả các API đều có thể là Web Service.

Điểm tương đồng giữa API và Web Service:

* Cả hai đều được truy cập thông qua HTTP/HTTPS để cho phép giao tiếp giữa nhà cung cấp dịch vụ và khách hàng.
* Cả hai đều gọi một hàm, xử lý dữ liệu và nhận phản hồi.

Sự khác biệt giữa API và Web Service:

* API có thể được lưu trữ trong ứng dụng hoặc IIS (Dịch vụ thông tin Internet), nhưng Web Service chỉ có thể được lưu trữ trên IIS.
* Web Service không phải là nguồn mở và được sử dụng để hiểu JSON (Ký hiệu đối tượng JavaScript) hoặc XML, trong khi API là nguồn mở và chỉ được sử dụng cho XML.
* API là một kiến trúc nhẹ (tốt nhất cho các thiết bị có băng thông hạn chế (ví dụ: điện thoại thông minh). Các Web Service không phải là kiến trúc nhẹ vì chúng yêu cầu SOAP để gửi và nhận dữ liệu mạng.
* API có thể sử dụng bất kỳ hình thức giao tiếp nào, nhưng Web Service chỉ sử dụng SOAP, REST và XML-RPC.
* API hỗ trợ URL, tiêu đề yêu cầu/phản hồi, bộ nhớ đệm, tạo phiên bản, định dạng nội dung. Các Web Service chỉ hỗ trợ HTTP.

1. **Ứng dụng của Web Service:**

Các Web Service được sử dụng cho nhiều ứng dụng, nhưng phổ biến nhất là sử dụng lại mã và kết nối các chương trình hiện có.

Phương pháp Web Service có thể giúp các nhà phát triển phân đoạn các ứng dụng thành các thành phần có thể được sử dụng và tái sử dụng cho các nhu cầu khác nhau. Ví dụ: nhiều chương trình có thể cần một công cụ chuyển đổi hoặc chức năng báo cáo. Điều này có thể thực hiện được nhờ các giao thức truyền thông phổ biến của Web Service.

Ngoài ra, các Web Service có thể được sử dụng để liên kết dữ liệu giữa các nền tảng khác nhau. Vì tất cả các ứng dụng có thể bao gồm một thành phần Web Service, điều này có thể biến bất kỳ chương trình cụ thể thành một chương trình có thể tương tác hoàn toàn.

**Bổ sung:** [Giới thiệu Web service – SOAP, WSDL và ASP.NET Web Service cơ bản | TopDev](https://topdev.vn/blog/gioi-thieu-web-service-soap-wsdl-va-asp-net-web-service-co-ban/)

1. **FIREBASE:**
2. **Firebase là gì?**

Firebase là nền tảng được xây dựng, cung cấp bởi Google, hoạt động trên nền tảng Cloud nhằm hỗ trợ phát triển các ứng dụng trên web và di động dựa trên nguyên tắc đơn giản hóa thao tác lập trình. Các CSDL có sẵn mang đến khả năng code nhanh chóng, thuận tiện hơn.

Dưới sự hỗ trợ của nền tảng này, lập trình viên hoàn toàn không cần để tâm đến backend hay server mà có cơ hội tập trung chủ yếu vào việc phát triển ứng dụng. Nền tảng này hướng đến đối tượng chủ yếu là di động, được biết đến như một backend as an server.

Below are some reputable organizations that rely on a firebase backend for its functioning:

The New York Times

Alibaba.com

Gameloft

Duolingo

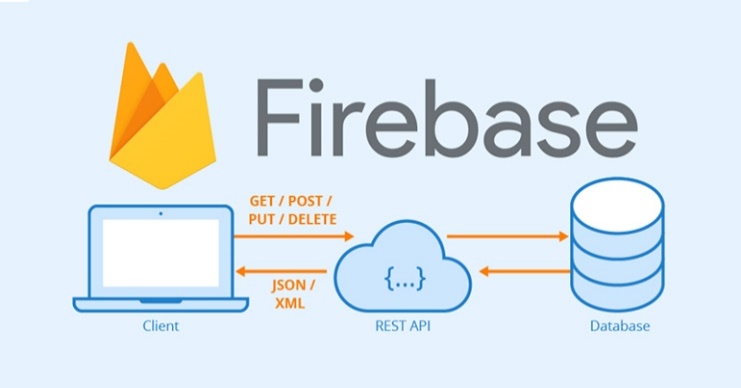
Trivago

Venmo

Lyft

A diagram of a server

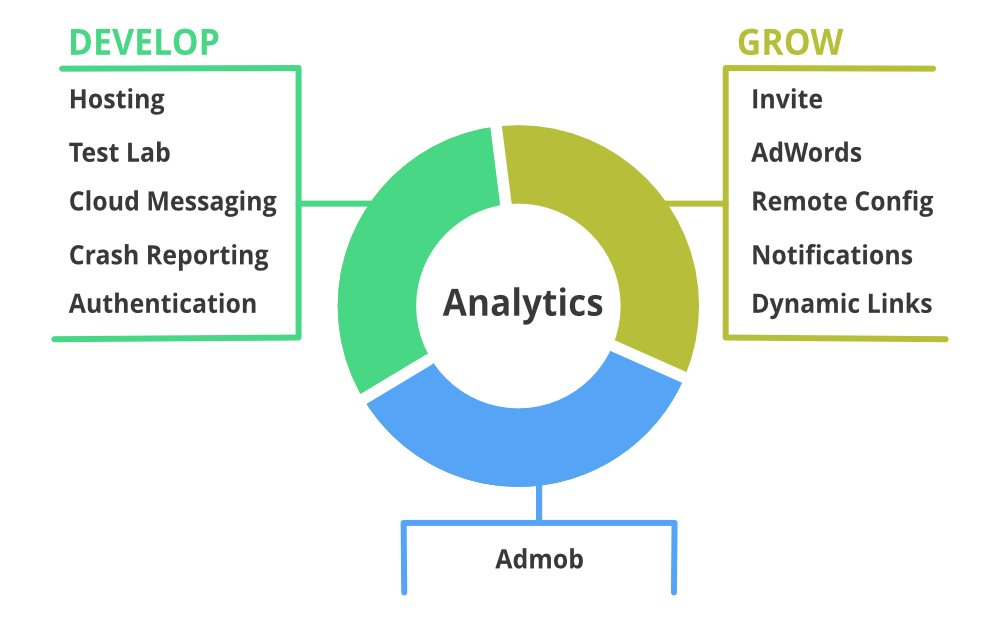
Description automatically generated



Lịch sử hình thành:

* Vào năm 2011, James Tamplin và Andrew Lee đã cho ra mắt Evolve. Đây là một nền tảng có cấu trúc khá đơn giản chuyên cung cấp các API cần thiết để tích hợp tính năng trò chuyện vào các trang web. Tuy nhiên, họ nhận ra rằng nền tảng này đang được sử dụng để truyền dữ liệu ứng dụng chứ không đơn giản là chat. Sau đó, họ đã phát triển Envole và tạo nên Firebase.
* Đến tháng 4 năm 2012, Firebase đã được công bố trên toàn cầu dưới dạng một công ty riêng biệt. Những năm sau đó, Firebase đã thực hiện nhiều cuộc huy động vốn và ra mắt các sản phẩm mới.
* Đến tháng 10 năm 2014, Firebase đã chính thức được Google mua lại và trở thành một ứng dụng đa năng trên nền tảng di động và website.

1. **Các tính năng chính:**



1. **Realtime Database:**

Realtime Database là một cơ sở dữ liệu thời gian thực. Ngay sau khi bạn đăng ký tài khoản trên Firebase, bạn sẽ nhận được Realtime Database được lưu trữ dưới dạng JSON và được đồng bộ hóa theo thời gian thực đối với mọi kết nối.

Đối với các ứng dụng được xây dựng trên đa nền tảng như Android, IOS và WebApp, tất cả client sẽ cùng sử dụng một cơ sở dữ liệu. Bên cạnh đó, hệ thống dữ liệu này sẽ tự động cập nhật khi lập trình viên phát triển ứng dụng. Sau đó, tất cả dữ liệu này sẽ được truyền tải thông qua các kết nối SSl có 2048 bit.

A diagram of a diagram

Description automatically generated with medium confidence

1. **Cloud Firestore:**
2. **Authentication:cv**
3. **Cloud Storage:**
4. **Hosting:**
5. **Cloud Functions:**
6. **Analytics**
7. **Chức năng:**
8. **Ưu điểm và hạn chế:**

Below listed are the advantages and disadvantages of using a Firebase backend:

Pros:

Free plans for beginners.

Real-time database is available.

Growing Community.

Numerous services are available.

Cons:

It uses NoSQL so, people migrating from SQL might feel difficulty.

It is still growing so, it is not tested to an extent.

Miễn phí

Firebase là nền tảng miễn phí cho phép đăng nhập bằng tài khoản Google.

Gói Spark cung cấp nhiều tính năng cho phép các nhà phát triển bắt đầu triển khai dự án miễn phí.

Gói Blaze dành cho các nhà phát triển có nhu cầu lớn hơn.

Tốc độ phát triển

Firebase cung cấp quyền truy cập vào máy chủ và máy chủ lưu trữ dành cho các nhà phát triển giúp tiết kiệm đáng kể thời gian triển khai ứng dụng.

Firebase dựa vào các nhà phát triển phụ trợ và giao diện người dùng để quản lý hay tối ưu hoá thời gian hoàn thành công việc.

Firebase bao gồm các dịch vụ sẵn hỗ trợ triển khai ứng dụng mà không cần tạo mã hay bắt đầu chương trình phụ trợ từ đầu.

Tích hợp nền tảng

Firebase có một danh sách hoàn chỉnh cho phép nhà phát triển tạo ra các sản phẩm hỗ trợ.

Firestore và Firebase’s Realtime Database là hai cơ sở dữ liệu cho phép lưu trữ đám mây và hỗ trợ triển khai ứng dụng dễ dàng hơn.

Firebase cung cấp các tính năng xây dựng, phát hành và theo dõi ứng dụng linh hoạt.

Cung cấp các công cụ để thu hút người dùng lựa chọn sử dụng ứng dụng đó.

Được cung cấp bởi Google

Firebase được cung cấp bởi Google có độ tin cậy cao.

Firebase trở thành một phần của Google Cloud Platform được tích hợp với các dịch vụ Google Cloud khác và một số dịch vụ của bên thứ ba.

Phát triển giao diện người dùng

Tập trung công việc tạo mã giao diện người dùng trên ứng dụng di động.

Tối ưu hoá thời gian tạo mã cho ứng dụng.

Tiết kiệm chi phí triển khai ứng dụng.

Không có máy chủ

Cung cấp một môi trường không sử dụng máy chủ.

Tích hợp với kiến trúc không máy chủ yêu cầu người dùng trả tiền.

Luôn trong trạng thái hoạt động.

Hỗ trợ mở rộng quy mô triển khai ứng dụng.

Machine Learning

Machine Learning được sử dụng phổ biến trong các dự án doanh nghiệp.

Cung cấp phương pháp phát triển tùy chọn.

Phù hợp với nhà phát triển iOS và Android.

Machine Learning tích hợp với API cung cấp nhiều tính năng như nhận dạng văn bản, nhận diện khuôn mặt, ghi nhãn hình ảnh hay quét mã vạch.

Tạo lưu lượng truy cập

Cho phép lập chỉ mục ứng dụng để thu hút người dùng sử dụng.

Các nhà phát triển sử dụng ứng dụng Android để truy cập các nội dung mong muốn.

Khắc phục lỗi màn hình

Giống như một công cụ cho phép tìm kiếm và khắc phục các sự cố nhanh chóng hơn.

Theo dõi các lỗi nghiêm trọng và không nghiêm trọng.

Bản sao lưu

Bảo vệ ứng dụng không bị mất dữ liệu với các bản sao lưu thường xuyên.

Gói Blaze giúp định cấu hình cơ sở dữ liệu hỗ trợ thực hiện các bản sao lưu tự động.