*Tên:* **Trần Huy Đức**

*MSSV*: **1911071**

**Môn học: Lập trình Web**

**LAB 04**

**Cookie, Session, Web Service**

***Bài 1:*** *Trình bày định nghĩa, ưu điểm và nhược điểm của các loại Web Service: REST và SOAP, trả lời vào file Word.*

1. **REST**

**REST** (viết tắt của *REpresentational State Transfer*) là một thuật ngữ được dùng để chỉ một kiểu kiến trúc được dùng trong hoạt động giao tiếp của máy tính (giữa máy tính cá nhân và máy chủ của website).

Nói theo một cách khác thì REST là kiểu kiến trúc sẽ cung cấp API (*Application Programming Interface*) thông qua đường truyền mạng để xử lý các hoạt động CRUD (*Create, Read, Update, và Delete*) trên dữ liệu. Trong vấn đề nghiên cứu và phát triển các ứng dụng Web Services sử dụng giao thức HTTP cho hoạt động giao tiếp thông qua kết nội mạng, REST được sử dụng nhiều hơn SOAP.

**Ưu điểm:**

* Sử dụng HTTP là giao thức cơ sở nên REST trở nên đơn giản hơn SOAP , REST thích ứng tốt với các công nghệ web mới, dễ dàng sử dụng và bảo trì.
* Tạo lập clients, phát triển các API, người dùng dễ dàng tiếp cận và nắm bắt dữ liệu và làm được mọi thứ mà [SOAP](https://bizflycloud.vn/tin-tuc/soap-la-gi-20181117112550075.htm) có thể làm được, thậm chí còn có thể làm dễ hơn và tốt hơn.
* Trong khi SOAP chỉ cho phép kiểu định dạng dữ liệu XML thì REST cho phép rất nhiều kiểu khác. JSON có độ phù hợp với dữ liệu hơn đồng thời thực hiện phân tích cú pháp nhanh hơn. REST cho phép hỗ trợ tốt hơn cho browser clients nhờ khả năng hỗ trợ của nó đối với JSON.
* REST có hiệu suất và khả năng mở rộng cao hơn. Các lần đọc dựa trên REST sẽ được lưu trữ còn các lần đọc dựa trên SOAP sẽ không được lưu trữ.
* Tách biệt rõ ràng giữa server và client, chỉ giao tiếp thông qua các phương thức HTTP và URI.

**Nhược điểm:**

1. **SOAP**

**SOAP** *(*viết tắt của *Simple Object Access Protocol)* có vẻ phổ biến hơn REST, nó được biết đến là hình thức truy xuất của Web Service hay cách thức mà Web Services truyền tải dữ liệu. SOAP sử dụng và dựa hoàn toàn vào XML để cung cấp các services truyền tin. SOAP được Microsoft phát triển để thay thế cho DCOM (*Distributed Component Object Mode*l) và CORBA (*Common Object Request Broker Architecture*) vì cách thức truyền tin XML của SOAP có khả năng làm việc tốt hơn qua internet. Nhờ việc dựa trên XML, SOAP tránh được việc phải phụ thuộc vào platform hay các ngôn ngữ lập trình.

**Ưu điểm:**

* Tuân theo cách xử lý trong hệ thống của các doanh nghiệp lớn.
* Được thực hiện ở tầng trên của các giao thức giao tiếp, cả với các giao thức không đồng bộ, không bắt buộc phải là HTTP.
* Các thông tin của đối tượng được giao tiếp với khách hàng.
* Bảo mật và xác thực luôn được đảm bảo trong giao thức (nhờ stateful).
* Có thể được miêu tả bằng WSDL.
* WS-Security SOAP: tương tự REST, SOAP hỗ trợ SSL song nó cũng hỗ trợ WS-Security để có thể tích hợp thêm những tính năng bảo mật doanh nghiệp (enterprise security features). SOAP hỗ trợ nhận diện thông qua trung gian chứ không chỉ dưới dạng point-to-point (SSL). Nó thường được dùng đến cho nhu cầu xây dựng Web Services đảm bảo tính tin cậy và an toàn vì thực tế nó hỗ trợ các công cụ bảo mật mà dịch vụ internet thông thường không cần đến. Ngoài ra, nó cũng cung cấp triển khai tiêu chuẩn để đảm bảo tính an toàn, tính toàn vẹn và tin cậy của dữ liệu.
* WS-AtomicTransaction: Giao dịch ACID trên service. Dù REST hỗ trợ các transaction song nó không thể đảm bảo tính toàn diện và không phù hợp với ACID. May mắn thay, ACID transaction hầu như không có nghĩa đối với các dịch vụ internet thông thường. Bản thân HTTP không thể cung cấp hai pha trên tài nguyên giao dịch phân tán nên REST bị giới hạn, ngược lại SOAP thì có thể. Các ứng dụng internet thường không đòi hỏi mức độ tin cậy giao dịch này song các ứng dụng doanh nghiệp đôi khi lại cần đến.
* WS-ReliableMessaging: REST không sở hữu hệ thống báo lỗi chuẩn và trông chờ clients ứng phó với lỗi communicate bằng cách thử lại (retrying). Trong khi đó, SOAP lại thành công trong tình huống này và cung cấp độ tin cậy end-to-end thông qua các trung gian SOAP.

**Nhược điểm:**

* Việc duy trì trạng thái (stateful) khiến tiêu tốn tài nguyên cho các metadata.
* Khó sử dụng và không phổ biến bằng REST trong các ứng dụng web hay mobile thông thường.
* Chỉ sử dụng XML.