

WEB SERVICE

swono@polibatam.ac.id

PENGANTAR

Awal Mula (1990-an):

- Perkembangan teknologi web dan pertumbuhan internet memicu kebutuhan untuk metode komunikasi antaraplikasi.
- XML (eXtensible Markup Language) menjadi populer sebagai format pertukaran data yang dapat dibaca oleh manusia dan mesin.

SOAP (Simple Object Access Protocol) (2000):

- SOAP muncul sebagai protokol standar untuk pertukaran pesan di antara aplikasi melalui HTTP dan protokol lainnya.
- SOAP menggunakan XML sebagai format pesan dan mendefinisikan aturan untuk pertukaran pesan yang terstruktur.

REST (Representational State Transfer) (2000-an):

- Pendekatan REST, yang menggunakan HTTP untuk pertukaran data, mulai mendapatkan popularitas.
- REST menggunakan metode standar HTTP seperti GET, POST, PUT, dan DELETE untuk berkomunikasi antara sistem.

WSDL (Web Services Description Language) dan UDDI (Universal Description, Discovery, and Integration) (2000-an):

- WSDL digunakan untuk mendefinisikan struktur data dan operasi yang ditawarkan oleh web service.
- UDDI digunakan untuk mencari dan mendaftarkan layanan web sehingga dapat ditemukan dan digunakan oleh aplikasi lain.

JSON (JavaScript Object Notation) (2010-an):

- Penggunaan JSON sebagai format pertukaran data dalam web service semakin meningkat.
- RESTful web service sering menggunakan JSON sebagai format data yang lebih ringan dan mudah dibaca oleh manusia.

DEFINISI

- Web service adalah teknologi yang memungkinkan aplikasi untuk berkomunikasi satu sama lain melalui internet.
- Merupakan metode standar untuk mengintegrasikan aplikasi yang berbeda, yang dapat berjalan pada platform dan bahasa pemrograman yang berbeda.

JENIS WEB SERVICE

SOAP Web Services:

- Berbasis pada protokol SOAP.
- Menggunakan XML sebagai format pesan.
- Mendefinisikan layanan dengan WSDL.

RESTful Web Services:

- Berbasis pada pendekatan REST.
- Menggunakan metode HTTP seperti GET, POST, PUT, DELETE.
- Format data dapat berupa XML atau JSON.

XML-RPC dan JSON-RPC:

- Protokol ringan untuk pertukaran data berbasis XML atau JSON.
- Tidak sekompleks SOAP, tetapi lebih sederhana untuk beberapa kasus.

KEUNTUNGAN WEB SERVICE

- Interoperabilitas: Memungkinkan integrasi antarplatform dan bahasa.
- Lock-In Vendor yang Rendah: Aplikasi dapat berkomunikasi tanpa terlalu tergantung pada teknologi atau vendor tertentu.
- Distribusi Aplikasi: Layanan dapat disebar dan diakses melalui internet.

PERTUKARAN DATA PADA WEB SERVICE

Permintaan (Request):

- **Inisiasi:** Proses dimulai ketika klien membuat permintaan (request) ke server.
- **Metode HTTP:** Permintaan dapat menggunakan metode HTTP seperti GET, POST, PUT, atau DELETE.
- **Endpoint:** Permintaan mencantumkan endpoint atau URL yang menunjuk ke sumber daya atau layanan yang diinginkan.

Pengkodean Data (Data Encoding):

- **Format Data:** Data yang akan dikirimkan dienkapsulasi dalam format tertentu, seperti JSON atau XML.
- **Parameter atau Payload:** Data dapat disertakan dalam parameter URL, dalam payload permintaan, atau menggunakan metode yang disepakati.

Pembuatan Respons (Response Creation):

- Format Respons: Server mengonversi hasil operasi bisnis ke dalam format data yang telah disepakati (misalnya, JSON atau XML).
- Status Code: Respons juga mencakup status code HTTP yang memberi tahu klien tentang hasil permintaan (200 untuk sukses, 404 untuk tidak ditemukan, dll.)

Pengkodean Respons (Response Encoding):

- Format Data: Respons dienkapsulasi dalam format yang dipahami oleh klien (JSON, XML).
- Pengiriman Data: Respons siap untuk dikirimkan kembali ke klien.

TOOLS

Postman:

<https://www.postman.com/downloads/>

Library REST:

<https://github.com/ardisaurus/old-rest-ci>

<https://github.com/chriskacerguis/codeigniter-restserver>

RESPON HTTP WEB SERVICE RESTFUL

HTTP_OK (200 OK):

- Deskripsi: Menunjukkan bahwa permintaan telah berhasil. Respons ini umumnya digunakan untuk operasi GET yang berhasil.
- Fungsi Umum: Memberikan konfirmasi bahwa permintaan berhasil dan menyertakan data yang diminta.

HTTP_CREATED (201 Created):

- Deskripsi: Menunjukkan bahwa permintaan telah berhasil dan sumber daya baru telah dibuat sebagai hasilnya.
- Fungsi Umum: Digunakan setelah permintaan POST yang sukses untuk menunjukkan bahwa sumber daya baru telah dibuat.

HTTP_NOT_FOUND (404 Not Found):

- Deskripsi: Menunjukkan bahwa sumber daya yang diminta tidak ditemukan di server.
- Fungsi Umum: Memberikan informasi bahwa endpoint atau sumber daya yang diminta tidak tersedia.

HTTP_BAD_REQUEST (400 Bad Request):

- Deskripsi: Menunjukkan bahwa server tidak dapat memahami permintaan klien karena sintaks atau struktur permintaan yang salah.
- Fungsi Umum: Digunakan ketika ada kesalahan dalam data atau parameter yang dikirim oleh klien.

HTTP_UNAUTHORIZED (401 Unauthorized):

- Deskripsi: Menunjukkan bahwa klien tidak diizinkan untuk mengakses sumber daya yang diminta tanpa autentikasi.
- Fungsi Umum: Digunakan untuk meminta autentikasi sebelum akses diberikan.

HTTP_FORBIDDEN (403 Forbidden):

- Deskripsi: Menunjukkan bahwa klien tidak diizinkan untuk mengakses sumber daya yang diminta bahkan setelah autentikasi.
- Fungsi Umum: Digunakan ketika ada autentikasi yang valid, tetapi klien tidak memiliki izin akses yang diperlukan.

HTTP_CONFLICT (409 Conflict):

- Deskripsi: Menunjukkan bahwa permintaan tidak dapat diselesaikan karena konflik dengan status saat ini dari sumber daya.
- Fungsi Umum: Digunakan ketika ada konflik data atau kondisi yang mencegah operasi berhasil.

HTTP_INTERNAL_SERVER_ERROR (500 Internal Server Error):

- Deskripsi: Menunjukkan bahwa terjadi kesalahan internal pada server yang mencegah pemenuhan permintaan.
- Fungsi Umum: Digunakan ketika ada kesalahan yang tidak terduga atau tidak dapat ditangani di sisi server.

HTTP_METHOD_NOT_ALLOWED (405 Method Not Allowed):

- Deskripsi: Menunjukkan bahwa metode HTTP yang digunakan dalam permintaan tidak diizinkan untuk sumber daya yang diminta.
- Fungsi Umum: Digunakan ketika klien menggunakan metode yang tidak diizinkan untuk operasi tertentu.

HTTP_NOT_ACCEPTABLE (406 Not Acceptable):

- Deskripsi: Menunjukkan bahwa server tidak dapat menghasilkan respons yang dapat diterima sesuai dengan header Accept yang diberikan oleh klien.
- Fungsi Umum: Digunakan ketika format respons yang diminta oleh klien tidak dapat didukung oleh server.

TERIMAKASIH