

**TUGAS PENDAHULUAN  
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL XIV  
DATA STORAGE  
'API'**



**Disusun Oleh :  
Muhammad Idham Cholid**

**2211104016**

**SE06-01**

**Asisten Praktikum :  
Muhammad Faza Zulian Gesit Al Barru  
Aisyah Hasna Aulia**

**Dosen Pengampu :  
Yudha Islami Sulistya, S.Kom., M.Cs.**

**PROGRAM STUDI S1 SOFTWARE ENGINEERING**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

## TUGAS PENDAHULUAN

### SOAL

A. Sebutkan dan jelaskan dua jenis utama **Web Service** yang sering digunakan dalam pengembangan aplikasi.

- SOAP (Simple Object Access Protocol):

SOAP adalah protokol berbasis XML yang digunakan untuk pertukaran informasi terstruktur dalam implementasi layanan web. SOAP mendefinisikan aturan untuk format pesan dan protokol komunikasi, serta mendukung berbagai protokol transportasi seperti HTTP, SMTP, dan lainnya. Keunggulan SOAP adalah kemampuannya untuk mendukung transaksi yang kompleks dan keamanan yang lebih baik melalui WS-Security. Namun, SOAP cenderung lebih rumit dan memerlukan lebih banyak bandwidth dibandingkan dengan REST.

- REST (Representational State Transfer):

REST adalah arsitektur yang menggunakan protokol HTTP untuk komunikasi antara klien dan server. REST memanfaatkan metode HTTP seperti GET, POST, PUT, dan DELETE untuk melakukan operasi pada sumber daya yang diwakili oleh URL. REST lebih sederhana dan lebih ringan dibandingkan SOAP, serta lebih mudah digunakan dan diintegrasikan dengan aplikasi web modern. REST juga mendukung berbagai format data, termasuk JSON dan XML, yang membuatnya lebih fleksibel.

B. Apa yang dimaksud dengan **Data Storage API**, dan bagaimana API ini mempermudah pengelolaan data dalam aplikasi?

Data Storage API adalah antarmuka pemrograman aplikasi yang memungkinkan pengembang menyimpan, membaca, memperbarui, dan menghapus data secara programatik dalam aplikasi. API ini bisa berupa layanan penyimpanan berbasis cloud (seperti Firebase atau AWS S3) atau database lokal.

- **Abstraksi:** Pengembang tidak perlu memahami detail implementasi penyimpanan data, sehingga dapat fokus pada logika aplikasi.
- **Kemudahan Penggunaan:** API menyediakan fungsi yang mudah digunakan untuk operasi umum seperti penyimpanan, pengambilan, dan penghapusan data.
- **Konsistensi:** API membantu menjaga konsistensi data dan memudahkan pengelolaan transaksi.
- **Skalabilitas:** Dengan menggunakan API, aplikasi dapat dengan mudah beradaptasi dengan perubahan dalam penyimpanan data tanpa mempengaruhi logika aplikasi.

C. Jelaskan bagaimana proses kerja komunikasi antara klien dan server dalam sebuah Web Service, mulai dari permintaan (*request*) hingga tanggapan (*response*).

- **Permintaan (Request)**

Klien mengirimkan permintaan ke server menggunakan protokol HTTP. Permintaan ini biasanya mencakup metode HTTP (GET, POST, PUT, DELETE), URL endpoint, dan mungkin juga header dan body yang berisi data tambahan.

- **Pengolahan Permintaan**

Server menerima permintaan dan memprosesnya. Ini melibatkan pengambilan data dari database, pemrosesan logika bisnis, dan persiapan tanggapan.

- **Tanggapan (Response)**

Setelah memproses permintaan, server mengirimkan tanggapan kembali ke klien. Tanggapan ini biasanya mencakup kode status HTTP (misalnya, 200 untuk sukses, 404 untuk tidak ditemukan), header, dan body yang berisi data yang diminta (dalam format JSON, XML, dll.).

- **Penerimaan Tanggapan**

Klien menerima tanggapan dari server dan dapat memproses data tersebut sesuai kebutuhan, seperti menampilkan informasi kepada pengguna atau menyimpan data untuk penggunaan lebih lanjut.

**D. Mengapa keamanan penting dalam penggunaan **Web Service**, dan metode apa saja yang dapat diterapkan untuk memastikan data tetap aman?**

Keamanan sangat penting dalam penggunaan Web Service karena:

- **Perlindungan Data Sensitif:** Web Service sering kali menangani data sensitif, seperti informasi pribadi, keuangan, atau kesehatan. Keamanan membantu melindungi data ini dari akses yang tidak sah.
- **Integritas Data:** Keamanan memastikan bahwa data tidak diubah atau dirusak selama transmisi antara klien dan server.
- **Kepercayaan Pengguna:** Pengguna lebih cenderung menggunakan aplikasi yang memiliki langkah-langkah keamanan yang kuat, yang dapat meningkatkan reputasi dan kepercayaan terhadap layanan.

Metode untuk Memastikan Keamanan:

- **Enkripsi:** Menggunakan protokol seperti HTTPS untuk mengenkripsi data yang ditransmisikan antara klien dan server.
- **Autentikasi dan Otorisasi:** Mengimplementasikan mekanisme autentikasi (seperti OAuth, JWT) untuk memastikan bahwa hanya pengguna yang berwenang yang dapat mengakses layanan.
- **Validasi Input:** Memastikan bahwa semua input dari pengguna divalidasi untuk mencegah serangan seperti SQL Injection atau Cross-Site Scripting (XSS).
- **Penggunaan Firewall dan IDS/IPS:** Menggunakan firewall dan sistem deteksi/pencegahan intrusi untuk melindungi server dari serangan eksternal.