TUGAS PENDAHULUAN
PEMROGRAMAN PERANGKAT BERGERAK

MODUL XIV

DATA STORAGE
‘API’

Disusun Oleh :

Mohammad Dhimas Afrizal / 2211104023

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

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

1. SOAP (Simple Object Access Protocol)

   SOAP adalah protokol berbasis XML yang digunakan untuk mengirim pesan antara aplikasi melalui jaringan. SOAP sangat cocok untuk sistem yang memerlukan keamanan tinggi dan transaksi kompleks.

   Kelebihan: Mendukung berbagai protokol komunikasi seperti HTTP, SMTP, dan TCP; menyediakan keamanan melalui WS-Security.

   Kekurangan: Lebih kompleks dan membutuhkan lebih banyak bandwidth dibandingkan REST.

2. REST (Representational State Transfer)

   REST adalah arsitektur yang menggunakan protokol HTTP untuk komunikasi. Data biasanya ditransmisikan dalam format JSON atau XML, namun JSON lebih populer karena ringan dan mudah dipahami.

   Kelebihan: Ringan, sederhana, cepat, dan cocok untuk aplikasi berbasis web dan mobile.

   Kekurangan: Kurang mendukung operasi yang sangat kompleks dibandingkan dengan SOAP.

b. Data Storage API adalah antarmuka pemrograman aplikasi yang memungkinkan pengembang untuk menyimpan, mengakses, memperbarui, dan menghapus data di layanan penyimpanan, baik lokal maupun berbasis cloud.

Cara Mempermudah Pengelolaan Data:

1. Akses yang Mudah: API menyediakan metode sederhana untuk berinteraksi dengan data tanpa harus menulis kode kompleks.
2. Skalabilitas: Mendukung penyimpanan dalam jumlah besar dan dapat menyesuaikan kebutuhan aplikasi.
3. Sinkronisasi Real-time: Data dapat diakses dan diperbarui secara real-time melalui API.

4. Keamanan Data: Banyak layanan Data Storage API menawarkan fitur keamanan bawaan, seperti enkripsi.
5. Integrasi: API memungkinkan integrasi antara berbagai platform dan teknologi.

Contoh: Firebase Realtime Database API, Amazon S3 API, atau API dari MongoDB.

c. Proses komunikasi antara klien dan server dalam Web Service melibatkan beberapa langkah:

1. Permintaan (Request):
   - Klien (misalnya aplikasi web atau mobile) mengirimkan permintaan ke server menggunakan protokol HTTP/HTTPS.
   - Permintaan ini biasanya mencakup:
     - URL Endpoint: Lokasi tujuan API.
     - Method: GET (membaca), POST (menulis), PUT (memperbarui), DELETE (menghapus).
     - Header: Informasi tambahan, seperti autentikasi atau format data.
     - Body (Opsional): Data yang dikirimkan (biasanya dalam format JSON/XML).
2. Pemrosesan di Server:

   Server menerima permintaan, memprosesnya, dan berinteraksi dengan database atau layanan lain sesuai kebutuhan.
3. Tanggapan (Response):

   Server mengirimkan tanggapan kembali ke klien dalam bentuk:
   - Kode Status HTTP: Misalnya 200 (OK), 404 (Not Found), 500 (Server Error).
   - Body: Data hasil pemrosesan dalam format JSON atau XML.
   - Header: Informasi tambahan tentang response.
4. Klien Menampilkan Data:

   Klien menerima tanggapan dari server, kemudian data diolah dan ditampilkan ke pengguna.

d. Keamanan dalam Web Service penting karena data yang dikirimkan antara klien dan server rentan terhadap serangan seperti Man-in-the-Middle (MITM), pencurian data, atau penyalahgunaan API. Jika keamanan tidak dijaga, data sensitif pengguna dapat bocor dan digunakan oleh pihak tidak bertanggung jawab.

Metode Keamanan yang Dapat Diterapkan:

- HTTPS (Hypertext Transfer Protocol Secure): Mengenkripsi komunikasi antara klien dan server menggunakan SSL/TLS.
- Token-based Authentication: Menggunakan token (seperti JWT atau OAuth) untuk memverifikasi identitas pengguna.
- API Key: Memberikan kunci unik untuk setiap pengguna atau aplikasi yang mengakses API.
- Rate Limiting: Membatasi jumlah permintaan API dalam jangka waktu tertentu untuk mencegah serangan DDoS.
- Enkripsi Data: Mengenkripsi data saat disimpan (at rest) dan saat dikirimkan (in transit).
- Firewall dan WAF (Web Application Firewall): Melindungi server dari akses tidak sah dan serangan injeksi (SQL Injection, XSS).
- Validasi Input Data: Memastikan input dari klien valid agar mencegah serangan seperti injection attacks.