TUGAS PENDAHULUAN
PEMROGRAMAN PERANGKAT BERGERAK

MODUL XIV

DATA STORAGE
'API'

Disusun Oleh :

Chayla Ravanelly

Quintitawati /

2211104082

SE 06 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.

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

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

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

JAWAB

a. Jenis Utama Web Service

1. **SOAP (Simple Object Access Protocol):**
   SOAP adalah protokol standar berbasis XML yang digunakan untuk mengirimkan data antar perangkat dalam jaringan. Protokol ini mendukung berbagai bahasa pemrograman dan platform, sehingga memudahkan interoperabilitas. SOAP menggunakan protokol seperti HTTP atau SMTP untuk mengirim pesan, dengan struktur pesan yang ketat dan terstandarisasi, membuatnya cocok untuk aplikasi enterprise yang membutuhkan keamanan tinggi.
2. **REST (Representational State Transfer):**
   REST adalah pendekatan arsitektural yang memanfaatkan protokol HTTP untuk komunikasi antara klien dan server. REST lebih ringan dibandingkan SOAP karena menggunakan format data yang fleksibel seperti JSON atau XML. REST sering digunakan untuk aplikasi web dan mobile karena sifatnya yang sederhana, cepat, dan mudah diimplementasikan.

b. **Pengertian Data Storage API**

Data Storage API adalah antarmuka pemrograman aplikasi yang menyediakan cara untuk menyimpan, mengambil, dan mengelola data secara efisien dalam sebuah aplikasi. API ini membantu pengembang untuk berinteraksi dengan sistem penyimpanan tanpa harus memahami detail teknis di baliknya, seperti database, file storage, atau cloud storage. Dengan menggunakan Data Storage API, pengelolaan

data menjadi lebih sederhana, karena pengembang dapat melakukan operasi CRUD (Create, Read, Update, Delete) secara langsung melalui fungsi-fungsi yang telah disediakan oleh API tersebut.

c. **Proses Kerja Komunikasi Klien dan Server dalam Web Service**

1. **Permintaan dari Klien:**
   Klien, biasanya sebuah aplikasi atau browser, mengirimkan permintaan (request) ke server menggunakan protokol seperti HTTP. Permintaan ini mencakup informasi seperti metode HTTP (GET, POST, PUT, DELETE) dan URL endpoint yang dituju.
2. **Pemrosesan di Server:**
   Server menerima permintaan tersebut, memprosesnya sesuai dengan logika bisnis atau data yang diminta, dan berinteraksi dengan sumber daya yang relevan seperti database.
3. **Pengiriman Tanggapan:**
   Setelah proses selesai, server mengirimkan tanggapan (response) kembali ke klien. Tanggapan ini biasanya berisi data dalam format tertentu (seperti JSON atau XML) serta kode status HTTP untuk menunjukkan hasil operasi (contoh: 200 untuk sukses, 404 untuk data tidak ditemukan).
4. **Penerimaan di Klien:**
   Klien menerima tanggapan dari server dan menampilkannya kepada pengguna atau memprosesnya lebih lanjut sesuai kebutuhan.

d. **Pentingnya Keamanan dalam Web Service**

Keamanan sangat penting dalam penggunaan Web Service untuk melindungi data dari akses tidak sah, mencegah manipulasi data, dan menjaga kerahasiaan informasi sensitif. Tanpa langkah-langkah keamanan yang memadai, data bisa rentan terhadap ancaman seperti serangan man-in-the-middle, SQL injection, atau pencurian identitas.

Metode Keamanan yang Dapat Diterapkan:

1. **Enkripsi Data:**
   Menggunakan protokol HTTPS untuk mengenkripsi data yang dikirim antara klien dan server sehingga tidak dapat dibaca oleh pihak yang tidak berwenang.

2. **Autentikasi dan Otorisasi:**
   Memastikan hanya pengguna yang sah yang dapat mengakses layanan melalui mekanisme seperti token API, OAuth, atau sistem login berbasis username dan password.
3. **Firewall dan Monitoring:**
   Melindungi server dengan firewall dan memonitor aktivitas untuk mendeteksi dan mencegah serangan yang mencurigakan.
4. **Input Validation:**
   Memastikan data yang dikirim oleh klien telah divalidasi untuk mencegah serangan injeksi, seperti SQL injection atau script injection.