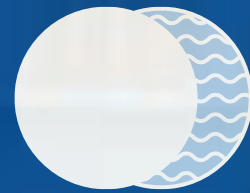


STMIK Kaputama

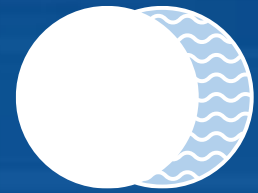


Pertemuan 12

PENYIMPANAN DATA LANJUTAN

Kristina Annatasia Br Sitepu, M.Kom





PEMBAHASAN

Pendahuluan

Arsitektur Client Server

Koneksi Android-MySQL melalui PHP

Implementasi CRUD



Pendahuluan

Dalam pengembangan aplikasi Android, penyimpanan dan pengelolaan data di luar perangkat sering diperlukan, terutama untuk aplikasi yang membutuhkan penyimpanan data dalam jumlah besar atau berbagi data antar pengguna. Salah satu pendekatan umum adalah menggunakan database server, dengan MySQL sebagai sistem manajemen basis data yang populer. Materi ini akan membahas cara menghubungkan aplikasi Android dengan database MySQL untuk pengelolaan data yang efisien.

Pengenalan MySql

MySQL adalah sistem manajemen basis data relasional (RDBMS) yang populer dan bersifat open-source. Menggunakan Structured Query Language (SQL) untuk pengelolaan data, MySQL sering dipakai dalam pengembangan aplikasi web dan mobile karena performa tinggi, reliabilitas baik, dan dukungan komunitas yang luas. MySQL cocok untuk menangani data dalam jumlah besar di server dan memungkinkan interaksi aplikasi dengan basis data melalui SQL.



Basis Data Relasional

MySQL menggunakan model relasional untuk menyimpan dan mengelola data.



Open-Source

MySQL tersedia secara gratis dan dapat dimodifikasi sesuai kebutuhan.



Performa Tinggi

MySQL dikenal dengan kecepatan dan efisiensi dalam mengelola data.

Arsitektur Client-Server di Aplikasi Android

Aplikasi Android yang mengelola data di database MySQL menggunakan arsitektur client-server. Aplikasi Android di perangkat pengguna berperan sebagai client, mengirim permintaan ke server untuk operasi data. Server adalah tempat MySQL berjalan, menyimpan data yang diakses oleh aplikasi. Selain MySQL, server juga menjalankan aplikasi PHP untuk membantu komunikasi antara aplikasi Android dan database MySQL.

Client (Android)

Mengirim permintaan ke server untuk operasi data seperti menyimpan, membaca, mengubah, dan menghapus.

Server (MySQL + PHP)

Menyimpan data, menjalankan MySQL, dan menggunakan PHP untuk memproses permintaan dari client.

Peran PHP sebagai Perantara

Android tidak dapat langsung terhubung ke database MySQL di server. Oleh karena itu, skrip PHP digunakan sebagai perantara. PHP adalah bahasa pemrograman server-side yang mudah diintegrasikan dengan MySQL. Skrip PHP bertindak sebagai API yang menerima permintaan dari aplikasi Android, meneruskannya ke MySQL, dan mengembalikan hasilnya ke aplikasi dalam format yang dapat dimengerti Android, seperti JSON.

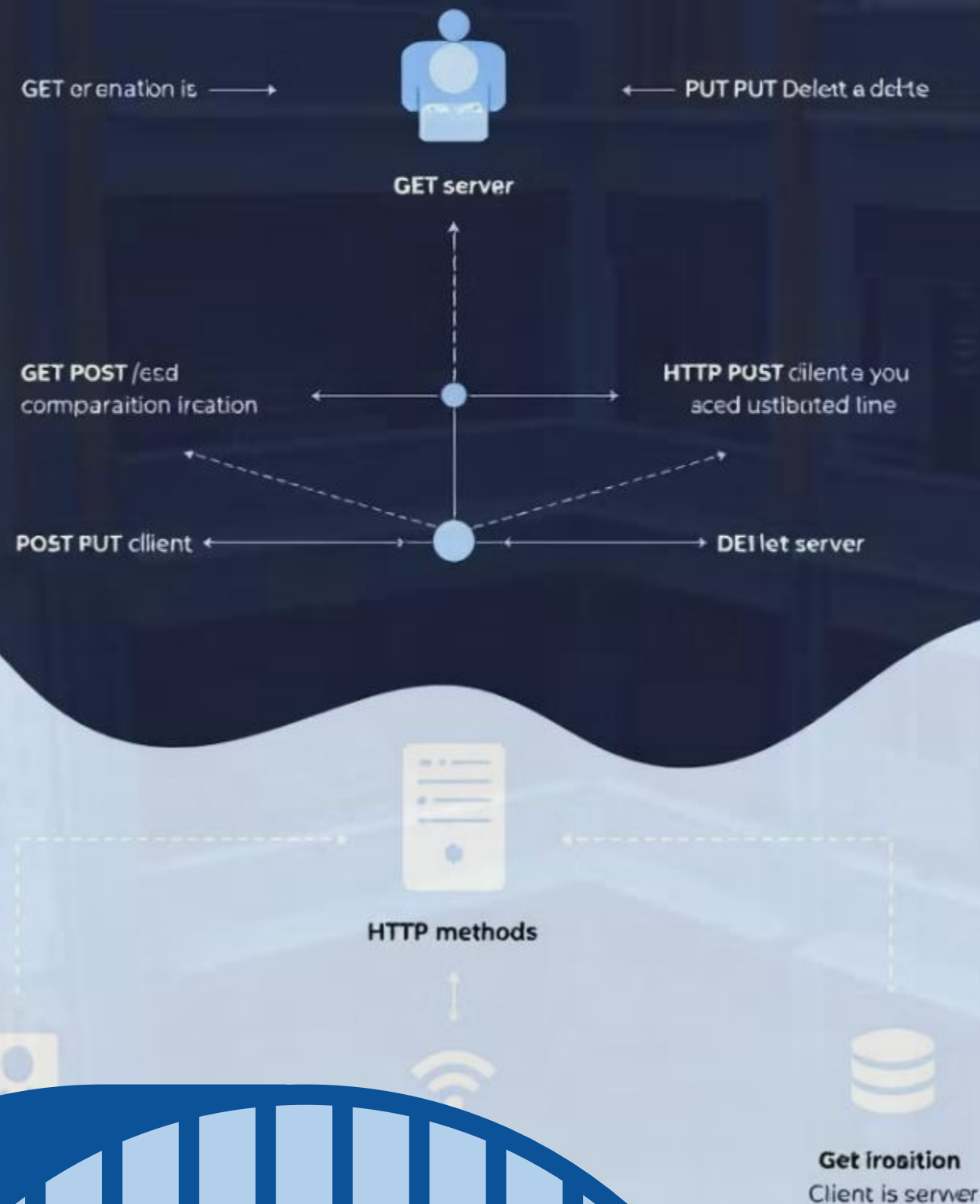


Peran PHP sebagai Perantara

- 1** — **Permintaan dari Android**
Aplikasi Android mengirim permintaan HTTP ke server.
- 2** — **Proses oleh PHP**
Skrip PHP menerima dan memproses permintaan dari Android.
- 3** — **Interaksi dengan MySQL**
PHP mengeksekusi query ke database MySQL.
- 4** — **Respons ke Android**
PHP mengirim hasil dalam format JSON kembali ke aplikasi Android.

REST API

What is set usn youry the swhen?



Teknik API REST

API (Application Programming Interface) adalah cara umum untuk memungkinkan aplikasi mengakses sumber daya data di server. REST API adalah gaya arsitektur yang banyak digunakan untuk membuat API, di mana setiap operasi dilakukan menggunakan HTTP request sederhana (GET, POST, PUT, DELETE). Dalam konteks ini, REST API digunakan untuk melakukan operasi CRUD (Create, Read, Update, Delete) dari aplikasi Android ke database MySQL.

GET

Digunakan untuk membaca data dari server.

POST

Digunakan untuk mengirim data baru ke server.

PUT

Digunakan untuk memperbarui data yang ada di server.

DELETE

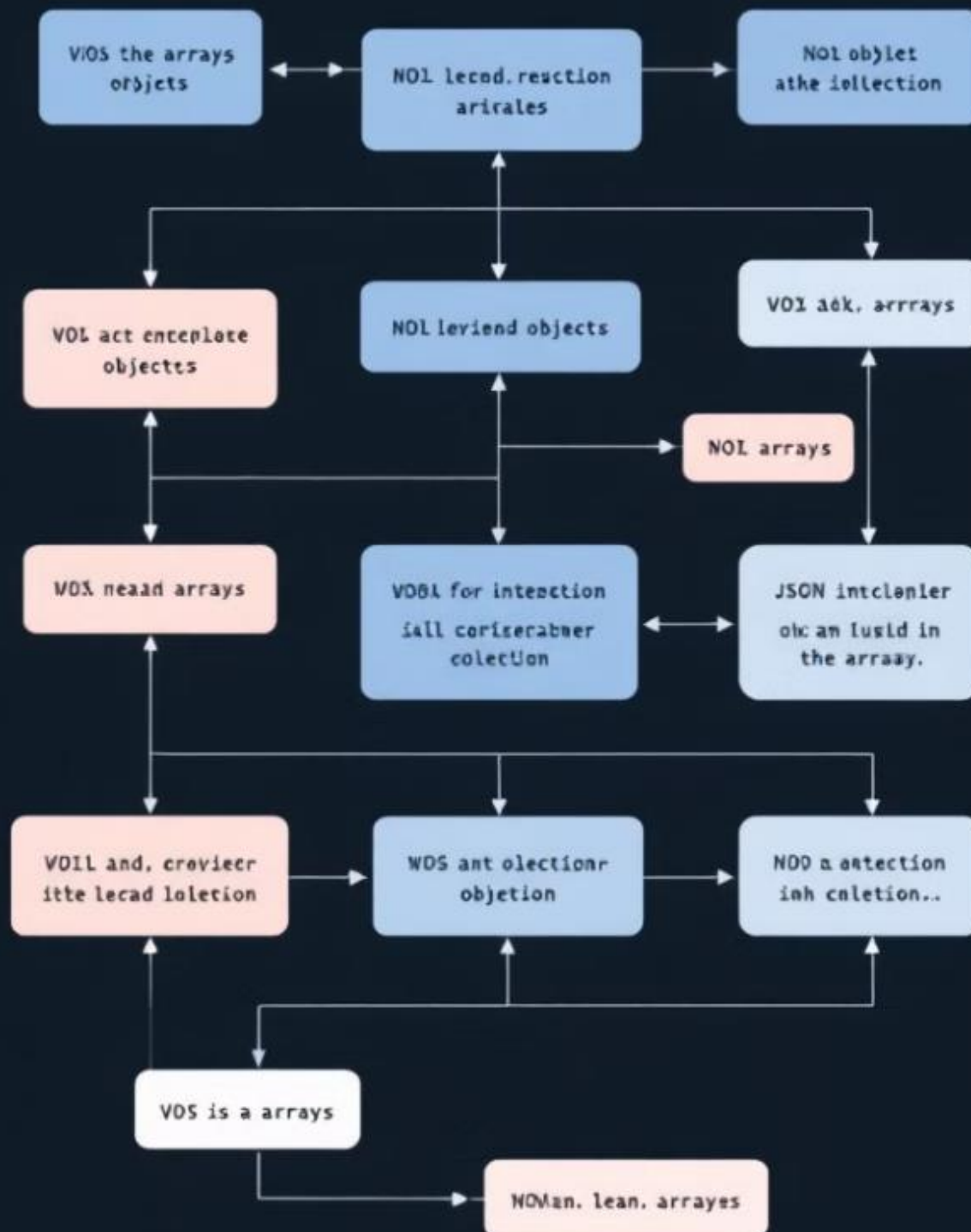
Digunakan untuk menghapus data dari server.

Format Data: JSON

Setelah data diambil dari MySQL melalui skrip PHP, data tersebut dikirim ke aplikasi Android dalam format JSON (JavaScript Object Notation). JSON adalah format data berbasis teks yang ringan dan mudah dibaca, cocok untuk pengiriman data melalui jaringan, dan mudah diproses oleh aplikasi Android. JSON sangat berguna dalam mengirimkan data yang kompleks antara aplikasi dan server dengan cara yang efisien.

- 1 Ringan**
 JSON memiliki struktur yang sederhana, membuatnya efisien untuk transfer data.
- 2 Mudah Dibaca**
 Format JSON mudah dipahami oleh manusia dan mesin.
- 3 Fleksibel**
 JSON dapat mewakili berbagai struktur data, dari yang sederhana hingga kompleks.

- 4 Kompatibel**
 Banyak bahasa pemrograman, termasuk Java di Android, memiliki dukungan bawaan untuk JSON.



Mekanisme Koneksi Android-MySQL melalui PHP

Proses koneksi dari aplikasi Android ke MySQL melibatkan beberapa tahapan. Pertama, aplikasi Android harus memiliki izin untuk mengakses internet. Kemudian, aplikasi mengirimkan request HTTP ke server, misalnya untuk meminta atau mengirim data. Server menerima request dan skrip PHP memprosesnya, berinteraksi dengan MySQL sesuai kebutuhan. Akhirnya, jika ada data yang diminta, server mengirimkan respons dalam format JSON yang akan diproses oleh aplikasi Android.

1

Koneksi Internet

Aplikasi Android memastikan koneksi internet aktif.

2

HTTP Request

Android mengirim permintaan ke server PHP.

3

Proses Server

PHP memproses permintaan dan berinteraksi dengan MySQL.

4

Respons JSON

Server mengirim data kembali ke Android dalam format JSON.

Implementasi CRUD

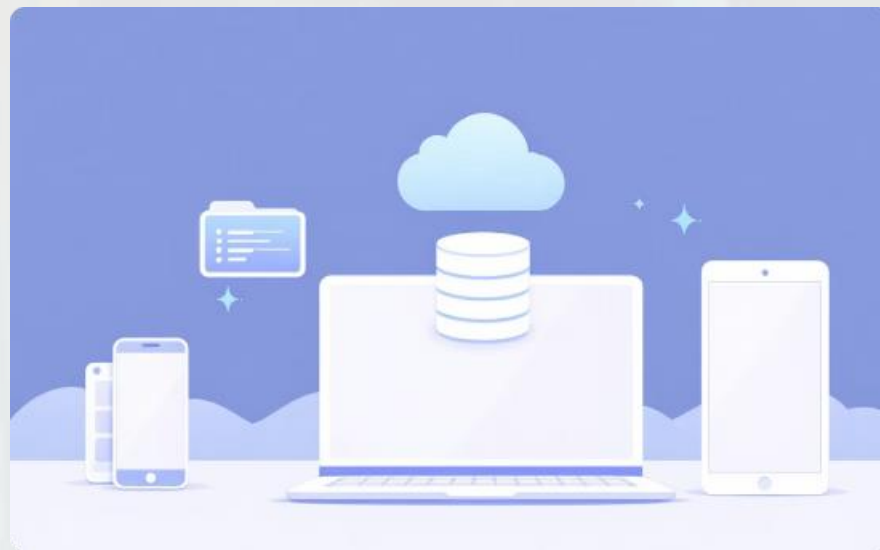


Operasi CRUD (Create, Read, Update, Delete) adalah inti dari pengelolaan data dalam database. Aplikasi Android akan mengirimkan permintaan CRUD melalui HTTP ke server, dan skrip PHP di server akan mengeksekusi perintah tersebut di MySQL. Create digunakan untuk menambah data baru, Read untuk membaca data, Update untuk mengubah data yang ada, dan Delete untuk menghapus data dari tabel.

Operasi	HTTP Method	Fungsi
Create	POST	Menambah data baru
Read	GET	Membaca data
Update	PUT	Mengubah data
Delete	DELETE	Menghapus data

MySQL Database

Menghubungkan Android Studio ke MySQL memberikan beberapa keuntungan penting. Aplikasi Android dapat menyimpan dan mengambil data dari server, memungkinkan data diakses oleh banyak pengguna. Data yang disimpan di server dapat di-backup dan diatur dengan lebih baik. Selain itu, data dapat diakses dari berbagai perangkat yang terhubung ke aplikasi, sehingga sangat cocok untuk aplikasi multi-user seperti e-commerce, media sosial, dan manajemen tugas.



Akses Multi-Perangkat

Data dapat diakses dari berbagai perangkat yang terhubung ke aplikasi.



Keamanan Data

Data yang disimpan di server dapat di-backup dan diatur dengan lebih aman.



Aplikasi Multi-User

Cocok untuk aplikasi seperti e-commerce, media sosial, dan manajemen tugas.

Kesimpulan



MySQL

Menghubungkan MySQL dengan Android Studio membuka berbagai kemungkinan dalam pengembangan aplikasi mobile yang membutuhkan pengelolaan data yang kuat. Dengan menggunakan arsitektur client-server, PHP sebagai perantara, dan format data JSON, pengembang dapat menciptakan aplikasi Android yang dapat mengakses dan memanipulasi data secara efisien di server MySQL. Pendekatan ini tidak hanya meningkatkan fungsionalitas aplikasi tetapi juga memungkinkan skalabilitas dan fleksibilitas yang lebih besar dalam pengelolaan data.



Aplikasi Mobile

Pengembangan aplikasi Android yang lebih canggih.



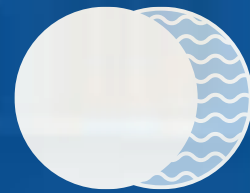
Pengelolaan Data

Manajemen data yang efisien dan terstruktur.



Aksesibilitas

Akses data dari berbagai perangkat dan lokasi.



se k i a n

TERIMA KASIH

