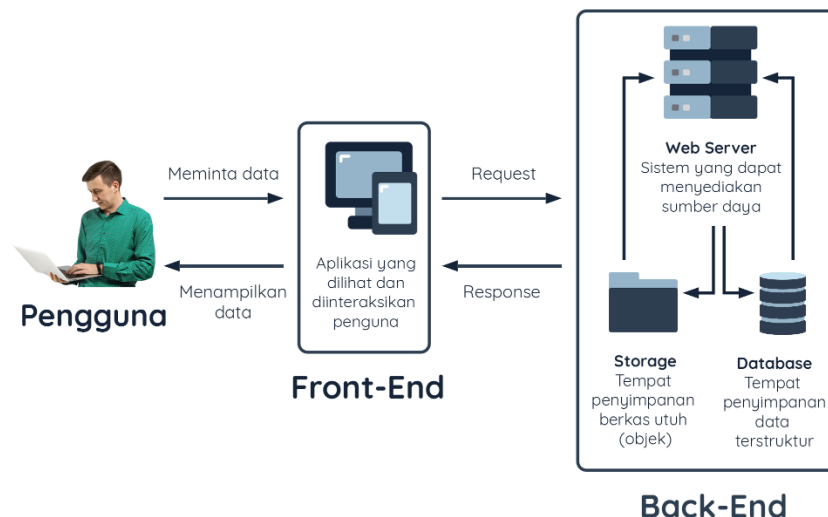


Room Database Bagian 1 (Create, Read)

Database atau basis data adalah kumpulan data yang dikelola sedemikian rupa berdasarkan ketentuan tertentu yang saling berhubungan sehingga mudah dalam pengelolaannya. Melalui pengelolaan tersebut pengguna dapat memperoleh kemudahan dalam mencari informasi, menyimpan informasi dan membuang informasi. Gampangnya sih, **database adalah sistem yang berfungsi sebagai mengumpulkan file, tabel, atau arsip yang terhubung dan disimpan dalam berbagai media elektronik.**

Dalam sebuah aplikasi, biasanya database akan dijadikan sebagai tempat penyimpanan, di mana data yang tersimpan di dalam database akan ditampilkan di dalam aplikasi. Berikut ini adalah contoh akses database yang tersimpan di dalam sebuah server.



Gambar 1. Ilustrasi Penggunaan Database Server dalam Aplikasi

Berdasarkan cara mengaksesnya, database terbagi menjadi 2 bagian, yang pertama yaitu database online, di mana kita membutuhkan server dan akses internet untuk mengakses database ini. Contoh dari database online adalah data akun sosial media milik kita.

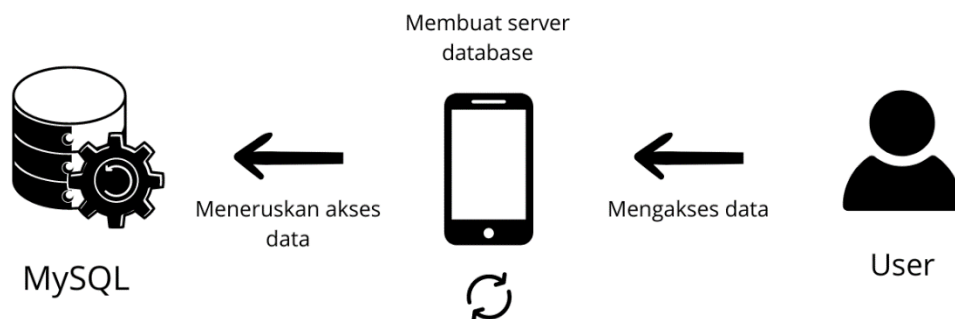
Yang kedua adalah database offline, di mana database tersebut jalan berjalan secara langsung di device kita, sehingga untuk mengaksesnya kita tidak memerlukan internet. Contoh dari database offline adalah history chat dari aplikasi whatsapp.

Berdasarkan jenis dan fungsi, database terbagi menjadi beberapa bagian, yaitu operational database, database warehouse, distributed database, relational database, dan end-user database.

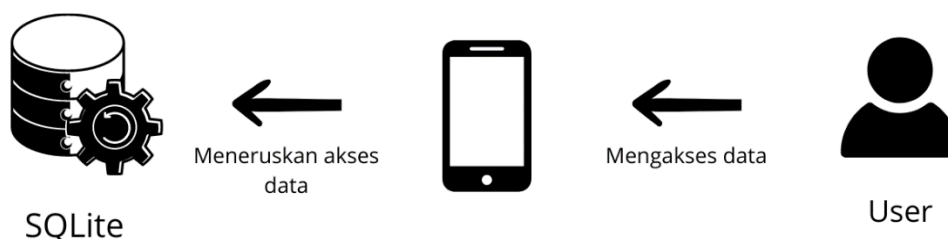
Mungkin kalian cukup familiar dengan relational database, seperti misalnya MySQL, PostgreSQL, dll. Namun kali ini kita akan mempelajari tentang end-user database.

End user database adalah jenis database yang berjalan dan tersimpan di perangkat pribadi milik pengguna (berbanding terbalik dengan database seperti MySQL yang dikenal di simpan di server). Contoh dari penggunaan end user database adalah aplikasi yang diakses secara offline seperti aplikasi catatan, atau salah satu penggunaan yang cukup populer adalah history chat whatsapp.

SQLite adalah salah satu database yang sering digunakan ketika developer ingin menerapkan konsep **End user database**. Alasannya adalah karena SQLite umumnya sudah built-in atau terinstall secara default di dalam sebuah aplikasi, sehingga pengguna tidak perlu untuk melakukan instalasi SQLite secara terpisah karena databasenya sudah tersedia secara langsung di dalam aplikasi yang ingin digunakan. FYI, SQLite dibuat menggunakan Bahasa C, penggunaannya sama seperti SQL database lainnya, namun yang membedakan SQLite dengan SQL database lainnya adalah ukurannya lebih kecil dan lebih cepat.



Gambar 2. Alur mengakses data menggunakan database MySQL



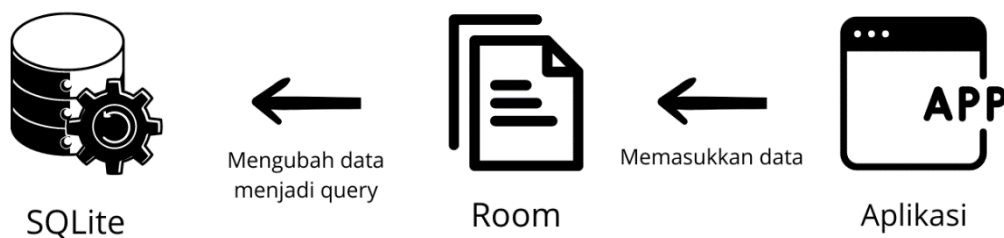
Gambar 3. Alur mengakses data menggunakan database SQLite

Room adalah library buatan Google, Room berjalan di atas SQLite, yang mana Room akan mempermudah kita dalam menggunakan SQLite karena Room memiliki banyak manfaat, salah satunya adalah membuat SQLite yang awalnya Query Oriented menjadi ORM (Object Relational Mapping). Apa itu ORM?

ORM (Object Relational Mapping) adalah konsep mengubah sebuah data menjadi sebuah query. Jadi nantinya proses mengelola data dalam database tidak memerlukan query, sehingga mempermudah bagi user yang menggunakannya.

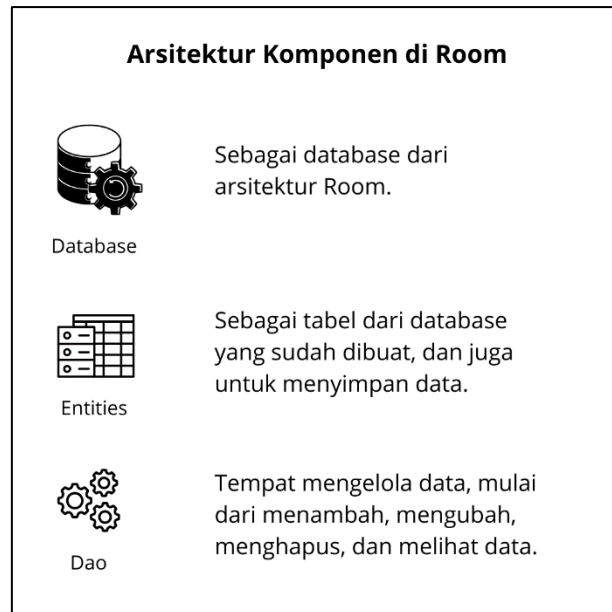


Gambar 4. Proses mengakses database tanpa ORM

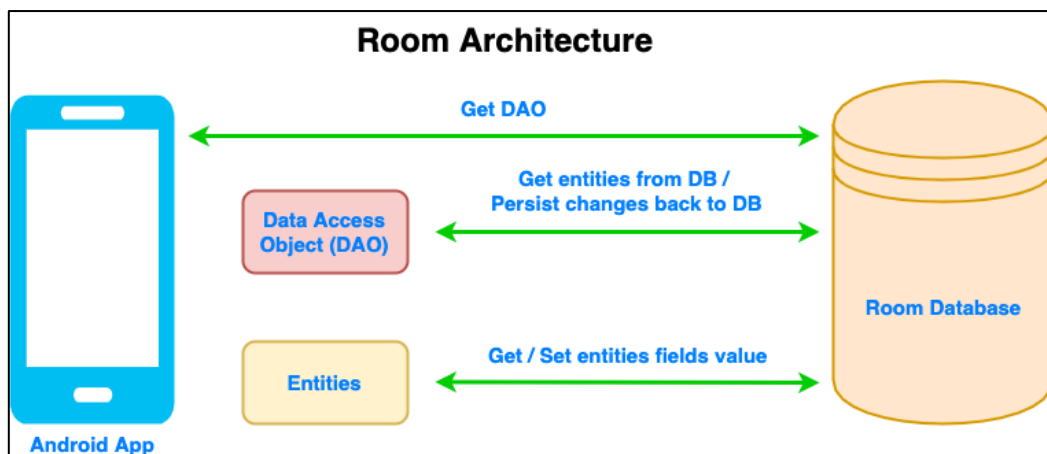


Gambar 5. Proses mengakses database dengan ORM

Manfaat lain Room adalah mengurangi kode boilerplate dan mempermudah migrasi database database.



Gambar 6. Arsitektur Komponen Room



Gambar 7. Alur pengelolaan data di Room

```
@Dao
interface AppDao {

    @Insert(onConflict = OnConflictStrategy.IGNORE)
    fun insertPlayer(player: PlayerDatabase)

    @Update
    fun updatePlayer(player: PlayerDatabase)

    @Delete
    fun deletePlayer(player: PlayerDatabase)

    @Query("SELECT * from playerdatabase ORDER BY player_name ASC")
    fun getAllPlayer(): LiveData<List<PlayerDatabase>>
}
```

Gambar 8. Komponen akses data Room

Ada empat komponen akses data di dalam Room, pada pertemuan kali ini, kita akan membahas komponen Insert dan Update.

Insert

Komponen Insert digunakan untuk memasukkan data ke dalam database. Karena Room mengadopsi konsep ORM, proses memasukkan data dilakukan secara otomatis sehingga kita cukup menuliskan data apa saja yang akan dimasukkan di dalam parameter, nantinya kotlin akan membuat query secara otomatis berdasarkan data yang kita tulis di dalam parameter.

Query

Berbeda dengan konsep ORM yang akan membuat query secara otomatis. Komponen query digunakan untuk membuat query secara manual, sehingga kita perlu menuliskan query yang spesifik berdasarkan kebutuhan kita. Biasanya anotasi query digunakan ketika kita berhubungan dengan melihat data (SELECT), sehingga proses SELECT data bisa dilakukan dengan berbagai macam variasi yang ada (Contoh: melihat data keseluruhan, melihat data berdasarkan nama, melihat data berdasarkan id, dan lain-lain). Namun biarpun begitu, anotasi query bisa digunakan untuk hal lain seperti misalnya Insert Data, Update data, dan masih banyak lagi.