Nama : Mita Pramesti

Nim : 2108096019

Kelas : TI-6A Sistem Komputer

Kuis Pertemuan ke 5

1. Jenis database yang dapat digunakan untuk membangun arsitektur IoT (Internet of Things) antara lain:
2. Static Database: Static databases, juga disebut batch databases, berfungsi untuk menyimpan dan mengelola data yang tidak berubah, yang biasanya disimpan dalam bentuk data yang dapat diakses oleh pengguna melalui sistem manajemen database (DBMS)[1].
3. Streaming Database: Streaming databases mengelola data yang bergerak, yang secara terus-menerus mengalir melalui database. Output dari database streaming biasanya disimpan di luar, seperti di cloud, dan dapat diakses melalui mekanisme query standar[1].
4. Time-Series Database: Time-series databases berfungsi untuk menyimpan dan mengelola data yang berubah secara terus-menerus, yang biasanya digunakan untuk menyimpan data dari perangkat IoT[1].
5. SQL Database: SQL (Structured Query Language) databases menggunakan SQL untuk menyimpan dan mengelola data, yang sering digunakan untuk mengelola data relasi[2].
6. NoSQL Database: NoSQL databases menggunakan teknologi yang berbeda untuk menyimpan

dan mengelola data, yang sering digunakan untuk mengelola data yang tidak dapat disimpan dalam bentuk relasi[2].

1. Graph Database: Graph databases digunakan untuk menyimpan dan mengelola data yang terhubung, yang sering digunakan untuk mengelola data yang dihasilkan oleh perangkat IoT[2].
2. SQL vs. NoSQL: Pilihan antara SQL dan NoSQL tergantung pada kebutuhan dan karakteristik dari proyek IoT. SQL databases lebih tepat untuk data yang dapat disimpan dalam bentuk relasi, sedangkan NoSQL databases lebih tepat untuk data yang tidak dapat disimpan dalam bentuk relasi[1].
3. Unified Database: Database unified merupakan kombinasi dari streaming dan static database, yang memiliki komponen streaming dan komponen statis. Komponen static berbasis SQL dan komponen streaming berbasis teknologi yang berbeda[1].
4. Cloud Database: Cloud database adalah database yang dapat diakses melalui internet, yang sering digunakan untuk mengelola data IoT yang tinggi[4].
5. No SQL Database: No SQL database adalah database yang tidak menggunakan SQL untuk menyimpan dan mengelola data, yang sering digunakan untuk mengelola data IoT yang tinggi[4].
6. SQL Database: SQL database adalah database yang menggunakan SQL untuk menyimpan dan mengelola data, yang sering digunakan untuk mengelola data IoT yang tinggi[4].
7. MQTT Cluster: MQTT (Message Queuing Telemetry Transport) Cluster adalah sistem pengiriman pesan yang digunakan untuk mengirim dan menerima data dari perangkat IoT, yang sering digunakan untuk mengelola komunikasi asinkron antara perangkat IoT dan platform cloud[4].
8. Business Intelligence Tool: Business Intelligence Tool adalah aplikasi yang digunakan untuk mengelola dan menganalisis data, yang sering digunakan untuk mengelola data IoT yang tinggi[4].

Contoh:

- Static Database: MySQL Cluster[3].

- Streaming Database: Amazon Kinesis[5].

- Time-Series Database: InfluxDB[5].

- SQL Database: Oracle, SQL Server, MySQL, IBM DB2, SQLite[4].

- NoSQL Database: MongoDB, Cassandra, Redis[5].

- Graph Database: Neo4j, Amazon Neptune[5].

- Cloud Database: Amazon DynamoDB, Amazon RDS, Amazon Aurora[5].

- MQTT Cluster: HiveMQ, Mosquitto[5].

- Business Intelligence Tool: Tableau, Power BI, QlikView[5].

Citations:

[1] <https://spada.uns.ac.id/mod/resource/view.php?id=159029>

[2] <https://www.researchgate.net/publication/351746535_Perancangan_Database_IoT_Berbasis_Cloud_dengan_Restful_API>

[3] <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/download/9465/4270/66454>

[4] <https://www.dicoding.com/blog/tipe-database/>

[5] <https://aws.amazon.com/id/nosql/>

1. Peran database pada arsitektur sistem IoT (Internet of Things) antara lain:
2. Mengelola Data Bergerak: Database dapat digunakan untuk mengelola data bergerak, yang sering digunakan dalam sistem IoT. Misalnya, database dapat digunakan untuk mengelola data yang bergerak dari sensor, yang dapat digunakan untuk mengirim data ke cloud atau ke lokasi lain[1]
3. Mengelola dan Menyimpan Data: Database dapat digunakan untuk menyimpan dan mengelola data yang dihasilkan oleh perangkat IoT, seperti sensor data, data perangkat lunak, dan data aplikasi. Ini memungkinkan pengguna untuk mengakses dan menganalisis data yang diperlukan untuk mengoptimalkan sistem IoT[2].
4. Mengoptimalkan Kinerja: Database dapat digunakan untuk mengoptimalkan kinerja sistem IoT. Misalnya, database dapat digunakan untuk mengurangi latensi dan mengurangi mengeluarkan data yang tidak diperlukan. Ini memungkinkan sistem IoT untuk beroperasi dengan lebih efisien[2].
5. Mengelola Data yang Berubah Secara Terus-Menerus
6. Mengelola Data yang Terstruktur
7. Mengelola Data yang Tidak Terstruktur
8. Mengelola Data yang Terhubung
9. Mengelola Data yang Bergerak dan Terhubung
10. Mengelola Data yang Bergerak, Terhubung, dan Terstruktur
11. Mengelola Data yang Bergerak, Terhubung, dan Tidak Terstruktur.
12. Mengelola Data yang Bergerak, Tidak Terhubung, dan Terstruktur
13. Mengelola Data yang Bergerak, Tidak Terhubung, dan Tidak Terstruktur

Citations:

[1] <https://www.researchgate.net/publication/351746535_Perancangan_Database_IoT_Berbasis_Cloud_dengan_Restful_API>

[2] <https://www.intel.co.id/content/www/id/id/artificial-intelligence/database.html>

1. Cara mengamankan sistem IoT agar tidak mudah terkena serangan atau kebocoran data antara lain:
2. Pembaruan Perangkat Lunak (Firmware): Pastikan perangkat IoT selalu menjalani pembaruan perangkat lunak terbaru, yang seringkali mencakup perbaikan keamanan yang penting[2] karena Software yang *out of date* biasanya rentan terkena bug.
3. Kata Sandi yang Kuat: Gunakan kata sandi yang kuat dan berlaku untuk mengelola akses ke perangkat IoT[2].
4. Membatasi fungsionalitas perangkat dan perangkat lunak yang dapat dijalankannya, kemungkinan penyalahgunaan perangkat menjadi terbatas
5. Enkripsi Data: Gunakan enkripsi data untuk mengelilingkan data yang diperlukan antara perangkat IoT dan server[2].
6. Otorisasi Akses: Gunakan sistem otorisasi akses, seperti OAuth, untuk memastikan bahwa hanya pihak yang sah memiliki izin untuk mengakses data atau mengontrol perangkat IoT[2].
7. Deteksi dan Respons terhadap Ancaman: Gunakan sistem deteksi ancaman yang efektif dan merespons serangan atau insiden keamanan dengan cepat[4].
8. Pemisahan Data Sensitif: Memisahkan dan mengenkripsi data sensitif yang diproses oleh perangkat IoT, sehingga data sensitif tetap terlindungi bahkan jika perangkat IoT terinfeksi atau dikompromikan[4].
9. Pendidikan dan Kesadaran Pengguna: Jaringan IoT dapat mengakses data pribadi pengguna, seperti informasi kesehatan, lokasi, atau preferensi. Jaringan IoT juga dapat menjadi sumber ancaman untuk privasi pengguna. Dengan mengajarkan pengguna tentang pentingnya keamanan IoT, termasuk cara terbaik untuk mengelola perangkat IoT dan menjaga keamanan data, kita dapat membantu mengurangi risiko privasi[4].
10. Penggunaan Blockchain: Penggunaan blockchain dapat membantu menciptakan catatan transaksi yang aman dan auditabilitas dalam jaringan IoT[4].
11. Perlindungan dan Metode Keamanan yang Kuat: Perlindungan dan metode keamanan yang kuat harus diperbarui secara teratur untuk melindungi sistem IoT dari serangan dan pelanggaran keamanan[4].
12. Pengamanan Perangkat Keras: Mempertahankan perangkat keras yang mengelilingkan perangkat IoT, seperti kunci fisik, pengamanan perangkat keras, atau penggunaan tempat penyimpanan yang aman, dapat mencegah orang yang tidak berhak mengaksesnya atau mencurinya[4].
13. Perancangan Sistem Keamanan: Membuat perancangan sistem keamanan yang tepat, yang meliputi perancangan antarmuka dan sistem, diagram alur, use case diagram, dan perancangan basis data, dapat membantu mengamankan sistem IoT[5].
14. Standar Keamanan: Mematuhi standar keamanan yang tepat, seperti "IoT Security Foundation" dan "Industrial Internet Consortium" (IIC), dapat membantu melindungi perangkat IoT dari serangan siber dan memastikan bahwa data sensitif dan privasi pengguna tetap terlindungi[2][4].

Citations:

[1] <https://www.intel.co.id/content/www/id/id/artificial-intelligence/database.html>

[2] <https://biztech.proxsisgroup.com/keamanan-internet-of-things-iot-menjaga-perangkat-anda-dari-serangan/>

[3] <https://spada.uns.ac.id/mod/resource/view.php?id=159029>

[4] <https://www.jsisfotek.org/index.php/JSisfotek/article/download/232/167>

[5] <https://dspace.uii.ac.id/bitstream/handle/123456789/28890/16523057%20Rauf%20Endro%20Widagdo.pdf?isAllowed=y&sequence=1>

1. Berdasarkan analisis saya, saya menemukan bahwa jenis database IoT yang digunakan di gedung FST UIN Walisongo adalah NoSQL karena jenis database ini lebih fleksibel untuk menangani data yang seringkali semi-struktural atau tidak terstruktur yang dihasilkan oleh sensor dan perangkat IoT yang kompleks di gedung FST. Berikut adalah alur kerjanya.
2. Pengumpulan Data

Data sensor dari perangkat IoT di gedung FST dikumpulkan secara terus-menerus. Pengumpulan data bisa dilakukan secara langsung dari perangkat sensor atau melalui gateway IoT.

1. Penyimpanan Data

Data yang dikumpulkan kemudian disimpan dalam basis data NoSQL. Contoh basis data NoSQL yang umum digunakan untuk IoT termasuk MongoDB, Cassandra, InfluxDB, dan Apache Cassandra.

1. Pemrosesan Data

Data yang disimpan dalam basis data NoSQL kemudian diproses untuk mempersiapkan data untuk analisis lebih lanjut. Pemrosesan data dapat mencakup transformasi data, normalisasi, penghapusan data duplikat, dan pemfilteran data yang tidak relevan.

1. Analisis Data

Setelah data diproses, analisis data dilakukan untuk mendapatkan wawasan yang berharga dari data sensor. Ini bisa meliputi analisis tren, deteksi anomali, pengidentifikasian pola, dan prediksi berdasarkan data historis.

1. Visualisasi Data

Hasil analisis data sering kali divisualisasikan untuk memudahkan pemahaman dan pengambilan keputusan. Grafik, diagram, dan peta sering digunakan untuk memvisualisasikan data sensor dalam konteks yang mudah dimengerti.

1. Aksi Otomatis

Dalam beberapa kasus, sistem IoT dapat dikonfigurasi untuk melakukan tindakan otomatis berdasarkan hasil analisis data. Contohnya, pengiriman notifikasi atau perintah otomatis kepada perangkat IoT untuk mengatur kondisi lingkungan tertentu berdasarkan kondisi yang dideteksi.

1. Monitoring dan Pemeliharaan

Selama dan setelah implementasi sistem IoT, monitoring terus-menerus dilakukan untuk memastikan kinerja sistem yang optimal. Pemeliharaan dan pemecahan masalah dilakukan secara teratur untuk menjaga keandalan dan keamanan sistem.