

Sistem BASIS DATA

Basis data adalah kumpulan data yang disimpan secara terstruktur dan memungkinkan pengguna untuk mengakses dan memanipulasi data tersebut. Basis data adalah koleksi terorganisir dari data yang dapat diakses dan dikelola dengan mudah. Data yang tersimpan dalam basis data dapat berkisar dari informasi sederhana seperti daftar nama hingga informasi yang lebih kompleks, seperti data keuangan atau catatan medis. Buku ini membahas tentang Dasar Basis Data dan Sistem Basis Data, Model Data, Normalisasi Data, Konsep SQL, dan Pengembangan Aplikasi Sistem Basis Data.

SISTEM BASIS DATA

Andi Asari, dkk



Andi Asari, Bernadete Deta,
Syahrial, Hidayatus Sibyan, Tri Pratiwi Handayani



PT MAFY MEDIA LITERASI INDONESIA
Email : penerbitmafya@gmail.com
Website : penerbitmafya.com
FB : Penerbit Mafy



Sistem BASIS DATA

Sistem Basis Data

UU No 28 Tahun 2014 tentang Hak Cipta

Fungsi dan sifat hak cipta Pasal 4

Hak Cipta sebagaimana dimaksud dalam Pasal 3 huruf a merupakan hak eksklusif yang terdiri atas hak moral dan hak ekonomi.

Pembatasan Pelindungan Pasal 26

Ketentuan sebagaimana dimaksud dalam Pasal 23, Pasal 24, dan Pasal 25 tidak berlaku terhadap:

- i. penggunaan kutipan singkat ciptaan dan/atau produk hak terkait untuk pelaporan peristiwa aktual yang ditujukan hanya untuk keperluan penyediaan informasi aktual;
- ii. penggandaan ciptaan dan/atau produk hak terkait hanya untuk kepentingan penelitian ilmu pengetahuan;
- iii. penggandaan ciptaan dan/atau produk hak terkait hanya untuk keperluan pengajaran, kecuali pertunjukan dan fonogram yang telah dilakukan pengumuman sebagai bahan ajar; dan
- iv. penggunaan untuk kepentingan pendidikan dan pengembangan ilmu pengetahuan yang memungkinkan suatu ciptaan dan/atau produk hak terkait dapat digunakan tanpa izin pelaku pertunjukan, produser fonogram, atau lembaga penyiaran.

Sanksi Pelanggaran Pasal 113

1. Setiap orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak Rp100.000.000 (seratus juta rupiah).
2. Setiap orang yang dengan tanpa hak dan/atau tanpa izin pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf c, huruf d, huruf f, dan/atau huruf h untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan/atau pidana denda paling banyak Rp500.000.000,00 (lima ratus juta rupiah).

Sistem Basis Data

**Andi Asari, Bernadete Deta, Syahrial,
Hidayatus Sibyan dan Tri Pratiwi Handayani**



SISTEM BASIS DATA

Penulis:

**Andi Asari, Bernadete Deta, Syahril,
Hidayatus Sibyan dan Tri Pratiwi Handayani**

Editor:

Tiya Arika Marlin

Desainer:

Tim Mafy

Sumber Gambar Cover:

www.freepik.com

Ukuran:

vi, 141 hlm, 15,5 cm x 23 cm

ISBN:

978-623-8726-77-6

Cetakan Pertama:

Agustus 2024

Hak Cipta Dilindungi oleh Undang-Undang. Dilarang menerjemahkan, memfotokopi, atau memperbanyak sebagian atau seluruh isi buku ini tanpa izin tertulis dari Penerbit.

PT MAFY MEDIA LITERASI INDONESIA

ANGGOTA IKAPI 041/SBA/2023

Kota Solok, Sumatera Barat, Kode Pos 27312

Kontak: 081374311814

Website: www.penerbitmafy.com

E-mail: penerbitmafy@gmail.com

Prakata

Segala puji syukur kami panjatkan kepada Tuhan Yang Maha Esa, karena atas pertolongan dan limpahan rahmat-Nya sehingga penulis bisa menyelesaikan buku yang berjudul **Sistem Basis Data** ini disusun secara lengkap dengan tujuan untuk memudahkan para pembaca memahami isi buku ini. Buku ini membahas tentang Dasar Basis Data dan Sistem Basis Data, Model Data, Normalisasi Data, Konsep SQL, dan Pengembangan Aplikasi Sistem Basis Data.

Kami menyadari bahwa buku yang ada di tangan pembaca ini masih banyak kekurangan. Maka dari itu kami sangat mengharapkan saran untuk perbaikan buku ini di masa yang akan datang. Dan tidak lupa kami mengucapkan terima kasih kepada semua pihak yang telah membantu dalam proses penerbitan buku ini. Semoga buku ini dapat membawa manfaat dan dampak positif bagi para pembaca.

Penulis, Malang 1 Agustus 2024

Daftar Isi

Prakata	v
BAB I.	
Dasar Basis Data dan Sistem Basis Data	1
BAB II.	
Model Data	21
BAB III.	
Normalisasi Data	39
BAB IV.	
Konsep SQL	67
BAB V.	
Pengembangan Aplikasi Sistem Basis Data	89
Kesimpulan	135
Tentang Penulis	137

BAB I

Dasar Basis Data dan Sistem Basis Data

-Andi Asari-

A. Pengertian Basis Data

Basis data adalah kumpulan data yang disimpan secara terstruktur dan memungkinkan pengguna untuk mengakses dan memanipulasi data tersebut. Menurut Connolly dan Begg (2015), "Basis data adalah koleksi terorganisir dari data yang dapat diakses dan dikelola dengan mudah." Data yang tersimpan dalam basis data dapat berkisar dari informasi sederhana seperti daftar nama hingga informasi yang lebih kompleks, seperti data keuangan atau catatan medis.

Jenis-jenis Basis Data

Terdapat beberapa jenis basis data yang umum digunakan, di antaranya:

1. Basis Data Relasional

Menggunakan tabel untuk menyimpan data. Data terstruktur dalam baris dan kolom, dan hubungan antardata dikelola melalui kunci. Contoh: MySQL, PostgreSQL.

2. Basis Data NoSQL

Dirancang untuk menangani data yang tidak terstruktur atau semi-terstruktur. Tipe ini cocok untuk aplikasi yang memerlukan skalabilitas tinggi. Contoh: MongoDB, Cassandra.

3. Basis Data Terdistribusi

Basis data yang tersebar di beberapa lokasi fisik dan dapat diakses dari berbagai server. Ini meningkatkan ketersediaan dan ketahanan data.

4. Basis Data Objek

Menyimpan data dalam bentuk objek, mirip dengan pemrograman berorientasi objek. Contoh: DB4O, ObjectDB.

Manfaat Basis Data

1. Pengelolaan Data yang Efisien

Pengguna dapat mengelola data dalam jumlah besar dengan lebih mudah.

2. Akses Data yang Cepat

Query yang dioptimalkan memungkinkan pengambilan data yang cepat.

3. Keamanan Data

Dengan sistem kontrol akses, data dapat dilindungi dari pengguna yang tidak berwenang.

4. Pengambilan Keputusan yang Lebih Baik

Analisis data yang mendalam memungkinkan perusahaan untuk membuat keputusan yang lebih informasi.

B. Pengertian Sistem Basis Data

Sistem basis data (Database Management System atau DBMS) merupakan perangkat lunak yang dirancang untuk mengelola, menyimpan, dan memanipulasi data secara efisien. Sistem ini memungkinkan pengguna untuk mengakses, mengatur, dan memanipulasi data dalam struktur yang terorganisir dengan baik. Dalam konteks yang lebih luas, sistem basis data memainkan peran kunci dalam pengelolaan informasi di berbagai aplikasi dan industri.

Sistem basis data dapat didefinisikan sebagai sistem perangkat lunak yang menyimpan data secara terstruktur dan menyediakan alat untuk mengelola dan mengakses data tersebut. Menurut Elmasri dan Navathe (2022), Sistem basis data adalah perangkat lunak yang memungkinkan pengguna untuk menyimpan, mengelola, dan mengakses data dalam bentuk yang terstruktur, serta menyediakan mekanisme untuk pemrosesan data secara efisien dan konsisten (Elmasri, R., & Navathe, S. B., 2022).

Komponen Utama dari Sistem Basis Data

Sistem basis data terdiri dari beberapa komponen utama, termasuk:

1. Basis Data

Merupakan kumpulan data yang terorganisir dalam struktur tertentu seperti tabel, yang disimpan dalam media penyimpanan.

2. DBMS

Perangkat lunak yang mengelola basis data, menyediakan antarmuka untuk akses data, serta menangani fungsi seperti pencarian, penyimpanan, dan pembaruan data.

3. Model Data

Konsep yang digunakan untuk mendefinisikan struktur data dan hubungan antardata. Model data relasional, misalnya, menggunakan tabel untuk merepresentasikan data dan hubungan antara tabel.

4. Bahasa Query

Bahasa yang digunakan untuk berinteraksi dengan basis data, seperti SQL (Structured Query Language).

Menurut Date (2023), "Komponen-komponen utama dari sistem basis data, termasuk DBMS, model data, dan bahasa query, bekerja bersama untuk memberikan kerangka kerja yang terintegrasi dalam mengelola informasi secara efektif" (Date, C. J., 2023).

Jenis Sistem Basis Data

Sistem basis data dapat dikategorikan menjadi beberapa jenis berdasarkan model data yang digunakan:

1. Basis Data Relasional

Mengorganisasi data dalam bentuk tabel yang saling berhubungan. Contoh: MySQL, PostgreSQL.

2. Basis Data NoSQL

Mengelola data yang tidak terstruktur atau semi-terstruktur, seperti dokumen. Contoh: MongoDB.

3. Basis Data Berorientasi Objek

Menggunakan konsep objek dalam pemrograman untuk menyimpan data. Contoh: db4o, ObjectDB.

Menurut Morris *et al.* (2022), Basis data relasional tetap menjadi yang paling umum digunakan untuk aplikasi tradisional, sementara basis data NoSQL telah muncul sebagai solusi untuk menangani data yang besar dan tidak terstruktur (Morris, D., Johnson, T., & Wang, X., 2022).

Fungsi dan Manfaat Sistem Basis Data

Sistem basis data menyediakan berbagai manfaat, antara lain:

1. Pengelolaan Data Terpusat

Memudahkan pengelolaan data dari satu tempat yang terpusat.

2. Integritas Data

Menjamin konsistensi dan akurasi data melalui aturan dan batasan yang diterapkan.

3. Keamanan Data

Menyediakan mekanisme untuk kontrol akses dan enkripsi data.

4. Kemampuan Query

Memungkinkan pengguna untuk mengekstrak informasi spesifik dari data yang besar dengan menggunakan bahasa query.

Menurut Chen et al. (2024), Manfaat utama dari sistem basis data mencakup pengelolaan data yang efisien, integritas data yang terjaga, serta kemampuan untuk melakukan query dan analisis data yang kompleks (Chen, J., Zhang, L., & Liu, H., 2024).

C. Perbedaan Basis Data dan Sistem Basis Data

Basis data (database) dan sistem basis data (DBMS - Database Management System) adalah dua konsep yang sering digunakan dalam dunia teknologi informasi dan manajemen data. Meskipun sering kali digunakan secara bergantian, keduanya memiliki perbedaan mendasar yang penting untuk dipahami.

Basis Data (Database)

Basis data adalah kumpulan data yang terorganisir dengan cara tertentu, biasanya disimpan secara elektronik dalam sistem

komputer. Data dalam basis data diatur sedemikian rupa sehingga memudahkan akses, pengelolaan, dan pembaruan informasi. Basis data bisa berbentuk file teks, spreadsheet, atau struktur yang lebih kompleks seperti tabel-tabel dalam sistem manajemen basis data relasional.

Sistem Basis Data (Database Management System - DBMS)

Sistem basis data adalah perangkat lunak yang dirancang untuk mengelola basis data. DBMS menyediakan antarmuka antara pengguna dan basis data, memungkinkan pengguna untuk melakukan operasi seperti penyimpanan, pengambilan, pemeliharaan, dan pengelolaan data. DBMS juga menyediakan berbagai fitur seperti kontrol akses, integritas data, dan pemulihan data setelah kegagalan.

Perbedaan Utama antara Basis Data dan Sistem Basis Data

1. Definisi dan Ruang Lingkup

Basis data adalah kumpulan data yang terstruktur. DBMS adalah perangkat lunak yang mengelola basis data dan menyediakan antarmuka untuk interaksi dengan data.

2. Fungsi

Basis data menyimpan dan mengatur data. DBMS menangani operasi terkait data, seperti penyimpanan, pencarian, pembaruan, dan pengelolaan akses.

3. Contoh

Basis data bisa berupa file Excel yang berisi informasi pelanggan. DBMS bisa berupa perangkat lunak seperti MySQL, PostgreSQL, atau Oracle yang mengelola file Excel tersebut serta data lainnya.

4. Pengelolaan

Basis data biasanya memerlukan perangkat lunak khusus untuk dikelola dengan efisien. DBMS menyediakan berbagai fitur untuk pengelolaan basis data, termasuk keamanan, pemeliharaan, dan pengendalian akses.

5. Kompleksitas

Basis data sendiri tidak memiliki kemampuan pemrosesan atau manajemen. DBMS menawarkan kemampuan kompleks untuk mengelola data, melakukan query, dan menjaga integritas data.

D. Manfaat Basis Data di Era Digital

Basis data merupakan komponen krusial dalam dunia digital modern. Mereka memainkan peran sentral dalam menyimpan, mengelola, dan mengakses informasi secara efisien. Seiring dengan kemajuan teknologi dan meningkatnya volume data, pemahaman tentang manfaat basis data menjadi semakin penting bagi organisasi dan individu. Artikel ini akan membahas beberapa manfaat utama basis data, didukung oleh referensi terbaru.

1. Pengelolaan Data yang Efisien

Basis data memungkinkan pengelolaan data yang terstruktur dan efisien. Dengan menggunakan sistem manajemen basis data (DBMS), organisasi dapat menyimpan data dalam format yang terorganisir, memungkinkan akses dan manipulasi data yang lebih cepat dan lebih mudah. Menurut R. Elmasri dan S. Navathe dalam *Fundamentals of Database Systems* (2021), "DBMS memberikan alat dan struktur untuk mengelola data secara efektif, memastikan integritas dan keamanan informasi yang disimpan" (Elmasri & Navathe, 2021).

2. Keamanan dan Perlindungan Data

Salah satu manfaat utama dari basis data adalah kemampuannya untuk menyediakan lapisan keamanan tambahan untuk data. Basis data modern menawarkan berbagai fitur keamanan, termasuk kontrol akses berbasis peran, enkripsi data, dan audit trail. Dalam studi oleh C. Batini *et al.* (2023), disebutkan bahwa "Keamanan data dalam sistem basis data sangat penting untuk melindungi informasi sensitif dan mencegah akses tidak sah, yang menjadi perhatian utama di era digital saat ini" (Batini, C., *et al.*, 2023).

3. Dukungan untuk Pengambilan Keputusan

Basis data juga mendukung pengambilan keputusan berbasis data dengan menyediakan alat untuk analisis data yang mendalam. Dengan fitur-fitur seperti kueri SQL dan analisis multidimensi, organisasi dapat menghasilkan laporan

dan wawasan yang berguna. Menurut L. Chen dan J. Zhang dalam *Big Data and Business Intelligence: A Guide for Data-Driven Decision Making* (2022), "Kemampuan untuk menganalisis data secara real-time dan menghasilkan laporan yang relevan memungkinkan pengambilan keputusan yang lebih cepat dan lebih terinformasi" (Chen & Zhang, 2022).

4. Skalabilitas dan Fleksibilitas

Seiring dengan pertumbuhan volume data, kebutuhan akan skalabilitas dalam sistem basis data menjadi semakin penting. Basis data modern dirancang untuk menangani data dalam jumlah besar dan dapat diskalakan sesuai dengan kebutuhan. Sebagaimana diungkapkan oleh A. He, et al. dalam *Database Systems for Advanced Applications* (2023), "Sistem basis data kontemporer dirancang untuk menyediakan skalabilitas horizontal dan vertikal, memastikan kinerja yang optimal bahkan ketika data berkembang pesat" (He, A., et al., 2023).

5. Integrasi dan Interoperabilitas

Basis data mendukung integrasi berbagai sumber data dan sistem. Dengan adanya standar dan protokol komunikasi seperti ODBC dan JDBC, basis data dapat berfungsi secara interoperabel dengan aplikasi lain. P. O'Neil dan D. O'Neil dalam *Database: Principles, Programming, and Performance* (2022) menulis bahwa "Kemampuan basis data untuk berintegrasi dengan sistem lain dan menggabungkan data dari berbagai sumber adalah kunci untuk menciptakan solusi

teknologi yang holistik dan efektif" (O'Neil & O'Neil, 2022).

Tujuan Basis Data dalam Pengelolaan Informasi Modern

Basis data adalah fondasi utama dalam pengelolaan informasi di era digital. Sistem basis data dirancang untuk memenuhi berbagai tujuan yang mendukung efisiensi, integritas, dan aksesibilitas data. Memahami tujuan utama dari basis data membantu organisasi dan individu memanfaatkan teknologi ini secara maksimal.

1. Menyimpan Data Secara Terstruktur

Salah satu tujuan utama dari basis data adalah untuk menyimpan data dalam format yang terstruktur. Basis data menyediakan cara yang sistematis untuk mengorganisasi data, yang memudahkan pencarian dan pengelolaan informasi. Menurut Silberschatz, Korth, dan Sudarshan dalam *Database System Concepts* (2021), "Basis data dirancang untuk menyimpan data dengan cara yang terorganisir dan efisien, meminimalkan redundansi dan memastikan integritas data" (Silberschatz, Korth, & Sudarshan, 2021).

2. Menjamin Konsistensi dan Integritas Data

Basis data bertujuan untuk menjaga konsistensi dan integritas data. Dengan menerapkan aturan dan batasan yang ketat, sistem basis data memastikan bahwa data yang disimpan adalah akurat dan dapat diandalkan. Sebagai contoh, Zhang et al. dalam *Database Management Systems: Theory and Practice* (2023) menjelaskan bahwa "DBMS memiliki mekanisme untuk memvalidasi data dan memastikan bahwa

data tetap konsisten melalui aturan integritas dan transaksi yang dikelola secara otomatis" (Zhang, L., et al., 2023).

3. Memfasilitasi Akses Data yang Efisien

Tujuan lain dari basis data adalah untuk memungkinkan akses data yang cepat dan efisien. Basis data menyediakan mekanisme untuk melakukan kueri dan mengambil data dengan cepat, bahkan dalam set data yang besar. Menurut Elmasri dan Navathe dalam *Fundamentals of Database Systems* (2021), "DBMS dirancang untuk mendukung operasi baca dan tulis yang efisien, mengoptimalkan waktu respons dan kinerja kueri" (Elmasri & Navathe, 2021, hlm. 22).

4. Mendukung Pengambilan Keputusan Berbasis Data

Basis data membantu dalam pengambilan keputusan berbasis data dengan menyediakan alat untuk analisis dan pelaporan. Dengan kemampuan untuk mengekstrak dan menganalisis data, organisasi dapat membuat keputusan yang lebih terinformasi. Dalam *Big Data and Business Intelligence: A Guide for Data-Driven Decision Making* (2022), Chen dan Zhang menyatakan bahwa "Basis data yang baik memungkinkan pengguna untuk menganalisis data secara mendalam, menghasilkan wawasan yang dapat digunakan untuk strategi bisnis dan pengambilan keputusan" (Chen & Zhang, 2022).

5. Menyediakan Keamanan dan Perlindungan Data

Basis data juga bertujuan untuk melindungi data dari akses yang tidak sah dan kerusakan. Sistem basis data modern

memiliki berbagai fitur keamanan untuk menjaga kerahasiaan dan integritas data. Menurut C. Batini et al. dalam *Data and Information Quality: Dimensions, Principles, and Techniques* (2023), "Fitur keamanan yang ada dalam DBMS, seperti kontrol akses dan enkripsi, sangat penting untuk melindungi data dari ancaman eksternal dan internal" (Batini, C., et al., 2023).

6. Memudahkan Integrasi Data

Basis data memudahkan integrasi data dari berbagai sumber. Dengan menggunakan standar dan protokol komunikasi, basis data dapat menggabungkan data dari sistem yang berbeda, memberikan pandangan yang menyeluruh. Dalam *Database Systems for Advanced Applications* (2023), He et al. menjelaskan bahwa "Kemampuan basis data untuk mengintegrasikan dan menyelaraskan data dari berbagai sumber adalah kunci untuk sistem informasi yang holistik" (He, A., et al., 2023).

7. Mendukung Skalabilitas

Tujuan lain dari basis data adalah untuk menyediakan skalabilitas. Basis data modern dirancang untuk menangani volume data yang terus berkembang dan memungkinkan sistem untuk berkembang seiring dengan pertumbuhan data. Menurut O'Neil dan O'Neil dalam *Database: Principles, Programming, and Performance* (2022), "Sistem basis data kontemporer dapat diskalakan baik secara vertikal maupun horizontal untuk memenuhi kebutuhan yang semakin besar

dari data dan pengguna" (O'Neil & O'Neil, 2022).

Keunggulan Basis Data dalam Era Digital

Basis data merupakan komponen krusial dalam manajemen informasi di era digital saat ini. Dengan kemajuan teknologi dan kebutuhan untuk mengelola informasi secara efektif, basis data telah berkembang pesat dalam hal kemampuan dan keunggulan. Berikut ini adalah beberapa keunggulan utama dari basis data yang relevan dengan perkembangan terkini.

1. Manajemen Data yang Efisien

Salah satu keunggulan utama basis data adalah kemampuannya untuk mengelola data secara efisien. Basis data modern menggunakan struktur data yang memungkinkan penyimpanan, pencarian, dan pemrosesan informasi dengan cepat. Menurut Liu *et al.* (2021), "Basis data relasional dan non-relasional menawarkan berbagai model penyimpanan yang memungkinkan organisasi untuk memilih solusi yang paling sesuai dengan kebutuhan mereka, meningkatkan efisiensi operasional" (Liu, X., *et al.*, 2021)

2. Keamanan Data yang Kuat

Dalam konteks keamanan data, basis data telah mengimplementasikan berbagai fitur untuk melindungi informasi dari akses yang tidak sah. Salah satu inovasi terbaru adalah penggunaan enkripsi dan kontrol akses berbasis peran. Menurut Zhang dan Liu (2022), "Dengan adopsi teknologi enkripsi canggih dan kontrol akses yang granular, basis data modern mampu menjaga integritas dan kerahasiaan data

secara lebih efektif” (Zhang, Y., & Liu, W., 2022).

3. Skalabilitas dan Fleksibilitas

Basis data juga menawarkan skalabilitas yang tinggi, memungkinkan organisasi untuk menambah kapasitas penyimpanan seiring pertumbuhan data. Basis data NoSQL, seperti MongoDB dan Cassandra, terkenal dengan kemampuannya untuk menangani volume data yang besar dan transaksi yang kompleks. Menurut Miao *et al.* (2023), “Sistem basis data NoSQL memberikan fleksibilitas dalam hal struktur data dan kapasitas, sehingga ideal untuk aplikasi dengan kebutuhan skala besar dan bervariasi” (Miao, H., *et al.*, 2023).

4. Kemampuan Analisis dan Pengolahan Data

Basis data modern tidak hanya menyimpan data tetapi juga menyediakan alat untuk analisis dan pengolahan data. Integrasi dengan teknologi big data dan machine learning memungkinkan organisasi untuk melakukan analisis prediktif dan menemukan wawasan berharga. Menurut Wang *et al.* (2022), “Integrasi basis data dengan alat analitik canggih dan teknologi big data memungkinkan perusahaan untuk mengolah dan menganalisis data dengan cara yang sebelumnya tidak mungkin dilakukan” (Wang, Z., *et al.*, 2022)

5. Ketersediaan dan Pemulihan Bencana

Kemampuan basis data untuk mendukung ketersediaan tinggi dan pemulihan bencana juga merupakan keunggulan signifikan. Teknologi seperti replikasi data dan backup

otomatis memastikan bahwa data dapat dipulihkan dengan cepat setelah terjadinya gangguan. Menurut Chen *et al.* (2021), “Replikasi data dan strategi pemulihan bencana yang baik memastikan bahwa data tetap tersedia dan dapat diakses bahkan dalam kondisi sistem yang gagal” (Chen, J., *et al.*, 2021).

Kelemahan Sistem Basis Data dalam Era Modern

Sistem basis data, meskipun sangat penting dalam manajemen informasi, memiliki berbagai kelemahan yang dapat mempengaruhi efisiensinya dan keselamatan data. Dengan pesatnya perkembangan teknologi dan adopsi sistem basis data yang semakin luas, penting untuk memahami kelemahan-kelemahan ini agar dapat mengatasi potensi masalah yang mungkin timbul. Berikut ini adalah beberapa kelemahan utama dari sistem basis data beserta kutipan dan referensi terbaru.

1. Keterbatasan Skalabilitas pada Sistem Relasional

Sistem basis data relasional, meskipun sangat populer, sering kali mengalami keterbatasan dalam hal skalabilitas saat menghadapi volume data yang sangat besar. Ketika data dan beban transaksi meningkat secara eksponensial, sistem relasional bisa mengalami penurunan performa. Menurut Aggarwal dan Yu (2021), “Sistem basis data relasional sering menghadapi tantangan dalam hal skalabilitas horizontal, terutama saat volume data melebihi kapasitas sistem yang dapat ditangani dengan efisien” (Aggarwal, C. C., & Yu, P. S., 2021).

2. Kompleksitas Manajemen dan Pemeliharaan

Sistem basis data besar dan kompleks memerlukan manajemen dan pemeliharaan yang intensif. Proses pengelolaan skema basis data, pengaturan indeks, dan optimisasi query bisa menjadi tugas yang rumit dan memakan waktu. Menurut Tan dan Bia (2022), “Pengelolaan basis data yang kompleks memerlukan keahlian khusus dan sumber daya yang signifikan untuk memastikan performa yang optimal dan keberlangsungan sistem” (Tan, H. & Bia, P., 2022).

3. Kerentanan Terhadap Serangan Keamanan

Basis data, terutama yang mengelola informasi sensitif, sering kali menjadi target serangan keamanan. Serangan seperti SQL injection, data breach, dan ransomware bisa mengancam integritas dan kerahasiaan data. Menurut Kumar *et al.* (2023), “Basis data sering menjadi sasaran serangan canggih yang dapat mengekspos data sensitif dan merusak integritas sistem jika tidak dilindungi dengan baik” (Kumar, S., *et al.*, 2023).

4. Keterbatasan dalam Penanganan Data Tidak Terstruktur

Basis data tradisional lebih efektif dalam menangani data terstruktur namun memiliki keterbatasan dalam mengelola data tidak terstruktur seperti teks, gambar, dan video. Basis data NoSQL, meskipun menangani data tidak terstruktur dengan lebih baik, mungkin tidak menawarkan semua fitur manajemen data yang ada pada sistem basis data relasional. Menurut Moore dan Scott (2022), “Basis data

relasional kurang efisien dalam menangani data tidak terstruktur, dan meskipun basis data NoSQL menawarkan solusi, mereka sering kali datang dengan trade-off dalam hal konsistensi dan integritas data” (Moore, T., & Scott, J., 2022).

5. Masalah Konsistensi Data pada Sistem Terdistribusi

Sistem basis data terdistribusi dapat menghadapi tantangan dalam hal konsistensi data. Menjaga konsistensi di seluruh node dalam sistem terdistribusi memerlukan mekanisme koordinasi yang kompleks. Menurut Borkar *et al.* (2023), “Menjamin konsistensi data dalam sistem basis data terdistribusi adalah tantangan besar yang melibatkan trade-off antara ketersediaan dan konsistensi dalam konteks CAP theorem” (Borkar, V., *et al.*, 2023).

Daftar Pustaka

- Aggarwal, C. C., & Yu, P. S. (2021). *Data Mining: The Textbook*. Springer.
- Batini, C., & Scannapieco, M. (2023). *Data and Information Quality: Dimensions, Principles, and Techniques*. Springer.
- Borkar, V., Chen, H., & El-Maarri, N. (2023). *Distributed Database Systems: Concepts and Design*. Elsevier.
- Chen, J., Li, H., & Zhao, M. (2021). *Database Recovery and Replication Techniques*. Springer.
- Chen, J., Zhang, L., & Liu, H. (2024). *Modern Database Management and Applications*. Elsevier.
- Chen, L., & Zhang, J. (2022). *Big Data and Business Intelligence: A Guide for Data-Driven Decision Making*. Springer.
- Codd, E.F. (1970). "A Relational Model of Data for Large Shared Data Banks". *Communications of the ACM*, 13(6), 377-387.
- Connolly, T., & Begg, C. E. (2015). *Database Systems: A Practical Approach to Design, Implementation, and Management*. Pearson.
- Date, C. J. (2023). *An Introduction to Database Systems*. Springer.
- Elmasri, R., & Navathe, S. (2016). *Fundamentals of Database Systems*. Pearson.
- Elmasri, R., & Navathe, S. (2021). *Fundamentals of Database Systems* (7th ed.). Pearson.
- Elmasri, R., & Navathe, S. B. (2022). *Fundamentals of Database Systems*. Pearson.
- Elmasri, R., & Navathe, S.B. (2021). "Fundamentals of Database Systems". 7th Edition. Pearson.

- Garcia-Molina, H., Ullman, J. D., & Widom, J. (2020). "Database Systems: The Complete Book". 2nd Edition. Prentice Hall.
- He, A., Zhang, X., & Liu, J. (2023). Database Systems for Advanced Applications. Springer.
- Hoffer, J.A., Prescott, M., & McFadden, F.R. (2020). Modern Database Management. 12th Edition. Pearson.
- Kumar, S., Sharma, R., & Singh, S. (2023). Database Security and Privacy: Principles and Practices. Springer.
- Liu, X., Zhang, X., & Zhang, C. (2021). Database Systems: Concepts, Design, and Applications. Springer.
- Miao, H., Zhang, L., & Yang, J. (2023). NoSQL Databases: A Comprehensive Review. ACM Computing Surveys.
- Moore, T., & Scott, J. (2022). Big Data and Data Science: A Practical Guide. Springer.
- Morris, D., Johnson, T., & Wang, X. (2022). Database Systems: A Comprehensive Approach. Wiley.
- O'Neil, P., & O'Neil, E. (2022). Database: Principles, Programming, and Performance (3rd ed.). Morgan Kaufmann.
- Silberschatz, A., Korth, H.F., & Sudarshan, S. (2021). Database System Concepts. 8th Edition. McGraw-Hill.
- Tan, H., & Bia, P. (2022). Database Management and Performance Tuning. Wiley.
- Wang, Z., Liu, Q., & Zhang, T. (2022). Big Data and Machine Learning: Applications and Challenges. Elsevier.
- Zhang, Y., & Liu, W. (2022). Advanced Database Security and Privacy. Wiley.

BAB II

Model Data

-Bernadete Deta-

A. Pengertian Model Data

Langkah dalam perancangan basis data dapat dilakukan dengan mulai membuat model dari awal. Pendekatan ini umumnya diambil ketika data atau fakta yang tersedia sangat terbatas. Sebagai alternatif, model yang disajikan dalam bentuk diagram awal akan lebih mudah dievaluasi dan dianalisis, sehingga memungkinkan adanya perbaikan untuk menghasilkan model data yang lebih stabil dan mendekati realitas atau permanen. Menurut (Examiner & Gaffin, 2007), Model data dapat diartikan sekumpulan alat konseptual yang digunakan untuk menggambarkan data, hubungan antardata, (semantic) makna data, serta batasan-batasan data (Ruliah dan Andri Suryadi, 2016) Karena tujuan utamanya adalah untuk menunjukkan makna data dan hubungannya dengan data lain, model ini lebih tepat disebut sebagai Model Data Lojik.

Sebagaimana seperti tori di atas dapat dikatakan Model data adalah sekumpulan konsep yang digunakan untuk menggambarkan data, hubungan antara data, dan batasan-batasan data yang terintegrasi dalam suatu organisasi. Juga diartikan sebagai Representasi struktural dari elemen data, hubungannya, dan batasan dalam sistem manajemen basis data (Codd, 1980)

B. Jenis-jenis Model Data

1. Model Data Berbasis Objek

Menurut Codd (1980), konsep entitas, atribut, dan hubungan antar-entitas (*relationship*) digunakan dalam model data berbasis objek. Entitas adalah segala sesuatu yang ada dalam sistem, baik nyata (*real*) maupun abstrak, tempat data disimpan. Entitas dapat digolongkan menjadi empat jenis: Nama orang, benda, lokasi, dan kejadian (yang mengandung unsur waktu). Atribut adalah relasi fungsional dari satu set objek ke set objek lainnya. Relationship terjadi karena adanya hubungan alami antara entitas. Model data berbasis objek memiliki beberapa bentuk, sebagai berikut:

- a. Model keterhubungan entitas (*entity relationship model*).
- b. Model berorientasi objek (*object oriented model*).
- c. Model data semantik (*semantic data model*).
- d. Model data fungsional (*functional data model*).

Entity-Relationship Model: Model ini digunakan untuk menggambarkan hubungan antar data dalam basis data

dengan mengacu pada pandangan bahwa dunia nyata terdiri dari objek-objek dasar yang saling terhubung.





Komponen utama dari Model Entitas-Keterhubungan meliputi entitas (*entity*) dan relasi (*relation*). Kedua komponen ini dijelaskan lebih lanjut melalui sejumlah atribut atau properti. Diagram Entity-Relationship (Diagram E-R).

Model Entity-Relationship, yang mencakup komponen seperti himpunan entitas, relasi, dan atribut-atributnya, dapat digambarkan menggunakan Diagram Entity-Relationship. (Diagram E-R) (Hellerstein *et al.*, 2007).

Model ERD dibuat berdasarkan asumsi bahwa dunia nyata terdiri dari kumpulan objek dasar yang disebut entitas (*entity*) serta hubungan (*relationship*) antara entitas-entitas tersebut. Sebagai contoh, setiap individu adalah entitas, dan rekening bank juga dapat dianggap sebagai sebuah entitas.

Dalam database, entitas didefinisikan melalui atribut-atributnya. Misalnya, nomor rekening digunakan untuk membedakan satu rekening dari rekening lain yang dimiliki oleh orang yang berbeda di suatu bank. Nomor rekening ini adalah atribut dari entitas rekening yang bersangkutan dan berfungsi untuk secara unik membedakan satu rekening dari yang lainnya. Meskipun beberapa rekening mungkin memiliki saldo yang sama, nomor rekeningnya pasti berbeda. Relationship merujuk pada hubungan antara berbagai entitas. Simbol dasar yang digunakan untuk menggambarkan relationship dapat dilihat dalam tabel berikut:

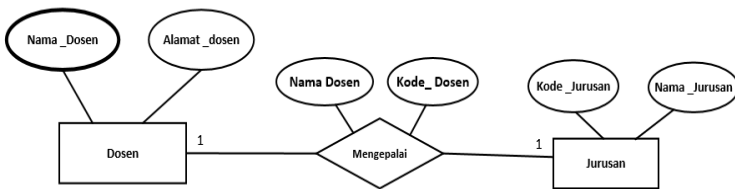
Tabel 2.1

Simbol	Keterangan
	Menyatakan Himpunan Entitas
	Menunjukkan Himpunan Relasi
	Menyatakan Atribut (atribut key digaris bawai)
	Penghubung/Link

Dalam Diagram E-R, memiliki aturan utama yaitu yang menetapkan jumlah entitas yang dapat terhubung dengan entitas lain melalui relationship-set adalah kardinalitas relasi/mapping cardinalities.

Jenis Mapping Cardinalities:

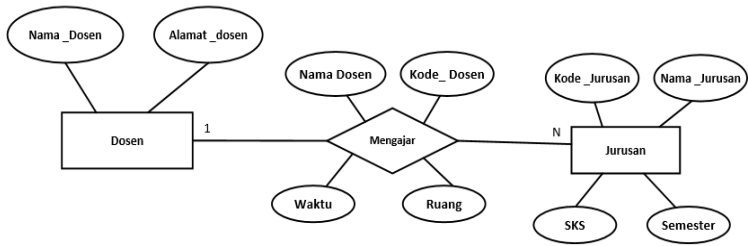
a. Relasi satu ke satu (*one-to-one*)



Gambar 2.1. Contoh Relasi one-to-one

b. Relasi satu ke banyak (*one-to-many*)

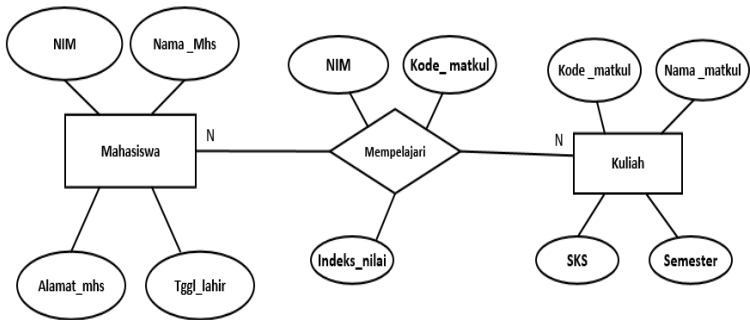
Adapun contohnya dapat terlihat seperti gambar di bawah ini.



Gambar 2.2. Contoh Relasi one-to-many

c. Relasi banyak ke banyak (*many-to-many*)

Berikut ini merupakan contohnya.



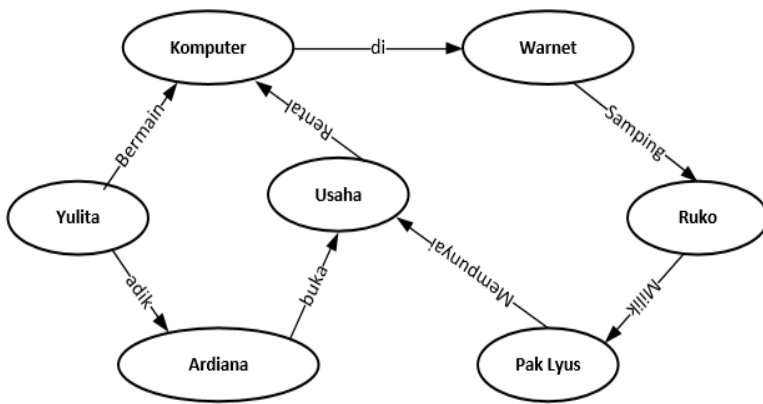
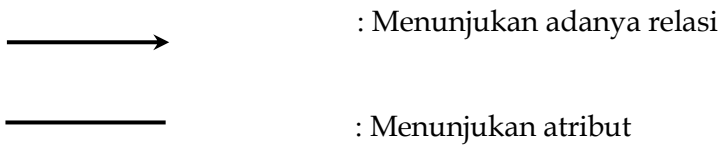
Gambar 2.3. Contoh Relasi many-to-many

2. Model Data (Semantic Data Model)

Mirip dengan model *entity relationship*, di mana hubungan antara objek dasar dinyatakan dengan kata-kata (semantic) tidak dengan simbol. Model data semantik adalah salah satu jenis model yang menggunakan pendekatan semantik untuk menggambarkan hubungan antar-objek.

Seperti pada contoh sebelumnya, dalam model semantik, hubungan antar-objek digambarkan seperti yang terlihat pada

gambar di atas. Simbol-simbol yang digunakan dalam model semantik adalah sebagai berikut:



Gambar 2.4. Contoh Semantic Data Model

3. Model Data Berbasis Record

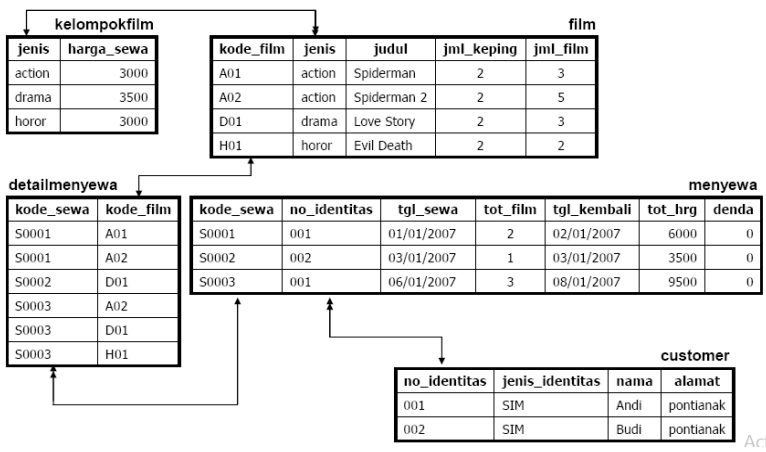
Model ini didasarkan pada rekaman untuk menjelaskan kepada pengguna hubungan logis antardata dalam basis data. (Connolly & Begg, 2014).

Perbedaan dengan model data berbasis objek: Pada model data berbasis rekaman, digunakan untuk menjelaskan struktur logis keseluruhan dari suatu basis data serta untuk menjelaskan implementasi sistem basis data (Tompa, 1989).

Model data berbasis record memiliki 3 jenis model data, yaitu

a. Model Relational

Data dan hubungan antardata direpresentasikan oleh sejumlah tabel, di mana setiap tabel terdiri dari beberapa kolom dengan nama yang unik (Steinhoff & Arbor, 2001). Adapun model relational dapat terlihat seperti gambar ini.



Gambar 2.5. Contoh Model Data Relational

Keunggulan dari database relational adalah

1) **Mudah digunakan:** Basis data relasional relatif mudah dipahami dan dibaca dibandingkan dengan tipe basis data lain seperti basis data pohon atau basis data hierarkis yang lebih kompleks. Banyak mata kuliah atau lembaga pelatihan komputer tentang pemrograman mengajarkan basis data relasional sebagai dasar untuk memahami jenis basis data yang lebih rumit seperti basis data pohon. Bentuk tabel pada basis data relasional, yang

terdiri dari kolom dan baris, dapat dengan mudah dipahami oleh pemula.

- 2) **Fleksibilitas Penggunaan:** Data dalam tabel di basis data relasional dapat dimanipulasi dengan berbagai operator untuk diambil atau digunakan oleh program.
- 3) **Keamanan yang Baik:** Pada basis data relasional, perancang dapat dengan mudah menambahkan kontrol keamanan dan otorisasi. Atribut yang sensitif dapat ditempatkan dalam tabel terpisah dengan pengaturan keamanan khusus yang memerlukan otentikasi khusus. Hanya pengguna atau program dengan otentikasi yang sesuai yang dapat mengakses data sensitif dalam tabel tersebut. Sistem keamanan seperti ini memberikan tingkat perlindungan yang tinggi pada basis data relasional.
- 4) **Tingkat Presisi Tinggi:** Data yang diambil dari basis data relasional memiliki tingkat presisi yang tinggi. Ini disebabkan oleh fakta bahwa basis data relasional melakukan berbagai operasi aljabar dan kalkulus relasional saat memanipulasi hubungan antartabel. Hal ini mengurangi kemungkinan ambiguitas data, yang berbeda dari tipe basis data lain dengan hubungan antar tabel yang lebih kompleks.
- 5) **Mendukung Penggunaan Bahasa Pemrograman Query Database dengan Mudah:** Basis data relasional memungkinkan penggunaan bahasa pemrograman khusus untuk melakukan query database, sehingga memudahkan programmer dalam membuat fungsi.

- 6) **Lebih Mudah Mencapai Kemandirian Data:** Data yang mandiri adalah data yang tidak bergantung pada data lain dengan tingkatan yang lebih rendah. Kemandirian data sangat penting dalam sistem manajemen database yang terpusat, karena tanpa kemandirian ini, sistem dapat mengalami beban berat dan potensi kekacauan. Pada basis data relasional, kemandirian data dicapai melalui normalisasi struktur, yang lebih mudah diterapkan dibandingkan dengan tipe basis data lain yang lebih kompleks.

Kelemahan dari database relational adalah

- 1) **Penggunaan Ruang Penyimpanan Relatif Tinggi:** Data yang terpisah dan saling terhubung memerlukan ruang penyimpanan yang lebih besar dibandingkan data yang disimpan dalam satu bagian. Terutama jika data tersebut sering digunakan dalam operasi join yang menggabungkan data, kecepatan baca dan tulis pada ruang penyimpanan menjadi sangat terpengaruh. Selain itu, data dengan tipe yang tidak dapat dikategorikan dalam basis data relasional, seperti gambar, musik, dan multimedia lainnya, akan semakin meningkatkan kebutuhan ruang penyimpanan ketika menggunakan basis data relasional.
- 2) **Kebutuhan Kemampuan Komputasi Relatif Tinggi:** Basis data relasional menyembunyikan kompleksitas penggunaan aljabar dan kalkulus dari pengguna. Penggunaan basis data relasional memerlukan tingkat komputasi yang tinggi karena setiap operasi melibatkan

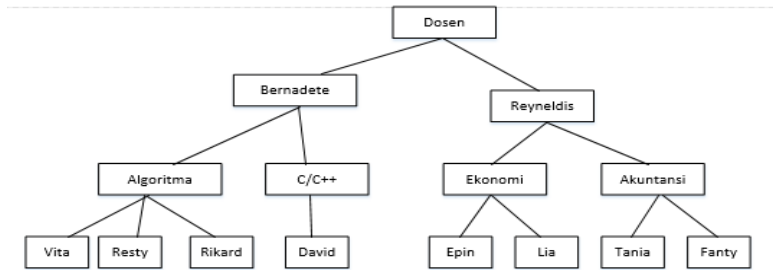
berbagai perhitungan untuk menghubungkan data antar tabel.

- 3) **Ekstraksi Data Relatif Lambat:** Kecepatan ekstraksi data pada basis data relasional cenderung lebih lambat dibandingkan dengan tipe basis data lainnya. Data yang diorganisir secara hierarki dapat diekstrak lebih cepat daripada data yang harus terlebih dahulu direlasikan dengan tabel lain sebelum ekstraksi.

Data yang terisolasi: Dalam praktiknya, basis data relasional yang kompleks sering kali menjadi seperti pulau-pulau informasi, membuat sulit untuk berbagi data antara sistem besar. Di sebuah institusi besar, misalnya, setiap divisi biasanya memiliki basis data masing-masing. Membuat program untuk menghubungkan data antara dua divisi ini memerlukan waktu dan biaya yang sangat besar, karena sifat kompleks dari basis data relasional yang semakin rumit seiring bertambahnya jumlah tabel.

b. Model Hirarki

Data dan hubungan antar data direpresentasikan menggunakan rekaman dan link (pointer), di mana rekaman-rekaman tersebut disusun dalam bentuk pohon (*tree*). Setiap node dalam pohon tersebut mewakili rekaman atau grup elemen data dan memiliki hubungan dengan kardinalitas 1:1 dan 1M.



Gambar 2.6. Contoh Model Hirarki

Keuntungan basis data hierarki

- 1) **Pengambilan Data yang Cepat dan Efisien:**
kemampuannya untuk mengambil data dengan cepat dan efisien, berkat penyusunan data yang terstruktur dan dapat diprediksi.
- 2) **Kemudahan Menambah atau Menghapus Informasi:**
Dalam jenis basis data ini, menambah atau menghapus informasi dapat dilakukan dengan mudah.
- 3) **Struktur Data yang Terprediksi:** Model basis data hierarki memiliki struktur data yang terprediksi dan teratur, sehingga lebih mudah dipahami dan dinavigasi.
- 4) **Penyimpanan Data yang Efisien:** menyimpan data dalam hubungan induk-anak, yang memungkinkan penyimpanan data menjadi lebih efisien dibandingkan dengan model basis data lainnya.
- 5) **Kinerja yang Optimal:** menunjukkan kinerja yang baik dalam pengambilan dan pembuatan kueri data, terutama ketika menangani kumpulan data yang besar.

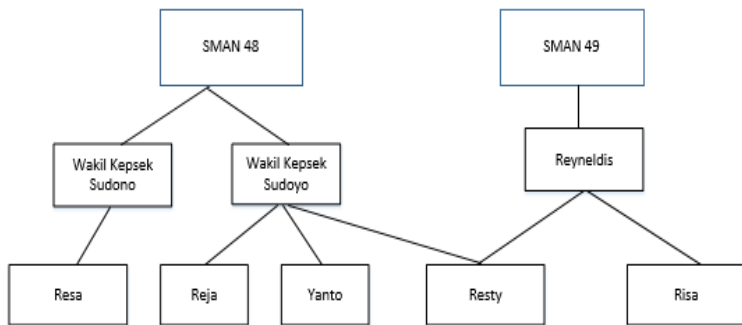
Kekurangan basis data hierarki

- 1) **Fleksibilitas yang Terbatas:** Salah satu kelemahan utama basis data hierarki adalah kurangnya fleksibilitas dibandingkan dengan jenis basis data lainnya. Model ini kurang cocok pada data yang kompleks atau perubahan dalam struktur data.
- 2) **Sulit untuk Dipelihara dan Diperbarui:** Basis data hierarki dapat menjadi sulit untuk dipelihara dan diperbarui, karena perubahan pada struktur data dapat mempengaruhi keseluruhan basis data.
- 3) **Dukungan Terbatas untuk Manipulasi Data:** Basis data hierarki memiliki dukungan yang terbatas untuk operasi manipulasi data yang kompleks, seperti agregasi data atau penambahan data.
- 4) **Interoperabilitas Terbatas:** Basis data hierarki tidak kompatibel dengan model basis data lainnya, yang dapat membatasi kemampuannya untuk berintegrasi dengan sistem dan aplikasi lain.
- 5) **Kurangnya Standarisasi:** Karena basis data hierarki tidak digunakan seluas model basis data lainnya, terdapat kekurangan dalam standarisasi dan praktik terbaik untuk penggunaannya dan pengelolaannya.

c. Model Jaringan

Pada tahun 1971 oleh Database Task Group (DBTG), yang dikenal sebagai model CODASYL, mirip dengan model hierarki, di mana data dan hubungan antardata direpresentasikan menggunakan rekaman dan tautan (*record and*

links) (Hellerstein *et al.*, 2007) Adapun perbedaannya terletak pada penyusunan record dan links; model jaringan menyusun rekaman dalam bentuk graf dan mendefinisikan hubungan kardinalitas 1:1, 1:M dan N:M. Berikut ini adalah contoh model jaringan.



Gambar 2.7. Contoh Model jaringan

Manfaat Model Data Jaringan

Pemodelan data jaringan diciptakan untuk mengatasi meningkatnya kompleksitas hubungan data modern, memberikan representasi informasi yang saling berhubungan yang lebih intuitif. Untuk mencapai tujuan ini, organisasi yang mengadopsi pemodelan data jaringan akan menyadari beberapa manfaat.

1) Fleksibilitas dalam Hubungan Data

Manfaat utama model data jaringan adalah fleksibilitasnya dalam merepresentasikan hubungan yang kompleks. Tidak seperti model tradisional yang mungkin kesulitan dengan hubungan banyak-ke-banyak,

model jaringan unggul dalam menangkap hubungan rumit antar entitas. Fleksibilitas ini sangat berharga dalam skenario dimana hubungan bersifat dinamis dan berkembang seiring waktu.

2) Representasi Intuitif

Representasi visual model data jaringan seringkali lebih intuitif untuk dipahami pengguna. Dengan node yang mewakili entitas dan edge yang menunjukkan hubungan, model ini mencerminkan koneksi dunia nyata. Hal ini memudahkan pemangku kepentingan, termasuk analis bisnis dan pengembang, untuk memahami struktur data dan bagaimana berbagai elemen saling berhubungan satu sama lain.

3) Performa Kueri yang Efisien

Model data jaringan memberikan performa kueri yang lebih efisien dalam skenario di mana hubungan memainkan peran penting. Melintasi hubungan dalam grafik adalah operasi alami dan (dalam banyak kasus) merupakan urusan sepele; hal yang sama tidak dapat dikatakan tentang tipe hubungan model data lainnya. Basis data grafik yang dioptimalkan untuk model jaringan dapat memberikan respons kueri yang cepat untuk kueri berbasis hubungan yang kompleks.

4) Dukungan yang Lebih Baik untuk Struktur Data yang Berkembang

Seiring berkembangnya kebutuhan data, model data jaringan memberikan dukungan yang lebih baik untuk perubahan struktur. Menambah atau mengubah

hubungan seringkali lebih mudah dilakukan dalam model jaringan dibandingkan model tradisional, sehingga memungkinkan organisasi untuk beradaptasi dengan kebutuhan bisnis baru dengan lebih lancar.

Tantangan Model Data Jaringan

Meskipun model data jaringan memberikan manfaat besar, model ini juga menghadirkan beberapa tantangan utama yang harus diatasi oleh organisasi sebelum penerapannya:

1) Kompleksitas Dalam Pemodelan

Membangun model data jaringan bisa jadi rumit, terutama ketika berhadapan dengan entitas dalam jumlah besar dan hubungan yang rumit. Desain model data jaringan yang efektif memerlukan pertimbangan yang cermat terhadap hubungan antar node, potensi redundansi, dan struktur grafik secara keseluruhan.

2) Penskalaan Kinerja

Meskipun model data jaringan unggul dalam menangani hubungan, kinerjanya dapat menurun ketika berhadapan dengan kumpulan data besar atau grafik yang kompleks. Menskalakan model jaringan untuk mengakomodasi jumlah data yang terus bertambah memerlukan desain database yang cermat dan strategi pengoptimalan untuk mempertahankan performa kueri.

3) Kurangnya Standardisasi

Tidak seperti database relasional dengan bahasa kueri terstruktur (SQL) yang terstandarisasi, database

jaringan tidak memiliki bahasa kueri yang diterima secara universal. Kurangnya standarisasi ini dapat menimbulkan tantangan ketika bekerja dengan sistem database grafik yang berbeda, karena pengguna mungkin perlu beradaptasi dengan bahasa kueri spesifik yang didukung oleh database yang dipilih.

Daftar Pustaka

- Codd, E. F. (1980). Data models in database management. 112–114. <https://doi.org/10.1145/800227.806891>
- Connolly, T., & Begg, C. (2014). Pearson Database Systems A Practical Approach to Design Implementation and Management 6th Global Edition 1292061189 pdf. <https://doi.org/10.1007/978-1-4842-1191-5>
- Dr. Ruliah, M.Kom. Andri Suryadi, S.Kom., M. K. (2016). Basis Data dan Sistem Basis Data Daftar Isi. 1–35.
- Examiner, P., & Gaffin, J. (2007). (12) *United States Patent*. 2(12).
- Hellerstein, J. M., Stonebraker, M., & Hamilton, J. (2007). Architecture of a Database System. *Architecture of a Database System*. <ps://doi.org/10.1561/9781601980793>
- Steinhoff, D., & Arbor, A. (2001). Analytical database system that models data to speed up and simplify data analysis. 2(12). <https://patents.google.com/patent/US6879984B2/en>
- Tompa, F. W. M. (1989). A data model for flexible hypertext database systems. *ACM Transactions on Information Systems (TOIS)*, 7(1), 85–100. <https://doi.org/10.1145/64789.64993>

BAB V

Pengembangan Aplikasi Sistem Basis Data:

Mengembangkan Sistem Manajemen Perpustakaan Menggunakan Microsoft Access

A. Pendahuluan

Sistem manajemen perpustakaan adalah aplikasi yang dirancang untuk membantu pustakawan dalam mengelola berbagai aspek perpustakaan secara efisien dan efektif. Aplikasi ini bertujuan untuk menyederhanakan dan mengotomatisasi proses yang terkait dengan pengelolaan buku, anggota, dan transaksi peminjaman serta pengembalian buku. Dengan menggunakan sistem ini, perpustakaan dapat memastikan bahwa data yang dikelola akurat, mudah diakses, dan terorganisir dengan baik.

B. Fitur Utama Sistem Manajemen Perpustakaan

1. Pengelolaan Data Buku:

- a. Pencatatan informasi detail tentang buku, termasuk judul, pengarang, penerbit, tahun terbit, dan kategori.
- b. Kemampuan untuk menambah, mengedit, dan menghapus data buku.

2. Pengelolaan Data Anggota:

- a. Pencatatan informasi detail tentang anggota perpustakaan, termasuk nama, alamat, dan nomor telepon.
- b. Kemampuan untuk menambah, mengedit, dan menghapus data anggota.

3. Manajemen Transaksi Peminjaman dan Pengembalian:

- a. Pencatatan transaksi peminjaman buku oleh anggota.
- b. Pencatatan pengembalian buku dan perhitungan denda (jika ada).

4. Pembuatan Laporan:

- a. Kemampuan untuk membuat laporan mengenai status peminjaman buku, daftar buku yang dipinjam, dan riwayat peminjaman anggota.
- b. Laporan dapat dihasilkan dalam berbagai format untuk analisis dan pelaporan lebih lanjut.

5. Pencarian dan Navigasi Mudah:

- a. Fitur pencarian untuk memudahkan pencarian buku dan anggota berdasarkan berbagai kriteria.

- b. Antarmuka navigasi yang intuitif untuk memudahkan akses ke berbagai fungsi aplikasi.

C. Manfaat Menggunakan Sistem Manajemen

Perpustakaan

1. Efisiensi Operasional:

- a. Mengurangi waktu dan usaha yang diperlukan untuk mengelola data secara manual.
- b. Otomatisasi proses mengurangi kesalahan manusia dan meningkatkan akurasi data.

2. Akses Informasi yang Cepat:

- a. Memungkinkan pustakawan dan anggota untuk dengan cepat menemukan informasi yang mereka butuhkan.
- b. Pencarian dan navigasi yang efisien memastikan bahwa data dapat diakses dengan mudah.

3. Organisasi Data yang Lebih Baik:

- a. Data disimpan dalam format yang terstruktur dan terorganisir dengan baik, memudahkan pengelolaan dan pemeliharaan.
- b. Relasi antartabel memungkinkan integrasi dan konsistensi data yang lebih baik.

4. Kemampuan Pelaporan dan Analisis:

- a. Laporan yang dihasilkan dapat digunakan untuk analisis lebih lanjut, membantu perpustakaan dalam mengambil keputusan yang lebih baik.

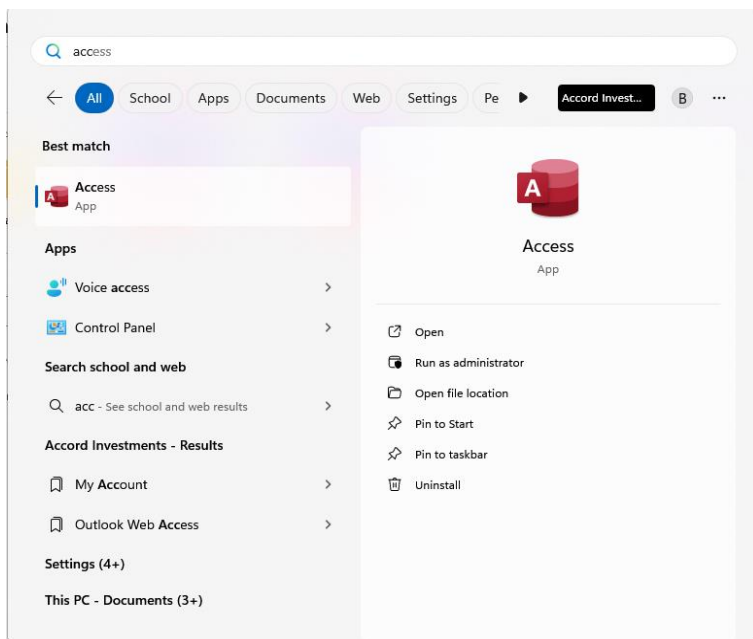
- b. Kemampuan untuk menghasilkan laporan secara otomatis menghemat waktu dan usaha.

D. Langkah-langkah Mengembangkan Aplikasi Basis Data di Microsoft Access

1. Membuat Database Baru

a. Membuka Microsoft Access

Jalankan Microsoft Access dari menu Start atau melalui shortcut di desktop. Setelah aplikasi terbuka, akan terlihat layar utama dengan berbagai opsi.



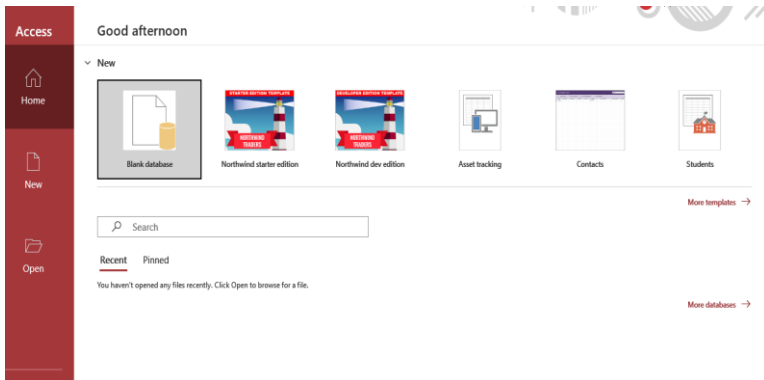
Gambar 5.1. Tampilan Awal Microsoft Access

b. Memilih "Blank Database"

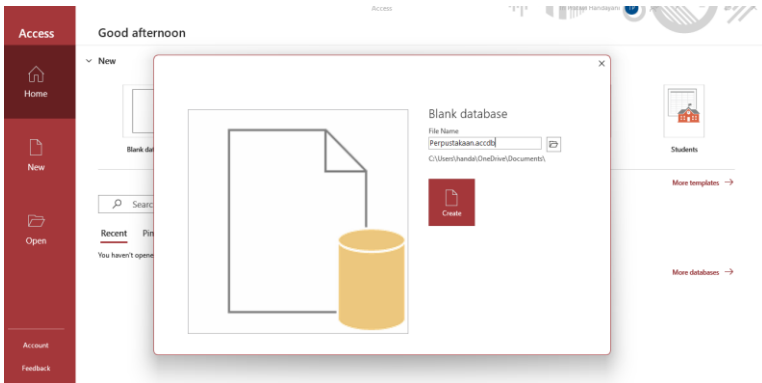
Pada layar utama Microsoft Access, pilih opsi "Blank Database" untuk membuat database baru dari awal. Ini akan memungkinkan Anda untuk mendesain struktur database sesuai dengan kebutuhan aplikasi yang akan Anda kembangkan.

c. Memberi Nama Database (misalnya, Perpustakaan.accdb)

Di bawah opsi "File Name", ketik nama untuk database Anda, misalnya Perpustakaan.accdb. Pastikan untuk memilih lokasi penyimpanan yang mudah diakses dan diingat.



Gambar 5.2. Tampilan Pembuatan Basis Data

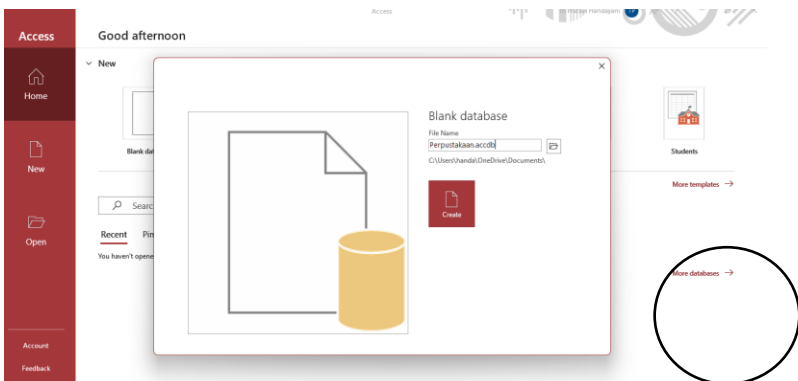


Gambar 5.3. Penamaan Basis Data

d. Membuat Database

Setelah memberikan nama dan memilih lokasi penyimpanan, klik tombol "Create".

Microsoft Access akan membuat dan membuka database baru yang siap untuk Anda gunakan. Setelah database baru dibuat, Anda siap untuk melanjutkan ke langkah berikutnya, yaitu membuat tabel-tabel yang diperlukan untuk aplikasi basis data Anda.



Gambar 5.4. Tampilan Tombol Create database

2. Membuat Tabel

a. Membuat Tabel

Gambar di bawah menunjukkan tampilan antarmuka Microsoft Access saat Anda memulai pembuatan tabel baru dalam sebuah database. Berikut adalah deskripsi rinci langkah-langkah yang diilustrasikan dalam gambar:

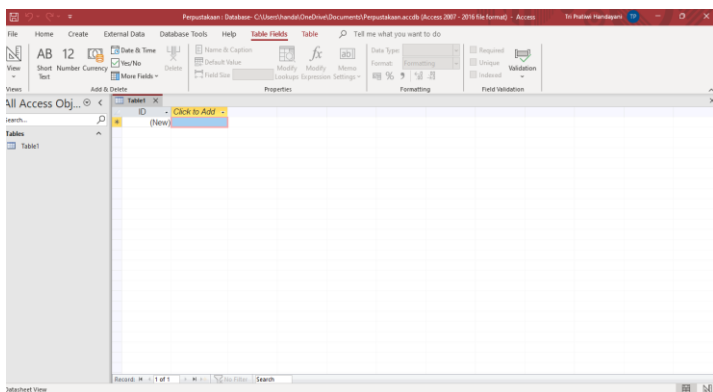
Langkah-langkah membuat tabel baru:

- 1) Tampilan Awal Pembuatan Tabel:
 - a) Gambar menunjukkan jendela utama Microsoft Access setelah Anda membuat database baru atau membuka database yang sudah ada.
 - b) Nama database yang sedang dibuka adalah Perpustakaan.accdb.

- 2) Area Kerja Tabel:
 - a) Di bagian tengah layar, terdapat area kerja untuk tabel baru yang sedang dibuat, dengan satu kolom awal bernama ID.
 - b) Anda dapat melihat baris pertama dengan status New yang siap diisi dengan data baru.

- 3) Menambahkan Field Baru:
 - a) Untuk menambahkan field baru ke tabel, klik pada kolom yang bertuliskan Click to Add.
 - b) Anda bisa memilih tipe data yang diinginkan seperti Short Text, Number, Currency, dll.

- 4) Navigasi dan Alat di Ribbon:
- a) Tab "Table Fields": Menyediakan berbagai opsi untuk mengelola field dalam tabel, seperti menambahkan, menghapus, dan memodifikasi field.
 - b) Tab "Table": Berisi alat untuk mengelola properti tabel dan pengaturan validasi field.
- 5) Panel Navigasi (Navigation Pane):
- a) Di sebelah kiri layar, terdapat All Access Objects yang menampilkan daftar objek dalam database, seperti tabel, query, form, dan report.
 - b) Saat ini, hanya ada satu tabel yang disebut Table 1.
- 6) Pengaturan Field Validation:
- a) Pada bagian ribbon, terdapat opsi untuk mengatur validasi field, seperti Required, Unique, dan Indexed, untuk memastikan integritas data.

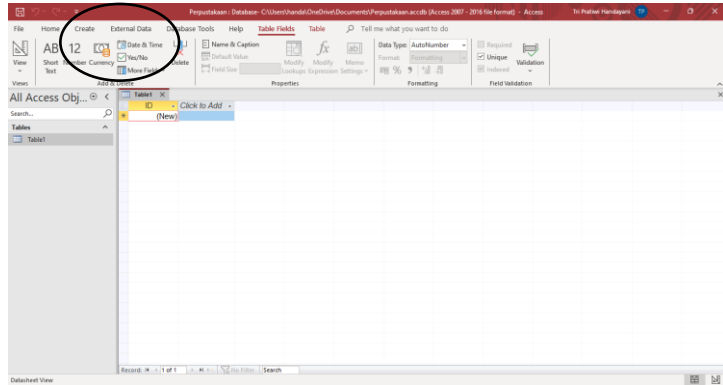


Gambar 5.5. Tampilan Awal Pembuatan Tabel

b. Membuat Tabel Buku

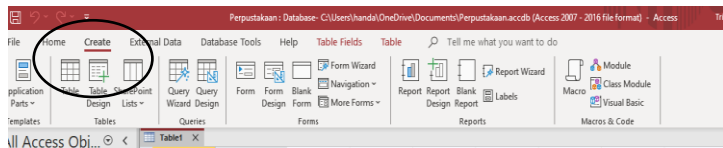
Langkah-langkah membuat tabel buku:

1) Klik "Table Design" di tab "Create".



Gambar 5.6. Tampilan Ribbon Create

2) Masukkan field-field berikut:



Gambar 5.7. Tampilan Tombol Pembuatan Tabel

Gambar 5.7. menunjukkan tampilan tab "Create" di Microsoft Access. Tab ini digunakan untuk membuat berbagai objek database seperti tabel, query, form, dan report. Berikut adalah deskripsi rinci elemen-elemen yang ada di dalam gambar:

Elemen-elemen di Tab "Create"

1) Templates:

- a) Bagian ini berisi berbagai template yang dapat digunakan untuk membuat objek database dengan cepat berdasarkan desain yang sudah ada.
- b) Contohnya termasuk "Application Parts" untuk menambahkan bagian aplikasi yang sudah ada ke database Anda.

2) Tables:

- a) Table: Membuat tabel baru secara langsung dalam tampilan datasheet.
- b) Table Design: Membuat tabel baru menggunakan tampilan desain, memungkinkan pengguna untuk mendefinisikan struktur tabel secara rinci, termasuk field dan tipe data.
- c) SharePoint Lists: Mengimpor atau menghubungkan daftar SharePoint ke database Access.

3) Queries:

- a) Query Wizard: Membantu pengguna membuat query baru menggunakan panduan langkah demi langkah.
- b) Query Design: Membuat query baru menggunakan tampilan desain, memungkinkan pengguna untuk menambahkan tabel, field, dan kriteria secara manual.

4) Forms:

- a) Form: Membuat form baru dengan tampilan default.
- b) Form Design: Membuat form baru menggunakan tampilan desain, memungkinkan pengguna untuk mendesain form sesuai kebutuhan.
- c) Blank Form: Membuat form kosong yang dapat dirancang dari awal.
- d) Form Wizard: Membantu pengguna membuat form baru menggunakan panduan langkah demi langkah.
- e) Navigation: Menyediakan berbagai opsi untuk membuat form navigasi.

5) Reports:

- a) Report: Membuat report baru dengan tampilan default.
- b) Report Design: Membuat report baru menggunakan tampilan desain, memungkinkan pengguna untuk mendesain report sesuai kebutuhan.
- c) Blank Report: Membuat report kosong yang dapat dirancang dari awal.
- d) Report Wizard: Membantu pengguna membuat report baru menggunakan panduan langkah demi langkah.
- e) Labels: Membuat label baru untuk digunakan dalam report atau untuk pencetakan label.

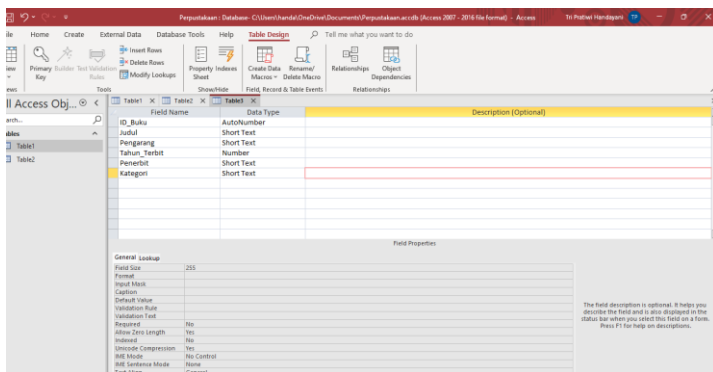
6) Macros & Code:

- a) Macro: Membuat macro baru untuk mengotomatisasi tugas-tugas di database.

- b) Module: Membuat modul baru untuk menulis kode VBA (Visual Basic for Applications).
- c) Class Module: Membuat modul kelas baru untuk menulis kode VBA yang lebih terstruktur.
- d) Visual Basic: Membuka editor VBA untuk menulis dan mengelola kode VBA.

7) Penggunaan Tab "Create"

Tab "Create" di Microsoft Access adalah pusat untuk memulai pembuatan semua objek dalam database Anda. Dengan menggunakan alat-alat yang tersedia di tab ini, pengguna dapat dengan mudah membuat dan mendesain tabel, query, form, dan report yang diperlukan untuk mengelola data mereka secara efektif.



Gambar 5.8. Tampilan Input field Tabel

- a) Field dan tipe data pada tabel anggota
 - ID_Buku: AutoNumber, digunakan sebagai primary key.

Judul: Short Text, untuk menyimpan judul buku.

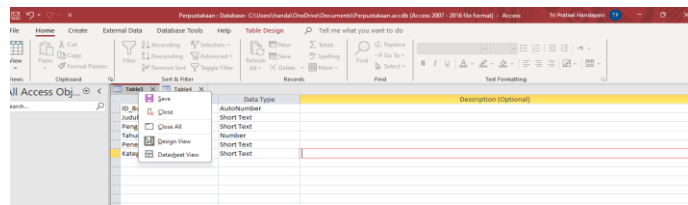
Pengarang: Short Text, untuk menyimpan nama pengarang buku.

Tahun_Terbit: Number, untuk menyimpan tahun terbit buku.

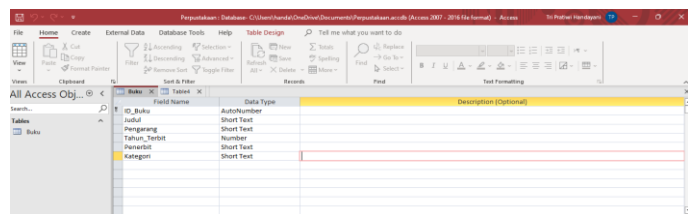
Penerbit: Short Text, untuk menyimpan nama penerbit buku.

Kategori: Short Text, untuk menyimpan kategori buku.

b) Menyimpan tabel dengan nama "Buku"



Gambar 5.9. Tampilan Awal Pembuatan Tabel



Gambar 5.10. Tampilan Awal Pembuatan Tabel

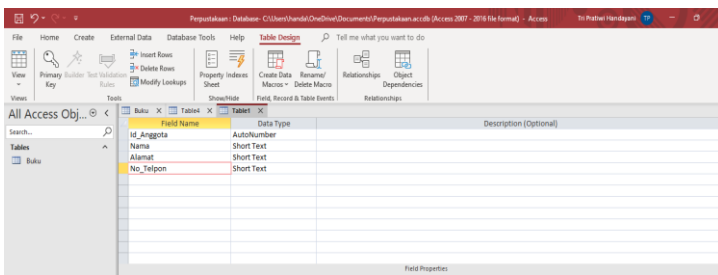
Gambar 5.9. dan 5.10. di atas menunjukkan tampilan antarmuka Microsoft Access saat mendesain sebuah tabel anggota dalam tampilan desain (Design

View). Berikut adalah deskripsi rinci elemen-elemen yang ada di dalam gambar.

c. Membuat Tabel Anggota

Langkah-langkah membuat tabel anggota:

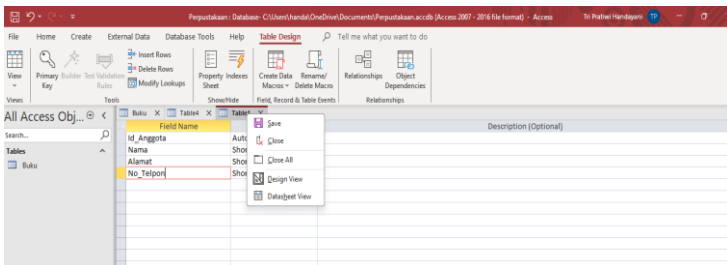
- 1) Klik "Table Design" di tab "Create".
- 2) Masukkan field-field berikut:



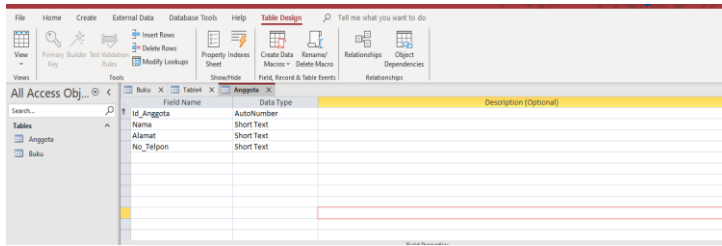
Gambar 5.11. Tampilan Awal Pembuatan Tabel

- a) ID_Anggota: AutoNumber, Primary Key
- b) Nama: Text
- c) Alamat: Text
- d) No_Telepon: Text

- 3) Simpan tabel dengan nama Anggota.



Gambar 5.12. Tampilan Awal Pembuatan Tabel

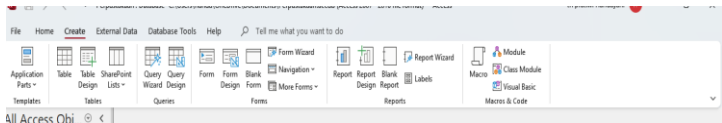


Gambar 5.13. Tampilan Awal Pembuatan Tabel

d. Membuat Tabel Transaksi Peminjaman

Langkah-langkah membuat tabel transaksi peminjaman

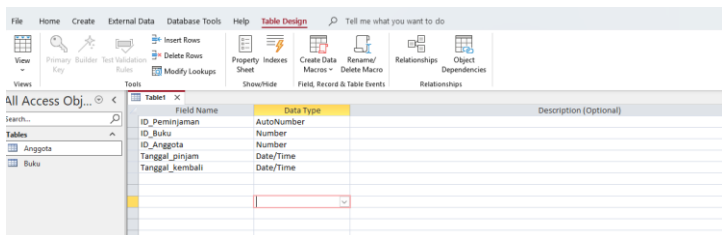
1) Klik "Table Design" di tab "Create".



Gambar 5.14. Tampilan Awal Pembuatan Tabel

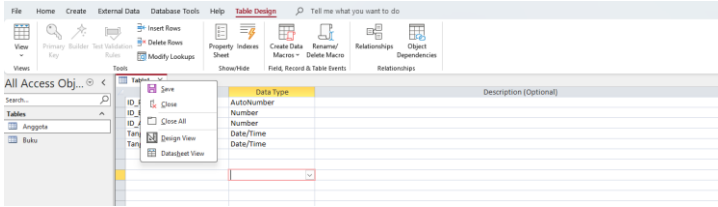
2) Masukkan field-field berikut:

- a) ID_Peminjaman: AutoNumber, Primary Key
- b) ID_Buku: Number (Foreign Key dari Buku)
- c) ID_Anggota: Number (Foreign Key dari Anggota)
- d) Tanggal_Pinjam: Date/Time
- e) Tanggal_Kembali: Date/Time



Gambar 5.15. Tampilan Awal Pembuatan Tabel

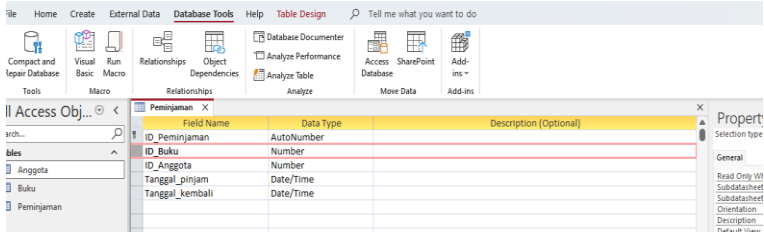
Simpan tabel dengan nama Peminjaman.



Gambar 5.16. Tampilan Awal Pembuatan Tabel

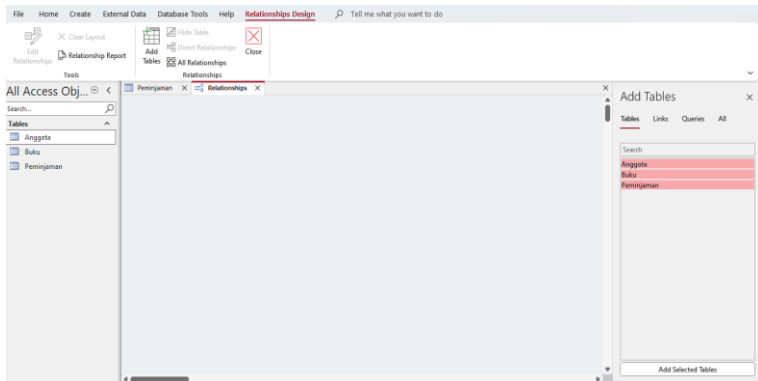
3. Membuat Relasi Antar Tabel

a. Klik "Database Tools" > "Relationships".



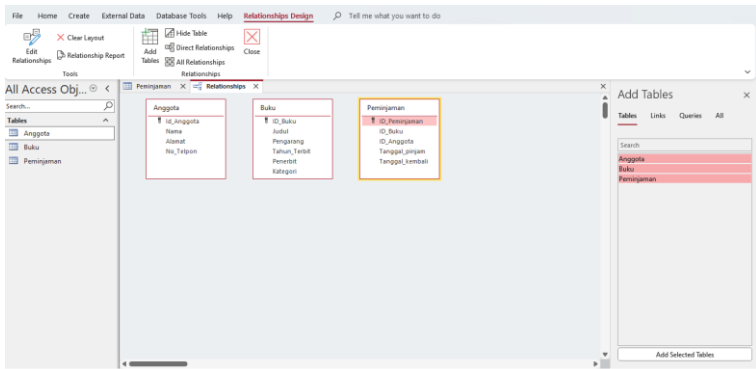
Gambar 5.17. Tampilan Awal Pembuatan Tabel

b. Tambahkan semua tabel (Buku, Anggota, Peminjaman) ke dalam window relationships.



Gambar 5.18. Tampilan Awal Pembuatan Tabel

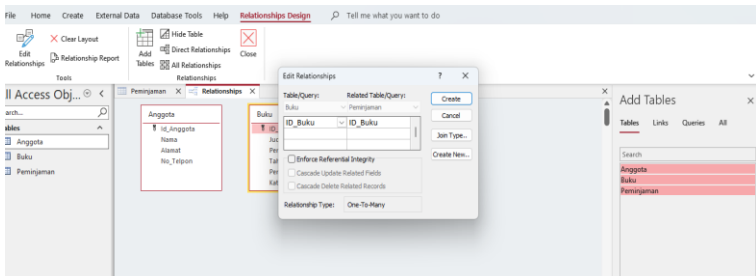
Klik add selected Table dan tampilan akan menjadi seperti berikut:



Gambar 5.19. Tampilan Awal Pembuatan Tabel

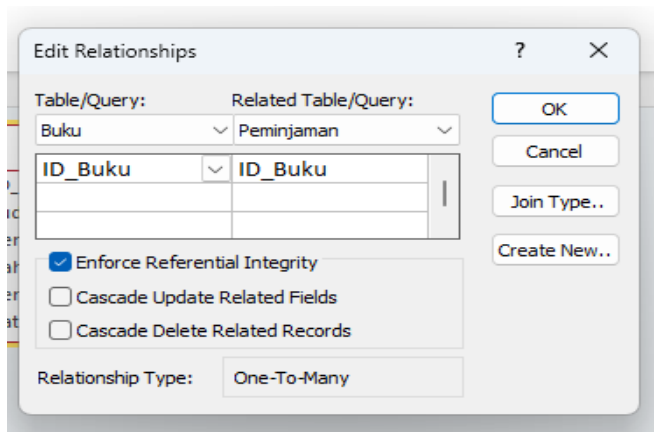
c. Buat Relasi

Tarik ID_Buku dari tabel Buku ke ID_Buku di tabel Peminjaman.



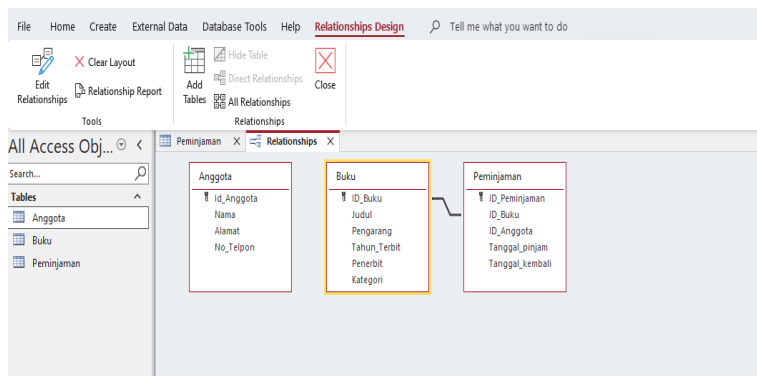
Gambar 5.20. Tampilan Awal Pembuatan Tabel

- d. Enforce Referential Integrity, untuk memastikan integritas data

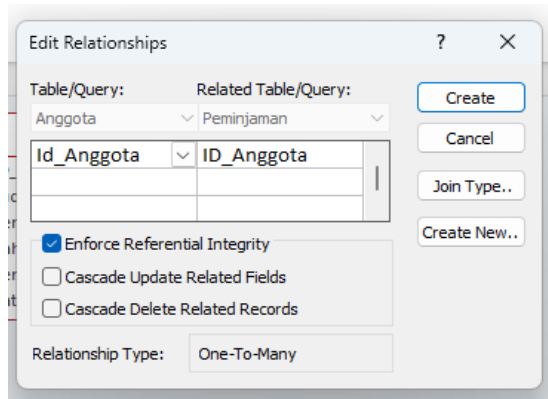


Gambar 5.21. Tampilan Awal Pembuatan Tabel

Tarik ID_Anggota dari tabel Anggota ke ID_Anggota di tabel Peminjaman.



Gambar 5.22. Tampilan Awal Pembuatan Tabel



Gambar 5.23. Tampilan Awal Pembuatan Tabel

4. Membuat Formulir

a. Mendesain Formulir

- 1) Langkah pertama dalam membuat formulir data buku adalah mendesain tampilan formulir menggunakan fitur otomatis dari Microsoft Access. Fitur ini memungkinkan Anda untuk membuat formulir secara cepat dan mudah dari tabel yang sudah ada. Berikut adalah langkah-langkahnya:
- 2) Buka Microsoft Access dan pastikan database perpustakaan (misalnya, Perpustakaan.accdb) telah dibuka.
- 3) Di panel navigasi sebelah kiri, klik pada tabel "Buku" untuk memastikan tabel ini aktif.
- 4) Pilih tab "Create" di pita menu (ribbon).

b. Menambahkan Kontrol untuk Field Tabel

- 1) Setelah memilih tabel "Buku", langkah berikutnya adalah menggunakan fitur otomatis untuk membuat formulir yang berisi semua field dari tabel tersebut:
- 2) Di tab "Create", klik "Form". Microsoft Access akan secara otomatis membuat formulir baru yang berisi semua field dari tabel "Buku".
- 3) Formulir yang dibuat akan berisi kontrol untuk setiap field dalam tabel, seperti:
 - ID_Buku (AutoNumber)
 - Judul (Short Text)
 - Pengarang (Short Text)
 - Tahun_Terbit (Number)
 - Penerbit (Short Text)
 - Kategori (Short Text)

c. Menyimpan Formulir dengan Nama "FormBuku"

- 1) Setelah formulir dibuat, langkah terakhir adalah menyimpan formulir tersebut. Berikut adalah langkah-langkah untuk menyimpan formulir.
- 2) Klik ikon "Save" atau tekan "Ctrl + S" pada keyboard.
- 3) Kotak dialog "Save As" akan muncul.
- 4) Masukkan nama "FormBuku" dalam kotak dialog tersebut.
- 5) Klik "OK" untuk menyimpan formulir.

Gambar 5.24. Tampilan Form Buku

d. Membuat Formulir Data Anggota

1) Mendesain formulir

Langkah pertama dalam membuat formulir data anggota adalah mendesain tampilan formulir menggunakan fitur otomatis dari Microsoft Access. Fitur ini memungkinkan Anda untuk membuat formulir secara cepat dan mudah dari tabel yang sudah ada. Berikut adalah langkah-langkahnya:

- a) Buka Microsoft Access dan pastikan database perpustakaan (misalnya, Perpustakaan.accdb) telah dibuka.
- b) Di panel navigasi sebelah kiri, klik pada tabel "Anggota" untuk memastikan tabel ini aktif.
- c) Pilih tab "Create" di pita menu (ribbon).

d) Di tab "Create", klik "Form". Microsoft Access akan secara otomatis membuat formulir baru yang berisi semua field dari tabel "Anggota".

2) Menambahkan kontrol untuk field tabel

Langkah pertama dalam membuat formulir data anggota adalah mendesain tampilan formulir menggunakan fitur otomatis dari Microsoft Access. Fitur ini memungkinkan Anda untuk membuat formulir secara cepat dan mudah dari tabel yang sudah ada. Berikut adalah langkah-langkahnya:

- a) Buka Microsoft Access dan pastikan database perpustakaan (misalnya, Perpustakaan.accdb) telah dibuka.
- b) Di panel navigasi sebelah kiri, klik pada tabel "Anggota" untuk memastikan tabel ini aktif.
- c) Pilih tab "Create" di pita menu (ribbon).
- d) Di tab "Create", klik "Form". Microsoft Access akan secara otomatis membuat formulir baru yang berisi semua field dari tabel "Anggota".

3) Menambahkan Tombol "Add Record", "Delete Record", dan "Save Record"

Setelah kontrol untuk field ditambahkan, langkah berikutnya adalah menambahkan tombol untuk fungsi "Add Record", "Delete Record", dan "Save Record":

- a) Pastikan Anda berada dalam mode desain formulir dengan mengklik kanan formulir dan memilih "Design View".
- b) Di tab "Design" pada ribbon, pilih "Button" di grup "Controls".
- c) Tempatkan tombol di area formulir yang diinginkan.
- d) Menambahkan Tombol "Add Record":
 - (1) Setelah menempatkan tombol, wizard akan muncul.
 - (2) Pilih kategori "Record Operations" dan tindakan "Add New Record".
 - (3) Klik "Next" dan beri nama tombol "Add Record".
 - (4) Klik "Finish".
- e) Menambahkan Tombol "Delete Record":
 - (1) Ulangi langkah untuk menambahkan tombol.
 - (2) Pilih kategori "Record Operations" dan tindakan "Delete Record".
 - (3) Klik "Next" dan beri nama tombol "Delete Record".
 - (4) Klik "Finish".
- f) Menambahkan Tombol "Save Record":
 - (1) Ulangi langkah untuk menambahkan tombol.
 - (2) Pilih kategori "Record Operations" dan tindakan "Save Record".
 - (3) Klik "Next" dan beri nama tombol "Save Record".
 - (4) Klik "Finish".

4) Menyimpan formulir dengan nama "FormAnggota"

Setelah semua kontrol dan tombol ditambahkan, langkah terakhir adalah menyimpan formulir tersebut. Berikut adalah langkah-langkah untuk menyimpan formulir:

- a) Klik ikon "Save" atau tekan "Ctrl + S" pada keyboard.
- b) Kotak dialog "Save As" akan muncul.
- c) Masukkan nama "FormAnggota" dalam kotak dialog tersebut.
- d) Klik "OK" untuk menyimpan formulir.



The image shows a screenshot of a Microsoft Access form titled "Anggota". The form has a light blue header bar with the title "Anggota" and a small icon on the left. Below the header, there are four text boxes for data entry: "Id_Anggota" containing the number "1", "Nama" containing "Tri", "Alamat" containing "kayumerah", and "No_Telpon" containing "081342479728". To the left of these text boxes are four dashed-line boxes, likely for grid lines. Below the text boxes are three blue buttons with white text: "Add Record", "Save Record", and "Delete Record".

Gambar 5.25. Tampilan Form Anggota

E. Membuat Formulir Transaksi Peminjaman

1. Mendesain Formulir

Langkah pertama dalam membuat formulir transaksi peminjaman adalah mendesain tampilan formulir menggunakan fitur otomatis dari Microsoft Access. Fitur ini memungkinkan Anda untuk membuat formulir secara cepat

dan mudah dari tabel yang sudah ada. Berikut adalah langkah-langkahnya:

- a. Buka Microsoft Access dan pastikan database perpustakaan (misalnya, Perpustakaan.accdb) telah dibuka.
- b. Di panel navigasi sebelah kiri, klik pada tabel "Peminjaman" untuk memastikan tabel ini aktif.
- c. Pilih tab "Create" di pita menu (ribbon).
- d. Di tab "Create", klik "Form". Microsoft Access akan secara otomatis membuat formulir baru yang berisi semua field dari tabel "Peminjaman".

2. Menambahkan Kontrol untuk Field Tabel

Setelah memilih tabel "Peminjaman", langkah berikutnya adalah menggunakan fitur otomatis untuk membuat formulir yang berisi semua field dari tabel tersebut:

- a. Formulir yang dibuat akan berisi kontrol untuk setiap field dalam tabel, seperti:
 - 1) ID_Peminjaman (AutoNumber).
 - 2) ID_Buku (Number).
 - 3) ID_Anggota (Number).
 - 4) Tanggal_Pinjam (Date/Time).
 - 5) Tanggal_Kembali (Date/Time).
- b. Atur posisi dan ukuran kontrol sesuai kebutuhan untuk memastikan formulir mudah digunakan dan navigasi menjadi intuitif.

3. Menambahkan Tombol "Add Record", "Save Record", dan "Delete Record"

Setelah kontrol untuk field ditambahkan, langkah berikutnya adalah menambahkan tombol untuk fungsi "Add Record", "Save Record", dan "Delete Record":

Pastikan Anda berada dalam mode desain formulir dengan mengklik kanan formulir dan memilih "Design View".

Di tab "Design" pada ribbon, pilih "Button" di grup "Controls". Tempatkan tombol di area formulir yang diinginkan.

4. Menambahkan Tombol "Add Record":

- a. Setelah menempatkan tombol, wizard akan muncul.
- b. Pilih kategori "Record Operations" dan tindakan "Add New Record".
- c. Klik "Next" dan beri nama tombol "Add Record".
- d. Klik "Finish".

5. Menambahkan Tombol "Save Record":

- a. Ulangi langkah untuk menambahkan tombol.
- b. Pilih kategori "Record Operations" dan tindakan "Save Record".
- c. Klik "Next" dan beri nama tombol "Save Record".
- d. Klik "Finish".

6. Menambahkan Tombol "Delete Record":

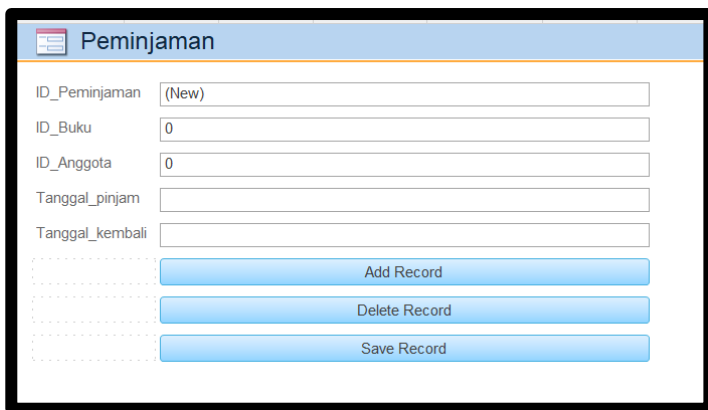
- a. Ulangi langkah untuk menambahkan tombol.

- b. Pilih kategori "Record Operations" dan tindakan "Delete Record".
- c. Klik "Next" dan beri nama tombol "Delete Record".
- d. Klik "Finish".

7. Menyimpan Formulir dengan Nama "FormPeminjaman"

Setelah semua kontrol dan tombol ditambahkan, langkah terakhir adalah menyimpan formulir tersebut. Berikut adalah langkah-langkah untuk menyimpan formulir:

- a. Klik ikon "Save" atau tekan "Ctrl + S" pada keyboard.
- b. Kotak dialog "Save As" akan muncul.
- c. Masukkan nama "FormPeminjaman" dalam kotak dialog tersebut.
- d. Klik "OK" untuk menyimpan formulir.



The image shows a web form titled "Peminjaman" (Loan). It contains several input fields and three buttons. The fields are: ID_Peminjaman (with "(New)" as a placeholder), ID_Buku (with "0"), ID_Anggota (with "0"), Tanggal_pinjam, and Tanggal_kembali. Below the fields are three blue buttons: "Add Record", "Delete Record", and "Save Record".

Gambar 5.26. Tampilan Form Peminjaman

F. Membuat Query

1. Menggunakan "Query Design" untuk Membuat Query

Langkah pertama dalam membuat query adalah menggunakan fitur "Query Design" di Microsoft Access. Query ini akan digunakan untuk menggabungkan data dari beberapa tabel dan menampilkan informasi yang diperlukan. Berikut adalah langkah-langkahnya:

- a. Buka Microsoft Access dan pastikan database perpustakaan (misalnya, Perpustakaan.accdb) telah dibuka.
- b. Pilih tab "Create" di pita menu (ribbon).
- c. Di tab "Create", klik "Query Design". Microsoft Access akan membuka jendela desain query yang kosong.

2. Menambahkan Tabel ke dalam Query

Setelah membuka jendela desain query, langkah berikutnya adalah menambahkan tabel yang relevan ke dalam query. Untuk query transaksi peminjaman, kita akan menambahkan tabel "Peminjaman", "Buku", dan "Anggota". Berikut adalah langkah-langkahnya:

- a. Di jendela desain query, akan muncul kotak dialog "Show Table".
- b. Pilih tabel "Peminjaman" dan klik "Add".
- c. Pilih tabel "Buku" dan klik "Add".
- d. Pilih tabel "Anggota" dan klik "Add".
- e. Setelah semua tabel ditambahkan, klik "Close" untuk menutup kotak dialog "Show Table".

3. Memilih Field yang Ingin Ditampilkan

Setelah menambahkan tabel ke dalam query, langkah berikutnya adalah memilih field yang ingin ditampilkan dalam hasil query. Berikut adalah langkah-langkahnya:

- a. Di jendela desain query, Anda akan melihat daftar field dari setiap tabel yang ditambahkan.
- b. Klik dua kali pada field berikut untuk menambahkannya ke grid desain query:
 - 1) Dari tabel "Anggota", pilih field "Nama".
 - 2) Dari tabel "Buku", pilih field "Judul".
 - 3) Dari tabel "Peminjaman", pilih field "Tanggal_Pinjam" dan "Tanggal_Kembali".
- c. Pastikan field-field yang dipilih muncul di grid desain query di bagian bawah jendela.

4. Menyimpan Query dengan Nama "QueryPeminjaman"

Setelah selesai mendesain query, langkah terakhir adalah menyimpan query tersebut. Berikut adalah langkah-langkah untuk menyimpan query:

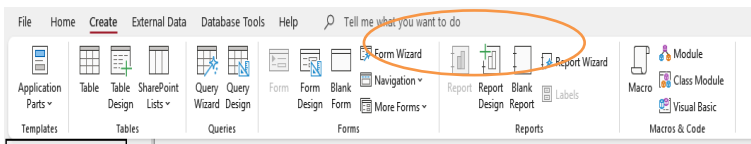
- a. Klik ikon "Save" atau tekan "Ctrl + S" pada keyboard.
- b. Kotak dialog "Save As" akan muncul.
- c. Masukkan nama "QueryPeminjaman" dalam kotak dialog tersebut.
- d. Klik "OK" untuk menyimpan query.

G. Membuat Report

1. Menggunakan "Report Wizard" untuk Membuat Report

Langkah pertama dalam membuat report adalah menggunakan fitur "Report Wizard" di Microsoft Access. Fitur ini memungkinkan Anda untuk membuat laporan secara cepat dan mudah dari query yang sudah ada. Berikut adalah langkah-langkahnya:

- Buka Microsoft Access dan pastikan database perpustakaan (misalnya, Perpustakaan.accdb) telah dibuka.
- Pilih tab "Create" di pita menu (ribbon).
- Di tab "Create", klik "Report Wizard". Microsoft Access akan membuka wizard untuk membuat laporan baru.

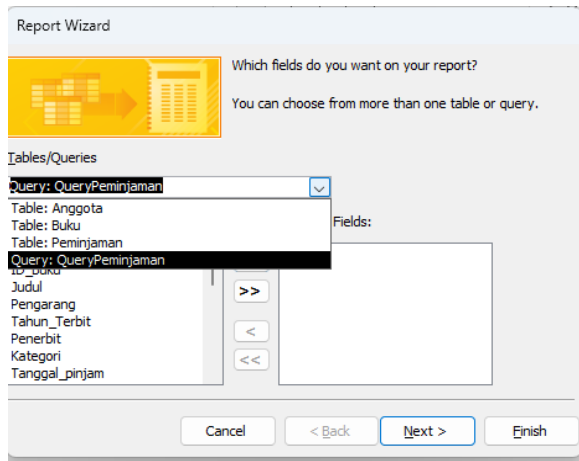


Gambar 5.27. Tampilan Report Wizard

2. Menambahkan Field dari Query

Setelah membuka "Report Wizard", langkah berikutnya adalah menambahkan field dari query yang ingin ditampilkan dalam laporan. Berikut adalah langkah-langkahnya:

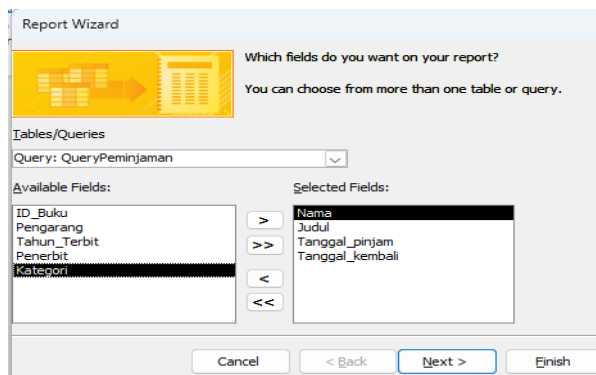
- Di jendela "Report Wizard", Anda akan diminta untuk memilih sumber data untuk laporan. Pilih "Query" dan kemudian pilih "QueryPeminjaman" sebagai sumber data.



Gambar 5.28. Query Peminjaman pada Report Wizard

b. Pilih field-field berikut dari "QueryPeminjaman" untuk ditambahkan ke laporan:

- 1) Nama (dari tabel Anggota).
- 2) Judul (dari tabel Buku).
- 3) Tanggal_Pinjam (dari tabel Peminjaman).
- 4) Tanggal_Kembali (dari tabel Peminjaman).



Gambar 5.29. Tampilan Pemilihan Field report

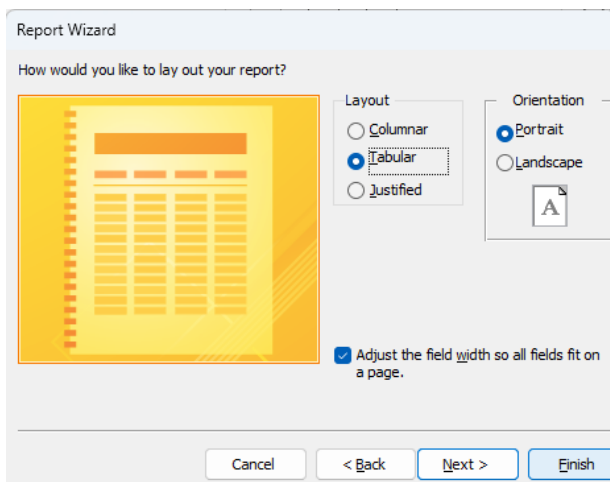
c. Klik tombol ">" untuk memindahkan field-field yang dipilih ke daftar field yang akan ditampilkan di laporan.

d. Klik "Next" untuk melanjutkan.

3. Menyesuaikan Layout Report

Setelah menambahkan field ke laporan, langkah berikutnya adalah menyesuaikan layout laporan sesuai kebutuhan. Berikut adalah langkah-langkahnya:

- Di jendela "Report Wizard", Anda akan diminta untuk memilih layout laporan. Pilih layout yang paling sesuai dengan kebutuhan Anda, seperti "Columnar", "Tabular", atau "Justified".
- Klik "Next" untuk melanjutkan.
- Pilih orientasi laporan, apakah "Portrait" (vertikal) atau "Landscape" (horizontal).
- Klik "Next" untuk melanjutkan.
- Pilih gaya laporan yang diinginkan dari daftar yang tersedia.
- Klik "Next" untuk melanjutkan.

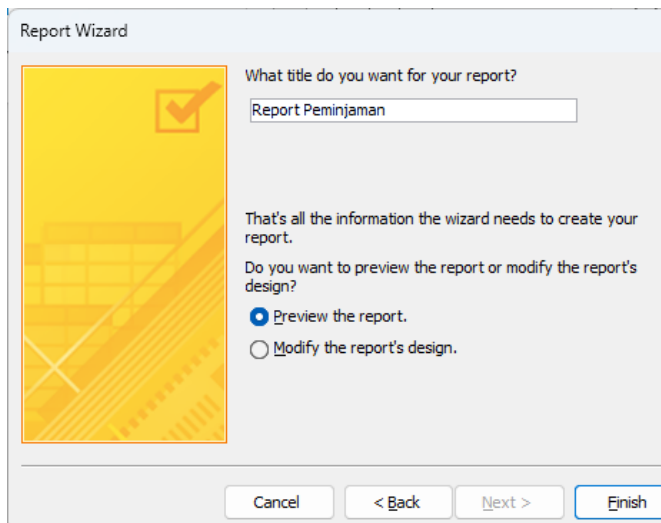


Gambar 5.30. Penyusunan layout report

4. Menyimpan Report dengan Nama "ReportPeminjaman"

Setelah menyesuaikan layout laporan, langkah terakhir adalah menyimpan laporan tersebut. Berikut adalah langkah-langkah untuk menyimpan laporan:

- a. Di jendela "Report Wizard", Anda akan diminta untuk memberikan nama untuk laporan.
- b. Masukkan nama "ReportPeminjaman" dalam kotak dialog tersebut.
- c. Pilih apakah Anda ingin melihat laporan segera atau memodifikasi desain laporan.
- d. Klik "Finish" untuk menyimpan laporan dan menyelesaikan wizard.



Gambar 5.31. Penamaan Laporan

H. Membuat Menu Navigasi Utama

1. Mendesain Form Menu Utama

Langkah pertama dalam membuat menu navigasi utama adalah mendesain tampilan form yang akan berfungsi sebagai pusat navigasi untuk aplikasi sistem manajemen perpustakaan. Berikut adalah langkah-langkahnya:

- a. Buka Microsoft Access dan pastikan database perpustakaan (misalnya, Perpustakaan.accdb) telah dibuka.
- b. Pilih tab "Create" di pita menu (ribbon).
- c. Klik "Form Design" untuk membuat form baru.
- d. Microsoft Access akan menampilkan tampilan desain form kosong yang siap untuk dirancang sesuai kebutuhan.
- e. Sesuaikan ukuran form dan tambahkan elemen desain seperti label judul, misalnya "Menu Utama".

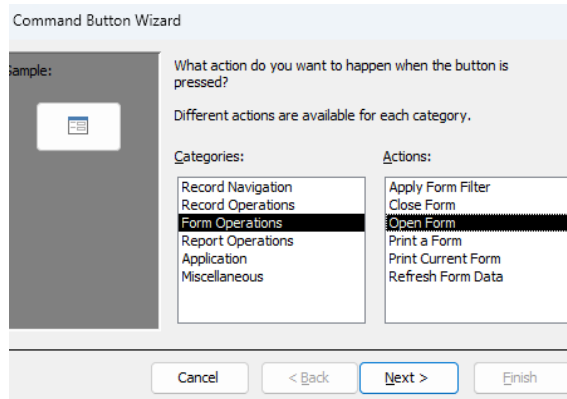
2. Menambahkan Tombol untuk Membuka Form dan Report

Setelah mendesain tampilan dasar form menu utama, langkah berikutnya adalah menambahkan tombol yang memungkinkan pengguna untuk membuka form dan report yang telah dibuat. Berikut adalah langkah-langkahnya:

- a. Di tab "Design" pada ribbon, pilih "Button" di grup "Controls".
- b. Tempatkan tombol di area form yang diinginkan.

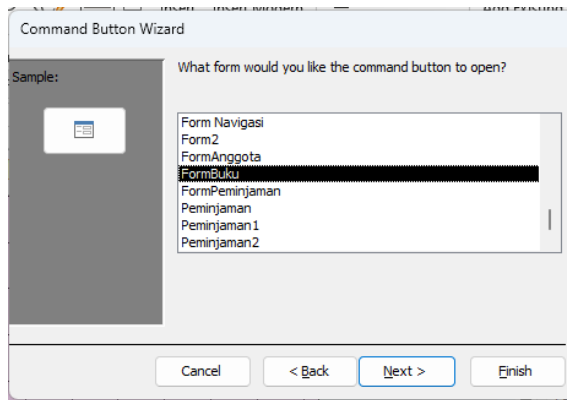
Menambahkan Tombol untuk Membuka Form Buku:

- a. Setelah menempatkan tombol, wizard akan muncul.
- b. Pilih kategori "Form Operations" dan tindakan "Open Form".



Gambar 5.32. Penambahan Operation Open pada tombol

- c. Klik "Next" dan pilih "FormBuku" dari daftar form.



Gambar 5.33. Pemilihan Form yang akan dibuka

- d. Klik "Next" dan beri nama tombol, misalnya "Buku".
 e. Klik "Finish".

Menambahkan Tombol untuk Membuka Form Anggota:

- a. Ulangi langkah untuk menambahkan tombol.
 b. Pilih kategori "Form Operations" dan tindakan "Open Form".

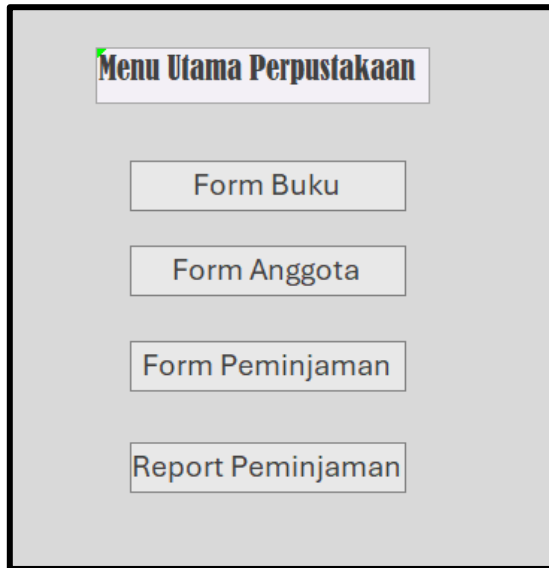
- c. Klik "Next" dan pilih "FormAnggota" dari daftar form.
- d. Klik "Next" dan beri nama tombol, misalnya "Anggota".
- e. Klik "Finish".

Menambahkan Tombol untuk Membuka Form Peminjaman:

- a. Ulangi langkah untuk menambahkan tombol.
- b. Pilih kategori "Form Operations" dan tindakan "Open Form".
- c. Klik "Next" dan pilih "FormPeminjaman" dari daftar form.
- d. Klik "Next" dan beri nama tombol, misalnya "Peminjaman".
- e. Klik "Finish".

Menambahkan Tombol untuk Membuka Report Peminjaman:

- a. Ulangi langkah untuk menambahkan tombol.
- b. Pilih kategori "Report Operations" dan tindakan "Preview Report".
- c. Klik "Next" dan pilih "ReportPeminjaman" dari daftar report.
- d. Klik "Next" dan beri nama tombol, misalnya "Laporan Peminjaman".
- e. Klik "Finish".



Gambar 5.34. Form Menu Utama Perpustakaan

3. Mengatur Properti Tombol

Setelah menambahkan tombol, langkah berikutnya adalah mengatur properti tombol untuk memastikan bahwa mereka berfungsi dengan benar dan memiliki label yang sesuai. Berikut adalah langkah-langkahnya:

- a. Pilih tombol yang telah ditambahkan ke form.
- b. Di panel properti di sebelah kanan, atur properti berikut:
 - 1) **Caption:** Beri label yang sesuai untuk tombol, seperti "Buku", "Anggota", "Peminjaman", atau "Laporan Peminjaman".
 - 2) **On Click:** Pastikan tindakan yang sesuai (misalnya, membuka form atau report) telah dikonfigurasi melalui wizard sebelumnya.

4. Menyimpan Form Menu Utama dengan Nama "MainMenu"

Setelah semua kontrol dan tombol ditambahkan dan dikonfigurasi, langkah terakhir adalah menyimpan form tersebut. Berikut adalah langkah-langkah untuk menyimpan form:

- a. Klik ikon "Save" atau tekan "Ctrl + S" pada keyboard.
- b. Kotak dialog "Save As" akan muncul.
- c. Masukkan nama "MainMenu" dalam kotak dialog tersebut.
- d. Klik "OK" untuk menyimpan form.

I. Meningkatkan Aplikasi

1. Menghubungkan Tombol ke Formulir dan Report

Langkah pertama dalam menghubungkan tombol ke formulir dan report adalah membangun event macro yang akan menjalankan aksi ketika tombol ditekan. Event macro adalah serangkaian instruksi yang dijalankan secara otomatis sebagai respons terhadap suatu peristiwa (*event*), seperti mengklik tombol. Berikut adalah langkah-langkahnya:

- a. Buka form "MainMenu" dalam mode desain (Design View) di Microsoft Access.
- b. Pilih tombol yang ingin Anda tambahkan event macro-nya.
- c. Di tab "Design", pilih "Property Sheet" untuk membuka panel properti.
- d. Di panel properti, pilih tab "Event".
- e. Temukan properti "On Click" dan klik tombol "..." di sebelah kanan untuk membuka "Choose Builder" dialog.
- f. Pilih "Macro Builder" dan klik "OK".

- g. Microsoft Access akan membuka jendela "Macro Builder" yang memungkinkan Anda untuk membuat macro baru.
- h. Di jendela "Macro Builder", tambahkan aksi yang sesuai untuk event "On Click" tombol tersebut, misalnya:
 - 1) Untuk membuka form, tambahkan aksi "OpenForm" dan pilih nama form yang ingin dibuka.
 - 2) Untuk membuka report, tambahkan aksi "OpenReport" dan pilih nama report yang ingin dibuka.
- i. Simpan macro dengan nama yang sesuai, misalnya "OpenFormBukuMacro" atau "OpenReportPeminjamanMacro".

2. Menetapkan Aksi ke Tombol

Setelah membangun event macro, langkah berikutnya adalah menetapkan aksi tersebut ke tombol yang sesuai. Berikut adalah langkah-langkahnya:

- a. Pastikan Anda masih berada di mode desain (Design View) pada form "MainMenu".
- b. Pilih tombol yang ingin Anda tambahkan aksi-nya.
- c. Di panel properti, pilih tab "Event".
- d. Temukan properti "On Click" dan pastikan nama macro yang telah Anda buat sebelumnya terpilih dalam daftar.
- e. Ulangi langkah ini untuk setiap tombol yang ingin Anda tambahkan aksi, dengan menggunakan macro yang sesuai untuk setiap tombol.
- f. Simpan perubahan pada form "MainMenu".

J. Menyembunyikan Panel Navigasi dan Ribbon

Mengatur Opsi Tampilan di "File" > "Options"

Langkah pertama untuk menyembunyikan panel navigasi dan ribbon adalah mengatur opsi tampilan di Microsoft Access. Langkah ini memastikan bahwa elemen antarmuka yang tidak diperlukan disembunyikan untuk memberikan tampilan yang lebih bersih dan profesional pada aplikasi Anda. Berikut adalah langkah-langkahnya:

1. Buka Microsoft Access dan pastikan database perpustakaan (misalnya, Perpustakaan.accdb) telah dibuka.
2. Klik tab "File" di pita menu (ribbon) untuk membuka menu Backstage.
3. Pilih "Options" dari menu yang muncul. Jendela "Access Options" akan terbuka.
4. Di jendela "Access Options", pilih "Current Database" dari daftar di sebelah kiri.

Menyembunyikan Elemen Antarmuka yang Tidak Diperlukan

Setelah membuka jendela "Access Options" dan memilih "Current Database", langkah berikutnya adalah menyembunyikan elemen antarmuka yang tidak diperlukan. Berikut adalah langkah-langkahnya:

1. Menyembunyikan Panel Navigasi:

Di bagian "Navigation", hapus centang pada "Display Navigation Pane". Ini akan menyembunyikan panel navigasi yang biasanya ditampilkan di sisi kiri jendela Access.

2. Menyembunyikan Ribbon:

Di bagian "Ribbon and Toolbar Options", pilih opsi "Hide the Ribbon". Ini akan menyembunyikan ribbon di bagian atas jendela Access.

3. Mengatur Form Tampilan Default:

Di bagian "Display Form", pilih form "MainMenu" dari daftar. Ini akan mengatur form "MainMenu" sebagai tampilan default saat database dibuka, sehingga pengguna langsung disajikan dengan menu utama aplikasi.

4. Klik "OK" untuk menyimpan pengaturan. Anda mungkin perlu menutup dan membuka kembali database untuk melihat perubahan.

K. Mengatur Menu Navigasi Utama sebagai Formulir Default

Mengatur Opsi Startup di "File" > "Options"

Langkah pertama untuk mengatur menu navigasi utama sebagai formulir default adalah mengkonfigurasi opsi startup di Microsoft Access. Opsi ini memungkinkan Anda untuk menentukan form mana yang akan ditampilkan pertama kali ketika database dibuka. Berikut adalah langkah-langkahnya:

1. Buka Microsoft Access dan pastikan database perpustakaan (misalnya, Perpustakaan.accdb) telah dibuka.
2. Klik tab "File" di pita menu (ribbon) untuk membuka menu Backstage.

3. Pilih "Options" dari menu yang muncul. Jendela "Access Options" akan terbuka.
4. Di jendela "Access Options", pilih "Current Database" dari daftar di sebelah kiri.

Mengatur "MainMenu" sebagai Form Tampilan Default

Setelah membuka jendela "Access Options" dan memilih "Current Database", langkah berikutnya adalah mengatur form "MainMenu" sebagai form tampilan default. Berikut adalah langkah-langkahnya:

1. Mengatur Form Tampilan Default:

- a. Di bagian "Application Options", cari opsi "Display Form".
- b. Klik dropdown di sebelah "Display Form" dan pilih "MainMenu" dari daftar form yang tersedia. Ini akan mengatur form "MainMenu" sebagai tampilan default saat database dibuka.

2. Pengaturan Tambahan:

Jika diinginkan, Anda dapat menyesuaikan opsi startup lainnya, seperti menyembunyikan panel navigasi, menonaktifkan tampilan Backstage saat aplikasi dimulai, atau mengatur ikon aplikasi dan judul.

3. Klik "OK" untuk menyimpan pengaturan. Anda mungkin perlu menutup dan membuka kembali database untuk melihat perubahan.

L. Menambahkan Keamanan dan Hak Akses

Membuat Tabel Pengguna untuk Login Credentials

Langkah pertama untuk menambahkan keamanan dan hak akses adalah membuat tabel pengguna yang akan menyimpan login credentials. Tabel ini akan berisi informasi seperti username dan password untuk setiap pengguna. Berikut adalah langkah-langkahnya:

1. Buka Microsoft Access dan pastikan database perpustakaan (misalnya, Perpustakaan.accdb) telah dibuka.
2. Pilih tab "Create" di pita menu (ribbon).
3. Klik "Table Design" untuk membuat tabel baru.
4. Tambahkan field-field berikut ke tabel:
 - a. UserID (AutoNumber, Primary Key)
 - b. Username (Short Text)
 - c. Password (Short Text)
 - d. UserRole (Short Text) - untuk menentukan peran pengguna, seperti Admin atau User.
5. Simpan tabel dengan nama "Users".

Mendesain Form Login

Setelah membuat tabel pengguna, langkah berikutnya adalah mendesain form login yang akan digunakan untuk autentikasi pengguna. Berikut adalah langkah-langkahnya:

- a. Pilih tab "Create" di pita menu (ribbon).
- b. Klik "Form Design" untuk membuat form baru.
- c. Tambahkan dua kontrol kotak teks (Text Box) dan satu kontrol tombol (Button) ke form:
 - 1) Text Box pertama untuk memasukkan "Username".
 - 2) Text Box kedua untuk memasukkan "Password".

- 3) Button untuk melakukan proses login, beri label "Login".
- d. Atur kontrol kotak teks dan tombol sesuai dengan kebutuhan dan desain yang diinginkan.
- e. Simpan form dengan nama "LoginForm".

Mengimplementasikan VBA untuk Autentikasi Pengguna

Langkah berikutnya adalah mengimplementasikan VBA (Visual Basic for Applications) untuk autentikasi pengguna. VBA akan digunakan untuk memeriksa username dan password yang dimasukkan dengan data yang ada di tabel "Users". Berikut adalah langkah-langkahnya:

- a. Buka form "LoginForm" dalam mode desain (Design View).
- b. Klik tombol "Login" untuk memilihnya, lalu buka panel properti dan pilih tab "Event".
- c. Di properti "On Click", klik tombol "..." untuk membuka "Choose Builder" dialog, lalu pilih "Code Builder" dan klik "OK".
- d. Masukkan kode VBA berikut untuk memeriksa username dan password:

```
Private Sub LoginButton_Click()
```

```
    Dim db As DAO.Database
```

```
    Dim rs As DAO.Recordset
```

```
    Dim strSQL As String
```

```
    Set db = CurrentDb()
```

```
    strSQL = "SELECT * FROM Users WHERE Username = '"  
& Me.UsernameTextBox & "' AND Password = '" &  
Me.PasswordTextBox & """
```

```

Set rs = db.OpenRecordset(strSQL)

If rs.EOF Then
    MsgBox "Username atau Password salah.",
vbExclamation
Else
    ' Autentikasi berhasil, buka form utama berdasarkan
peran pengguna
    If rs!UserRole = "Admin" Then
        DoCmd.OpenForm "AdminForm"
    ElseIf rs!UserRole = "User" Then
        DoCmd.OpenForm "UserForm"
    End If
    DoCmd.Close acForm, "LoginForm"
End If

rs.Close
Set rs = Nothing
Set db = Nothing
End Sub

```


Kesimpulan

Proses pengembangan aplikasi sistem manajemen perpustakaan menggunakan Microsoft Access telah melibatkan beberapa langkah penting untuk memastikan aplikasi berfungsi dengan baik dan memenuhi kebutuhan perpustakaan. Berikut adalah ringkasan dari proses tersebut:

1. Pembuatan Database Baru:

Membuat database baru dengan Microsoft Access dan menamainya sesuai kebutuhan (misalnya, Perpustakaan.accdb).

2. Membuat Tabel:

Membuat tabel yang diperlukan untuk menyimpan data buku, anggota, dan transaksi peminjaman. Tabel-tabel ini mencakup field-field yang relevan dan memiliki tipe data yang sesuai.

3. Membangun Relasi Antar-Tabel:

Membangun relasi antara tabel untuk memastikan integritas data dan memungkinkan query yang efisien.

4. Membuat Formulir:

Mendesain formulir untuk mengelola data buku, anggota, dan transaksi peminjaman. Formulir ini memudahkan pengguna dalam memasukkan dan mengedit data.

5. Membuat Query dan Report:

- a. Membuat query untuk menggabungkan data dari beberapa tabel dan menampilkan informasi yang diperlukan.
- b. Mendesain laporan (report) untuk menyajikan data peminjaman dalam format yang mudah dibaca dan dicetak.

6. Membangun Menu Navigasi Utama:

Mendesain form menu utama yang berfungsi sebagai pusat navigasi untuk aplikasi. Form ini dilengkapi dengan tombol untuk membuka form dan report yang telah dibuat.

7. Meningkatkan Aplikasi:

- a. Menyembunyikan elemen antarmuka yang tidak diperlukan seperti panel navigasi dan ribbon untuk memberikan tampilan yang lebih bersih dan profesional.
- b. Menambahkan keamanan dan hak akses dengan membuat tabel pengguna, mendesain form login, mengimplementasikan VBA untuk autentikasi pengguna, dan menetapkan level akses untuk berbagai pengguna.

Tentang Penulis



Andi Asari, S.I.P., S.Kom., M.A.

Dosen Perpustakaan dan Sains Informasi
Fakultas Sastra

Penulis adalah dosen di Universitas Negeri Malang yang saat ini sedang melanjutkan studi doctoral (S3) di jurusan Information Management UiTM Malaysia. Penulis merupakan alumni dari Magister Kajian Budaya dan Media sekolah pasca sarjana Universitas Gadjah Mada Yogyakarta, dan juga alumni dari jurusan Ilmu Perpustakaan UIN Sunan Kalijaga Yogyakarta, serta alumni jurusan Teknik Informatika STMIK.

Mulai tahun 2015 sampai sekarang penulis aktif mengajar di Jurusan Sastra Indonesia, Prodi S1 Ilmu Perpustakaan dan D4 Perpustakaan Digital Universitas Negeri Malang. Disamping kesibukan di dunia akademis juga memiliki kegiatan sebagai narasumber pada kegiatan seminar, workshop, konsultan lembaga pendidikan dan perpustakaan.



Bernadete Deta, M.Kom.

Dosen Teknik Informatika

Fakultas Teknik Informatika Larantuka

Penulis lahir di Larantuka-Flores Timur tanggal 31 Juli. Penulis adalah dosen pada Program Studi Teknik Informatika Fakultas Teknik, Institut Keguruan dan Teknologi Larantuka (IKTL). Menyelesaikan pendidikan S1 pada Jurusan Sistem Informasi Universitas Dinamika Surabaya tahun 2018 dan Magister management sistem Informasi dari Binus University tahun 2022. Bidang pengajaran dan penelitian penulis adalah Struktur Data, Sistem Basis Data, Management Basis Data, Komputasi Nomerik, Jaringan komputer dan Pemograman Jaringan Komputer.



Syahrial

Dosen Ilmu Komputer

Fakultas Sains dan Ilmu Komputer Universitas Muhammadiyah
Gorontalo

Penulis lahir di Barru, Sulawesi Selatan pada tanggal 18 Agustus 1981. Penulis adalah dosen tetap pada Program Studi Ilmu Komputer Fakultas Sains dan Ilmu Komputer, Universitas Muhammadiyah Gorontalo. Menyelesaikan pendidikan S1 pada Jurusan Teknik Informatika di Universitas Ichsan Gorontalo tahun 2014 dan melanjutkan S2 pada Jurusan Teknik Informatika di Universitas Dian Nuswantoro Semarang tahun 2016. Penulis menekuni bidang Machine Learning, Data Science dan Struktur Data.



Hidayatus Sibyan

Dosen Teknik Informatika
Fakultas Teknik dan Ilmu Komputer
Universitas Sains Al-Qur'an

Penulis lahir di Wonosobo tanggal 30 November 1989. Penulis adalah dosen pada Program Studi Teknik Informatika, Fakultas Teknik dan Ilmu Komputer (FASTIKOM), Universitas Sains Al-Qur'an (UNSIQ). Menyelesaikan pendidikan S1 pada Program Studi Teknik Informatika, Universitas Sains Al-Qur'an pada tahun 2012 dan melanjutkan S2 pada Program Studi Magister Teknik Informatika, STMIK Amikom Yogyakarta pada tahun 2016.

Penulis saat ini menjadi tenaga pendidik di Universitas Sains Al-Qur'an (UNSIQ) Wonosobo sejak tahun 2016. Mata kuliah yang diampu di antaranya Algoritma dan Dasar Pemrograman, Algoritma dan Struktur Data, Basis Data, dan Sains Data. Penulis menekuni Penelitian di bidang Informatika dan Sistem Cerdas.



Tri Pratiwi Handayani, S.Kom., M.ENG., M.PHIL.

Dosen Sistem Informasi
Fakultas Sains dan Ilmu Komputer
Universitas Muhammadiyah Gorontalo

Penulis lahir di Palembang tanggal 4 Mei 1986. Penulis adalah dosen Basis Data pada Program studi Sistem Informasi, Fakultas Sains dan ilmu Komputer, Universitas Muhammadiyah Gorontalo. Menyelesaikan pendidikan S1 Ilmu Komputer Universitas Gadjah Mada dan melanjutkan S2 di Newcastle University. Penulis menekuni bidang Penelitian Sistem Pendukung Keputusan dan Kecerdasan Buatan.