



UNIVERSITAS BHAYANGKARA JAKARTA RAYA
FAKULTAS ILMU KOMPUTER

Kampus I: Jl. Harsono RM No. 67, Ragunan, Pasar Minggu, Jakarta Selatan, 12550
Telepon: (021) 27808121 – 27808882
Kampus II: Jl. Raya Perjuangan, Marga Mulya, Bekasi Utara, Jawa Barat, 17142
Telepon: (021) 88955882, Fax.: (021) 88955871
Web: fasilkom.ubharajaya.ac.id, E-mail: fasilkom@ubharajaya.ac.id

SURAT TUGAS

Nomor: ST/1182/X/2024/FASILKOM-UBJ

1. Dasar: Kalender Akademik Universitas Bhayangkara Jakarta Raya Tahun Akademik 2024/2025.
2. Dalam rangka mewujudkan Tri Dharma Perguruan Tinggi untuk Dosen di Universitas Bhayangkara Jakarta Raya maka dihimbau untuk melakukan Penelitian.
3. Sehubungan dengan hal tersebut di atas, maka Dekan Fakultas Ilmu Komputer Universitas Bhayangkara Jakarta Raya menugaskan:

NO.	NAMA	NIDN	JABATAN
1.	Rafika Sari, S.Si., M.Si.	0329098902	Dosen Tetap Prodi Informatika
2.	Dr. Rakhmi Khalida, S.T., M.M.S.I.	0304099201	Dosen Tetap Prodi Informatika
3.	Aida Fitriyani, S.Kom., M.M.S.I.	0302078508	Dosen Tetap Prodi Informatika

Membuat Buku dengan judul **“Pengantar Ilmu Komputer”** yang diterbitkan oleh U ME Publishing, Cetakan Pertama: Desember 2024, ISBN: 978-623-10-6199-7.

Ditetapkan di : Jakarta

Pada tanggal : 21 Oktober 2024

DEKAN FAKULTAS ILMU KOMPUTER
UNIVERSITAS BHAYANGKARA JAKARTA RAYA



Dr. Dra. Tyastuti Sri Lestari, M.M.
NIP. 1408206

SERTIFIKAT

UME.025/K.002.003/I/2025

Diberikan Kepada:

Aida Fitriyani

Atas partisipasinya menjadi PENULIS di Penerbit U ME PUBLISHING dengan judul :

PENGANTAR ILMU KOMPUTER

ISBN : 978-623-10-6199-7



Direktur CV. U ME Publishing



U ME
PUBLISHING

Eldi Miharni



PENGANTAR ILMU KOMPUTER



Penulis :

Rafika Sari, Rakhmi Khalida, Aida Fitriyani,
Achmad Ardiansyah, Mukrodin, Mepa Kurniasih

PENGANTAR ILMU KOMPUTER

**Rafika Sari
Rakhmi Khalida
Aida Fitriyani
Achmad Ardiansyah
Mukrodin
Mepa Kurniasih**



PENGANTAR ILMU KOMPUTER

Penulis :

Rafika Sari
Rakhmi Khalida
Aida Fitriyani
Achmad Ardiansyah
Mukrodin
Mepa Kurniasih

Editor : Alfauzain, S.kom, M.kom

Penyunting : Muhammad Ikhlas Al Kuthsi, S.Kom, M.M

Desain Sampul dan Tata Letak : Yayang Tineza Erwanda, S.E

Diterbitkan oleh :

U ME Publishing

Anggota IKAPI No. 059/SBA/2024

Jl. Perumdam 4 Blok H No.2 Tunggul Hitam Kota Padang,
Sumatera Barat

Email : kontak@umepublishing.com

Website : umepublishing.com

ISBN : 978-623-10-6199-7

Cetakan pertama, Desember 2024

Hak cipta dilindungi undang-undang.

Dilarang keras memperbanyak, memfotokopi, Sebagian atau seluruh isi buku tanpa izin tertulis dari penerbit.

KATA PENGANTAR

Dengan mengucapkan puji syukur kehadirat Allah SWT, atas limpahan rahmat dan hidayah-Nya, maka Penulisan Buku dengan judul Pengantar Ilmu Komputer dapat diselesaikan. Buku ini membahas tentang Pendahuluan Pengantar Ilmu Komputer, Komponen-Komponen Komputer, Sistem Operasi, Pemrograman, Database, Tren dan Masa Depan Teknologi Komputer.

Oleh karena itu, kami sangat mengaharapkan kritik dan saran demi perbaikan dan kesempurnaan buku ini selanjutnya. Kami mengucapkan terima kasih kepada berbagai pihak yang telah membantu dalam proses penyelesaian Buku ini. Semoga Buku ini dapat menjadi sumber referensi dan literatur yang mudah dipahami.

Padang, November 2024

Penulis

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI	ii
DAFTAR GAMBAR	v
BAB 1 PENGANTAR ILMU KOMPUTER	1
1.1 Definisi dan Ruang Lingkup Ilmu Komputer	1
1.1.1 Pengertian Ilmu Komputer	1
1.1.2 Hubungan Ilmu Komputer- Komputer dengan Disiplin Ilmu Lainnya	2
1.1.3 Peran Ilmu Komputer dalam Berbagai Bidang	3
1.2 Sejarah dan Perkembangan Ilmu Komputer	5
1.2.1 Sejarah Awal Komputasi	5
1.2.2 Perkembangan Perangkat Keras dan Perangkat Lunak	6
1.2.3 Munculnya Sub-Bidang dalam Ilmu Komputer	7
1.3 Hubungan Ilmu Komputer dengan Teknologi Informasi	8
1.4 Peran dan Dampak Ilmu Komputer di Masyarakat	11
1.4.1 Dampak Sosial, Ekonomi dan Budaya dari Teknologi Komputer	11
1.4.2 Etika dalam Ilmu Komputer	12
DAFTAR PUSTAKA	14
BAB 2 KOMPONEN-KOMPONEN KOMPUTER	15
2.1 Macam Jenis <i>Hardware</i> Komputer	15
2.2 Macam Jenis <i>Software</i> Komputer	26
2.3 Komponen Komputer pada Tujuan Khusus	27
DAFTAR PUSTAKA	31
BAB 3 PENGENALAN SISTEM OPERASI	33
3.1 Pengenalan Sistem Operasi	33

3.1.1 Definisi Sistem Operasi.....	33
3.1.3 Sejarah Sistem Operasi.....	33
3.2 Proses	34
3.2.1 Definisi Proses.....	35
3.2.2 Keadaan Proses	35
3.2.3 Operasi-operasi Terhadap Proses	36
3.3 Thread	36
3.3.1 Definisi Thread.....	37
3.3.2 Model Multithreading.....	37
3.4 Penjadwalan Cpu.....	38
3.4.1 Konsep Dasar	38
3.4.2 Kriteria Penjadwal.....	39
3.5 Sinkronisasi Proses	39
3.5.1 Definisi Sinkronisasi	39
3.5.2 Race Condition	40
3.5.3 Critical Section.....	40
3.5.4 Semaphore.....	40
3.6 <i>Deadlock</i>	41
3.6.1 Definisi <i>Deadlock</i>	41
3.6.2 Strategi Menghadapi <i>Deadlock</i>	42
3.6.3 Mencegah <i>Deadlock</i>	42
3.7 Sistem Berkas.....	43
3.7.1 Konsep Dasar	43
3.7.2 Operasi Pada Berkas.....	44
3.7.3 Jenis Berkas.....	45
3.7.4 Struktur Berkas	45
DAFTAR PUSTAKA.....	46
BAB 4 PEMROGRAMAN	47
4.1 Konsep Dasar Pemrograman	47
4.2 Bahasa Pemrograman	53
4.3 Struktur Data dan Algoritma.....	56
4.4 Pemrograman Berbasis Web	60
4.5 Kesalahan Pemrograman dan Debugging.....	64

DAFTAR PUSTAKA	67
BAB 5 DATABASE	71
5.1 Konsep Dasar Basisdata	71
5.2 Hirarki Database	73
5.3 Komponen Basisdata	74
5.4 Tujuan Basisdata	76
5.5 Manfaat Basisdata	79
5.6 Normalisasi	80
5.7 Normalisasi pada Entity Relationship Diagram	82
5.8 <i>Structured Query Language (SQL)</i>	86
DAFTAR PUSTAKA	91
BAB 6 TREN DAN MASA DEPAN TEKNOLOGI	
KOMPUTER.....	93
6.1 Kecerdasan Buatan dan Pembelajaran Mesin.....	93
6.2 Komputasi Kuantum.....	98
6.3 <i>Internet of Things (IoT)</i>	105
6.4 <i>Realitas Virtual (VR)</i> dan <i>Augmented Reality</i> (AR).....	109
6.5 Blockchain dan Keamanan Digital.....	113
6.6 Robotika dan Otomasi.....	122
6.7 Keberlanjutan dalam Teknologi Komputer	129
6.8 Kesimpulan	135
DAFTAR PUSTAKA	136
BIODATA PENULIS	

DAFTAR GAMBAR

Gambar 3.1. Sejarah Sistem Operasi.....	33
Gambar 3.2. Sejarah Sistem Operasi.....	35
Gambar 3.3. Thread.....	36
Gambar 3.4. Multithreading.....	37
Gambar 3.5. Urutan Kedatangan Proses	38
Gambar 3.6. Sinkronisasi Sistem Operasi.....	39
Gambar 3.7. Deadlock Sistem Operasi.....	41
Gambar 3.8. Sistem Berkas.....	43
Gambar 5.1. Lemari arsip/ data manual	72
Gambar 5.2. Arsip/ data digital.....	72
Gambar 5.3. Hiartki Database	73
Gambar 5.4. Tampilan XAMPP	88
Gambar 5.5. Tampilan Window Command Prompt.....	88
Gambar 5.6. Arahkan ke directory C:\xampp\mysql\bin.....	89
Gambar 5.7. Login Ke MySQL	89
Gambar 5.8. Menampilkan Database yang ada di Server	90

BAB 1

PENGANTAR ILMU KOMPUTER

Oleh Rafika Sari

1.1 Definisi dan Ruang Lingkup Ilmu Komputer

1.1.1 Pengertian Ilmu Komputer

Ilmu komputer adalah disiplin akademik yang berfokus pada studi teori, pengembangan, dan aplikasi sistem komputer. Ini mencakup berbagai aspek, mulai dari pemrograman, algoritma, hingga perangkat keras dan perangkat lunak yang digunakan untuk memproses informasi. Secara umum, ilmu komputer dapat didefinisikan sebagai studi tentang komputasi, termasuk teori, desain, pengembangan, dan penerapan teknologi komputer. (Kadir and Triwahyuni, 2013)

Ilmu komputer bukan hanya sekadar belajar bagaimana menggunakan komputer, tetapi juga mencakup pemahaman tentang bagaimana komputer bekerja, bagaimana sistem dikembangkan, dan bagaimana algoritma dirancang untuk menyelesaikan berbagai masalah. Disiplin ini menggabungkan aspek matematika, logika, dan teknik untuk menciptakan solusi inovatif. Dalam pengertian yang lebih luas, ilmu komputer mencakup dua area utama:

1. **Teori Komputasi:** Mempelajari dasar-dasar matematis komputasi dan kemampuan serta batasan dari sistem komputasi.
2. **Pengembangan Sistem Komputer:** Termasuk pengembangan perangkat lunak, perangkat keras, jaringan, dan sistem yang mendukung operasi teknologi modern.

Beberapa konsep dasar dalam ilmu komputer meliputi algoritma, struktur data, teori bahasa pemrograman, sistem operasi, jaringan komputer, keamanan informasi, kecerdasan buatan, dan komputasi kuantum.

1.1.2 Hubungan Ilmu Komputer- Komputer dengan Disiplin Ilmu Lainnya

Ilmu komputer sering dianggap sebagai disiplin yang terpisah, tetapi memiliki hubungan yang erat dengan banyak bidang lain. Berikut adalah beberapa hubungan ilmu komputer dengan disiplin lainnya:(Sedgewick and Wayne, 2017)

1. Matematika: Ilmu komputer sangat bergantung pada matematika, terutama dalam bidang algoritma, kriptografi, teori graf, dan probabilitas. Konsep seperti logika matematika dan teori himpunan menjadi dasar dalam pengembangan algoritma dan perangkat lunak.
2. Fisika: Dalam pengembangan perangkat keras komputer, fisika memainkan peran penting. Konsep seperti mekanika kuantum digunakan dalam pengembangan komputasi kuantum dan pemrosesan data tingkat tinggi.
3. Biologi: Bioinformatika adalah contoh bagaimana ilmu komputer digunakan untuk memecahkan masalah biologi, seperti analisis DNA dan simulasi evolusi. Bidang ini menggabungkan algoritma komputer dengan data biologis untuk menghasilkan wawasan yang lebih mendalam.
4. Psikologi dan Neurosains: Bidang kecerdasan buatan dan pembelajaran mesin sering mengambil inspirasi dari cara kerja otak manusia. Studi tentang antarmuka otak-komputer (brain-computer interface) juga merupakan hasil kolaborasi antara ilmu komputer dan psikologi.
5. Ekonomi dan Bisnis: Ilmu komputer memainkan peran penting dalam bidang ekonomi, khususnya dalam analisis

data besar (big data), pengambilan keputusan berbasis data, dan otomatisasi proses bisnis. Sistem manajemen basis data (DBMS) dan teknologi blockchain adalah contohnya.

6. Seni dan Desain: Grafik komputer, animasi, dan pengembangan game adalah contoh aplikasi ilmu komputer dalam seni dan desain. Algoritma komputer digunakan untuk menciptakan efek visual yang kompleks.
7. Hukum dan Etika: Dengan meningkatnya masalah privasi dan keamanan data, hubungan antara ilmu komputer, hukum, dan etika menjadi semakin penting. Peraturan seperti GDPR di Uni Eropa adalah contoh dari penerapan hukum dalam bidang teknologi.

Hubungan ini menunjukkan bahwa ilmu komputer adalah disiplin yang interdisipliner dan sangat relevan dengan berbagai bidang ilmu pengetahuan dan teknologi lainnya.

1.1.3 Peran Ilmu Komputer dalam Berbagai Bidang

Ilmu komputer telah merevolusi cara manusia bekerja, belajar, dan berkomunikasi. Berikut adalah beberapa bidang utama di mana ilmu komputer memainkan peran penting:

1. Pendidikan: Teknologi komputer telah mengubah cara pendidikan disampaikan. Dengan e-learning, video konferensi, dan perangkat lunak pembelajaran adaptif, ilmu komputer memungkinkan akses pendidikan yang lebih luas dan personalisasi pembelajaran.
2. Kesehatan: Dalam bidang kesehatan, ilmu komputer digunakan untuk memproses data medis, mendiagnosis penyakit, dan mengembangkan perangkat medis pintar. Contohnya termasuk penggunaan kecerdasan buatan untuk analisis gambar radiologi dan simulasi medis.

3. Industri dan Manufaktur: Otomasi industri, robotika, dan Internet of Things (IoT) adalah hasil langsung dari kemajuan dalam ilmu komputer. Teknologi ini meningkatkan efisiensi, mengurangi biaya, dan meningkatkan keselamatan di tempat kerja.
4. Keuangan dan Perbankan: Sistem perbankan modern bergantung pada ilmu komputer untuk pengelolaan data nasabah, analisis risiko, dan keamanan transaksi. Algoritma pembelajaran mesin digunakan untuk mendeteksi penipuan dan memberikan rekomendasi investasi.
5. Transportasi: Ilmu komputer mendukung pengembangan kendaraan otonom, sistem navigasi, dan manajemen lalu lintas. Aplikasi seperti Google Maps menggunakan algoritma kompleks untuk menyediakan rute tercepat berdasarkan kondisi lalu lintas secara real-time.
6. Komunikasi: Internet, media sosial, dan aplikasi komunikasi seperti WhatsApp dan Zoom adalah hasil dari kemajuan dalam ilmu komputer. Mereka memungkinkan komunikasi instan lintas benua.
7. Sains dan Penelitian: Dalam bidang penelitian ilmiah, simulasi komputer dan analisis data besar membantu ilmuwan memahami fenomena kompleks, seperti perubahan iklim, astrofisika, dan epidemiologi.
8. Keamanan dan Militer: Ilmu komputer digunakan dalam pengembangan sistem keamanan siber untuk melindungi data sensitif. Selain itu, teknologi ini digunakan untuk simulasi militer, sistem drone, dan analisis data intelijen.
9. Hiburan: Dalam dunia hiburan, ilmu komputer mendukung pengembangan animasi, efek visual, game, dan platform streaming seperti Netflix. Teknologi realitas virtual (VR) dan augmented reality (AR) juga merupakan kontribusi ilmu komputer.

10. Pertanian: Ilmu komputer membantu petani dengan teknologi seperti pertanian presisi, yang menggunakan data untuk meningkatkan hasil panen dan efisiensi penggunaan sumber daya seperti air dan pupuk.

Dengan perannya yang luas, ilmu komputer telah menjadi salah satu pilar utama dalam pembangunan ekonomi dan sosial. Inovasi dalam disiplin ini terus mendorong kemajuan di berbagai sektor, memastikan bahwa ilmu komputer tetap menjadi inti dari transformasi global.

1.2 Sejarah dan Perkembangan Ilmu Komputer

1.2.1 Sejarah Awal Komputasi

Komputasi telah menjadi bagian integral dari kehidupan manusia sejak zaman kuno. Sejarahnya dimulai dengan alat-alat sederhana untuk membantu perhitungan hingga munculnya komputer modern. Abacus, yang ditemukan sekitar 2400 SM, adalah alat bantu hitung pertama yang digunakan untuk perhitungan dasar seperti penjumlahan dan pengurangan. Alat ini digunakan secara luas di Mesir, Yunani, dan Tiongkok. (Glenn Brookshear and Brylow, 2020)

Munculnya Mesin Mekanik, Pada abad ke-17, ilmuwan seperti Blaise Pascal dan Gottfried Wilhelm Leibniz mengembangkan mesin hitung mekanik. Pascal menciptakan Pascaline (1642), alat yang dapat menjumlahkan dan mengurangi angka. Leibniz menyempurnakan konsep ini dengan menciptakan mesin yang mampu melakukan perkalian dan pembagian. Revolusi Komputasi oleh Charles Babbage. Pada abad ke-19, Charles Babbage merancang mesin analitik (Analytical Engine), yang dianggap sebagai cikal bakal komputer modern. Mesin ini menggunakan konsep pemrograman dengan kartu berlubang (punch card) yang kemudian diadopsi oleh Herman Hollerith untuk tabulasi data

sensus di AS. (Glenn Brookshear and Brylow, 2020)

Awal Komputer Elektronik. Pada tahun 1940-an, komputer elektronik pertama, seperti ENIAC (Electronic Numerical Integrator and Computer), mulai dikembangkan. ENIAC digunakan untuk perhitungan militer selama Perang Dunia II. Komputer generasi pertama ini berbasis tabung vakum, berukuran besar, dan sangat mahal. Evolusi Komputer Modern. Komputer generasi kedua menggunakan transistor (1950-an), yang lebih kecil dan hemat energi. Generasi ketiga beralih ke sirkuit terpadu (IC) pada 1960-an, diikuti dengan mikroprosesor pada 1970-an, yang membuka jalan bagi personal computer (PC). Era modern ditandai dengan komputasi awan, perangkat pintar, dan pengembangan kecerdasan buatan. (Glenn Brookshear and Brylow, 2020)

1.2.2 Perkembangan Perangkat Keras dan Perangkat Lunak

1. Perangkat Keras (*Hardware*)
 - a. Tabung Vakum: Digunakan pada komputer generasi pertama. Contohnya ENIAC dan UNIVAC I.
 - b. Transistor: Menggantikan tabung vakum, memungkinkan ukuran komputer lebih kecil.
 - c. Sirkuit Terpadu (IC): Membuat komputer lebih cepat dan ekonomis.
 - d. Mikroprosesor: Intel 4004, prosesor pertama, memperkenalkan komputer pribadi seperti Altair 8800.
 - e. Komputasi Modern: Komputer saat ini menggunakan arsitektur multi-core, GPU, dan teknologi kuantum yang sedang berkembang.
2. Perangkat Lunak (*Software*)
 - a. Pemrograman Awal: Menggunakan bahasa mesin (kode biner). Contoh bahasa pertama adalah Assembly.

- b. Bahasa Pemrograman Tingkat Tinggi: FORTRAN, COBOL, dan C dikembangkan pada tahun 1950-1970-an, mempermudah pengembangan perangkat lunak.
- c. Sistem Operasi: Awalnya berupa CLI (*Command Line Interface*), berkembang menjadi GUI (*Graphical User Interface*) seperti Windows, macOS, dan Linux.

Perangkat Lunak Modern: Fokus pada pengembangan berbasis AI, cloud computing, dan aplikasi mobile. (Tanenbaum S. and Austin, 2013)

1.2.3 Munculnya Sub-Bidang dalam Ilmu Komputer

Ilmu komputer telah berkembang menjadi disiplin yang luas dengan berbagai sub-bidang, di antaranya:

1. Algoritma dan Struktur Data: Fokus pada pengembangan metode untuk pengolahan data secara efisien. Penting dalam pengembangan perangkat lunak dan sistem operasi.
2. Sistem Operasi: Mengelola perangkat keras dan perangkat lunak. Contohnya adalah Windows, Linux, dan macOS.
3. Jaringan Komputer: Berfokus pada komunikasi data antar komputer. Teknologi seperti internet, cloud computing, dan IoT berkembang dari sub-bidang ini.
4. Kecerdasan Buatan (AI): Bertujuan menciptakan sistem yang dapat berpikir dan belajar seperti manusia. Contoh aplikasi: chatbot, mobil otonom, dan analisis data prediktif.
5. Keamanan Siber: Berfokus pada melindungi sistem komputer dari ancaman. Penting dalam era digital untuk melindungi data pribadi dan perusahaan.
6. Ilmu Data dan Pembelajaran Mesin: Mengolah data besar untuk mendapatkan wawasan yang berharga. Digunakan

dalam analisis bisnis, prediksi cuaca, dan personalisasi layanan.

7. Multimedia dan Grafik Komputer: Berhubungan dengan pembuatan gambar, video, dan animasi. Penting dalam industri kreatif seperti film dan game.
8. Pengembangan Perangkat Lunak: Meliputi desain, implementasi, dan pemeliharaan perangkat lunak. Sub-bidang ini mencakup metodologi seperti Agile dan DevOps.
9. Komputasi Awan: Menyediakan layanan komputasi melalui internet. Mengubah cara perusahaan mengelola data dan aplikasi.
10. Robotika: Kombinasi ilmu komputer dan teknik untuk menciptakan mesin cerdas. Aplikasinya mencakup industri, kedokteran, dan eksplorasi luar angkasa.

Pengembangan sub-bidang ini terus berlangsung seiring dengan kebutuhan masyarakat dan kemajuan teknologi, menjadikan ilmu komputer sebagai salah satu bidang studi yang paling dinamis dan inovatif.

1.3 Hubungan Ilmu Komputer dengan Teknologi Informasi

Hubungan antara ilmu komputer, teknologi informasi, dan sistem informasi serta bagaimana ilmu komputer berperan dalam pengembangan teknologi modern, yaitu: (Kadir and Triwahyuni, 2013)

1. Ilmu Komputer

Ilmu komputer adalah bidang studi yang berfokus pada teori, desain, dan pengembangan perangkat lunak serta perangkat keras. Bidang ini mencakup aspek mendalam seperti algoritma, struktur data, kecerdasan buatan, komputasi awan, dan sistem operasi. Ciri utama ilmu

komputer:

- a. Berorientasi pada dasar-dasar teoretis.
- b. Memanfaatkan matematika sebagai fondasi, seperti logika, statistik, dan probabilitas.
- c. Menyediakan dasar untuk membangun teknologi informasi dan sistem informasi.

Contoh penerapan ilmu komputer meliputi:

- a. Pengembangan algoritma pencarian (Google Search).
- b. Pembelajaran mesin untuk analisis data.
- c. Pengembangan perangkat lunak seperti sistem operasi Windows atau Linux.

2. Teknologi Informasi (TI)

Teknologi informasi adalah aplikasi teknologi untuk mengelola dan mendukung kebutuhan informasi. TI melibatkan implementasi perangkat keras dan perangkat lunak untuk menyimpan, memproses, dan mengakses data. Fokusnya adalah pada bagaimana teknologi diterapkan untuk memecahkan masalah praktis dalam organisasi atau masyarakat. Karakteristik teknologi informasi:

- a. Berfokus pada solusi praktis berbasis teknologi.
- b. Memanfaatkan produk yang dihasilkan oleh ilmu komputer, seperti perangkat keras, perangkat lunak, dan jaringan.
- c. Berorientasi pada efisiensi dan efektivitas penyampaian informasi.

Contoh penerapan TI meliputi:

- a. Sistem manajemen data perusahaan (ERP).
- b. Infrastruktur jaringan komunikasi.
- c. Keamanan data pada aplikasi perbankan.

3. Sistem Informasi

Sistem informasi adalah bidang yang lebih luas yang menggabungkan teknologi, manusia, dan proses untuk mendukung pengambilan keputusan di organisasi. Fokus utama sistem informasi adalah mengelola data dan informasi yang relevan bagi organisasi untuk mendukung operasional dan strategi bisnis. Karakteristik sistem informasi: (Kristanto, 2018)

- a. Menitikberatkan pada interaksi antara teknologi dan manusia.
- b. Mengintegrasikan ilmu komputer dan teknologi informasi untuk mendukung proses bisnis.
- c. Berorientasi pada pengelolaan informasi untuk tujuan organisasi.

Contoh penerapan sistem informasi meliputi:

- a. Sistem manajemen pelanggan (CRM).
- b. Sistem pendukung keputusan (DSS).
- c. Aplikasi sistem informasi geografis (GIS).

Keterkaitan antara ketiganya yaitu Ilmu Komputer memberikan fondasi teoritis dan teknologi inti (seperti algoritma dan perangkat lunak). Teknologi Informasi menerapkan teori tersebut untuk menciptakan solusi teknologi. Sistem Informasi mengintegrasikan teknologi untuk mendukung kebutuhan organisasi atau masyarakat. Sebagai contoh, pengembangan aplikasi berbasis web memanfaatkan algoritma (ilmu komputer), framework dan platform teknologi (teknologi informasi), serta penyesuaian dengan kebutuhan pengguna (sistem informasi).

1.4 Peran dan Dampak Ilmu Komputer di Masyarakat

1.4.1 Dampak Sosial, Ekonomi dan Budaya dari Teknologi Komputer

Teknologi komputer telah membawa dampak besar dalam berbagai aspek kehidupan manusia. Namun, dampak tersebut memiliki sisi positif dan negatif yang harus dipertimbangkan.

1. Dampak Sosial

- a. Peningkatan Konektivitas: Teknologi komputer telah menciptakan dunia yang lebih terhubung. Media sosial, email, dan aplikasi pesan instan memungkinkan komunikasi instan tanpa batas geografis.
- b. Perubahan Pola Hidup: Kehadiran komputer dan internet memengaruhi cara orang bekerja, belajar, dan berinteraksi. Banyak pekerjaan yang kini dapat dilakukan secara jarak jauh, dan pembelajaran daring semakin populer.
- c. Ketergantungan Teknologi: Di sisi lain, meningkatnya ketergantungan pada teknologi dapat menyebabkan masalah seperti isolasi sosial dan gangguan kesehatan mental akibat penggunaan perangkat secara berlebihan.

2. Dampak Ekonomi

- a. Peningkatan Efisiensi dan Produktivitas: Teknologi komputer telah membantu perusahaan meningkatkan produktivitas melalui automasi, analisis data, dan komunikasi yang lebih efisien.
- b. Penciptaan Lapangan Kerja Baru: Bidang seperti pengembangan perangkat lunak, analitik data, dan keamanan siber menciptakan peluang kerja baru.
- c. Disrupsi di Pasar Kerja: Automasi dan AI juga menyebabkan hilangnya beberapa jenis pekerjaan

tradisional, sehingga memerlukan tenaga kerja untuk beradaptasi dan belajar keterampilan baru.

3. Dampak Budaya

- a. Globalisasi Budaya: Teknologi komputer memungkinkan penyebaran budaya secara cepat. Musik, film, dan tradisi dari berbagai negara kini lebih mudah diakses.
- b. Ancaman terhadap Identitas Lokal: Dominasi budaya global dapat mengancam keberlangsungan budaya lokal jika tidak diimbangi dengan pelestarian tradisi dan nilai-nilai lokal.

Perubahan Gaya Hidup: Kehadiran teknologi komputer mengubah cara manusia mengonsumsi informasi, bekerja, dan menghabiskan waktu luang, seperti bermain video game atau streaming konten daring.

1.4.2 Etika dalam Ilmu Komputer

Seiring dengan perkembangan teknologi komputer, muncul tantangan etis yang harus dihadapi oleh para profesional dan pengguna teknologi. Etika dalam ilmu komputer berperan penting dalam memastikan bahwa teknologi digunakan secara bertanggung jawab dan memberikan manfaat bagi masyarakat. Beberapa isu etika yang sering dibahas meliputi: (Reynold, 2019)

1. Privasi dan Keamanan Data: Dalam dunia yang semakin terhubung, perlindungan data pribadi menjadi prioritas utama. Etika dalam ilmu komputer mengharuskan pengembang teknologi untuk melindungi privasi pengguna dan mencegah penyalahgunaan data.
2. Keadilan dan Aksesibilitas: Teknologi komputer harus dirancang untuk inklusivitas, memastikan bahwa semua orang memiliki akses yang sama terhadap teknologi tanpa diskriminasi.

3. tanggung Jawab Sosial: Pengembang teknologi memiliki tanggung jawab untuk mempertimbangkan dampak sosial dari inovasi mereka. Misalnya, pengembangan AI harus dilakukan dengan mempertimbangkan dampaknya terhadap pekerjaan manusia.
4. Penggunaan AI dan Automasi: Etika dalam penggunaan AI mencakup transparansi dalam algoritma, menghindari bias, dan memastikan bahwa teknologi ini tidak disalahgunakan untuk tujuan yang merugikan.
5. Hak Kekayaan Intelektual: Hak cipta dan paten merupakan isu penting dalam ilmu komputer, terutama dalam pengembangan perangkat lunak. Pelanggaran hak kekayaan intelektual dapat menimbulkan kerugian besar bagi penciptanya.

Peran dan dampak ilmu komputer dalam masyarakat sangat signifikan, terutama di era Revolusi Industri 4.0. Teknologi komputer memberikan manfaat besar dalam berbagai aspek kehidupan, tetapi juga menimbulkan tantangan yang perlu diatasi melalui pendekatan etis dan inovatif. Dengan memahami peran, dampak, dan etika ilmu komputer, kita dapat memanfaatkan teknologi ini untuk menciptakan dunia yang lebih baik.

DAFTAR PUSTAKA

- GLENN BROOKSHEAR, J. and BRYLOW, D. (2020) *Computer Science An Overview THIRTEENTH EDITION*.
- KADIR, A. and TRIWAHYUNI, T.C. (2013) *Pengantar Teknologi Informasi*. Revisi. Yogyakarta: ANDI.
- KRISTANTO, A. (2018) *Perancangan Sistem Informasi dan Aplikasinya*. 1st ed. Yogyakarta: Gava Media.
- REYNOLD, G.S. (2019) *Ethics in Information Technology*. 6th ed. Boston: Cengage Learning.
- SEDGEWICK, R. and WAYNE, K. (2017) *Computer Science: An Interdisciplinary Approach*. Princeton: Princeton Education.
- TANENBAUM S., A. and AUSTIN, T. (2013) *STRUCTURED COMPUTER ORGANIZATION SIXTH EDITION*. 6th ed. USA: Pearson Education.

BAB 2

KOMPONEN-KOMPONEN KOMPUTER

Oleh Rakhmi Khalida

2.1 Macam Jenis *Hardware* Komputer

Hardware atau perangkat keras merupakan komponen dalam sistem komputer yang secara fisik dapat dilihat dan diraba keberadaannya. Komponen *hardware* memiliki peran yang sangat penting sehingga apabila tidak terdapat komponen ini pada sistem komputer, maka tidak akan ada yang dioperasikan.

Berdasarkan fungsinya, perangkat keras terdiri dari perangkat *input* dan *output*, perangkat penyimpanan, perangkat pemroses, perangkat tambahan (aksesoris).

A. Perangkat *Input* (unit masukan)

Unit masukan merupakan perangkat yang berfungsi untuk memasukkan data ke dalam komputer kemudian data tersebut nantinya diolah dalam suatu proses. Perangkat ini memungkinkan pengguna berinteraksi dengan komputer dengan cara memberikan perintah, mengirim informasi, atau memasukkan data yang akan diolah oleh sistem komputer. Unit masukan menjadi kunci untuk mengubah kebutuhan, ide, dan perintah pengguna menjadi tindakan yang dijalankan oleh komputer, menjadikannya komponen vital bagi seluruh sistem. Berikut adalah beberapa perangkat masukan yang umum digunakan:

1. Keyboard

Perangkat utama untuk memasukkan data berupa

teks dan memberikan perintah ke komputer. Terdiri dari tombol alfanumerik, tombol fungsi, dan tombol navigasi.

2. Mouse

Alat yang digunakan untuk menggerakkan kursor di layar dan memilih atau menjalankan perintah. Terdapat jenis mouse mekanik, optik, dan laser, serta versi nirkabel (*wireless*).

3. Touchpad

Biasanya terdapat pada laptop sebagai pengganti mouse. Menggunakan gerakan jari untuk mengontrol kursor, klik, atau fungsi lain melalui gestur.

4. Scanner

Perangkat yang digunakan untuk mengonversi dokumen fisik atau gambar menjadi data digital. Ada beberapa jenis scanner, seperti flatbed scanner dan handheld scanner.

5. Microphone

Digunakan untuk memasukkan data suara atau merekam audio ke komputer. Umumnya digunakan dalam aplikasi komunikasi (misalnya Zoom, Skype) atau untuk merekam suara.

6. Webcam

Perangkat untuk menangkap gambar atau video, biasanya digunakan untuk video call atau konferensi. Umumnya terdapat pada laptop, tetapi juga tersedia sebagai perangkat eksternal untuk desktop.

7. Barcode Scanner

Perangkat yang membaca kode batang (barcode) dan mengubahnya menjadi data digital. Biasa digunakan di kasir atau gudang untuk inventarisasi barang.

8. Graphic Tablet (Drawing Tablet)

Digunakan untuk menggambar atau membuat desain

langsung di komputer. Biasanya dilengkapi dengan stylus pen untuk menggambar dengan lebih presisi.

9. Touchscreen

Layar yang dapat mendeteksi sentuhan langsung dari jari atau stylus. Umumnya terdapat pada perangkat seperti tablet, smartphone, dan monitor dengan layar sentuh.

10. Light Pen

Perangkat yang digunakan dengan layar CRT untuk menggambar atau menandai bagian layar. Kini jarang digunakan karena layar sentuh dan graphic tablet lebih umum.

11. Sensor atau Perangkat IoT (Internet of Things)

Sensor yang mendeteksi data fisik (misalnya suhu, kelembapan, gerakan) dan mengirimkannya ke komputer. Umumnya digunakan pada sistem otomatisasi dan smart home.

Perangkat masukan ini memungkinkan pengguna berinteraksi dengan komputer melalui berbagai cara, seperti mengetik, menunjuk, merekam suara, atau bahkan mendeteksi kondisi lingkungan.

B. Perangkat *Output* (unit keluaran)

Unit keluaran (*output*) pada komponen komputer adalah perangkat yang menampilkan hasil dari pemrosesan data oleh komputer, baik dalam bentuk visual, audio, atau fisik. Unit keluaran ini memungkinkan komputer memberikan hasil dari proses data dalam bentuk yang bisa dilihat, didengar, atau dirasakan oleh pengguna sesuai kebutuhan dan lingkungan kerja. Berikut ini beberapa perangkat keluaran yang umum digunakan:

1. Monitor

Perangkat utama untuk menampilkan output visual dari komputer, seperti gambar, teks, dan video.

Terdapat berbagai jenis monitor seperti LCD, LED, dan OLED, dengan berbagai resolusi dan ukuran.

2. Printer

Perangkat untuk mencetak teks, gambar, atau dokumen dari komputer ke media fisik, seperti kertas. Jenis printer yang umum mencakup inkjet, laser, dan dot matrix dan thermal printer.

3. Speaker

Perangkat yang menghasilkan output audio dari komputer. Digunakan untuk mendengarkan musik, audio dari video, atau suara lainnya dari aplikasi computer.

4. Headphones/Earphones

Perangkat audio pribadi yang menghasilkan output suara langsung ke telinga pengguna. Dapat berupa versi dengan kabel atau nirkabel (Bluetooth).

5. Projector

Alat untuk menampilkan output visual pada permukaan besar, seperti dinding atau layar khusus. Biasa digunakan untuk presentasi, menonton film, atau pertemuan skala besar.

6. Plotter

Perangkat cetak khusus yang menghasilkan output berupa gambar teknis atau grafik dalam ukuran besar. Umumnya digunakan dalam desain arsitektur, teknik, dan CAD (Computer-Aided Design).

7. LCD Display atau Digital Signage

Layar digital yang menampilkan informasi atau iklan secara terus-menerus. Biasa digunakan di tempat umum, seperti pusat perbelanjaan atau terminal.

8. VR Headset (Virtual Reality Headset)

Perangkat untuk menampilkan lingkungan virtual tiga dimensi yang interaktif. Digunakan dalam permainan,

pelatihan simulasi, dan aplikasi imersif lainnya.

9. Braille Display

Perangkat khusus untuk tunanetra yang mengubah teks digital menjadi huruf Braille. Menampilkan karakter dalam bentuk Braille yang bisa dirasakan dengan jari.

10. Sound Card Output

Bagian dari sistem komputer yang mengubah data digital menjadi sinyal audio. Digunakan sebagai output suara pada speaker atau headphone.

11. Robotik dan Aktuator (Perangkat Keluaran Fisik)

Perangkat yang menghasilkan output fisik, seperti gerakan atau reaksi mekanis. Umumnya digunakan dalam sistem otomatisasi atau robotika yang dikontrol oleh computer.

12. Haptic Devices (Perangkat Sentuhan)

Perangkat yang memberikan umpan balik berupa getaran atau sentuhan, biasanya dalam aplikasi gaming atau simulasi. Contoh: joystick atau controller game dengan fitur getar.

C. Perangkat pemroses

Perangkat pemroses pada komponen komputer adalah perangkat yang berfungsi mengolah data dan menjalankan instruksi dari perangkat lunak. Komponen pemrosesan ini bekerja bersama untuk memastikan komputer mampu mengolah data dengan efisien, menjalankan aplikasi, dan merespon perintah pengguna dengan cepat. Berikut adalah beberapa komponen utama yang termasuk dalam perangkat pemroses:

1. CPU (*Central Processing Unit*)

Dikenal sebagai "otak" komputer, CPU adalah komponen utama yang bertugas mengeksekusi perintah dan mengolah data. CPU memiliki beberapa

komponen internal penting, seperti **ALU (Arithmetic Logic Unit)** untuk melakukan operasi matematika dan logika, serta **Control Unit (CU)** yang mengontrol aliran data antara CPU dan komponen lain. CPU juga memiliki **core** (inti pemrosesan) yang dapat berjumlah satu atau lebih, dengan inti yang lebih banyak umumnya meningkatkan kemampuan multitasking.

2. GPU (*Graphics Processing Unit*)

GPU adalah prosesor khusus yang dirancang untuk menangani pemrosesan grafis, seperti rendering gambar, video, dan 3D. Dapat berupa **GPU terintegrasi** (bagian dari CPU, seperti pada prosesor Intel dan AMD tertentu) atau **GPU diskrit** (kartu grafis terpisah seperti Nvidia GeForce atau AMD Radeon). GPU juga berperan dalam pemrosesan paralel untuk tugas yang memerlukan banyak operasi matematis sekaligus, seperti machine learning atau aplikasi sains.

3. RAM (*Random Access Memory*)

RAM adalah memori sementara yang digunakan untuk menyimpan data dan instruksi sementara selama komputer berjalan. Bertindak sebagai penyimpanan sementara yang memungkinkan akses cepat oleh CPU, sehingga mempercepat kinerja saat multitasking.

4. Cache Memory

Cache adalah memori kecil yang sangat cepat dan terintegrasi dalam CPU atau di dekatnya. Mempercepat akses data dan instruksi yang sering digunakan dengan menyimpannya sementara untuk pemrosesan yang lebih cepat dibandingkan RAM.

5. **Motherboard (Papan Induk)**
Motherboard adalah komponen yang menghubungkan semua perangkat pemroses dan perangkat keras lainnya dalam satu sistem. Meskipun bukan pemroses langsung, motherboard mengatur aliran data antara CPU, GPU, RAM, dan perangkat lain sehingga mendukung proses komputasi.
6. **Chipset**
Chipset adalah sekumpulan chip pada motherboard yang mengontrol komunikasi antara CPU, RAM, penyimpanan, dan perangkat lainnya. Umumnya, chipset terdiri dari dua bagian, yaitu Northbridge (untuk CPU dan RAM) dan Southbridge (untuk I/O dan penyimpanan).
7. ***Digital Signal Processor (DSP)***
DSP adalah prosesor khusus yang digunakan untuk mengolah sinyal digital, seperti audio dan video, dengan cepat. Umumnya ditemukan dalam perangkat yang memerlukan pemrosesan suara atau gambar secara *real-time*, seperti smartphone, perangkat audio, dan kamera.
8. **TPU (Tensor Processing Unit)**
TPU adalah jenis prosesor yang dikembangkan untuk mempercepat proses machine learning dan kecerdasan buatan. Umumnya digunakan pada server data center untuk aplikasi AI dan deep learning, terutama dalam lingkungan Google Cloud.
9. ***ASIC (Application-Specific Integrated Circuit)***
ASIC adalah sirkuit terintegrasi yang dirancang khusus untuk melakukan satu jenis tugas tertentu dengan sangat efisien. Contoh penggunaannya adalah pada perangkat untuk penambangan *cryptocurrency* dan router jaringan yang memiliki fungsi spesifik.

10. FPGA (*Field-Programmable Gate Array*)

FPGA adalah sirkuit yang bisa diprogram ulang untuk menjalankan tugas tertentu, memungkinkan fleksibilitas dalam melakukan pemrosesan khusus. Digunakan dalam aplikasi yang memerlukan pemrosesan cepat dan dapat disesuaikan, seperti penelitian ilmiah, jaringan, dan perangkat elektronik khusus.

11. NPU (*Neural Processing Unit*)

NPU adalah prosesor yang dirancang khusus untuk mempercepat tugas AI yang terinspirasi oleh jaringan saraf (*neural networks*). Biasanya ditemukan dalam perangkat modern seperti smartphone untuk mempercepat aplikasi kecerdasan buatan, termasuk pengenalan wajah dan pengenalan gambar.

D. Perangkat Penyimpanan

Perangkat penyimpanan pada komponen komputer adalah perangkat yang berfungsi untuk menyimpan data secara permanen atau sementara. Perangkat ini penting untuk menyimpan sistem operasi, program, dan data pengguna. Perangkat penyimpanan ini membantu komputer menyimpan dan mengelola data dengan efektif, baik dalam jumlah kecil maupun besar, dan memungkinkan pengguna untuk mengakses dan mengamankan data sesuai kebutuhan. Berikut adalah jenis-jenis perangkat penyimpanan pada komputer:

1. *Hard Disk Drive* (HDD)

Penyimpanan mekanis yang menggunakan piringan magnetis untuk menyimpan data. Kapasitas HDD umumnya besar dan tersedia dalam berbagai ukuran, seperti 500GB, 1TB, hingga 10TB atau lebih. Cocok untuk penyimpanan data besar dengan harga per GB yang relatif murah, namun kecepatan baca/tulisnya

lebih lambat dibandingkan SSD.

2. *Solid State Drive (SSD)*

Penyimpanan berbasis flash memory yang lebih cepat daripada HDD. Tidak memiliki bagian bergerak, sehingga lebih tahan terhadap guncangan dan memiliki waktu akses yang jauh lebih cepat. Umumnya digunakan untuk sistem operasi dan aplikasi yang memerlukan akses cepat, meski kapasitasnya biasanya lebih mahal per GB dibandingkan HDD.

3. *NVMe SSD (Non-Volatile Memory Express SSD)*

Jenis SSD yang menggunakan interface NVMe melalui jalur PCIe, membuatnya jauh lebih cepat dibandingkan dengan SSD SATA. Cocok untuk pengguna yang membutuhkan kecepatan tinggi, seperti dalam pengeditan video, gaming, dan aplikasi yang membutuhkan kinerja tinggi.

4. *Hybrid Drive (SSHD - Solid State Hybrid Drive)*

Kombinasi antara HDD dan SSD dalam satu perangkat, menggabungkan kapasitas besar dari HDD dengan kecepatan akses dari SSD. Data yang sering diakses disimpan pada bagian SSD, sementara data lainnya disimpan pada bagian HDD. Memberikan performa lebih baik dibandingkan HDD biasa namun lebih terjangkau daripada SSD.

5. *Optical Drive (CD/DVD/Blu-ray)*

Digunakan untuk membaca dan menulis data ke cakram optik seperti CD, DVD, dan Blu-ray. Umumnya digunakan untuk media penyimpanan permanen atau cadangan data yang jarang diakses. Optik drive kini kurang umum dalam komputer modern, tetapi masih digunakan pada beberapa laptop dan desktop.

6. *Flash Drive (USB Drive)*

Penyimpanan portabel yang berbasis flash memory

dan dapat dicolokkan ke port USB. Umumnya digunakan untuk memindahkan data antar komputer atau sebagai penyimpanan sementara. Kapasitasnya bervariasi, mulai dari 4GB hingga 1TB, dengan kecepatan akses yang cukup baik.

7. SD Card dan MicroSD

Penyimpanan berbasis flash memory yang sering digunakan pada perangkat portabel seperti smartphone, kamera digital, dan tablet. Bisa digunakan pada komputer melalui card reader atau port SD yang tersedia.

8. External Hard Drive

HDD atau SSD yang dikemas dalam casing portabel dan dihubungkan ke komputer melalui port USB atau Thunderbolt. Digunakan untuk penyimpanan eksternal atau sebagai cadangan (backup) data.

9. Cloud Storage

Penyimpanan berbasis internet yang di-hosting oleh penyedia layanan seperti Google Drive, Dropbox, atau OneDrive. Data disimpan di server jarak jauh dan dapat diakses dari perangkat apa pun yang terhubung ke internet. Berguna untuk cadangan data, berbagi file, dan kolaborasi online.

E. Perangkat tambahan

Perangkat tambahan (peripheral) pada komponen komputer adalah perangkat eksternal yang menambah fungsi atau fitur pada komputer. Perangkat ini tidak diperlukan untuk menjalankan komputer, tetapi dapat meningkatkan kegunaan dan fleksibilitasnya. Perangkat tambahan ini menambah fungsionalitas komputer, meningkatkan kenyamanan, produktivitas, dan kemampuan multimedia. Setiap perangkat memiliki kegunaan spesifik sesuai kebutuhan pengguna.

Berikut adalah beberapa perangkat tambahan yang umum digunakan:

1. *External Hard Drive*

Penyimpanan eksternal yang dapat digunakan untuk menyimpan atau mencadangkan (backup) data. Terhubung melalui USB atau Thunderbolt, dan dapat berupa HDD atau SSD.

2. *Card Reader*

Perangkat yang digunakan untuk membaca kartu memori, seperti SD card atau microSD. Sering digunakan oleh fotografer dan pengguna perangkat portabel untuk mentransfer data dari kartu memori ke komputer.

3. *Game Controller*

Perangkat yang digunakan untuk bermain game, seperti joystick, gamepad, atau steering wheel. Menyediakan kontrol yang lebih intuitif dan menyenangkan dalam bermain game dibandingkan dengan keyboard dan mouse.

4. *Docking Station*

Perangkat yang memungkinkan laptop terhubung dengan beberapa perangkat sekaligus, seperti monitor, keyboard, dan mouse. Memudahkan pengguna laptop yang ingin mengubahnya menjadi workstation yang lebih besar di meja kerja.

5. Modem dan Router

Modem menghubungkan komputer ke jaringan internet, sementara router memungkinkan banyak perangkat terhubung ke internet atau ke satu jaringan lokal (LAN). Beberapa perangkat menyatukan fungsi modem dan router dalam satu alat.

6. *Uninterruptible Power Supply (UPS)*

Perangkat yang menyediakan daya cadangan saat

terjadi pemadaman listrik. Melindungi komputer dari kehilangan data akibat mati listrik mendadak dan menjaga kestabilan daya.

7. *External Optical Drive*

Digunakan untuk membaca dan menulis CD, DVD, atau Blu-ray pada komputer yang tidak memiliki drive optik bawaan. Berguna untuk keperluan instalasi software lama atau memutar media fisik.

8. Hubs dan USB Extender

Alat yang memperbanyak jumlah port USB yang tersedia pada komputer. Berguna untuk pengguna yang memiliki banyak perangkat USB namun keterbatasan jumlah port di komputer.

2.2 Macam Jenis *Software* Komputer

Software atau perangkat lunak memungkinkan perangkat keras menjalankan fungsi yang berguna, membuat pengguna berinteraksi dengan sistem komputer secara intuitif. Perangkat lunak komputer terus berkembang melalui pembaruan yang memperbaiki bug, meningkatkan performa, dan menambah fitur baru, sehingga komputer tetap relevan dan efisien. Perangkat lunak adalah jiwa dari sistem komputer yang membuat perangkat keras berfungsi sesuai kebutuhan pengguna, membuat operasi yang efisien, dan menjaga sistem tetap aman dan terbaru. Tanpa perangkat lunak, komputer tidak dapat berfungsi sebagai alat yang berguna bagi manusia. Berikut adalah beberapa perangkat lunak yang wajib ada pada sebuah komputer untuk menjalankan fungsinya dengan baik:

1. Sistem Operasi (*Operating System*)

Sistem operasi adalah perangkat lunak paling penting yang memungkinkan komputer untuk menjalankan aplikasi lain dan mengelola perangkat keras. Contoh

- sistem operasi yaitu Windows, macOS, atau Linux.
2. Driver Perangkat Keras
Driver adalah perangkat lunak yang memungkinkan sistem operasi berkomunikasi dengan perangkat keras seperti kartu grafis, printer, atau perangkat *input/output* lainnya. Contoh driver perangkat keras diantaranya driver graphics card, printer, dan perangkat lain.
 3. Browser Web
Browser web sangat penting untuk mengakses internet dan mencari informasi online. Contoh browser web yaitu Google Chrome, Mozilla Firefox, Microsoft Edge.
 4. Perangkat Lunak Keamanan
Jika pengguna melakukan pemrograman, IDE atau text editor khusus diperlukan untuk menulis dan menjalankan kode.

Perangkat lunak ini membantu komputer menjalankan berbagai tugas penting, mulai dari produktivitas hingga keamanan dan hiburan.

2.3 Komponen Komputer pada Tujuan Khusus

Saat komputer digunakan untuk sebuah tujuan khusus, komputer memiliki minimal spesifikasi komponen. Komponen komputer dengan tujuan khusus adalah komponen perangkat keras yang dirancang untuk menjalankan fungsi atau tugas tertentu yang tidak bisa atau kurang efisien dilakukan oleh komponen umum. Biasanya, komponen ini memiliki arsitektur atau kemampuan khusus yang mengoptimalkan performa untuk jenis pekerjaan spesifik. Berikut adalah beberapa contoh spesifikasi minimal untuk berbagai tujuan khusus:

- a. Gaming
 - a. CPU : Minimal quad-core (4 inti) atau lebih tinggi, seperti Intel Core i5 atau AMD Ryzen 5.

- b. GPU : Dedicated GPU dengan memori minimal 4GB, seperti NVIDIA GTX 1650 atau AMD RX 570.
- c. RAM**: 8GB (16GB lebih direkomendasikan untuk performa lebih baik).
- d. Storage**: SSD 256GB (lebih baik untuk loading game), dengan HDD tambahan untuk penyimpanan data.
- e. Power Supply**: PSU dengan watt yang cukup untuk mengakomodasi GPU dan komponen lainnya (biasanya 500W ke atas)

2. Desain Grafis dan Multimedia

- a. CPU: Minimal quad-core atau lebih tinggi dengan clock speed tinggi untuk rendering cepat, seperti Intel Core i7 atau AMD Ryzen 7.
- b. GPU: GPU dedicated dengan memori minimal 6GB, seperti NVIDIA GTX 1660 atau AMD RX 580.
- c. RAM: Minimal 16GB, untuk menangani aplikasi desain yang memakan banyak memori.
- d. Storage: SSD 512GB untuk aplikasi, dengan tambahan HDD untuk penyimpanan file media besar.
- e. Monitor: Resolusi tinggi (2K atau 4K) dan akurasi warna yang baik, seperti panel IPS.

3. Machine Learning dan AI

- a. CPU: Minimal 8-core, seperti Intel Core i9 atau AMD Ryzen 9, atau prosesor server seperti Intel Xeon.
- b. GPU: GPU yang mendukung CUDA (untuk NVIDIA) dengan memori besar, seperti NVIDIA RTX 3060 ke atas atau Tensor Processing Unit (TPU).
- c. RAM: Minimal 32GB, namun 64GB atau lebih direkomendasikan untuk proyek besar.
- d. Storage: SSD minimal 1TB untuk kecepatan baca/tulis

- data yang tinggi, dengan penyimpanan tambahan untuk dataset besar.
- e. Power Supply: PSU dengan watt tinggi sesuai kebutuhan GPU dan CPU, biasanya 750W atau lebih.
4. Server atau Pusat Data
- a. CPU: Prosesor server multi-core seperti Intel Xeon atau AMD EPYC, minimal 8-core.
 - b. RAM: Minimal 32GB, namun 64GB atau lebih sering digunakan untuk menangani beberapa proses sekaligus.
 - c. Storage: Sistem RAID untuk redundansi data, menggunakan SSD untuk kecepatan atau HDD untuk kapasitas besar.
 - d. NIC (Network Interface Card): Kartu jaringan gigabit atau 10GbE untuk koneksi cepat.
 - e. Power Supply: PSU dengan watt tinggi dan cadangan untuk keandalan.
5. *Virtual Reality (VR) Development* dan Simulasi
- a. CPU: Minimal 6-core, seperti Intel Core i7 atau AMD Ryzen 7.
 - b. GPU: Dedicated GPU kelas atas dengan minimal 6GB VRAM, seperti NVIDIA RTX 2070 atau AMD RX 5700 XT.
 - c. RAM: Minimal 16GB, namun 32GB lebih optimal untuk mengelola simulasi yang kompleks.
 - d. Storage: SSD minimal 512GB untuk loading yang cepat.
 - e. *VR Headset Compatibility*: Sistem yang mendukung headset VR dan aksesoris VR lainnya.
6. Video Editing dan Rendering
- a. CPU: Minimal 8-core dengan clock speed tinggi,

seperti Intel Core i9 atau AMD Ryzen 9.

- b. GPU: GPU dedicated dengan memori minimal 8GB, seperti NVIDIA RTX 3060 atau AMD RX 6700
- c. RAM**: Minimal 32GB, terutama untuk pengeditan video beresolusi tinggi (4K dan lebih).
- d. Storage: SSD 1TB untuk kecepatan, dengan tambahan HDD untuk penyimpanan file video besar.
- e. Monitor: Resolusi tinggi dengan akurasi warna baik, seperti panel IPS 4K.

DAFTAR PUSTAKA

Ikbal, Iskandar (2010) Pengantar Ilmu Komputer.

Ikbal, Iskandar (2011) Bab Iii Komputer Dan Bagian-bagiannya.
[Teaching Resource]

Wijaya Indra, Sefriani Rini, Menrisal (2020) Pemeliharaan
Perangkat Komputer. Surabaya: Scopindo Media
Pustaka

Kemendikbud, 2013. Perakitan Komputer. Jakarta: Balai
Pustaka.

BAB 3

PENGENALAN SISTEM OPERASI

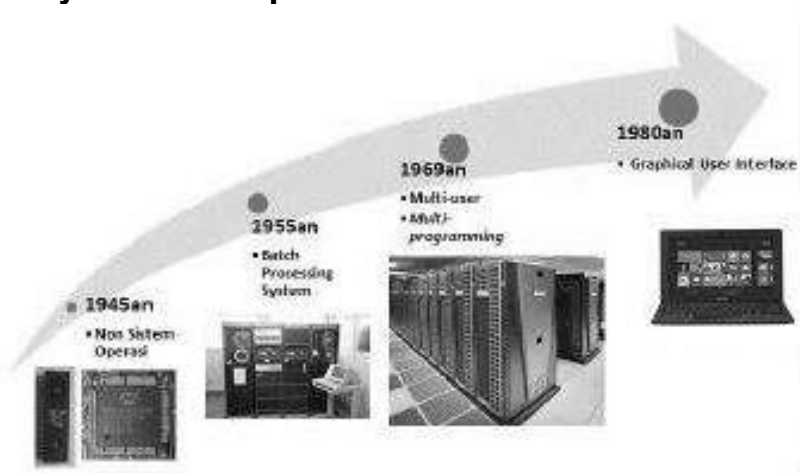
Oleh Aida Fitriyani

3.1 Pengenalan Sistem Operasi

3.1.1 Definisi Sistem Operasi

Sistem operasi merupakan sebuah penghubung antara pengguna dari komputer dengan perangkat keras komputer. Sebelum ada sistem operasi, orang hanya menggunakan komputer dengan menggunakan sinyal analog dan sinyal digital.

3.1.3 Sejarah Sistem Operasi



Gambar 3.1. Sejarah Sistem Operasi

Menurut Tanenbaum, sistem operasi mengalami perkembangan yang sangat pesat, yang dapat dibagi ke dalam empat generasi:

1. Generasi Pertama (1945-1955)

Generasi pertama merupakan awal perkembangan sistem komputasi elektronik sebagai pengganti sistem komputasi mekanik, hal itu disebabkan kecepatan manusia untuk menghitung terbatas dan manusia sangat mudah untuk membuat kecerobohan, kekeliruan bahkan kesalahan. Pada generasi ini belum ada sistem operasi, maka sistem komputer diberi instruksi yang harus dikerjakan secara langsung.

2. Generasi Kedua (1955-1965)

Generasi kedua memperkenalkan Batch Processing System, yaitu Job yang dikerjakan dalam satu rangkaian, lalu dieksekusi secara berurutan. Pada generasi ini sistem komputer belum dilengkapi sistem operasi, tetapi beberapa fungsi sistem operasi telah ada, contohnya fungsi sistem operasi ialah FMS dan IBSYS.

3. Generasi Ketiga (1965-1980)

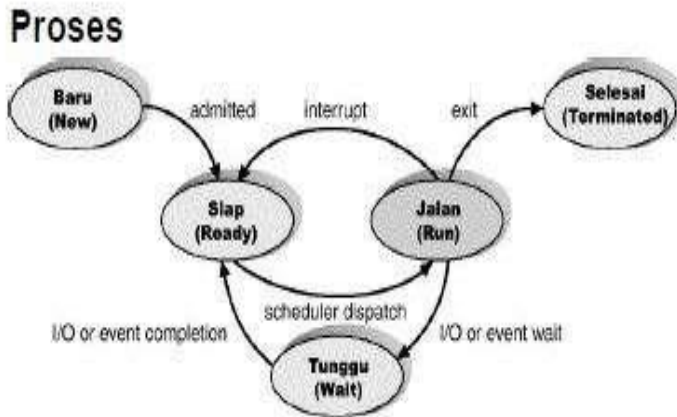
Pada generasi ini perkembangan sistem operasi dikembangkan untuk melayani banyak pemakai sekaligus, di mana para pemakai interaktif berkomunikasi lewat terminal secara online ke komputer, maka sistem operasi menjadi multi-user (digunakan banyak pengguna sekaligus) dan multi-programming (melayani banyak program sekaligus).

4. Generasi Keempat (Pasca 1980an)

Di generasi keempat, sistem operasi sudah dipergunakan untuk jaringan komputer dimana pemakai menyadari keberadaan komputer-komputer yang saling terhubung satu sama lainnya.

3.2 Proses

3.2.1 Definisi Proses



Gambar 3.2. Sejarah Sistem Operasi

Secara informal, proses adalah program dalam eksekusi. Suatu proses adalah lebih dari kode program, dimana kadang kala dikenal sebagai bagian tulisan. Proses juga termasuk aktivitas yang sedang terjadi, sebagaimana digambarkan oleh nilai pada program counter dan isi dari daftar prosesor/*processor's register*.

3.2.2 Keadaan Proses

Sebagaimana proses bekerja, maka proses tersebut merubah state (keadaan statis/ asal). Status dari sebuah proses didefinisikan dalam bagian oleh aktivitas yang ada dari proses tersebut. Tiap proses mungkin adalah satu dari keadaan berikut ini:

1. New: Proses sedang dikerjakan/ dibuat.
2. Running: Instruksi sedang dikerjakan.
3. Waiting: Proses sedang menunggu sejumlah kejadian untuk terjadi (seperti sebuah penyelesaian I/O atau penerimaan sebuah tanda/ signal).
4. Ready: Proses sedang menunggu untuk ditugaskan pada

sebuah prosesor.

5. Terminated: Proses telah selesai melaksanakan tugasnya/ mengeksekusi.

3.2.3 Operasi-operasi Terhadap Proses

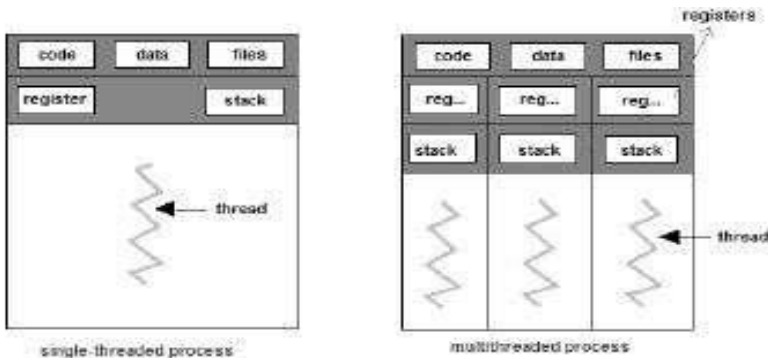
Penghancuran/terminasi proses (*destroy a process*): Melibatkan pembebasan proses dari sistem, yaitu:

1. Sumber daya-sumber daya yang dipakai dikembalikan
2. Proses dihancurkan dari senarai atau tabel system
3. PCB dihapus (ruang memori PCB dikembalikan ke pool bebas).

Penghancuran lebih rumit bila proses telah menciptakan proses-proses lain. Terdapat dua pendekatan, yaitu:

1. Pada beberapa sistem, proses-proses turunan dihancurkan saat proses-proses induk dihancurkan secara otomatis.
2. Beberapa sistem lain menganggap proses anak independen terhadap proses induk sehingga proses anak tidak secara otomatis dihancurkan saat proses induk dihancurkan.

3.3 Thread



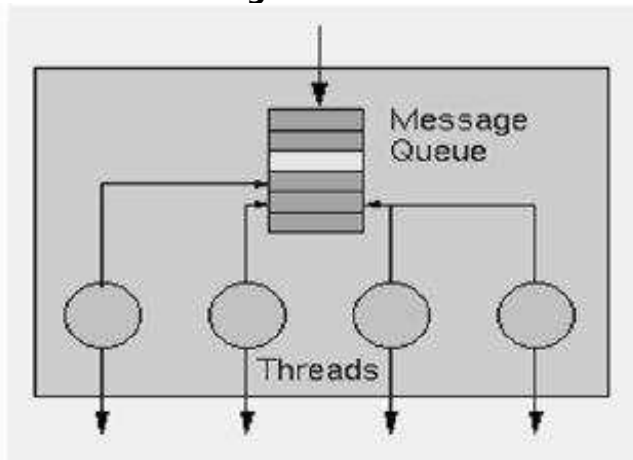
Gambar 3.3. Thread

3.3.1 Definisi Thread

Thread merupakan unit dasar dari penggunaan CPU, yang terdiri dari Thread_ID, program counter, register set, dan stack. Sebuah thread berbagi code section, data section, dan sumber daya sistem operasi dengan Thread lain yang dimiliki oleh proses yang sama. Thread juga sering disebut lightweight process.

Secara informal, proses adalah program yang sedang dieksekusi. Ada dua jenis proses, proses berat (heavyweight) atau biasa dikenal dengan proses tradisional, dan proses ringan atau kadang disebut thread. Thread saling berbagi bagian program, bagian data, dan sumber daya sistem operasi dengan thread lain yang mengacu pada proses yang sama.

3.3.2 Model Multithreading



Gambar 3.4. Multithreading

Implementasi Multithreading dalam sistem operasi terdiri dari:

1. *Model Many to One*

Model many to one ini memetakan beberapa tingkatan

pengguna thread hanya kesatu buah kernel thread.

2. *Model One to One*

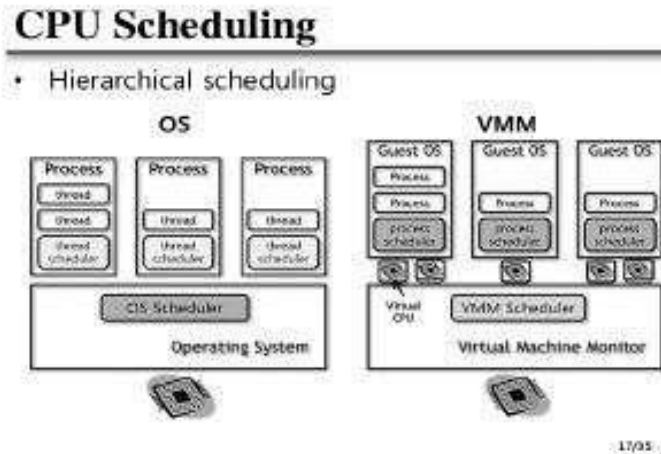
Model one to one memetakan setiap thread pengguna ke dalam satu kernel thread.

3. *Model Many to Many*

Beberapa tingkatan thread pengguna dapat menggunakan jumlah kernel thread yang lebih kecil atau sama dengan jumlah thread pengguna.

3.4 Penjadwalan Cpu

3.4.1 Konsep Dasar



Gambar 3.5. Urutan Kedatangan Proses

Penjadwalan CPU adalah basis dari multi programming sistem operasi dengan cara men-switch CPU di antara proses. Akibatnya sistem operasi bisa membuat komputer produktif. Tujuan dari multi programming adalah agar proses bisa berjalan secara bersamaan dengan tujuan untuk memaksimalkan kinerja dari CPU. Penjadwalan CPU mungkin akan dijalankan ketika terjadi proses:

1. Berubah dari running ke waiting state.
2. Berubah dari running ke ready state.

3. Berubah dari waiting ke ready.
4. Terminates.

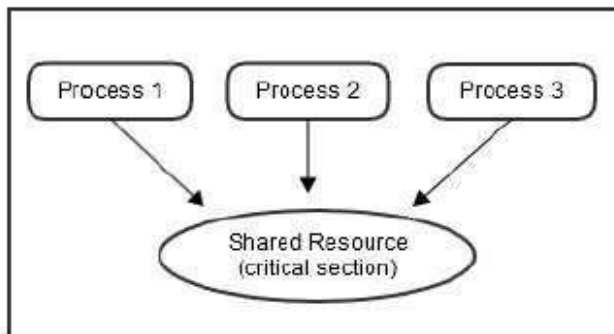
3.4.2 Kriteria Penjadwal

Banyak kriteria yang dianjurkan untuk membandingkan penjadwalan CPU algoritma. Kriteria yang biasanya digunakan dalam memilih adalah:

1. CPU utilization, artinya menjaga CPU sesibuk mungkin. CPU utilization akan mempunyai range dari 0 ke 100 persen.
2. Throughput artinya, jika CPU mengeksekusi suatu proses, berarti CPU telah melaksanakan sebuah kerja.
3. Turnaround time yaitu, interval dari waktu yang diizinkan dengan waktu yang dibutuhkan untuk menyelesaikan sebuah proses.
4. Waiting time, artinya dalam penjadwalan CPU algoritma penjadwal CPU tidak mempengaruhi waktu untuk melaksanakan proses tersebut atau I/O.
5. Response time. Di sistem yang interaktif, turnaround time mungkin bukan waktu yang terbaik untuk kriteria.

3.5 Sinkronisasi Proses

3.5.1 Definisi Sinkronisasi



Gambar 3.6. Sinkronisasi Sistem Operasi

Sinkronisasi merupakan suatu proses pengaturan jalannya beberapa proses pada waktu yang bersamaan untuk menyamakan waktu dan data supaya tidak terjadi inconsistensi (ketidak konsistenan) data akibat adanya akses data secara konkuren agar hasilnya bagus dan sesuai dengan apa yang diharapkan. Disini sinkronisasi diperlukan agar data tersebut tetap konsisten.

3.5.2 Race Condition

Race Condition merupakan situasi dimana beberapa proses mengakses dan memanipulasi data secara bersamaan. Nilai terakhir dari data bergantung dari proses mana yang selesai terakhir. Untuk menghindari *Race Condition*, proses-proses secara bersamaan harus di-sinkronisasikan.

3.5.3 Critical Section

Critical Section adalah sebuah segmen kode dimana sebuah proses memiliki sumber daya bersama yang diakses terdiri dari:

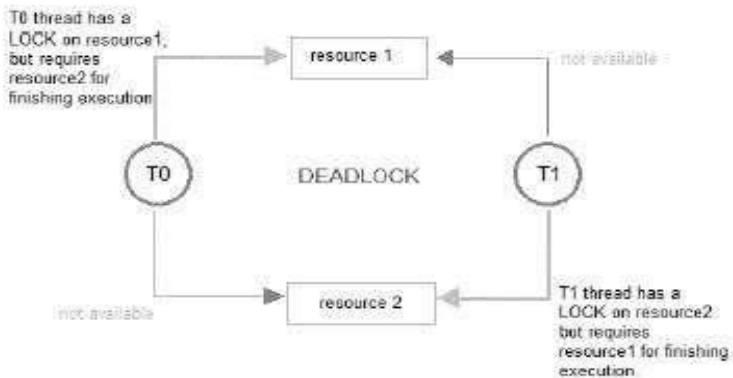
1. *Entry Section*, kode yang digunakan untuk masuk ke dalam *critical section*
2. *Critical Section*, kode dimana hanya ada satu proses yang dapat dieksekusi pada satu waktu
3. *Exit Section*, akhir dari *critical section*, mengizinkan proses lain
4. *Remainder Section*, merupakan kode istirahat setelah masuk ke *critical section*.

3.5.4 Semaphore

Semaphore adalah pendekatan yang diajukan oleh Dijkstra, dengan prinsip bahwa dua proses atau lebih dapat bekerja sama dengan menggunakan penanda-penanda sederhana. Seperti proses dapat dipaksa berhenti pada

suatu saat, sampai proses mendapatkan penanda tertentu itu.

3.6 *Deadlock*



Gambar 3.7. Deadlock Sistem Operasi

Dua program tersebut memegang kontrol terhadap sumber daya yang dibutuhkan oleh program yang lain. Tidak ada yang dapat melanjutkan proses masing-masing sampai program yang lain memberikan sumber dayanya, tetapi tidak ada yang mengalah. Kondisi inilah yang disebut Deadlock atau pada beberapa buku disebut *Deadly Embrace*.

3.6.1 Definisi *Deadlock*

Deadlock yang mungkin dapat terjadi pada suatu proses disebabkan proses itu menunggu suatu kejadian tertentu yang tidak akan pernah terjadi. Dua atau lebih proses dikatakan berada dalam kondisi *deadlock*, bila setiap proses yang ada menunggu suatu kejadian yang hanya dapat dilakukan oleh proses lain dalam himpunan tersebut.

3.6.2 Strategi Menghadapi *Deadlock*

Strategi untuk menghadapi *deadlock* dapat dibagi menjadi tiga pendekatan, yaitu:

1. Mengabaikan adanya *deadlock*.
2. Memastikan bahwa *deadlock* tidak akan pernah ada, baik dengan metode pencegahan, dengan mencegah empat kondisi *deadlock* agar tidak akan pernah terjadi.
3. Membiarkan *deadlock* untuk terjadi, pendekatan ini membutuhkan dua metode yang saling mendukung, yaitu:
 - a. Pendeteksian *deadlock*, yaitu untuk meng-identifikasi ketika *deadlock* terjadi.
 - b. Pemulihan *deadlock*, yaitu untuk meng-embalikan kembali sumber daya yang dibutuhkan pada proses yang memintanya.

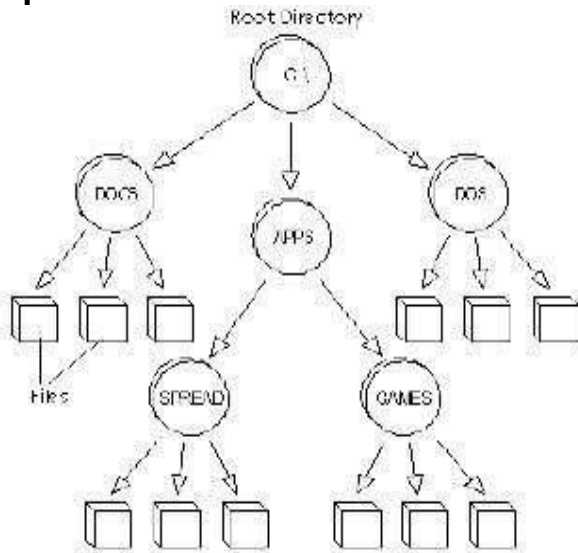
3.6.3 Mencegah *Deadlock*

Langkah yang digunakan adalah dengan spooling sumber daya, yaitu dengan mengantrikan job-job pada antrian dan akan dilayani satu-satu.

Cara kedua dengan meniadakan kondisi hold and wait terlihat lebih menjanjikan. Jika suatu proses yang sedang menggunakan sumber daya dapat dicegah agar tidak dapat menunggu sumber daya yang lain, maka *deadlock* dapat dicegah.

3.7 Sistem Berkas

3.7.1 Konsep Dasar



Gambar 3.8. Sistem Berkas

Sistem berkas merupakan mekanisme penyimpanan on-line serta untuk akses, baik data maupun program yang berada dalam Sistem Operasi. Terdapat dua bagian penting dalam sistem berkas, yaitu:

1. Kumpulan berkas, sebagai tempat penyimpanan data.
2. Struktur direktori, yang mengatur dan menyediakan informasi mengenai seluruh berkas dalam sistem.

Sistem Operasi mengabstraksi properti fisik dari media penyimpanannya dan mendefinisikan unit penyimpanan logis, yaitu berkas.

3.7.2 Operasi Pada Berkas

Berikut dapat kita lihat apa yang harus dilakukan sistem operasi pada keenam operasi dasar pada berkas:

1. Membuat sebuah berkas.

Pertama, tempat baru di dalam sistem berkas harus di alokasikan untuk berkas yang akan dibuat. Kedua, sebuah direktori harus mempersiapkan tempat untuk berkas baru, kemudian direktori tersebut akan mencatat nama berkas dan lokasinya pada sistem berkas.

2. Menulis pada sebuah berkas.

Untuk menulis pada berkas, kita menggunakan system call beserta nama berkas yang akan ditulisi dan informasi apa yang akan ditulis pada berkas. Ketika diberi nama berkas, sistem mencari ke direktori untuk mendapatkan lokasi berkas.

3. Membaca sebuah berkas.

Untuk dapat membaca berkas, kita menggunakan system call beserta nama berkas dan di blok memori mana berkas berikutnya diletakkan. Sama seperti menulis, direktori mencari berkas yang akan dibaca, dan sistem menyimpan penunjuk baca pada berkas dimana pembacaan berikutnya akan terjadi.

4. Menempatkan kembali sebuah berkas.

Direktori yang bertugas untuk mencari berkas yang bersesuaian, dan mengembalikan lokasi berkas pada saat itu. Menempatkan berkas tidak perlu melibatkan proses I/O. Operasi sering disebut pencarian berkas.

5. Menghapus sebuah berkas.

Untuk menghapus berkas kita perlu mencari berkas tersebut di dalam direktori. Setelah ditemukan kita membebaskan tempat yang dipakai berkas tersebut (sehingga dapat digunakan oleh berkas lain) dan menghapus tempatnya di direktori.

6. Memendekkan berkas.

Ada suatu keadaan dimana pengguna menginginkan atribut dari berkas tetap sama tetapi ingin menghapus isi dari berkas tersebut.

3.7.3 Jenis Berkas

Pertimbangan utama dalam perancangan sistem berkas dan seluruh sistem operasi, apakah sistem operasi harus mengenali dan mendukung jenis berkas. Jika suatu sistem operasi mengenali jenis dari berkas, maka ia dapat mengoperasikan berkas tersebut.

Sistem menggunakan akhiran tersebut untuk mengindikasikan jenis berkas dan jenis operasi yang dapat dilakukan pada berkas tersebut. Sebagai contoh hanya berkas yang berakhiran .bat, .exe atau .com yang bisa dijalankan (eksekusi).

3.7.4 Struktur Berkas

Beberapa sistem operasi menerapkan dan mendukung struktur berkas sedikit struktur berkas. Pendekatan ini digunakan pada MS-DOS dan UNIX. UNIX menganggap setiap berkas sebagai urutan 8-bit byte, tidak ada interpretasi sistem operasi terhadap dari bit-bit ini. Skema tersebut menawarkan fleksibilitas tinggi tetapi dukungan yang terbatas.

DAFTAR PUSTAKA

- John Wiley & Sons. Stalling, William, 2001, Operating Systems. Fourth Edition. Prentice Hall.
- Hariyanto, B. 1997. Sistem Operasi, Informatika, Bandung.
- William A. 1993. Introduction to Operating System. New York: HarperCollins College Publishers.
- Naghizadeh. 2005. Operating System: Concepts and Techniques. Amazon.
- Ron White. 1998. How Computers Work, Fourth Edition, Que corporation, A Division of Macmillan Computer Publishing. USA.
- Bob DuCharme. 2001. The Operating System Handbook or, Fake Your Way Through Minis and Mainframes. Singapore: McGraw-Hill Book Co.
- Silberschatz, A., dan Galvin P. 2003. Operating System Concept. Sixth Edition. Massachusetts: Addison.
- Silberschatz, Avi. Galvin, Peter & Gagne, Rag. 2000. Applied Operating Systems: First Edition. John Wiley & Sons.

BAB 4

PEMROGRAMAN

Oleh Achmad Ardiansyah

Pemrograman adalah salah satu keterampilan dasar yang diperlukan dalam dunia teknologi komputer. Bahasa pemrograman memungkinkan pengembang untuk membuat perangkat lunak yang memecahkan masalah, mengotomatiskan proses, dan menciptakan pengalaman interaktif. Bab ini akan memperkenalkan konsep dasar pemrograman, berbagai jenis bahasa pemrograman, dan pendekatan untuk menulis kode yang efisien.

4.1 Konsep Dasar Pemrograman

Logika dan Algoritma: Bagaimana program dikembangkan berdasarkan urutan logis dari instruksi, serta peran algoritma dalam menyelesaikan masalah. Konsep dasar pemrograman adalah prinsip-prinsip dasar yang harus dipahami oleh seorang programmer untuk membuat program komputer yang efektif dan efisien. Berikut adalah beberapa konsep dasar pemrograman yang umum (Admin LAB ICT, 2015):

1. Pengertian Pemrograman:

Pemrograman adalah suatu kumpulan urutan perintah ke komputer untuk mengerjakan sesuatu, dimana instruksi tersebut menggunakan bahasa yang dimengerti oleh komputer atau dikenal dengan bahasa pemrograman.

2. Model IPO (Input, Proses, Output):

- a. Input: Proses pemasukan data ke dalam komputer

melalui peralatan input.

- b. Proses: Pengolahan data yang meliputi menghitung, membandingkan, mengurutkan, mengklasifikasikan, mengendalikan, dan mencari di storage.
- c. Output: Proses untuk menghasilkan keluaran dari proses pengolahan data ke peralatan output seperti informasi (monitor, speaker, dll).
- d. Distribution: Penyebaran informasi kepada pihak-pihak yang berhak dan membutuhkan informasi.
- e. Storage: Perekaman hasil pengolahan data yang dapat digunakan untuk input proses selanjutnya.

3. Prinsip Bahasa Pemrograman:

Sintaks: Menjelaskan bagaimana struktur program yang benar. Tipe Sistem dan Semantik: Menunjukkan tipe nilai yang dapat dimanipulasi oleh program dan arti semantik dari program, mencakup aturan penamaan entitas (variabel, fungsi, class, parameter, dll).

Manajemen Memori: Menunjuk kepada sekumpulan teknik yang membantu memahami pemetaan letak nilai, struktur data, dan struktur program di dalam memori.

Exception Handling: Mengenai penanganan hal-hal yang tak terduga seperti kesalahan input ketika menjalankan program.

4. Algoritma dan Logika Pemrograman:

Pemrograman melibatkan membuat algoritma, serangkaian instruksi yang diikuti oleh komputer untuk menyelesaikan tugas tertentu. Pemahaman logika pemrograman membantu mengembangkan algoritma yang efisien dan tepat (UNMAHA, 2023).

5. Tipe Data dan Variabel:

Dalam pemrograman, Anda akan berurusan dengan berbagai jenis data seperti angka, teks, boolean, dan lainnya. Variabel adalah cara menyimpan dan mengelola

data.

6. Struktur Data, Algoritma, dan Pengendalian Alur Program (Thabroni, 2022):

Struktur data merujuk pada cara data diatur dan diolah dalam program. Algoritma adalah prosedur atau langkah-langkah yang digunakan untuk menyelesaikan masalah tertentu. Pengendalian alur program merujuk pada cara program menentukan urutan eksekusi instruksi.

7. Pemrograman Berorientasi Objek:

Melibatkan konsep abstraksi, pewarisan, dan polimorfisme untuk memungkinkan pembuatan program yang lebih efisien dan mudah dimengerti.

Dengan memahami konsep dasar pemrograman, seorang programmer dapat membuat program yang efektif, efisien, dan lebih mudah dipelihara dan ditingkatkan di masa depan.

Dalam pemrograman, sintaks, semantik, dan pragmatik tidak secara langsung terkait dengan konsep-konsep dasar pemrograman seperti algoritma, tipe data, atau struktur kontrol. Namun, perbedaan antara sintaks, semantik, dan pragmatik dalam konteks linguistik dapat memberikan pemahaman yang lebih luas tentang bagaimana bahasa dan komunikasi berfungsi, yang juga relevan dalam pemrograman sebagai cara untuk memahami dan mengkomunikasikan ide-ide program.

Sintaks

Sintaks dalam linguistik merujuk pada struktur formal bahasa, yaitu bagaimana kata-kata diatur untuk membentuk kalimat yang benar. Dalam pemrograman, sintaks merujuk pada aturan-aturan yang mengatur struktur kode program, seperti penulisan fungsi, perulangan, dan kondisi. Contoh:

sintaks bahasa pemrograman seperti Python memiliki aturan-aturan yang spesifik untuk penulisan kode yang valid (Rahmawati, 2019).

Semantik

Semantik dalam linguistik mempelajari makna kata dan kalimat. Dalam konteks pemrograman, semantik berarti memahami apa yang dilakukan oleh kode program. Ini melibatkan memahami tipe data, operasi yang dapat dilakukan, dan bagaimana variabel digunakan. Contoh: dalam bahasa pemrograman, semantik melibatkan memahami bahwa `x = 5` berarti variabel `x` sekarang memiliki nilai 5 (Prayogi, 2012).

Pragmatik

Pragmatik dalam linguistik mempelajari bagaimana makna bahasa digunakan dalam konteks tertentu. Dalam konteks pemrograman, pragmatik berarti memahami bagaimana kode program digunakan dalam situasi nyata. Ini melibatkan memahami konteks penggunaan, seperti bagaimana sebuah fungsi digunakan dalam aplikasi yang lebih besar. Contoh: dalam pengembangan aplikasi, pragmatik melibatkan memahami bahwa sebuah fungsi `login()` tidak hanya tentang memeriksa kredensial, tetapi juga tentang bagaimana fungsi tersebut digunakan dalam proses autentikasi yang lebih kompleks (Prayogi, 2012; Yudhistira, 2021).

Kesimpulan

Sintaks mengatur struktur formal kode program.

Semantik memahami apa yang dilakukan oleh kode program.

Pragmatik memahami bagaimana kode program digunakan dalam konteks nyata.

Dengan memahami perbedaan antara sintaks, semantik, dan pragmatik, seorang programmer dapat membuat program yang lebih efektif dan mudah dipahami oleh pengguna.

Sintaks dan semantik dalam pemrograman saling terkait dalam beberapa cara, tetapi juga memiliki perbedaan yang signifikan. Berikut adalah bagaimana sintaks mempengaruhi semantik dalam pemrograman:

Sintaks dan Semantik

1. Struktur Formal:

- a. Sintaks mengatur struktur formal bahasa pemrograman, yaitu bagaimana kata-kata dan simbol digunakan untuk membentuk ekspresi yang valid. Contoh: dalam bahasa Python, sintaks seperti `x = 5` adalah valid karena mengikuti aturan penulisan variabel dan operator.
- b. Semantik, di sisi lain, mempelajari makna dari ekspresi tersebut. Dalam konteks di atas, semantik akan menafsirkan bahwa `x` sekarang memiliki nilai 5 dan dapat digunakan dalam program.

2. Penafsiran Ekspresi:

Sintaks memberikan batasan atas apa yang dapat ditulis dalam kode, sedangkan semantik menentukan apa yang sebenarnya dilakukan oleh kode tersebut. Contoh: sintaks `x += 5` valid dalam banyak bahasa pemrograman, tetapi semantiknya berbeda-beda. Dalam bahasa Python, `x += 5` akan menambahkan 5 ke nilai `x`, sedangkan dalam bahasa C++, `x += 5` akan menambahkan 5 ke nilai `x` dan kemudian mengembalikan nilai `x`.

3. Keterkaitan Sintaks dan Semantik:

- a. Sintaks yang tidak tepat dapat menghasilkan kesalahan semantik. Contoh: sintaks `x = 5; y = x + z`

akan gagal jika `z` belum dideklarasikan, karena sintaksnya tidak valid. Semantiknya juga tidak dapat dipahami karena variabel `z` tidak didefinisikan.

- b. Sebaliknya, sintaks yang tepat tetapi semantiknya salah juga dapat menyebabkan kesalahan. Contoh: sintaks `x = 5; y = x + z` akan gagal jika `z` memiliki nilai yang tidak dapat ditambahkan dengan `x`, karena semantiknya tidak tepat.

Contoh Praktis

Contoh 1:

- a. Sintaks: `x = 5; y = x + z`
- b. Semantik: Jika `z` belum dideklarasikan, maka sintaks ini tidak valid karena variabel `z` tidak didefinisikan.
- c. Kesimpulan: Sintaks yang tidak tepat dapat menghasilkan kesalahan semantik.

Contoh 2:

- a. Sintaks: `x = 5; y = x + z`
- b. Semantik: Jika `z` memiliki nilai yang tidak dapat ditambahkan dengan `x`, maka sintaks ini tidak valid karena semantiknya tidak tepat.
- c. Kesimpulan: Sintaks yang tepat tetapi semantiknya salah juga dapat menyebabkan kesalahan.

Kesimpulan

Sintaks mempengaruhi semantik dalam pemrograman dengan menentukan apa yang dapat ditulis dalam kode dan apa yang sebenarnya dilakukan oleh kode tersebut. Sintaks yang tidak tepat dapat menghasilkan kesalahan semantik, sedangkan sintaks yang tepat tetapi semantiknya salah juga dapat menyebabkan kesalahan. Dengan demikian, pemahaman yang baik tentang sintaks dan semantik sangat penting untuk menulis kode yang efektif dan efisien (Educa Studio, 2023; Kandaga, 2005).

4.2 Bahasa Pemrograman

Terdapat banyak bahasa pemrograman, masing-masing dengan karakteristik dan kelebihanannya. Sub bab ini akan menjelaskan beberapa bahasa pemrograman utama, termasuk:

Bahasa Tingkat Tinggi dan Rendah: Perbedaan antara bahasa yang lebih dekat ke bahasa mesin (*assembly*) dan yang lebih dekat ke bahasa manusia (*Python, Java*).

Pemrograman Berbasis Objek: Konsep OOP (*Object-Oriented Programming*), serta prinsip inheritance, encapsulation, dan polymorphism.

Bahasa Pemrograman Populer: Ringkasan bahasa pemrograman yang paling banyak digunakan seperti Python, Java, C++, dan JavaScript, serta kasus penggunaannya.

Bahasa pemrograman adalah sistem notasi yang digunakan untuk menulis program komputer. Berikut adalah beberapa poin penting tentang bahasa pemrograman:

Pengertian Bahasa Pemrograman

Definisi: Bahasa pemrograman adalah sistem komunikasi antara manusia dan mesin yang digunakan untuk menginstruksikan komputer dalam melakukan tugas tertentu.

Fungsi: Bahasa pemrograman memungkinkan pengembang untuk menulis kode yang terstruktur dan dapat dipahami oleh perangkat komputer. Fungsi utamanya adalah memberikan instruksi kegiatan komputasi, mengatur kontrol perangkat internal ke perangkat eksternal, dan mengkomunikasikan keinginan pengguna komputer secara presisi.

Komponen Bahasa Pemrograman

Sintaksis: Sintaksis adalah struktur formal bahasa pemrograman, yaitu bagaimana kata-kata dan simbol digunakan untuk membentuk ekspresi yang valid. Contoh:

sintaks bahasa Python memiliki aturan-aturan yang spesifik untuk penulisan kode yang valid (Wikipedia, 2020).

Semantik: Semantik mempelajari makna dari ekspresi bahasa pemrograman. Contoh: pernyataan `x += y` memiliki makna yang sama di berbagai bahasa pemrograman seperti C, C++, Java, dan Python, tetapi mungkin ditulis dengan sintaks yang berbeda.

Jenis-Jenis Bahasa Pemrograman

Bahasa Tingkat Tinggi: Bahasa tingkat tinggi seperti Python, Java, dan JavaScript memungkinkan programmer untuk menulis kode dengan abstraksi yang lebih tinggi, sehingga lebih mudah dipahami dan digunakan.

Bahasa Tingkat Rendah: Bahasa tingkat rendah seperti *Assembly Language* memungkinkan programmer untuk mengakses perangkat keras secara langsung, tetapi lebih sulit digunakan karena memerlukan pengetahuan yang lebih spesifik tentang arsitektur komputer (Wikipedia, 2020).

Contoh Bahasa Pemrograman Populer

JavaScript: Digunakan untuk pengembangan web interaktif dan dinamis.

Python: Digunakan untuk berbagai aplikasi, termasuk pengembangan web, analisis data, dan *machine learning* (Binus University, 2023; Educa Studio, 2023a).

Java: Digunakan untuk pengembangan aplikasi berbasis desktop dan mobile, serta pengembangan web (Binus University, 2023; Educa Studio, 2023a).

Dengan demikian, bahasa pemrograman adalah alat yang sangat penting dalam pengembangan perangkat lunak dan aplikasi, dan memiliki berbagai jenis dan karakteristik yang berbeda-beda.

Bahasa pemrograman dapat dikelompokkan menjadi dua kategori utama berdasarkan tingkat abstraksinya: bahasa pemrograman tingkat tinggi dan bahasa pemrograman tingkat rendah. Berikut adalah perbedaan antara keduanya:

Bahasa Pemrograman Tingkat Tinggi

Tingkat Abstraksi: Tingkat abstraksi yang lebih tinggi dibandingkan dengan bahasa pemrograman tingkat rendah. Bahasa tingkat tinggi menggunakan instruksi yang mirip dengan bahasa manusia, seperti "print", "if", dan "input" (Iyan Ramadhani, 2023).

Mudah Dipahami: Bahasa tingkat tinggi lebih mudah dipahami oleh manusia karena menggunakan perintah yang lebih dekat dengan bahasa Inggris dan memiliki sintaks yang lebih terstruktur.

Komunikasi dengan Perangkat Keras: Meskipun tidak berkomunikasi langsung dengan perangkat keras, bahasa tingkat tinggi menyediakan abstraksi yang memungkinkan programmer untuk fokus pada logika program tanpa perlu memikirkan detail perangkat keras.

Fitur yang Lengkap: Bahasa tingkat tinggi seringkali menyediakan fitur-fitur yang lengkap seperti rutin penanganan string, fitur pemrograman berorientasi objek, dan input/output terhadap berkas.

Kompilator atau Interpreter: Bahasa tingkat tinggi membutuhkan kompilator atau interpreter untuk menerjemahkan kode program menjadi kode mesin yang dapat dipahami oleh komputer.

Bahasa Pemrograman Tingkat Rendah

Tingkat Abstraksi: Tingkat abstraksi yang lebih rendah dibandingkan dengan bahasa pemrograman tingkat tinggi. Bahasa tingkat rendah menggunakan instruksi dasar yang

dikenali oleh komputer, seperti kode biner atau kode mesin.

Komunikasi Langsung dengan Perangkat Keras: Bahasa tingkat rendah dapat berkomunikasi langsung dengan perangkat keras seperti memori, register, dan instruksi mesin.

Sulit Dipahami: Bahasa tingkat rendah lebih sulit dipahami oleh manusia karena menggunakan instruksi dasar yang tidak mirip dengan bahasa manusia.

Lebih Efisien: Bahasa tingkat rendah cenderung lebih hemat memori dan dapat menghasilkan kode yang lebih efisien karena tidak ada lapisan abstraksi tambahan.

Tidak Membutuhkan Kompiler atau Interpreter: Bahasa tingkat rendah tidak membutuhkan kompiler atau interpreter karena kode programnya sudah dalam bentuk yang dapat dieksekusi langsung oleh komputer (Admin G2, 2023).

Dengan demikian, bahasa pemrograman tingkat tinggi memungkinkan programmer untuk menulis kode yang lebih mudah dipahami dan lebih fokus pada logika program, sedangkan bahasa pemrograman tingkat rendah memberikan kontrol yang lebih tinggi terhadap perangkat keras dan dapat menghasilkan kode yang lebih efisien.

4.3 Struktur Data dan Algoritma

Struktur Data Dasar: Seperti array, linked list, stack, dan queue. Struktur data dan algoritma adalah dua konsep penting dalam ilmu komputer yang saling terkait dalam pengembangan perangkat lunak. Berikut adalah penjelasan tentang masing-masing konsep dan bagaimana mereka berinteraksi:

Struktur Data

Struktur data adalah cara menyimpan dan mengatur data secara terstruktur pada sistem komputer atau database, sehingga data tersebut lebih mudah diakses dan diperbarui. Struktur data dapat berupa array, linked list, stack, queue, tree,

dan graph, masing-masing memiliki kelebihan dan kekurangan yang spesifik.

Jenis Struktur Data:

Struktur Data Linear:

Array: Kumpulan data yang disimpan pada lokasi memori yang berdekatan, dengan indeks yang digunakan untuk mengakses setiap elemen (Anendya, 2024; Pradana, 2022).

Linked List: Urutan data linier yang dapat dihubungkan satu sama lain, dengan setiap elemen memiliki pointer ke elemen berikutnya.

Stack dan Queue: Struktur data yang memungkinkan akses elemen terakhir yang ditambahkan (stack) atau elemen pertama yang ditambahkan (queue).

Struktur Data Non-Linear:

Tree: Struktur data yang memiliki hubungan hierarkis antara elemen-elemen, dengan setiap node memiliki anak-anak dan/atau orang tua.

Graph: Struktur data yang terdiri dari kumpulan node dan koneksi antara node-node tersebut, sering digunakan untuk mewakili jaringan.

Algoritma

Algoritma adalah serangkaian instruksi yang diikuti oleh komputer untuk menyelesaikan tugas tertentu. Algoritma memungkinkan komputer untuk mengolah data dengan cara yang sistematis dan efisien.

Keterkaitan dengan Struktur Data:

Penggunaan Struktur Data dalam Algoritma:

Struktur data memainkan peran penting dalam algoritma karena mereka menyediakan cara yang efisien untuk menyimpan, mengakses, dan memproses data. Contoh, dalam penggunaan array, algoritma dapat dengan mudah

mengakses dan memanipulasi elemen-elemen array untuk mencapai hasil yang diinginkan.

Desain Algoritma yang Efektif:

Pemahaman yang mendalam tentang struktur data diperlukan untuk mengembangkan algoritma yang efektif. Struktur data yang tepat dapat meningkatkan kualitas algoritma dan membuatnya lebih efisien dalam penggunaan sumber daya komputer.

Kesimpulan

Struktur data dan algoritma saling terkait dalam pengembangan perangkat lunak. Struktur data menyediakan cara yang efisien untuk menyimpan dan mengatur data, sedangkan algoritma menggunakan struktur data untuk menyelesaikan tugas-tugas yang kompleks. Dengan memahami baik struktur data dan algoritma, seorang programmer dapat mengembangkan program yang lebih efektif dan efisien (Nurchaliza, 2024).

Struktur data dalam ilmu komputer dapat dikelompokkan menjadi dua jenis utama: linear dan non-linear. Berikut adalah penjelasan tentang jenis-jenis struktur data linear dan non-linear:

Struktur data linear adalah jenis struktur data yang elemennya disusun secara berurutan. Berikut adalah beberapa jenis struktur data linear yang umum digunakan (Putri, 2023):

1. Array

Array adalah kumpulan elemen dengan tipe data yang sama, disusun dalam urutan terdefinisi. Cara Kerja: Elemen di dalam array dapat diakses langsung berdasarkan indeks.

Contoh: Contoh array adalah penyimpanan nilai mahasiswa, di mana setiap data nilai menempati satu slot di array dan setiap slot di cek berdasarkan indeksnya.

2. Linked List

Linked list adalah struktur data yang terdiri dari simpul-simpul yang saling terhubung. Cara Kerja: Setiap simpul terbagi menjadi dua, yaitu data dan referensi ke simpul berikutnya. Ini memungkinkan alokasi memori dinamis dan fleksibilitas dalam pengelolaan data.

3. Stack

Pengertian: Stack adalah struktur data yang mengikuti prinsip LIFO (*Last-In-First-Out*). Cara Kerja: Elemen pertama yang dimasukkan adalah elemen terakhir yang digunakan. Stack sering digunakan dalam penanganan function call dan task stacks.

4. Queue

Pengertian: Queue adalah struktur data yang mengikuti prinsip FIFO (*First-In-First-Out*). Cara Kerja: Elemen pertama yang dimasukkan adalah yang keluar pertama. Queue digunakan dalam situasi antrian tugas, seperti pengolahan pesanan di toko.

Struktur Data Non-Linear

Struktur data non-linear adalah jenis struktur data di mana elemen-elemennya tidak disusun secara berurutan. Mereka menciptakan hierarki atau jaringan kompleks antar elemen (Azzukhruf, 2024).

1. Tree

Pengertian: Tree adalah struktur data hierarkis yang terdiri dari simpul-simpul yang saling terhubung dalam pola pohon. Cara Kerja: Setiap simpul memiliki satu elemen induk (*parent*) dan beberapa komponen anak (*children*). Tree sering digunakan dalam pencarian data cepat atau artificial intelligence.

2. Graph

Graph adalah struktur data yang terdiri dari simpul-simpul

yang terhubung oleh edge. Cara Kerja: Setiap node disebut sebagai vertices, dan koneksi antar node disebut sebagai edges. Graph sering digunakan untuk merepresentasikan jenis hubungan antar objek, seperti jaringan sosial atau komputer.

Dengan demikian, struktur data linear seperti array, linked list, stack, dan queue disusun secara berurutan, sedangkan struktur data non-linear seperti tree dan graph menciptakan hierarki atau jaringan kompleks antar elemen.

4.4 Pemrograman Berbasis Web

Dengan berkembangnya internet, pemrograman web menjadi keterampilan penting. Pemrograman berbasis web adalah proses pembuatan dan pengembangan situs web atau aplikasi web menggunakan bahasa pemrograman. Berikut adalah penjelasan lebih lanjut tentang pemrograman web dan bahasa-bahasa pemrograman yang umum digunakan:

Pengertian Pemrograman Web

Pemrograman web adalah proses pembuatan sekaligus pengembangan situs web atau aplikasi dengan menggunakan bahasa pemrograman. Tujuan utamanya adalah membuat situs web yang fungsional, interaktif, dan menarik secara visual (Educa Studio, 2023a).

Komponen Utama Pemrograman Web (Clara, 2024; Educa Studio, 2023a; Intern, 2019)

1. HTML (*HyperText Markup Language*):

Digunakan untuk struktur dasar dan tampilan situs web. HTML mendeskripsikan struktur konten web dan elemen-elemen yang akan ditampilkan di halaman.

2. CSS (*Cascading Style Sheets*):

Digunakan untuk mengatur tampilan dan gaya halaman

- web. CSS memungkinkan pengembang untuk menambahkan desain-desain tertentu pada halaman web agar tampilannya menarik.
3. JavaScript:
Digunakan untuk membuat interaksi dinamis pada halaman web. JavaScript memungkinkan pengembang untuk menambahkan logika dan interaksi yang kompleks pada situs web.

Bahasa Pemrograman Lainnya

Selain HTML, CSS, dan JavaScript, ada beberapa bahasa pemrograman lain yang umum digunakan dalam pengembangan web:

1. PHP: Bahasa pemrograman domain-specific yang digunakan untuk mengolah data dalam Relational Database Management System (RDBMS). PHP sering digunakan untuk membuat aplikasi web yang kompleks.
2. Java: Bahasa pemrograman yang fleksibel dan dapat digunakan pada banyak platform. Java sering digunakan untuk membuat aplikasi web yang besar dan kompleks (Wahab, 2022).
3. Python: Bahasa pemrograman yang populer digunakan dalam pengembangan web, terutama dengan framework seperti Django dan Flask.

Proses Pembuatan Situs Web

Pembuatan situs web melibatkan beberapa tahap, termasuk:

1. Desain dan Layout: Menggunakan HTML dan CSS untuk membuat struktur dan tampilan dasar situs web.
2. Interaksi Dinamis: Menggunakan JavaScript untuk menambahkan interaksi dinamis pada halaman web.
3. Pengolahan Data: Menggunakan PHP atau bahasa pemrograman lainnya untuk mengolah data dan

membuat aplikasi web yang kompleks.

Dengan memahami bahasa-bahasa pemrograman web dan komponen utamanya, pengembang dapat membuat situs web yang fungsional, interaktif, dan menarik secara visual.

Penggunaan API (Antarmuka Pemrograman Aplikasi) sangat penting dalam menghubungkan layanan web dan integrasi teknologi cloud dalam pemrograman. Berikut adalah beberapa cara dan manfaat penggunaan API dalam konteks ini:

1. Integrasi Layanan Web

API memungkinkan berbagai aplikasi dan layanan web berkomunikasi dan berbagi data secara efisien. Contohnya:

Sinkronisasi Data: API digunakan untuk sinkronisasi data otomatis antara klien dan server, seperti sinkronisasi data galeri gambar ponsel ke cloud (Amazon, 2023).

Komunikasi Aplikasi: API memfasilitasi pertukaran data dan komunikasi antar aplikasi tanpa campur tangan manusia, meningkatkan kelancaran berbagi data dan mengintegrasikan aplikasi yang sedang digunakan (Santoso, 2023).

2. Integrasi Teknologi Cloud

API juga digunakan untuk menghubungkan aplikasi yang berjalan di lingkungan komputasi awan (cloud) agar dapat berbagi data dan fungsionalitas. Contohnya:

WhatsApp Cloud API: Memungkinkan bisnis menengah dan besar untuk berinteraksi dengan pelanggan menggunakan versi yang ada di-host di cloud Meta, seperti integrasi profil lead, kompatibilitas digital pipeline, dan chatbot.

Apigee Integration: Membantu perusahaan

menghubungkan data dan aplikasi yang ada, serta menampilkannya sebagai API yang mudah diakses. Ini memungkinkan developer untuk menghubungkan berbagai aplikasi secara bersamaan dan mem-build beberapa API dengan lancar.

4. Manfaat Penggunaan API

Menghemat Waktu dan Biaya: Integrasi API memungkinkan alur integrasi untuk didefinisikan dan digunakan kembali oleh banyak pihak di dalam dan di luar organisasi, menghemat waktu dan biaya dalam hal kekinian informasi dan keakuratan data.

Mengurangi Risiko: API membantu mengurangi risiko yang terkait dengan tantangan konektivitas data dengan menggunakan solusi terpadu yang memungkinkan developer untuk tidak hanya menghubungkan aplikasi yang ada, tetapi juga membangun dan mengelola API dalam antarmuka yang sama.

Meningkatkan Efisiensi Operasional: API memungkinkan perusahaan untuk mengotomatisasi sistem, meningkatkan efisiensi operasional, dan analisis data dengan lebih baik. Contohnya, integrasi e-commerce, keuangan, aplikasi cloud, dan aplikasi internal dapat meningkatkan efisiensi operasional dan analisis data (UG Admin, 2023).

5. Cara Kerja API

Aplikasi Mengakses API: Ketika pengguna mengakses sebuah aplikasi, maka secara otomatis API menyediakan layanan yang dituju pengguna. **API Melakukan Request ke Server:** Jika telah berhasil mengakses alamat API, permintaan diteruskan menuju server layanan tujuan. **Server Memberikan Respon Ke API:** Saat data permintaan telah ditemukan, server

menghubungi API untuk memberikan informasi rincian layanan. API Meneruskan Respon Ke Aplikasi: API melanjutkan informasi dari server menuju aplikasi/website Anda. Pengguna pun dapat langsung menerima informasi dan rincian layanan yang dicari.

Dengan demikian, penggunaan API sangat penting dalam menghubungkan layanan web dan integrasi teknologi cloud, karena memungkinkan berbagai aplikasi dan layanan berkomunikasi dan berbagi data secara efisien, menghemat waktu dan biaya, serta meningkatkan efisiensi operasional dan analisis data.

4.5 Kesalahan Pemrograman dan Debugging

Kesalahan pemrograman dan debugging adalah dua konsep yang saling terkait dalam pengembangan perangkat lunak. Berikut adalah penjelasan lebih lanjut tentang kesalahan pemrograman dan proses debugging (Amazon, 2022; Pasla, 2024; Prasatya, 2024):

Kesalahan Pemrograman

Kesalahan pemrograman, juga dikenal sebagai "bug," adalah masalah yang terjadi dalam kode sumber perangkat lunak yang menyebabkan program tidak berfungsi dengan benar. Kesalahan ini dapat berupa kesalahan sintaks, kesalahan semantik, atau kesalahan logika.

Proses Debugging

Debugging adalah proses menemukan dan memperbaiki kesalahan atau bug di dalam kode sumber perangkat lunak. Berikut adalah langkah-langkah umum dalam proses debugging:

1. Identifikasi Kesalahan:

Laporan Bug: Pengguna, pengembang, atau pengujian melaporkan bug yang mereka temukan saat menguji atau menggunakan perangkat lunak. Pencarian Baris Kode:

- Pengembang menemukan baris kode atau modul kode yang tepat yang menyebabkan bug.
2. Analisis Kesalahan:
Rekam Perubahan Status Program: Pengembang merekam semua perubahan status program dan nilai data untuk memahami alur kerja program. Prioritaskan Perbaikan: Pengembang memprioritaskan perbaikan bug berdasarkan dampaknya pada fungsionalitas perangkat lunak.
 3. Perbaiki dan Validasi:
Perbaikan Bug: Pengembang memperbaiki bug dan menjalankan uji untuk memastikan perangkat lunak terus bekerja seperti yang diharapkan. Uji Baru: Mungkin menulis uji baru untuk memeriksa apakah bug akan kembali terjadi di masa mendatang.

Strategi Debugging

Berikut beberapa strategi debugging yang umum digunakan:

1. Cetak Pernyataan:
Menggunakan pernyataan cetak di berbagai titik dalam kode untuk melacak nilai variabel dan mengidentifikasi di mana kesalahan terjadi.
2. Alat Debugger:
Menggunakan alat debugger untuk melangkah melalui kode baris demi baris, memeriksa nilai variabel, dan menemukan titik masalah yang tepat.
3. Logika Bertahap:
Sederhanakan kode langkah demi langkah untuk melihat di mana kesalahan itu muncul, seperti membongkar mobil menjadi beberapa bagian untuk menemukan suku cadang yang rusak.
4. Pendekatan Sistematis:
Menggunakan pendekatan sistematis dalam debugging

melibatkan langkah demi langkah yang logis dan terstruktur untuk menemukan dan memperbaiki bugs.

5. Alat Debugging:

Menggunakan alat debugging yang tersedia, seperti debugger dalam *Integrated Development Environment* (IDE) atau alat eksternal yang lebih khusus, untuk membantu dalam proses debugging.

Pentingnya Debugging

Debugging memiliki peran yang sangat penting dalam pengembangan perangkat lunak karena:

1. Menghemat Waktu dan Tenaga:
 - Mendeteksi dan memperbaiki kesalahan sejak dini akan jauh lebih mudah dan cepat dibandingkan ketika program sudah jadi dan penuh dengan bug.
2. Menghasilkan Program yang Berkualitas:

Program yang bebas bug lebih stabil, dapat diandalkan, dan mudah dipelihara, terutama untuk program yang digunakan secara luas atau program mission-critical yang menangani data sensitif.
3. Meningkatkan Kemampuan Problem-Solving:

Proses debugging melatih kemampuan untuk berpikir kritis, memecahkan masalah secara sistematis, dan menganalisis kode secara detail.
4. Membangun Kebiasaan Coding yang Baik:

Debugging akan mendorong pengembang untuk menulis kode yang lebih bersih, terstruktur, dan mudah dibaca, sehingga program lebih mudah dipahami dan didebug tidak hanya oleh pengembang sendiri, tetapi juga oleh pengembang lain.

Dengan demikian, kesalahan pemrograman dan proses debugging adalah bagian integral dalam pengembangan perangkat lunak yang efektif dan efisien.

DAFTAR PUSTAKA

- Admin G2. (2023, July 22). Bahasa Pemrograman Tingkat Rendah & Tingkat Tinggi - Blog - G2Academy. Blog - G2Academy.
<https://insights.g2academy.co/educational/bahasa-pemrograman-tingkat-rendah-tingkat-tinggi/>
- Admin LAB ICT. (2015, July 29). Konsep Dasar Pemrograman – Laboratorium ICT Terpadu Universitas Budi Luhur. Budiluhur.ac.id. <https://labict.budiluhur.ac.id/konsep-dasar-pemrograman/>
- Amazon. (2023). Apa itu API? - Panduan Pemula API - AWS. Amazon Web Services, Inc.
<https://aws.amazon.com/id/what-is/api/>
- Anendya, A. (2024, February 15). Apa Itu Struktur Data? Manfaat, Jenis, dan Contohnya. Blog Dewaweb.
<https://www.dewaweb.com/blog/struktur-data-untuk-pemrograman/>
- Azzukhruf, K. (2024, January 31). Mengenal Apa Pengertian Struktur Data dan Jenis-Jenisnya. Dibimbing.id; dibimbing.id.
<https://dibimbing.id/blog/detail/mengenal-apa-pengertian-struktur-data-dan-jenis-jenisnya>
- Binus University. (2022, July 18). Pengertian Blockchain beserta Manfaat dan Cara Kerjanya. BINUS Online.
<https://online.binus.ac.id/2022/07/18/pengertian-blockchain-serta-manfaat-dan-cara-kerjanya/>
- Clara, A. (2024, April 22). Pengertian & Penjelasan Pemrograman Website Secara Lengkap. D3 Teknologi Komputer Telkom University.
<https://dce.telkomuniversity.ac.id/pengertian-penjelasan-pemrograman-website-secara-lengkap/>
- Educa Studio. (2023a). 11 Bahasa Pemrograman Web Terbaik

dan Paling Populer untuk Pemula | Berita | Gamelab Indonesia. Gamelab.ID.

<https://www.gamelab.id/news/2462-11-bahasa-pemrograman-web-terbaik-dan-paling-populer-untuk-pemula>

Intern, D. (2019, December 10). Berkenalan Dengan Pemrograman Web. Dicoding Blog. <https://www.dicoding.com/blog/kenal-pemrograman-web/>

Kandaga, T. (2005). Analisis dan Perancangan Bahasa Pemrograman Paralel Beserta Pembuatan Prototipe Kompilatornya. <https://media.neliti.com/media/publications/219172-analisis-dan-perancangan-bahasa-pemrogra.pdf>

Nurchaliza, R. (2024). Peran Penting Memahami Struktur Data dalam Ilmu Komputer. D3 Rekayasa Perangkat Lunak Aplikasi. <https://dif.telkomuniversity.ac.id/en/peran-penting-memahami-struktur-data/>

Pasla, B. N. (2024, April 19). Apa itu Debugging?: Sejarah, Jenis, dan Langkah-Langkah - BAMS. BAMS. <https://pasla.jambiprov.go.id/apa-itu-debugging-sejarah-jenis-dan-langkah-langkah/>

Pradana, T. P. (2022, January 27). Pengertian Struktur Data dan Tipenya dalam Pemrograman! IDwebhost. <https://idwebhost.com/blog/struktur-data-dan-tipenya/>

Prasatya. (2024). Debugging dalam Pemrograman: Penting Bagi Programmer? - CODEPOLITAN. Codepolitan.com. <https://www.codepolitan.com/blog/debugging-dalam-pemrograman-penting-bagi-programmer/>

Prayogi, I. (2012, April 9). Mengenal Perbedaan Semantik dan Pragmatik - Kompasiana.com. KOMPASIANA; Kompasiana.com.

- <https://www.kompasiana.com/icukprayogi/550f5eb6a33311ab2dba85e9/mengenal-perbedaan-semantik-dan-pragmatik>
- Putri, S. (2023). Struktur Data: Fungsi, Karakteristik, dan 2 Jenisnya. Kelas.work. <https://kelas.work/blogs/ketahui-berbagai-karakteristik-dan-jenis-jenis-struktur-data>
- Rahmawati, V. (2019, September 30). Informasi Sintaksis, Semantik Dan Pragmatik. Blogspot.com; Blogger. <https://vianarahmawati100.blogspot.com/2019/09/informasi-sintaksis-semantik-dan.html>
- Santoso, A. F. (2023, November 9). Integrasi API: Langkah Praktis Optimalkan Teknologi Bisnis - PhinCon. PhinCon. <https://phincon.com/articles/integrasi-api/>
- Sari, R. P. (2024). Apa itu Blockchain? Pengertian, Cara Kerja dan Penerapannya. Cloud Computing Indonesia. <https://www.cloudcomputing.id/pengetahuan-dasar/blockchain-pengertian-cara>
- Thabroni, G. (2022, February 24). Pemrograman Dasar - Konsep, Logika, Prinsip & Langkah - serupa.id. Serupa.id. <https://serupa.id/pemrograman-dasar-konsep-logika-prinsip-langkah/>
- UG Admin. (2023, September 12). Apa itu API Integration? Pengertian hingga Contoh Layanannya! - PT Usaha Gedung Mandiri. PT Usaha Gedung Mandiri. <https://ugmandiri.co.id/apa-itu-layanan-api-integration/>
- UNMAHA. (2023a, August 16). Panduan Pemula: Memahami Konsep Dasar Pemrograman. Blog | Universitas Mahakarya Asia | UNMAHA. <https://blog.unmaha.ac.id/panduan-pemula-memahami-konsep-dasar-pemrograman/>
- Wahab, A. (2022, October 14). Panduan Belajar Pemrograman

Web Bagi Pemula. Niagahoster Blog.
<https://www.niagahoster.co.id/blog/pemrograman-web/>

Wikipedia. (2020, May 6). Bahasa pemrograman. Wikipedia.
https://id.wikipedia.org/wiki/Bahasa_pemrograman

Yudhistira. (2021, April 4). Perbedaan Semantik dan Pragmatika - Narabahasa. Narabahasa.
<https://narabahasa.id/artikel/linguistik-umum/pragmatik/perbedaan-semantik-dan-pragmatika/>

BAB 5

DATABASE

Oleh Mukrodin

5.1 Konsep Dasar Basisdata

Basisdata merujuk pada kumpulan file yang berelasi, terorganisasi secara konsisten dan berkualitas tinggi yang disertakan dalam media elektronik untuk di manfaatkan Bersama-sama. Relasi tersebut dapat diterapkan pada setiap file yang ada. Basisdata tunggal menyajikan kumpulan data dari sebuah bisnis atau lembaga. Salah satu faktor yang sangat penting dalam penambangan data adalah basisdata. Karena itu, adalah mungkin untuk memahami bagaimana data diperoleh, diakses, ditangani, dan dikendalikan. Ada bidang dan catatan dalam satu file basisdata. Satu catatan terdiri dari data bidang demi bidang yang secara konsisten bekerja sama untuk menunjukkan bahwa setiap bidang merupakan bagian dari studi komprehensif yang disertakan dalam satu catatan...

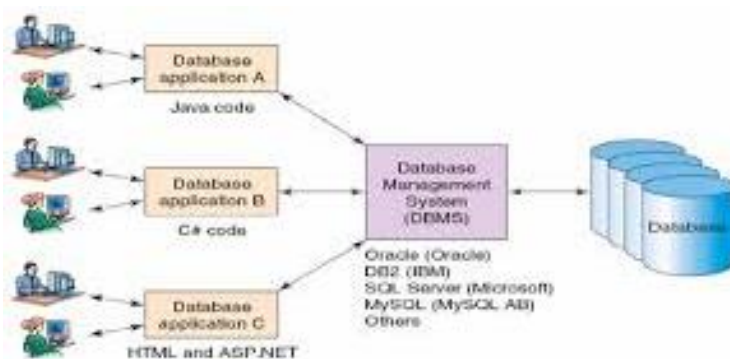
Istilah "Batabase" berasal dari kata "Basis" dan "Data," yang memiliki makna sebagai berikut: Basis: dapat dipahami sebagai tempat berkumpul atau bersarang, serta kantor pusat atau gudang. Data adalah representasi fakta aktual yang dapat berbentuk teks, angka, karakter, simbol, foto, suara, atau kombinasi dari semuanya. Contoh fakta dunia nyata ini meliputi orang (pekerja, mahasiswa, pembeli, pelanggan), objek, hewan, peristiwa, konsep, keadaan, dan sebagainya. Berdasarkan kedua interpretasi ini, dapat disimpulkan bahwa basis data adalah sekelompok catatan atau tabel terkait yang disimpan di media penyimpanan elektronik. Basis data dapat dianggap sebagai gudang digital atau lokasi untuk data yang

dapat digabungkan untuk mencapai tujuan..

Manfaat dibentuknya basis data pada sebuah organisasi adalah kecepatan dan kemudahan dalam proses pengambilan data. Pada gambar dibawah menunjukkan perbedaan antara penyimpanan data secara manual dan penyimpanan data secara digital. Masalah yang dihadapi lemari arsip adalah lambatnya dalam mencari data yang ada pada lemari arsip. Sedangkan pada basis data, seluruh data tersimpan secara digital dan tersimpan berdasarkan file masing-masing serta terpusat pada satu sistem, sehingga sangat cepat dan mudah dalam menelusuri data. Hal ini akan mempercepat proses penyajian data dan informasi yang diinginkan.



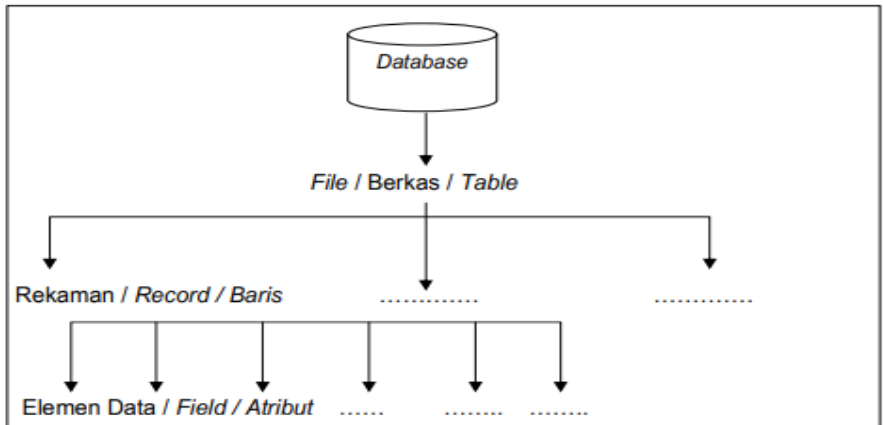
Gambar 5.1. Lemari arsip/ data manual



Gambar 5.2. Arsip/ data digital

5.2 Hirarki Database

Hirarki database adalah model data yang menyimpan data dalam bentuk catatan dan mengorganisasikannya dalam struktur seperti pohon, atau struktur induk-anak. Dalam struktur ini, satu simpul induk dapat memiliki banyak simpul anak yang terhubung melalui tautan.



Gambar 5.3. Hirarki Database

Berdasarkan fungsi dan manfaatnya database memiliki tingkatan data yang dapat disusun kedalam sebuah hirarkisebagai berikut :

1. Basisdata adalah kumpulan berbagai jenis catatan dengan hubungan di antara data.
2. Berkas adalah kumpulan catatan informasi yang berkaitan dengan suatu item.
3. Catatan adalah sekelompok bidang dan karakteristik yang saling terkait untuk suatu objek tertentu dengan panjang yang ditetapkan.
4. Unit data terkecil, yang dikenal sebagai bidang atau atribut, tidak dapat diubah.
5. Bagian terkecil yang dapat dialamatkan dalam memori adalah byte. Byte adalah sekelompok bit yang, dalam

kebanyakan kasus, terdiri dari delapan bit berbeda yang bersama-sama mewakili karakter dalam memori (1 byte = 1 karakter).

- 6. Bit sistem biner terdiri dari dua jenis nilai: 0 dan 1. Fondasi untuk komunikasi manusia-mesin adalah sistem bilangan biner, yang.

Berikut ini ilustrasi dari dari penjelasan file, *field*, dan *record*/isi data:



5.3 Komponen Basisdata

Sistem basis data adalah pengelompokan catatan atau tabel terkait dalam basis data sistem komputer, serta pengelompokan aplikasi (DBMS/Sistem Manajemen Basis Data). seperti : Ms.SQL Server, PostgreSQL, MYSQL, Oracle dll.). komponen utama basisdata sebagai berikut :

- 1. Perangkat Keras

Bagian utama penyimpanan basis data pada disk drive server sebagai media penyimpanan adalah perangkat keras.

- 2. Sistem Operasi

Sistem operasi, yang menggerakkan komputer klien dan

server, diibaratkan sebagai jiwa dari sebuah komputer. Sistem operasi mengendalikan semua sumber daya sistem dan menjalankan tugas-tugas rutin seperti manajemen berkas. Jika sistem operasi yang dipilih sesuai, program manajemen basis data akan aktif atau berfungsi. Berdasarkan fungsinya, sistem operasi dapat dibagi menjadi dua kategori: sistem operasi klien dan sistem operasi server. Sistem operasi komputer hadir dalam berbagai bentuk, seperti Microsoft Windows, Linux, Mac OS, dan Unix.

3. Basisdata

Basis data adalah pengelompokan data yang telah disusun sedemikian rupa sehingga mudah disimpan dan digunakan. Beberapa basis data dapat digunakan dalam sistem basis data. Setiap basis data dapat berisi file, rekaman, karakteristik, dsb.

4. Sistem Manajemen Basis Data, atau DBMS

Berbagai aplikasi yang digunakan untuk pembuatan dan pengelolaan basis data disebut sistem manajemen basis data. Sistem manajemen basis data (DBMS) terdiri dari sekelompok aplikasi dan kumpulan data. Perangkat lunak yang disebut sistem manajemen basis data (DBMS) mengontrol bagaimana data disusun, disimpan, dimodifikasi, dan diambil. Selain itu, konsistensi data, pengguna data bersama, dan langkah-langkah keamanan data diterapkan oleh perangkat lunak ini. Microsoft Access, SQL Server, MySQL atau MySQLi, DB2, Oracle, PostgreSQL, SQLite, Mongo DB, dan Maria DB adalah beberapa program yang merupakan bagian dari DBMS..

5. Pengguna

Ada berbagai tingkatan pengguna dalam basis data, mulai dari pengguna pemula atau pengguna dasar hingga pengguna tingkat lanjut, seperti:

- a. Pengguna yang melakukan pemrograman
Pengguna yang menggunakan DML (Data Manipulation Language) untuk berinteraksi dengan basis data dikenal sebagai pemrogram aplikasi, sedangkan pemrogram membuat aplikasi menggunakan berbagai bahasa pemrograman, seperti Pascal, Cobol, Basic, C++, PHP, JavaScript, Android, dan Python.
- b. Pengguna Lanjutan:
Mereka yang ahli dalam menangani sistem manajemen basis data, seperti MySQL, Oracle, dan sebagainya, dianggap sebagai pengguna tingkat lanjut.
- c. Pengguna Umum:
Bergantung pada tanggung jawab dan wewenangnya, pengguna ini bertugas mengawasi aplikasi yang telah selesai dan melakukan operasi program.

5.4 Tujuan Basisdata

keberhasilan sebuah basisdata tergantung bagaimana perusahaan mampu mengaplikasikan program database dimana pada saat proses pencarian data bisa dengan cepat dan akurat. Hal ini berkaitan dengan penggunaan data untuk kebutuhan analisis dan dukungan keputusan, serta dengan operasi dan administrasi data yang efisien.. sehingga mampu mengorganisasikan semua komponen database dengan baik sesuai harapan perusahaan, beberapa tujuan basisdata di antara :

1. Ketidakkonsistenan dan redundansi/duplikasi data
pada saat membangun sebuah program aplikasi, programmer akan membuat banyak tabel yang dimungkinkan terjadinya pengulangan data yang sama dan dalam waktu yang cukup lama maka akan ada bagian tabel yang mengalami penggandaan tabel pada suatu database. Contoh : nim, nama, tanggal lahir, alamat dll di

tabel mahasiswa pada database sistem informasi akademik, dan itu bisa terjadi pada saat membuat KRS dan juga pada Keuangan. Satu tabel mahasiswa dapat digunakan oleh beberapa sub-basis data yang membutuhkannya dalam sistem berbasis jaringan.

Redundansi adalah praktik menyimpan data yang sama di beberapa lokasi, yang membuang-buang kapasitas penyimpanan dan meningkatkan biaya akses. Ketidakkonsistenan juga dapat terjadi dengan terus-menerus menyimpan data yang sama di banyak lokasi dalam basis data. Hal ini terjadi karena semua tabel yang menyimpan data perlu diperbarui jika mahasiswa pindah ke alamat baru. Hal ini menjadi tidak konsisten jika hanya satu yang diperbarui..

2. Akses terhadap data sulit

pada saat kampus ingin mengecek data mahasiswa berdasarkan kota tertentu, misalnya data mahasiswa yang berada di kota "Semarang". maka kesulitan akan muncul, dengan demikian perlu adanya penyelesaian yang cepat dan mudah untuk mengerjakannya. dengan DBMS seperti : Mysql, Oracle dll mampu menyelesaikan persoalan yang terjadi dengan baik dan mudah.

3. Standarisasi

jika terjadi beberapa tabel yang tersebar dalam sebuah database dengan format tabel dan type data yang berbeda akan sangat menyulitkan dalam menuliskan program aplikasi untuk mengambil dan menyimpan data. maka perlu adanya standarisasi atau kesamaan data yang dibangun oleh programmer.

4. Banyak Pengguna

membangun database dalam sebuah organisasi/ perusahaan memiliki tujuan untuk mempercepat dan mempermudah dalam proses pengambilan data atau

proses perubahan data, akan tetapi yang lebih penting lagi bagaimana penggunaan database dapat dimanfaatkan bersama-sama dalam waktu yang bersama dengan tempat yang berbeda tanpa mengenal batas waktu. maka perlu membangun sebuah aplikasi yang dapat dimanfaatkan secara bersamaan seperti contoh : sistem informasi akademik, sistem informasi resevasi tiket pesawat/ kereta/ hotel, media sosial, marketplace dll

5. Kekhawatiran tentang keamanan

Dalam membangun sebuah sistem informasi yang baik perlu adanya keamanan sistem yang terintegrasi sehingga pengguna merasa nyaman dan aman, maka perlu adanya pembatasan hak akses pengguna, misal pengguna sebagai bagian kepegawaian tidak boleh mengakses bagian keuangan, mahasiswa tidak boleh mengakses data dosen, bagian gudang tidak boleh mengakses bagian kasir dll, hal tersebut untuk membatasi fungsi dan wewenangnya sebagai pengguna.

6. Masalah integrasi

Database memiliki banyak file dan tabel yang berbeda akan tetapi bagaimana file dan tabel tersebut dapat terhubung dan terkoneksi satu dengan yang lainnya dengan baik tanpa adanya kesalahan untuk mempercepat proses pencarian dan proses update data, misalnya : tabel mahasiswa dapat dipakai pada tabel dosen, tabel nilai dan tabel-tabel yang lain.

7. Kekhawatiran tentang independensi data

Setiap kali kita perlu melihat data, kita cukup menggunakan daftar utilitas, menambahkan data menggunakan Append (misalnya, DBMS Foxpro, Dbase), mengubah struktur tabel menggunakan Design Table, atau melakukan pencarian data menggunakan query (misalnya, Access, SQL Server, MySql, atau Oracle). Ini

menunjukkan bahwa operasi paket DBMS bersifat independen dari database. Tidak peduli bagaimana database berubah, semua perintah akan tetap stabil dan tidak terpengaruh.

5.5 Manfaat Basisdata

Setelah kita mempelajari konsep dan tujuan serta jenis-jenis dari database itu sendiri, dengan demikian perlu kita pahami manfaat dari pengguna database. maka apa saja manfaat database bagi pengguna sistem informasi.

1. Menjamin sistem keamanan data

Seorang pengguna database merasa nyaman dan aman jika pada saat menggunakan sistem database tidak ada keraguan dalam menggunakan terutama pada saat menjalankannya, dengan memberikan hak akses pada masing-masing pengguna sesuai dengan jobdescription maka pengguna lain tidak bisa mengakses data yang ada pada sistem tersebut, hal ini kan memberikan kenyamanan dan keamanan.

2. Akses banyak pengguna

Sistem database merupakan sistem yang terintegrasi yang memungkinkan banyak pengguna bisa melakukan aktivitas secara bersama-sama dengan waktu yang sama dan tempat yang berbeda.

3. Akses data cepat

Database mampu memilah/ menyeleksi data dengan cepat dan akurat secara sistematis dengan banyak pengguna secara bersamaan,

4. Menghemat Biaya Perangkat

Sistem basisdata mampu memberikan akses cepat, mudah dimana saja pada perusahaan yang ingin membangun cabang perusahaan, hal ini membutuhkan biaya investasi yang sangat besar dan butuh waktu yang lama, dengan

memanfaatkan sistem basisdata perusahaan hanya cukup membangun membuat webserver dengan koneksi internet, data dapat dimanfaatkan secara bersama-sama sehingga mengurangi biaya investasi untuk pengembangan yang lain.

5. Pengendalian data

Sistem basisdata mampu menyediakan layanan untuk mengendalikan data cepat dan mudah, dengan membangun webserver perusahaan bisa dengan mudah mengendalikan data secara cepat dan mudah.

5.6 Normalisasi

Normalisasi database adalah proses mengelompokkan tabel umum menjadi tabel-tabel yang lebih sederhana, mudah dalam proses pengelompokannya. sehingga diperoleh data yang sangat berkualitas, normalisasi merupakan pendekatan yang logis dalam membangun desain database yang rasional dengan menerapkan beberapa aturan-aturan yang diberlakukan untuk memperoleh data yang lebih sederhana, dalam Normalisasi dikenal dengan Bentuk Normal 1 (1NF), Bentuk Normal Ke 2 (2NF), Bentuk Normal Ke 3 (3NF). untuk mengurangi redudansi perlu adanya normalisasi..

Tujuan Normalisasi dibangun adalah untuk mengurangi dan menghilangkan adanya redudansi yang ada pada tabel database sehingga tercipta tabel yang sangat sederhana, jika pada tabel belum dinormalisasi maka kemungkinan terburuknya adanya anomali (perubahan yang tidak diinginkan) yang dapat merugikan sistem, anomali tersebut antara lain :

1. Anomai Insert

keadaan dimana tidak diharapkan adanya beberapa data dimasukan langsung karena akan mempengaruhi tabel yang lain. contoh : data tabel yang di inputkan adalah

primary key sehingga ketika dimasukan akan mempengaruhi data yang lain

2. Anomali Update

keadaan di mana nilai yang dimodifikasi menyebabkan ketidakkonsistenan basis data, yang menunjukkan bahwa data yang dimodifikasi bukanlah yang dimaksudkan. Adanya data yang berlebihan mengakibatkan anomali ini..

3. Anomali Delete

Proses penghapusan data yang tidak diharapkan, harusnya data yang tidak dihapus akan ikut terhapus.

Berikut ini merupakan contoh penerapan normalisasi, pada struk penjualan table belum normal.

kode_faktur	tanggal	kode_barang	nama_barang	harga	qty
KD_001	13/07/2020	BRG_001	Indomie Goreng	2500	5
		BRG_002	Indomie Goreng Jumbo	3000	8
		BRG_003	Mie Sedaap Goreng	2500	9
		BRG_004	Mie Sedaap Soto	2300	3
KD_002	13/07/2020	BRG_005	Intermie Goreng	1500	6
		BRG_006	Intermie Soto	1500	6
KD_003	14/07/2020	BRG_007	Pop Mie Ayam	4500	3

Pada contoh struk penjualan merupakan data yang belum di normalisasi pada tahap peratama 1NF. Dengan demikian, bentuk normal data adalah 1NF jika dan hanya jika setiap atribut berisi satu nilai dalam satu baris .seperti contoh berikut :

kode_faktur	tanggal	kode_barang	nama_barang	harga	qty
KD_001	13/07/2020	BRG_001	Indomie Goreng	2500	5
KD_001	13/07/2020	BRG_002	Indomie Goreng Jumbo	3000	8
KD_001	13/07/2020	BRG_003	Mie Sedaap Goreng	2500	9
KD_001	13/07/2020	BRG_004	Mie Sedaap Soto	2300	3
KD_002	13/07/2020	BRG_005	Intermie Goreng	1500	6
KD_002	13/07/2020	BRG_006	Intermie Soto	1500	6
KD_003	14/07/2020	BRG_007	Pop Mie Ayam	4500	3

Normalisasi 1NF adalah dimana tidak boleh ada

duplikasi data pada tabel. sedangkan pada 2NF adalah tidak diperbolehkan adanya fungsional dependensi yaitu. Atribut apa pun yang secara efektif bergantung pada kunci utama tetapi bukan kunci dikatakan bergantung secara fungsional. Hal yang penting adalah bahwa tabel perlu dipartisi menggunakan kunci utama pada titik ini dalam proses normalisasi 2NF. Dengan demikian, bentuk normalisasi 2NF tabel adalah sebagai berikut. :

Tabel Barang			Tabel Transaksi			
kode_barang	nama_barang	harga	kode_faktur	tanggal	kode_barang	qty
BRG_001	Indomie Goreng	2500	KD_001	13/07/2020	BRG_001	5
BRG_002	Indomie Goreng Jumbo	3000	KD_001	13/07/2020	BRG_002	8
BRG_003	Mie Sedaap Goreng	2500	KD_001	13/07/2020	BRG_003	9
BRG_004	Mie Sedaap Soto	2300	KD_001	13/07/2020	BRG_004	3
BRG_005	Intermie Goreng	1500	KD_002	13/07/2020	BRG_005	6
BRG_006	Intermie Soto	1500	KD_002	13/07/2020	BRG_006	6
BRG_007	Pop Mie Ayam	4500	KD_003	14/07/2020	BRG_007	3

Data parsial dan dependensi transitif dilarang dalam tabel 3NF. Bila terdapat tiga atribut—A, B, dan C—dalam suatu kondisi atau tabel hasil relasi, dependensi transitif dapat muncul. $A \rightarrow B$ dan $B \Rightarrow C$ adalah kondisinya. Setelah itu, C dianggap sebagai ketergantungan transitif pada A melalui B.

5.7 Normalisasi pada Entity Relationship Diagram

Diagram penghubungan Entitas, atau ERD, adalah jenis desain basis data yang menggunakan diagram hubungan untuk memudahkan penyajian data yang memiliki hubungan dalam bentuk diagram. Desainer basis data merasa lebih mudah menyajikan data menggunakan Diagram Hubungan Entitas, yang membuat proses pengembangan jauh lebih sederhana dan lebih menarik.

Jenis model data yang akan digunakan harus diputuskan sebelum kita dapat membangun sistem basis data yang sesuai. sedangkan pengembangan sistem sesuai dengan

persyaratan proses bisnis akan sangat terpengaruh oleh hal ini. Semua jenis arsitektur data dalam suatu organisasi dapat didokumentasikan dengan baik dengan penggunaan Diagram Hubungan Entitas konseptual. Satu atau beberapa jenis model data logis dapat diterapkan pada model ini. Membangun struktur metadata untuk data induk entitas dan set model ER logis adalah tujuan pengembangannya.

Penyimpanan data dapat ditampilkan dengan menghubungkan ERD ke Diagram Alir Data (DFD). Tujuannya adalah untuk menggambarkan interkoneksi dan kemampuan konstruksi data relasional dari proses data. Penjelasan tentang kata-kata dan elemen dalam ERD dapat ditemukan di bawah ini.

1. Entitas

Entitas adalah sesuatu yang ada dengan cara yang berbeda dan unik. Entitas direpresentasikan oleh persegi panjang dalam ERD. Selanjutnya, persegi panjang kecil di dalam persegi panjang besar akan merepresentasikan entitas lemah. Entitas lemah dimaksudkan untuk merepresentasikan entitas yang terhubung dengan entitas lain. Karyawan, dosen, mahasiswa, guru, karyawan, barang, dan lain-lain adalah beberapa contoh entitas.

2. Atribut

Karena setiap entitas memiliki atributnya sendiri yang berfungsi untuk menyampaikan detail tentang atribut entitas tersebut. Karena simbol elips menunjukkan atribut yang krusial, maka atribut tersebut berfungsi sebagai pembeda antara entitas dan atribut yang mana. Misalnya, properti entitas mahasiswa meliputi NIM, nama, tempat lahir, tanggal lahir, alamat, dan jenis kelamin; atribut entitas dosen meliputi NIDN, nama dosen, alamat, dan jenis kelamin; dan atribut entitas barang meliputi ID

barang, kode barang, nama barang, dan jenis barang. Kualitas tersebut terbagi dalam beberapa jenis berikut:

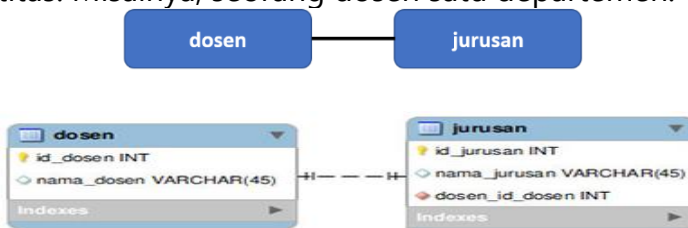
- a. Atribut utama adalah atribut yang menonjol dari yang lain atau memiliki kualitas khusus. Data atribut utama sering kali berbentuk nilai numerik. NIM (Nomor Induk Mahasiswa), No. KTP, SIM, NPWP, dan sebagainya adalah beberapa contohnya.
- b. Atribut sederhana, juga dikenal sebagai atribut atomik, adalah atribut dengan nilai tunggal yang tidak dapat dipecah lebih lanjut. Contohnya termasuk alamat tempat tinggal, tempat usaha, nama penerbit, tahun penerbitan jurnal, dll.
- c. Atribut bernilai banyak adalah atribut yang diberi nilai untuk setiap entitas. Daftar nama penulis dalam sebuah novel adalah contoh atribut bernilai banyak..
- d. Atribut Gabungan (*Composite*), atribut yang memiliki susunan paling kecil pada atribut tertentu. Contohnya adalah data terkait nama lengkap, yang terdiri dari nama depan, tengah, dan belakang, alamat yang terdiri dari jalan, kecamatan, kota desa dll

3. Relasi

Relasi adalah hubungan antara beberapa tipe entitas yang terdapat dalam berbagai himpunan entitas. Bentuk relasi digambarkan dengan diagram belah ketupat. Ada berbagai macam relasi, seperti::

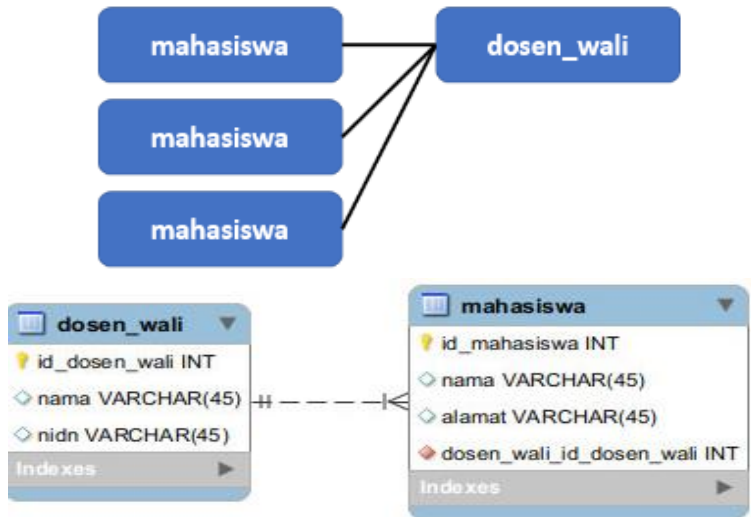
a. One to One

Ini berarti bahwa hanya ada satu hubungan antara dua entitas. Misalnya, seorang dosen satu departemen.



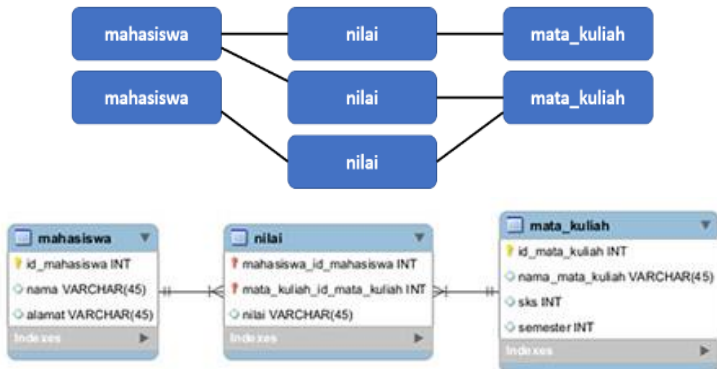
b. One to Many

Ini adalah hubungan antara satu hal dengan banyak entitas lain, begitu juga sebaliknya. Misalnya, seorang dosen dapat membantu banyak mahasiswa.



c. Many to Many




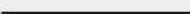
Ini adalah hubungan dengan beberapa hubungan di antara banyak entitas. Misalnya, seorang mahasiswa dapat mendaftar di beberapa mata kuliah, dan beberapa siswa dapat mendaftar di satu mata kuliah.



4. Garis

Garis mewakili hubungan entitas antara atribut.. Saat ini ada beberapa cara untuk membuat ERD, dengan beberapa aplikasi baik offline maupun online antara lain : Draw.io, Dbdiagram.io, ucidchart, QuickDBD, SQLDBM dan masih banyak lagi.

Tabel 5.1. Diagram ERD

Notasi	Makna
 <p>ENTITAS</p>	Entitas adalah Suatu objek yang dapat diidentifikasi dalam lingkungan pengguna. Entitas dapat berupa: -Sebuah elemen lingkungan pada perusahaan -Suatu sumber daya -Suatu arus informasi
 <p>RELASI</p>	Relasi menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.
 <p>ATRIBUT</p>	Atribut berfungsi untuk memberikan gambaran karakteristik dari suatu entitas. Atribut yang berfungsi sebagai Key diberi garis bawah.
	Garis berperan sebagai penghubung antara atribut dengan entitas dan kumpulan entitas dengan relasi.

5.8 Structured Query Language (SQL)

Salah satu DBMS dengan bahasa basisdata terstruktur seperti *Structured Query Language* (SQL). Informasi disimpan dalam basis data relasional sebagai tabel, di mana properti data yang berbeda dan hubungan antara nilai data direpresentasikan oleh baris dan kolom. Informasi dapat disimpan, diperbarui, dihapus, dicari, dan diambil dari basis data menggunakan perintah SQL. SQL juga dapat digunakan untuk mengoptimalkan dan memelihara kinerja basis data. Secara umum, manajer basis data menggunakan tiga perintah SQL mendasar: DDL (*Data Definition Language*), DML (*Data Manipulation Language*), dan DQL (*Data Query Language*).

Berikut ini adalah beberapa jenis perintah SQL yang dapat digunakan secara maksimal.

1. *Data Manipulation Language* (DML)

Salah satu jenis perintah terpenting yang bertanggung jawab atas semua modifikasi basis data adalah perintah bahasa SQL ini, yang berguna untuk membuat perubahan pada basis data. Perintah DML meliputi INSERT, UPDATE, dan DELETE.

2. *Data Definition Language* (DDL)

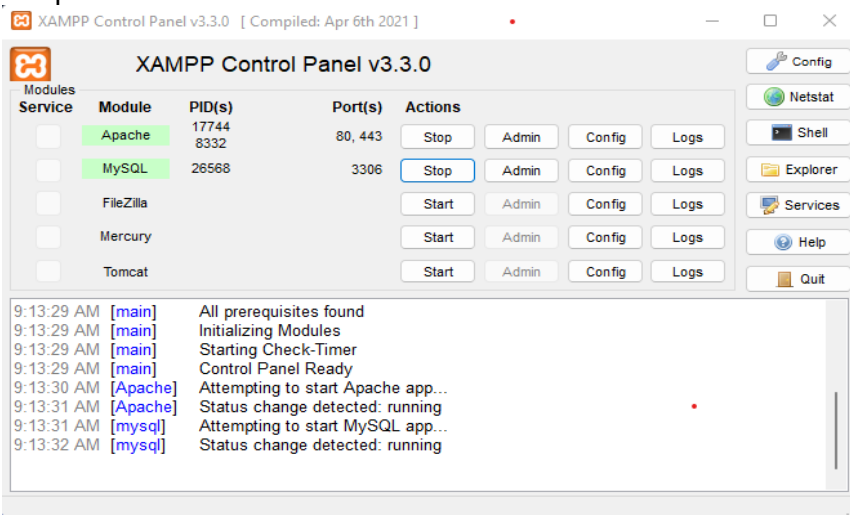
Perintah bahasa SQL ini mendefinisikan skema basis data tertentu yang dapat dengan mudah diubah melalui struktur objek basis data. Menggunakan perintah seperti CREATE dan DROP untuk menyimpan semua perusahaan dalam basis data juga bermanfaat.

3. *Data Control Language* (DCL)

Jenis perintah bahasa SQL ini memiliki fungsi untuk memberikan kontrol terhadap berbagai hal seperti izin hingga hak dari sebuah sistem *database*. Contoh dari jenis perintah ini seperti GRANT dan REVOKE

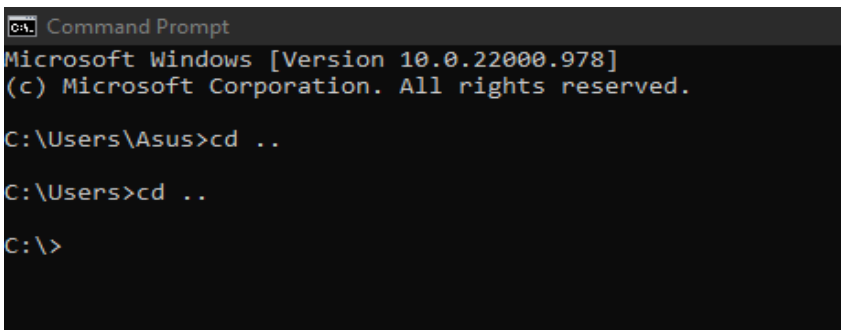
Langkah-langka mengoperasikan SQL, Pastikan aplikasi Xampp sudah terinstal dikomputer anda , selanjutnya ikuti Langkah-langkah sebagai berikut :

Langkah 1: Verifikasi bahwa server MySQL aktif dan beroperasi.



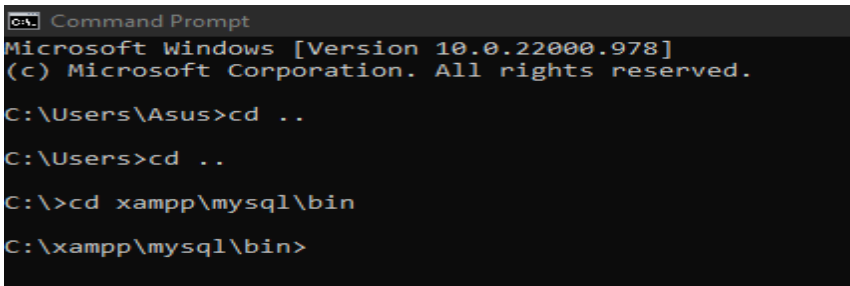
Gambar 5.4. Tampilan XAMPP

Langkah 2: Luncurkan program prompt perintah, masukkan "cd\" dan tekan "Enter" untuk menavigasi ke direktori C:\..\



Gambar 5.5. Tampilan Window Command Prompt

Langkah 3: Selanjutnya, akses folder "C:\xampp\mysql\bin>",



```
Command Prompt
Microsoft Windows [Version 10.0.22000.978]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Asus>cd ..

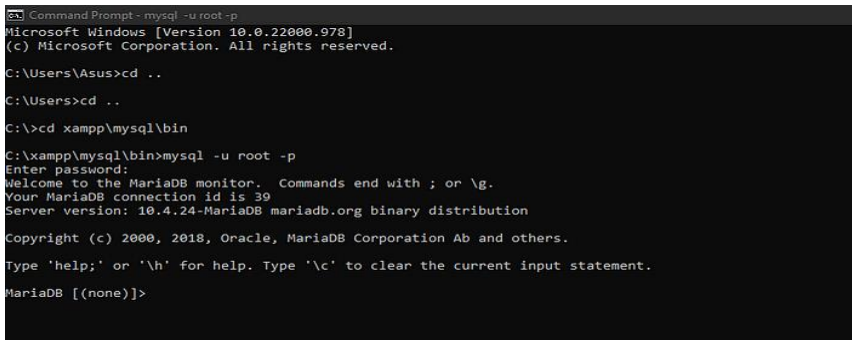
C:\Users>cd ..

C:\>cd xampp\mysql\bin

C:\xampp\mysql\bin>
```

Gambar 5.6. Arahkan ke directory C:\xampp\mysql\bin

Langkah 4: Masukkan mysql setelah Anda membuka folder "C:\xampp\mysql\bin>". "Mysql -u root -p" adalah tipenya. Tekan tombol Enter. Jika ada kata sandi, masukkan di sini lalu tekan tombol Enter sekali lagi (kata sandi root kosong secara default).



```
Command Prompt - mysql -u root -p
Microsoft Windows [Version 10.0.22000.978]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Asus>cd ..

C:\Users>cd ..

C:\>cd xampp\mysql\bin

C:\xampp\mysql\bin>mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 39
Server version: 10.4.24-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Gambar 5.7. Login Ke MySQL

Langkah 5: Anda telah berhasil masuk ke MySQL sebagai pengguna root saat layar tampak seperti gambar yang menyertai. Anda dapat mengetik 'show databases' untuk melihat database di server. Ingatlah untuk selalu mengakhiri perintah dengan ":".

```
Command Prompt - mysql -u root -p
Microsoft Windows [Version 10.0.22000.978]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Asus>cd ..
C:\Users>cd ..
C:\>cd xampp\mysql\bin
C:\xampp\mysql\bin>mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 39
Server version: 10.4.24-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| coba_many_to_many |
| cobaanime |
| information_schema |
| mysql |
| perbankan |
| perbankanbaru |
| performance_schema |
| phpmyadmin |
| test |
| wordpress |
+-----+
10 rows in set (0.001 sec)

MariaDB [(none)]>
```

Gambar 5.8. Menampilkan Database yang ada di Server

DAFTAR PUSTAKA

- Ardie, 2004. *Manajemen Database dan MySql*. Penerbit Andi : Yogyakarta.
- Jogiyanto, H,M. 2001, Analisis dan Desain Sistem Infomasi, Edisi kedua. Penerbit Andi : Yogyakarta.
- Kristianto, H. 2003. *Konsep dan Perancangan Database*. Andi : Yogyakarta.
- Sutanto, E. 2003. *Sistem Basis Data*. Graha Ilmu : Yogyakarta

BAB 6

TREN DAN MASA DEPAN TEKNOLOGI KOMPUTER

Oleh Mepa Kurniasih

Teknologi komputer berkembang dengan sangat cepat, menghadirkan peluang dan tantangan baru. Tren masa depan di bidang ini mencakup berbagai inovasi yang akan membentuk cara kita menggunakan komputer dalam kehidupan sehari-hari. Bab ini akan mengulas beberapa tren utama yang diprediksi akan mendominasi masa depan teknologi komputer.

6.1 Kecerdasan Buatan dan Pembelajaran Mesin

Kecerdasan buatan (*Artificial Intelligence, AI*) dan pembelajaran mesin (*Machine Learning, ML*) adalah dua konsep yang saling terkait dalam ilmu komputer. Berikut adalah penjelasan lebih lanjut tentang keduanya:

Kecerdasan Buatan (*Artificial Intelligence*)

Kecerdasan buatan adalah simulasi dari kecerdasan yang dimiliki oleh manusia yang dimodelkan di dalam mesin dan diprogram agar bisa berpikir seperti halnya manusia. AI dapat melakukan tugas-tugas yang umumnya membutuhkan kecerdasan atau intervensi manusia, seperti mengenali suara, wajah, dan teks, serta membuat keputusan berdasarkan input data waktu nyata (IBM, 2023; Wikipedia, 2023).

AI digunakan dalam berbagai bidang, termasuk ekonomi, sains, obat-obatan, teknik, dan militer. Contoh aplikasi AI meliputi sistem pakar, permainan komputer, logika

kabur, jaringan saraf tiruan, dan robotika.

Pembelajaran Mesin (*Machine Learning*)

Pembelajaran mesin adalah suatu cabang kecerdasan buatan yang memungkinkan mesin untuk belajar dari data dan membuat prediksi atau klasifikasi berdasarkan pola yang ditemukan dalam data tersebut. ML menggunakan algoritma untuk mengidentifikasi pola dan membuat model yang dapat digunakan untuk membuat prediksi atau klasifikasi.

Proses pembelajaran mesin melibatkan beberapa langkah utama:

1. Data Koleksi: Mengumpulkan data yang relevan untuk model.
2. Data Preprocessing: Mengolah data untuk memastikan kualitas dan konsistensi.
3. Model Pembelajaran: Menggunakan algoritma ML untuk membuat model yang dapat belajar dari data.
4. Evaluasi Model: Menguji model untuk memastikan akurasi dan keefektifannya.
5. Penggunaan Model: Menggunakan model yang telah dibuat untuk membuat prediksi atau klasifikasi baru.

Hubungan Kecerdasan Buatan dan Pembelajaran Mesin

Pembelajaran Mesin sebagai Bagian dari Kecerdasan Buatan:

Pembelajaran mesin adalah salah satu teknologi yang digunakan dalam kecerdasan buatan untuk membuat sistem yang dapat belajar dan beradaptasi dengan lingkungan sekitar.

AI seringkali menggunakan ML untuk mengembangkan kemampuan belajar dan adaptasi yang cerdas dalam sebuah mesin.

Contoh aplikasi yang menggunakan kombinasi keduanya adalah pengenalan objek/muka, permainan catur, dan analitik

bisnis. AI dapat menggunakan ML untuk mengenali pola dan membuat keputusan berdasarkan data yang diperoleh.

Dengan demikian, kecerdasan buatan dan pembelajaran mesin adalah dua konsep yang saling terkait dalam pengembangan sistem yang cerdas dan beradaptasi. Pembelajaran mesin adalah salah satu teknologi yang digunakan dalam kecerdasan buatan untuk membuat sistem yang dapat belajar dan beradaptasi dengan lingkungan sekitar.

Pembelajaran mesin (*Machine Learning*, ML) adalah salah satu cabang ilmu dari kecerdasan buatan (*Artificial Intelligence*, AI) yang memungkinkan mesin untuk belajar dari data dan membuat prediksi atau klasifikasi berdasarkan pola yang ditemukan dalam data tersebut. Berikut adalah cara kerja pembelajaran mesin dalam kecerdasan buatan:

Cara Kerja Pembelajaran Mesin

1. Pengumpulan Data:

Data Relevan: Mengumpulkan data yang relevan dengan masalah yang ingin dipecahkan. Data ini dapat berupa teks, gambar, suara, atau data numerik lainnya (Patria, 2023; Verihubs, 2023).

2. Pengolahan Data:

Data Preprocessing: Mengolah data untuk memastikan kualitas dan konsistensi. Ini termasuk menghilangkan data yang tidak relevan, mengubah format data, dan menghilangkan noise[2][3].

3. Pemilihan Model:

Teknik Pembelajaran: Memilih teknik pembelajaran yang sesuai, seperti supervised learning, unsupervised learning, atau reinforcement learning. Supervised learning melibatkan model yang dilatih dengan data yang telah ditandai, sedangkan unsupervised learning melibatkan

model yang dilatih dengan data yang tidak ditandai.

4. Pelatihan Model:

Pelatihan Model: Memberikan pelatihan terhadap model yang dipilih menggunakan data yang telah diproses. Model ini belajar dari pola-pola dalam data untuk membuat prediksi atau klasifikasi.

5. Evaluasi Model:

Mengukur Akurasi: Mengukur akurasi model dengan menggunakan metrik yang relevan seperti akurasi, presisi, recall, dan F1-score. Evaluasi ini membantu mengetahui seberapa baik model dalam membuat prediksi atau klasifikasi.

6. Implementasi dan Uji Coba:

Implementasi: Mengimplementasikan model yang telah dilatih ke dalam aplikasi atau sistem yang sesuai. Model ini dapat digunakan untuk membuat prediksi atau klasifikasi baru berdasarkan data yang diperoleh.

Contoh aplikasi pembelajaran mesin adalah merekomendasikan produk kepada pelanggan berdasarkan riwayat pembelian mereka. Model pembelajaran mesin menganalisis data pelanggan dan mengidentifikasi pola dalam perilaku pembelian mereka untuk memprediksi produk mana yang paling mungkin diminati pelanggan.

Contoh lainnya adalah deteksi penipuan di bank. Model pembelajaran mesin menganalisis riwayat transaksi pelanggan dan mencari pola yang mungkin menunjukkan aktivitas penipuan seperti pembelian dalam jumlah besar yang tidak biasa atau transaksi yang dilakukan dari lokasi berbeda.

Dengan demikian, pembelajaran mesin adalah teknologi yang mampu mempelajari data yang ada dan melakukan tugas-tugas tertentu sesuai dengan apa yang ia pelajari, sehingga sangat berguna dalam berbagai aplikasi

seperti rekomendasi produk, deteksi penipuan, pengenalan gambar, dan lain-lain.

Perbedaan antara supervised learning dan unsupervised learning dalam pembelajaran mesin (machine learning) adalah sebagai berikut (Chairunnisa, 2024; Revou, 2024; Trivusi, 2022):

Supervised Learning

Supervised learning adalah metode pembelajaran mesin di mana model dilatih menggunakan data yang telah ditandai atau berlabel. Tujuan utamanya adalah untuk membuat model yang dapat memprediksi output berdasarkan input data yang diberikan bersamaan dengan label yang tepat.

Cara Kerja:

Data Berlabel: Data input diberikan bersamaan dengan output yang benar.

Model Dilatih: Model dilatih untuk menemukan fungsi pemetaan antara variabel input (X) dengan variabel output (Y).

Umpan Balik: Model menerima umpan balik langsung untuk memeriksa apakah menghasilkan output yang benar atau tidak.

Manfaat:

Akurasi Tinggi: Model supervised learning dapat menghasilkan hasil yang akurat dan dapat dipercaya.

Kendali Penuh: Pengguna memiliki kendali penuh untuk menentukan jumlah kelas yang diinginkan dalam data yang dilatih.

Aplikasi: Cocok untuk mendeteksi penipuan, prakiraan cuaca, dan prediksi harga.

Unsupervised Learning

Unsupervised learning adalah metode pembelajaran mesin di mana model dilatih menggunakan data yang tidak berlabel. Tujuan utamanya adalah untuk menemukan struktur dan pola

dari data input tanpa intervensi manusia.

Cara Kerja:

Data Tidak Berlabel: Hanya data input yang diberikan tanpa label.

Model Belajar Mandiri: Model dirancang untuk bisa “belajar mandiri” dalam mengumpulkan informasi dan menemukan pola tersembunyi dalam data.

Tidak Mengambil Umpan Balik: Model tidak mengambil umpan balik karena tidak ada label yang diberikan.

Manfaat:

Pola Tersembunyi: Model unsupervised learning dapat menemukan pola tersembunyi dalam data.

Analisis Asosiasi: Digunakan untuk menemukan asosiasi antara variabel dalam dataset besar.

Aplikasi: Ideal untuk mendeteksi anomali, buyer persona, dan medical imaging.

Dengan demikian, perbedaan antara supervised learning dan unsupervised learning terletak pada cara kerja dan tujuan mereka. Supervised learning digunakan untuk memprediksi output berdasarkan data yang telah ditandai, sedangkan unsupervised learning digunakan untuk menemukan pola tersembunyi dalam data tanpa label (Laraswati, 2022).

6.2 Komputasi Kuantum

Komputasi kuantum adalah salah satu area yang sedang berkembang dan diprediksi akan mengubah lanskap komputasi. Komputasi kuantum adalah jenis komputasi yang memanfaatkan fenomena mekanika kuantum untuk memecahkan masalah yang terlalu rumit untuk komputer klasik. Berikut adalah penjelasan lebih lanjut tentang komputasi kuantum:

Pengertian Komputasi Kuantum

Komputasi kuantum menggunakan prinsip-prinsip mekanika kuantum, seperti superposisi, interferensi, dan keterikatan, untuk melakukan operasi komputasi yang lebih cepat dan efisien daripada komputer klasik. Komputer kuantum dapat memanfaatkan qubit (bit kuantum) yang dapat berada dalam superposisi, memungkinkan mereka menjadi 1 dan 0 pada waktu yang sama, sehingga memungkinkan komputer kuantum untuk menentukan jawaban yang benar secara simultan dan lebih efisien (Wikipedia, 2022).

Perangkat Keras dan Perangkat Lunak

Komputer kuantum memiliki perangkat keras dan perangkat lunak yang mirip dengan komputer klasik. Perangkat keras kuantum melibatkan qubit yang dapat disimpan dan dimanipulasi menggunakan teknologi seperti transmon, perangkap ion, dan komputer kuantum topologi. Perangkat lunak komputer kuantum melibatkan algoritma kuantum yang dirancang untuk memanfaatkan sifat-sifat kuantum seperti superposisi dan interferensi kuantum.

Aplikasi Komputasi Kuantum

Komputasi kuantum memiliki berbagai aplikasi yang potensial, termasuk:

Optimasi: Algoritma komputasi kuantum efektif dalam menyelesaikan masalah optimisasi, seperti menemukan jalur terpendek atau mengoptimalkan alokasi sumber daya.

Kriptografi: Komputasi kuantum dapat memecahkan algoritma enkripsi klasik, memerlukan pengembangan teknik kriptografi baru yang tahan serangan kuantum.

Simulasi dan Pemodelan: Komputer kuantum dapat mensimulasikan sistem kompleks dengan lebih akurat,

bermanfaat dalam simulasi reaksi kimia, ilmu material, pemodelan iklim, dan penemuan obat.

Pengenalan Pola dan Machine Learning: Komputasi kuantum dapat menangani kumpulan data besar dan meningkatkan pengenalan pola, menghasilkan prediksi dan wawasan yang lebih akurat dalam bidang keuangan, perawatan kesehatan, dan kecerdasan buatan (Fuad, 2024).

Sejarah dan Perkembangan

Sejarah komputasi kuantum dimulai pada 1980-an dengan fisikawan seperti Richard Feynman, David Deutsch, dan Paul Benioff yang mengembangkan konsep dasar dalam komputasi kuantum. Pada 1994, Peter Shor mengusulkan algoritma Shor yang mampu memecahkan masalah faktorisasi secara efisien dengan komputasi kuantum, menunjukkan potensi besar komputasi kuantum dalam bidang kriptografi (Fuad, 2024).

Dengan demikian, komputasi kuantum adalah teknologi yang sangat berpotensi dalam memecahkan masalah kompleks yang tidak dapat diselesaikan oleh komputer klasik, dengan menggunakan prinsip-prinsip mekanika kuantum untuk meningkatkan kecepatan dan efisiensi operasi komputasi.

Komputasi kuantum didasarkan pada beberapa prinsip dasar mekanika kuantum yang memungkinkan perhitungan yang lebih cepat dan akurat daripada komputer klasik. Berikut adalah prinsip-prinsip dasar komputasi kuantum (Darmawan, 2023):

1. Superposisi

Superposisi adalah kemampuan qubit (bit kuantum) untuk berada dalam beberapa keadaan secara bersamaan. Dalam komputasi kuantum, qubit dapat berada dalam superposisi 0 dan 1, sehingga memungkinkan beberapa

perhitungan dilakukan secara bersamaan.

Implikasi: Superposisi memberikan paralelisme yang melekat pada komputer kuantum, memungkinkannya untuk memproses jutaan operasi secara bersamaan.

2. Keterikatan (*Entanglement*)

Keterikatan adalah fenomena di mana dua atau lebih partikel dapat berkorelasi sedemikian rupa sehingga keadaan kuantumnya saling terkait. Dalam komputasi kuantum, qubit yang terjerat dapat digunakan untuk melakukan perhitungan tertentu dengan lebih cepat daripada komputer klasik.

Implikasi: Keterikatan memungkinkan komputer kuantum untuk melakukan operasi yang lebih kompleks dengan lebih efisien.

3. Prinsip Ketidakpastian (*Uncertainty Principle*)

Prinsip ketidakpastian menyatakan bahwa tidak mungkin untuk mengetahui posisi dan momentum sebuah partikel dengan akurasi penuh. Prinsip ini memiliki implikasi penting untuk komputasi kuantum, karena ini berarti bahwa pengukuran pada qubit dapat mengubah keadaannya. Prinsip ketidakpastian membatasi kemampuan pengukuran pada qubit, sehingga memungkinkan komputer kuantum untuk melakukan perhitungan yang lebih kompleks dengan lebih efisien.

4. Dekohensi

Dekohensi adalah proses di mana keadaan kuantum qubit berubah menjadi keadaan klasik. Dekohensi dapat terjadi karena gangguan eksternal atau karena proses pengukuran. Dekohensi membatasi waktu hidup qubit dan memungkinkan komputer kuantum untuk melakukan perhitungan yang lebih cepat sebelum keadaan kuantum berubah menjadi klasik.

Dengan demikian, prinsip-prinsip dasar komputasi kuantum adalah superposisi, keterikatan, prinsip ketidakpastian, dan dekoherensi. Masing-masing prinsip ini memungkinkan komputer kuantum untuk melakukan perhitungan yang lebih cepat dan akurat daripada komputer klasik.

Komputasi kuantum memiliki berbagai aplikasi yang signifikan dalam berbagai bidang, termasuk kriptografi, simulasi molekuler, dan pengoptimalan masalah kompleks. Berikut adalah detail penggunaan komputasi kuantum dalam masing-masing bidang tersebut (Bravo, 2024):

1. Kriptografi

Komputasi kuantum berpotensi mematahkan banyak algoritme kriptografi saat ini yang digunakan untuk mengamankan komunikasi dan transaksi. Namun, komputasi kuantum juga dapat digunakan untuk mengembangkan metode enkripsi tahan kuantum baru yang lebih aman. Jika komputasi kuantum dapat memecahkan algoritme kriptografi yang ada, maka akan memerlukan perubahan signifikan dalam sistem keamanan digital. Oleh karena itu, penelitian untuk mengembangkan metode enkripsi tahan kuantum sangat penting untuk memastikan keamanan data di masa depan.

2. Simulasi Molekuler

Komputasi kuantum dapat mensimulasikan perilaku material kompleks pada tingkat molekuler, memungkinkan penemuan material baru dengan sifat yang diinginkan seperti superkonduktivitas atau penyimpanan energi yang lebih baik. Simulasi molekuler dengan komputasi kuantum dapat membantu dalam merancang obat-obatan yang lebih efektif dan bahan-bahan dengan sifat yang diinginkan. Contohnya, simulasi

fotosintesis dan formasi molekul yang kompleks dapat membantu dalam penemuan material baru dengan sifat yang diinginkan.

3. Pengoptimalan Masalah Kompleks

Banyak masalah di dunia nyata yang melibatkan pencarian solusi optimal dari sejumlah besar solusi yang mungkin. Komputasi kuantum dapat digunakan untuk memecahkan masalah optimasi ini dengan lebih efisien daripada komputer klasik, memungkinkan solusi yang lebih cepat dan lebih akurat. Pengoptimalan masalah kompleks dengan komputasi kuantum dapat membantu dalam berbagai bidang seperti logistik, ekonomi, dan ilmu material. Contohnya, komputasi kuantum dapat membantu dalam menemukan rute pengiriman yang paling efisien atau menemukan kombinasi bahan yang paling optimal untuk suatu produk.

Dengan demikian, komputasi kuantum menawarkan solusi inovatif dalam kriptografi, simulasi molekuler, dan pengoptimalan masalah kompleks, membuka peluang baru untuk penelitian dan pengembangan di berbagai bidang.

Perusahaan teknologi telah mencapai beberapa kemajuan signifikan dalam mengembangkan komputer kuantum. Berikut adalah beberapa contoh kemajuan yang telah dicapai (Citra Pramesti, 2023):

1. Pengembangan Arsitektur Komputer Kuantum

Komputer kuantum memiliki arsitektur yang terdiri dari lima lapisan, yaitu Physical Layer, Virtual Layer, Error Correction Layer, Logical Layer, dan Application Layer. Setiap lapisan memiliki peran khusus dalam menjalankan operasi kuantum. Lapisan Physical Layer mencakup qubit fisik dan berbagai perangkat keras yang mengelola operasi fisik pada qubit. Lapisan Virtual Layer mengubah

state kuantum menjadi informasi primitif seperti qubit dan kuantum register

2. Pengembangan Algoritma Kuantum

Pada tahun 1994, Peter Shor mengusulkan algoritma Shor yang dapat memecahkan masalah faktorisasi secara efisien dengan komputasi kuantum. Algoritma ini menunjukkan potensi besar komputasi kuantum dalam bidang kriptografi. Selain algoritma Shor, ada beberapa algoritma kuantum lain yang dikembangkan untuk berbagai aplikasi, seperti simulasi molekuler, pengoptimalan, dan analisis data.

3. Pengembangan Teknologi Qubit

Qubits pada komputer kuantum dapat berada dalam superposisi, memungkinkan mereka menjadi 1 dan 0 pada waktu yang sama. Ini berarti, alih-alih memeriksa satu per satu, komputer kuantum dapat menentukan jawaban yang benar secara simultan dan lebih efisien. Pada tahun 1998, tim di IBM berhasil melakukan operasi dasar pada qubit menggunakan molekul. Ini merupakan salah satu langkah penting dalam pengembangan teknologi qubit.

4. Pengembangan Sistem Komputer Kuantum

Beberapa perusahaan teknologi seperti IBM telah mengembangkan komputer kuantum yang beroperasi dengan baik. Mereka telah melakukan eksperimen dan pengujian untuk meningkatkan kinerja dan keandalan sistem. Meskipun mendapatkan waktu pada komputer quantum fisik bisa sulit dan mahal, simulator qubit memungkinkan perusahaan untuk memulai proses eksperimen ini tanpa melebihi anggaran. Ini membantu dalam pengembangan teknologi qubit dan komputer kuantum secara lebih efisien.

Dengan demikian, perusahaan teknologi telah mencapai kemajuan signifikan dalam mengembangkan komputer kuantum, termasuk pengembangan arsitektur komputer kuantum, algoritma kuantum, teknologi qubit, dan sistem komputer kuantum.

6.3 Internet of Things (IoT)

IoT telah memperluas konsep komputasi dengan menghubungkan perangkat fisik ke internet. Sub bab ini menjelaskan:

Evolusi IoT: Bagaimana perangkat pintar seperti rumah pintar dan kota pintar berkembang melalui IoT.

Keamanan IoT: Tantangan dalam mengamankan jaringan perangkat yang terhubung.

Aplikasi IoT dalam Industri: Penggunaan IoT dalam otomasi industri, pertanian pintar, dan kesehatan.

Internet of Things (IoT) adalah konsep di mana berbagai perangkat, seperti sensor, perangkat elektronik, dan objek lainnya, terhubung dan berkomunikasi dengan satu sama lain melalui jaringan internet. Berikut adalah penjelasan lebih lanjut tentang IoT:

Pengertian Internet of Things

IoT adalah sebuah konsep yang mengacu pada jaringan objek fisik yang terhubung ke internet dan dapat saling bertukar data tanpa perlu campur tangan manusia. Contoh perangkat IoT meliputi sensor, kamera, lampu, pintu otomatis, dan lain-lain (Sofyan, 2019; Wikipedia, 2019)

Unsur Pembentuk Internet of Things

IoT terbentuk dari beberapa unsur utama, yaitu (Gillis, 2023; meilinaeka, 2023; Wikipedia, 2019):

1. **Sensor atau Device:** Perangkat yang dapat mengumpulkan data dari lingkungan sekitar, seperti sensor suhu, sensor cahaya, dan sensor suara.
2. **Konektivitas:** Teknologi nirkabel seperti WiFi, Bluetooth, cellular, Zigbee, dan LoRaWAN yang memungkinkan perangkat IoT berkomunikasi dengan jaringan internet.
3. **Artificial Intelligence (AI) dan Machine Learning (ML):** Teknologi AI dan ML digunakan untuk mengidentifikasi pola dan anomali dalam data, serta membuat keputusan berdasarkan analisis data.
4. **Cloud Computing:** Data yang dikumpulkan oleh perangkat IoT dikirim ke cloud untuk diproses dan dianalisis, sehingga dapat diakses dari mana saja dan kapan saja.

Manfaat Internet of Things

IoT memiliki berbagai manfaat, termasuk (meilinaeka, 2023):

1. **Efisiensi dan Produktivitas:** IoT dapat membantu meningkatkan efisiensi dan produktivitas melalui pengumpulan dan analisis data secara real-time.
2. **Peningkatan Kualitas Hidup:** IoT dapat membantu memantau kesehatan dan keamanan seseorang, serta menghubungkan mereka dengan teknologi yang dapat membantu meningkatkan kualitas hidup mereka.
3. **Peningkatan Efisiensi Energi:** IoT dapat digunakan untuk mengontrol dan memantau konsumsi energi dalam sebuah bangunan atau gedung, dan kemudian dapat digunakan untuk mengoptimalkan konsumsi energi dan meningkatkan efisiensi.
4. **Kemudahan Pengoperasian:** IoT membuat perangkat yang terhubung lebih mudah untuk dioperasikan dan dikontrol, terutama melalui aplikasi mobile atau web yang mudah digunakan.

5. Kemampuan untuk Membuat Pengambilan Keputusan yang Lebih Baik: IoT dapat membantu organisasi atau individu untuk membuat keputusan yang lebih baik dan lebih cepat melalui analisis data secara real-time.
6. Optimasi *Supply Chain*: IoT data dapat digunakan untuk mengoptimalkan supply chain dan *inventory management processes*, sehingga dapat mengurangi biaya dan meningkatkan kepuasan pelanggan.
7. Automasi Tugas: IoT dapat mengotomasi tugas-tugas untuk meningkatkan kualitas layanan dan mengurangi kebutuhan intervensi manusia.
8. Pengalaman Pelanggan yang Ditingkatkan: IoT dapat mengembangkan produk dan layanan yang disesuaikan dengan preferensi dan kebutuhan individu, seperti smart home devices dan wearable technology.

Dengan demikian, IoT adalah teknologi yang sangat berdampak pada berbagai aspek kehidupan, dari meningkatkan efisiensi dan produktivitas hingga meningkatkan kualitas hidup dan pengalaman pelanggan.

Contoh penerapan *Internet of Things* (IoT) di rumah tangga sangat beragam dan dapat meningkatkan kenyamanan, efisiensi, dan keamanan. Berikut adalah beberapa contoh penerapan IoT di rumah tangga:

1. Smart Home (Rumah Pintar)

Menggunakan teknologi pendeteksi gerakan, pecahan kaca, dan kamera keamanan untuk memantau keadaan rumah. Sistem ini dapat terhubung ke cloud dan gawai pintar, memungkinkan pemilik rumah untuk memantau keadaan rumah dari mana saja dan kapan saja. Menggunakan termostat pintar yang dapat diprogram untuk menyesuaikan suhu dan kelembapan sesuai dengan preferensi individu. Termostat ini dapat mendeteksi dan

menyesuaikan suhu secara otomatis (Verihubs, 2024; Versha, 2023)

Menggunakan lampu pintar yang dapat diatur secara otomatis berdasarkan waktu hari atau perintah suara melalui virtual assistant seperti Alexa atau Google Assistant. Menggunakan kulkas pintar yang dilengkapi dengan Food AI, memungkinkan pengguna untuk mengecek isi kulkas lewat gawai pintar dan bahkan memesan bahan makanan yang sudah habis (Ismail, 2022). Menggunakan kunci pintu pintar yang dapat dibuka secara otomatis dengan kode akses atau sidik jari, memungkinkan pengguna untuk mengontrol akses ke rumah.

2. Detektor Keamanan

Menggunakan detektor panas, asap, dan air yang dikombinasikan dengan bantalan alarm keamanan untuk mendeteksi kebakaran dan banjir. Sistem ini dapat mengirimkan notifikasi ke pengguna melalui aplikasi ponsel atau komputer.

3. Kamera Keamanan

Menggunakan kamera keamanan yang dapat terhubung ke cloud dan gawai pintar, memungkinkan pengguna untuk memantau keadaan rumah secara real-time dari mana saja.

4. Alarm Keamanan

Menggunakan alarm keamanan yang dapat diaktifkan secara otomatis ketika terdeteksi gerakan tidak diinginkan atau pecahan kaca. Sistem ini dapat mengirimkan notifikasi ke pengguna melalui aplikasi ponsel atau komputer.

Dengan demikian, penerapan IoT di rumah tangga dapat meningkatkan kenyamanan, efisiensi, dan keamanan,

serta memudahkan pengguna dalam mengontrol dan memantau berbagai aspek kehidupan sehari-hari.

6.4 *Realitas Virtual (VR)* dan *Augmented Reality (AR)*

VR dan AR adalah teknologi yang semakin populer dalam berbagai bidang seperti pendidikan, hiburan, dan bisnis.

Realitas Virtual (VR)

Realitas Virtual (VR) adalah lingkungan simulasi yang diciptakan oleh komputer, di mana pengguna dapat merasakan dunia virtual yang tampak nyata. VR secara sepenuhnya membenamkan pengguna dalam dunia yang dihasilkan komputer, memblokir pandangan dunia nyata dan menggunakan headset untuk menampilkan gambar stereoskopik (PuskoMedia Indonesia, 2024; Wikipedia, 2006).

Cara Kerja:

Pengguna mengenakan headset yang menampilkan gambar stereoskopik dan memblokir pandangan dunia nyata. Pelacak gerakan melacak pergerakan kepala dan tangan pengguna, memungkinkan mereka untuk berinteraksi dan menjelajahi dunia virtual secara intuitif.

Aplikasi:

VR digunakan dalam berbagai aplikasi hiburan, seperti game dan film virtual. VR juga digunakan dalam perawatan kesehatan untuk terapi dan pelatihan pasien. VR dapat digunakan dalam pendidikan untuk membuat pengalaman belajar yang lebih interaktif dan imersif.

Augmented Reality (AR)

Augmented Reality (AR) adalah teknologi yang menambahkan sentuhan virtual ke lingkungan dunia nyata. AR menjaga dunia nyata tetap terpusat tetapi memperindahkannya

dengan detail digital lainnya, memberi lapisan persepsi tingkat baru (Metaverse, 2024)

Cara Kerja:

AR menambahkan informasi digital pada elemen dunia nyata, seperti menambahkan label virtual pada objek fisik. AR dapat diakses melalui berbagai perangkat, mulai dari smartphone hingga kacamata khusus, seperti ARKit dan RealityKit untuk rendering.

Aplikasi:

AR digunakan dalam penggunaan sehari-hari, seperti aplikasi Pokémon GO yang menambahkan objek virtual ke lingkungan sekitar. AR semakin dianggap sebagai alat untuk pasar konsumen umum, bukan hanya untuk gamer.

Dengan demikian, VR dan AR memiliki peran yang berbeda dalam memanipulasi pengalaman pengguna, dengan VR menciptakan lingkungan virtual yang sepenuhnya imersif dan AR menambahkan informasi digital ke lingkungan dunia nyata.

Perbedaan utama antara *Virtual Reality* (VR) dan *Augmented Reality* (AR) adalah cara mereka memanipulasi dan menambahkan informasi ke pengalaman pengguna. Berikut adalah penjelasan lebih lanjut tentang perbedaan tersebut:

1. Pengertian dan Tujuan

Virtual Reality adalah teknologi yang menciptakan lingkungan virtual yang sepenuhnya imersif, mengaburkan garis antara dunia nyata dan virtual. Tujuannya adalah untuk membuat pengguna merasakan sensasi di dalam dunia digital yang sangat nyata. Augmented Reality adalah teknologi yang meningkatkan pengalaman pengguna di dunia nyata dengan menambahkan informasi digital di atas dunia nyata. Tujuannya adalah untuk memperindah dan memperkaya pengalaman

pengguna dengan elemen digital (Surti, 2023).

2. Cara Kerja

VR menggunakan perangkat keras seperti headset VR untuk memblokir pandangan dunia nyata dan menampilkan gambar stereoskopik. Pengguna dapat sepenuhnya tenggelam dalam dunia digital yang diciptakan oleh komputer. AR menggunakan perangkat khusus seperti kamera untuk mengidentifikasi objek dan lingkungan di dunia nyata, lalu menambahkan informasi digital ke atasnya. Data yang dikumpulkan kemudian dioptimalkan dengan grafik, audio, atau video.

3. Interaksi dan Pengalaman

Interaksi di VR sangat imersif, memungkinkan pengguna untuk berinteraksi dengan lingkungan virtual menggunakan pengontrol khusus dan headset VR. Pengalaman ini sangat realistis dan keterlibatan yang tinggi. Interaksi di AR lebih fokus pada meningkatkan pengalaman di dunia nyata. Pengguna dapat melihat objek virtual yang ditambahkan ke atas objek fisik, seperti label virtual pada produk atau karakter virtual dalam game.

4. Aplikasi

VR memiliki berbagai aplikasi di berbagai industri, seperti hiburan, pendidikan, dan pelatihan. Contoh aplikasi VR adalah game yang memungkinkan pemain untuk benar-benar memasuki dunia virtual dan merasakan sensasi pertempuran atau petualangan. AR juga memiliki berbagai aplikasi, seperti penggunaan sehari-hari seperti aplikasi Pokémon GO yang menambahkan objek virtual ke lingkungan sekitar. AR juga digunakan dalam industri seperti pemasaran dan pembuatan untuk meningkatkan pengalaman pengguna.

Dengan demikian, perbedaan utama antara VR dan AR adalah cara mereka memanipulasi dan menambahkan informasi ke pengalaman pengguna. VR menciptakan lingkungan virtual yang sepenuhnya imersif, sedangkan AR meningkatkan pengalaman di dunia nyata dengan menambahkan elemen digital.

Perusahaan teknologi telah mencapai beberapa kemajuan signifikan dalam mengembangkan komputer kuantum. Berikut adalah beberapa contoh kemajuan yang telah dicapai:

1. Pengembangan Arsitektur Komputer Kuantum

Komputer kuantum memiliki arsitektur yang terdiri dari lima lapisan, yaitu Physical Layer, Virtual Layer, Error Correction Layer, Logical Layer, dan Application Layer. Setiap lapisan memiliki peran khusus dalam menjalankan operasi kuantum. Lapisan Physical Layer mencakup qubit fisik dan berbagai perangkat keras yang mengelola operasi fisik pada qubit. Lapisan Virtual Layer mengubah state kuantum menjadi informasi primitif seperti qubit dan kuantum register (Citra Pramesti, 2023).

2. Pengembangan Algoritma Kuantum

Pada tahun 1994, Peter Shor mengusulkan algoritma Shor yang dapat memecahkan masalah faktorisasi secara efisien dengan komputasi kuantum. Algoritma ini menunjukkan potensi besar komputasi kuantum dalam bidang kriptografi[4]. Selain algoritma Shor, ada beberapa algoritma kuantum lain yang dikembangkan untuk berbagai aplikasi, seperti simulasi molekuler, pengoptimalan, dan analisis data[2][4].

3. Pengembangan Teknologi Qubit

Qubits pada komputer kuantum dapat berada dalam superposisi, memungkinkan mereka menjadi 1 dan 0 pada waktu yang sama. Ini berarti, alih-alih memeriksa

satu per satu, komputer kuantum dapat menentukan jawaban yang benar secara simultan dan lebih efisien. Pada tahun 1998, tim di IBM berhasil melakukan operasi dasar pada qubit menggunakan molekul. Ini merupakan salah satu langkah penting dalam pengembangan teknologi qubit.

4. Pengembangan Sistem Komputer Kuantum

Beberapa perusahaan teknologi seperti IBM telah mengembangkan komputer kuantum yang beroperasi dengan baik. Mereka telah melakukan eksperimen dan pengujian untuk meningkatkan kinerja dan keandalan sistem[1][4]. Meskipun mendapatkan waktu pada komputer quantum fisik bisa sulit dan mahal, simulator qubit memungkinkan perusahaan untuk memulai proses eksperimen ini tanpa melebihi anggaran. Ini membantu dalam pengembangan teknologi qubit dan komputer kuantum secara lebih efisien.

Dengan demikian, perusahaan teknologi telah mencapai kemajuan signifikan dalam mengembangkan komputer kuantum, termasuk pengembangan arsitektur komputer kuantum, algoritma kuantum, teknologi qubit, dan sistem komputer kuantum.

6.5 Blockchain dan Keamanan Digital

Blockchain telah merevolusi cara penyimpanan dan pertukaran data, khususnya dalam konteks transaksi digital.

1. Transparansi dan Keandalan

Setiap transaksi dan perubahan data yang dicatat dalam blockchain dapat dilihat oleh semua pengguna dalam jaringan. Hal ini mengurangi risiko manipulasi atau pemalsuan data karena setiap perubahan harus disetujui oleh mayoritas pengguna dalam jaringan. Blockchain

memberikan tingkat keandalan yang tinggi karena setiap transaksi yang dicatat tidak dapat diubah atau dihapus. Semua perubahan harus diverifikasi oleh banyak komputer yang saling terhubung, menciptakan sistem yang sangat sulit ditembus (PuskoMedia Indonesia, 2024a)

2. Enkripsi dan Keamanan Data

Data dalam blockchain dienkripsi menggunakan algoritma kriptografi yang canggih. Ini berarti bahwa data yang disimpan dalam blockchain sangat sulit untuk diakses atau dimodifikasi oleh pihak yang tidak berwenang. Karena data tidak terpusat pada satu server atau pihak tertentu, risiko pencurian atau kehilangan data juga berkurang secara signifikan. Ini membuat blockchain sangat ideal untuk melindungi data sensitif.

3. Keamanan Identitas Digital

Blockchain meningkatkan keamanan identitas digital melalui mekanisme yang membuat catatan transaksi tidak dapat diubah dan terdistribusi luas. Ini menciptakan lingkungan yang lebih aman dan tepercaya untuk pengelolaan identitas digital.

4. Kontrol Data

Blockchain memberi pelanggan kontrol penuh atas data pribadi mereka, memungkinkan mereka untuk membagikannya dengan perusahaan hanya jika mereka setuju. Ini meningkatkan privasi data dan mengurangi risiko data breach.

5. Transparansi dalam Transaksi

Blockchain digunakan dalam berbagai sektor untuk meningkatkan transparansi dalam transaksi, seperti dalam pembuatan mata uang digital seperti Bitcoin. Setiap transaksi yang terjadi dapat dilihat oleh semua pengguna dalam jaringan, membuat sistem lebih transparan dan

aman.

Dengan demikian, teknologi blockchain menjadi solusi yang efektif dalam meningkatkan keamanan digital melalui transparansi, keandalan, enkripsi, pengamanan identitas, kontrol data, dan transparansi dalam transaksi.

Blockchain meningkatkan transparansi dalam keamanan digital melalui beberapa cara yang signifikan:

1. **Transparansi dalam Catatan Transaksi:** Setiap transaksi yang terjadi dalam blockchain dicatat dalam sebuah buku besar yang terdistribusi di seluruh jaringan. Ini berarti bahwa semua pengguna dalam jaringan memiliki akses ke semua transaksi yang pernah terjadi, sehingga meningkatkan transparansi.
2. **Imutabilitas Data:** Setiap blok dalam blockchain mengandung informasi tentang transaksi yang terjadi dan tidak dapat diubah atau dihapus. Hal ini memastikan bahwa catatan transaksi tetap dan dapat diandalkan, sehingga meningkatkan transparansi dan keandalan data.
3. **Konsensus Jaringan:** Jaringan blockchain menggunakan mekanisme konsensus untuk memverifikasi transaksi. Ini berarti bahwa setiap transaksi harus disetujui oleh mayoritas pengguna dalam jaringan sebelum dapat ditambahkan ke blockchain. Hal ini memastikan bahwa hanya transaksi yang valid yang dapat ditambahkan, meningkatkan transparansi dan keamanan.
4. **Tanda Tangan Digital:** Tanda tangan digital digunakan untuk memverifikasi keaslian identitas digital dan mencegah pemalsuan. Dengan menggunakan tanda tangan digital, setiap transaksi dapat diverifikasi secara otentik, meningkatkan transparansi dan keamanan identitas digital.
5. **Desentralisasi:** Blockchain tidak dimiliki atau dikendalikan

oleh organisasi tertentu. Sebaliknya, salinan dari seluruh rantai blok dimiliki oleh setiap anggota jaringan. Hal ini mengurangi risiko kecurangan dan meningkatkan transparansi karena tidak ada titik kelemahan tunggal.

Dengan demikian, blockchain meningkatkan transparansi dalam keamanan digital melalui catatan terdistribusi, imutabilitas data, konsensus jaringan, tanda tangan digital, dan desentralisasi (dewan studio, 2024).

Blockchain dapat mengurangi risiko serangan siber melalui beberapa cara yang efektif (Andhika R, 2024):

1. Pencatatan Data yang Aman dan Tidak Dapat Diubah: Blockchain mencatat transaksi dan perubahan data secara real-time, membuatnya sulit untuk diubah atau dihapus oleh pihak yang tidak berwenang. Hal ini memastikan bahwa data asli dan otentik tetap ada, sehingga memungkinkan perusahaan untuk memulihkan data tanpa membayar tebusan, seperti dalam kasus serangan ransomware.
2. Transparansi dan Imutabilitas: Setiap transaksi dalam blockchain dicatat dalam sebuah buku besar yang terdistribusi di seluruh jaringan. Ini membuat semua pengguna dalam jaringan memiliki akses ke semua transaksi yang pernah terjadi, meningkatkan transparansi dan memungkinkan deteksi lebih cepat dari aktivitas mencurigakan.
3. Konsensus Jaringan: Jaringan blockchain menggunakan mekanisme konsensus untuk memverifikasi transaksi, memastikan bahwa hanya transaksi yang valid yang dapat ditambahkan ke blockchain. Hal ini mengurangi risiko transaksi palsu dan memperkuat keamanan data.
4. Enkripsi dan Kriptografi: Data dalam blockchain dienkripsi dengan teknik kriptografi yang kuat, sehingga hanya

pihak yang berwenang yang dapat mengaksesnya. Ini memungkinkan pelacakan dan audit yang lebih baik, sehingga setiap aktivitas mencurigakan dapat segera terdeteksi dan ditindaklanjuti.

5. Pengamanan Identitas Digital: Blockchain dapat digunakan untuk mengamankan identitas digital dengan menerapkan identitas digital yang berbasis blockchain. Hal ini memastikan bahwa hanya pengguna yang sah yang memiliki akses ke sistem dan data sensitif, mengurangi risiko serangan yang sering kali dimulai dengan pencurian kredensial pengguna.

Dengan demikian, blockchain menawarkan solusi inovatif yang dapat meningkatkan keamanan digital dan mengurangi risiko serangan siber melalui transparansi, imutabilitas, konsensus jaringan, enkripsi, dan pengamanan identitas digital.

Prinsip kerja blockchain sebagai sistem terdesentralisasi dan aman dapat dijelaskan melalui beberapa komponen utama (Binus University, 2022; Rezkhita, 2023; Sari, 2024):

1. Desentralisasi

Blockchain beroperasi sebagai jaringan peer-to-peer, di mana setiap node dalam jaringan memiliki salinan lengkap dari blockchain. Ini berarti tidak ada pusat data tunggal yang dapat diakses oleh pihak tertentu, melainkan setiap node memiliki informasi yang sama. Setiap node dalam jaringan blockchain memiliki peran aktif dalam memverifikasi dan menambahkan blok baru ke dalam rantai. Hal ini memastikan bahwa tidak ada satu entitas yang memiliki kontrol penuh atas data, membuat sistem lebih terdesentralisasi.

2. Kriptografi

Setiap blok dalam blockchain memiliki hash unik yang mengidentifikasi blok tersebut dan semua informasi yang terkandung di dalamnya. Hash ini dihasilkan melalui proses enkripsi yang kuat, membuat sulit bagi pihak manapun untuk mengubah atau memanipulasi data. Karena setiap blok dienkripsi dengan hash unik, data dalam blockchain tidak dapat diubah atau dihapus. Jika salah satu blok dirusak, hashnya akan berubah, membuat semua blok berikutnya tidak valid. Ini memastikan keamanan data yang tinggi.

3. Konsensus Jaringan

Setiap node dalam jaringan blockchain harus menyetujui bahwa transaksi yang dicatat adalah valid. Ini dilakukan melalui proses konsensus yang berbeda-beda tergantung pada jenis blockchain, seperti Proof of Work (PoW) atau Proof of Stake (PoS). Konsensus jaringan memastikan bahwa semua node dalam jaringan memiliki versi yang sama dari blockchain. Jika ada perbedaan, node akan menolak untuk menambahkan blok baru ke dalam rantai, sehingga memastikan integritas data.

4. Imutabilitas Data

Rantai Blok yang Tidak Dapat Diubah**: Setiap blok dalam blockchain dihubungkan dengan hash dari blok sebelumnya, menciptakan rantai yang tidak dapat diubah. Jika salah satu blok dirusak, semua blok berikutnya akan menjadi tidak valid, sehingga memastikan imutabilitas data.

5. Transparansi dan Akuntabilitas

Blockchain menawarkan transparansi yang tinggi karena semua transaksi dapat dilihat oleh siapa saja. Ini memungkinkan pengguna untuk melacak sejarah dan jejak semua transaksi, meningkatkan akuntabilitas dan

keamanan data.

Dengan demikian, prinsip kerja blockchain sebagai sistem terdesentralisasi dan aman didasarkan pada penggunaan kriptografi, konsensus jaringan, imutabilitas data, dan transparansi. Ini membuat blockchain menjadi teknologi yang sangat aman dan efektif untuk merekam dan mengelola transaksi digital.

Blockchain diterapkan dalam sektor logistik, kontrak pintar, dan verifikasi identitas dengan cara yang berbeda-beda, tetapi semua memiliki tujuan untuk meningkatkan efisiensi, keamanan, dan transparansi. Berikut adalah detail aplikasi blockchain dalam masing-masing bidang tersebut (CoinMarketCap, 2021; Plisio, 2024):

1. Logistik

Blockchain digunakan untuk mengintegrasikan sensor dan aktuator dalam sistem logistik, memungkinkan pengawasan dan kontrol yang lebih akurat. Contohnya, sensor dapat mendeteksi kondisi barang, suhu, dan kelembaban, sementara aktuator dapat mengontrol proses pengiriman secara otomatis. Blockchain membantu dalam optimasi rantai pasokan dengan menganalisis data real-time dari berbagai sumber, seperti sensor dan perangkat lainnya. Hal ini memungkinkan perusahaan untuk mengidentifikasi masalah sebelum mereka terjadi dan mengambil tindakan yang tepat untuk meningkatkan efisiensi.

Blockchain menawarkan transparansi yang tinggi karena semua transaksi dapat dilihat oleh siapa saja. Ini memungkinkan pengguna untuk melacak sejarah dan jejak semua transaksi, meningkatkan akuntabilitas dan keamanan data dalam proses logistik.

2. Kontrak Pintar (*Smart Contract*)

Kontrak pintar adalah protokol komputer yang secara otomatis mengeksekusi, memverifikasi, dan menegakkan ketentuan-ketentuan yang telah ditetapkan di dalamnya. Mereka ditulis dalam bahasa pemrograman khusus seperti Solidity untuk Ethereum dan Vyper untuk kontrak pintar yang lebih sederhana. Kontrak pintar memiliki potensi penggunaan yang luas dalam berbagai industri. Misalnya, dalam industri logistik, kontrak pintar dapat digunakan untuk memastikan pengiriman barang sesuai jadwal dengan otomatisasi pemantauan dan pembayaran yang terkait.

Kontrak pintar memfasilitasi transaksi tepercaya antara pihak independen tanpa memerlukan otoritas pusat atau sistem hukum. Kode mereka disimpan di blockchain, menjadikannya transparan dan dapat diverifikasi secara publik, memungkinkan siapa pun memeriksa kode kontrak dan status operasionalnya saat ini.

3. Verifikasi Identitas

Blockchain dapat digunakan untuk mengamankan identitas digital dengan menerapkan identitas digital yang berbasis blockchain. Hal ini memastikan bahwa hanya pengguna yang sah yang memiliki akses ke sistem dan data sensitif, mengurangi risiko serangan yang sering kali dimulai dengan pencurian kredensial pengguna. Blockchain menawarkan transparansi yang tinggi karena semua transaksi dapat dilihat oleh siapa saja. Data yang disimpan dalam blockchain tidak dapat diubah atau dihapus, memastikan imutabilitas data dan keamanan identitas digital.

Dengan demikian, blockchain diterapkan dalam sektor logistik untuk meningkatkan efisiensi dan transparansi, dalam kontrak pintar untuk memfasilitasi transaksi tepercaya dan otomatisasi, dan dalam verifikasi identitas untuk mengamankan data sensitif dengan transparansi dan imutabilitas (Al Berlant Ghulam, 2023).

Implikasi keamanan yang lebih kuat dari teknologi blockchain, serta tantangan yang masih ada, dapat dijelaskan sebagai berikut:

1. Implikasi Keamanan yang Lebih Kuat

Blockchain menawarkan keamanan data yang sangat tinggi karena database dalam blockchain bersifat append-only, yang berarti data hanya dapat ditambahkan dan tidak dapat diubah atau dihapus. Hal ini membuat sulit bagi peretas untuk meretas data. Blockchain menawarkan sistem transparan yang tinggi, di mana semua pihak dapat melihat akses publik selama proses transaksi tanpa harus login terlebih dahulu. Ini meningkatkan keamanan dan transparansi data.

Blockchain memberikan kualitas audit yang lebih baik dengan memberikan akses pengguna untuk melacak sejarah dan jejak semua transaksi. Ini mengurangi risiko penyelewengan dana di lingkungan bisnis.

2. Tantangan yang Masih Ada

Kebanyakan orang masih belum memahami teknologi blockchain dan bagaimana hal itu dapat membantu bisnis dan ekonomi. Hal ini membuat sulit bagi beberapa organisasi untuk menerapkan teknologi ini karena kurangnya edukasi dan pemahaman. Implementasi teknologi blockchain dapat sangat kompleks, terutama dalam hal integrasi dengan sistem yang sudah ada. Hal ini memerlukan kerja sama antar pemerintah, industri, dan sektor publik untuk mengatasi tantangan dan hambatan

ini.

Meskipun teknologi blockchain menawarkan banyak manfaat, biaya implementasi yang diperlukan untuk mengembangkan dan memelihara sistem blockchain masih cukup tinggi. Hal ini dapat menjadi tantangan bagi beberapa organisasi yang memiliki anggaran terbatas.

3. Tantangan Khusus dalam Implementasi

Implementasi blockchain hibrida dan konsorsium dapat menambah kompleksitas dalam hal keamanan dan kontrol akses. Perusahaan harus mengatur sistem berbasis izin privat bersama dengan sistem publik, yang memerlukan kontrak pintar untuk memungkinkan anggota publik memeriksa transaksi privat. Jika salah satu blok dalam rantai blockchain dirusak, maka semua blok berikutnya akan menjadi tidak valid. Untuk mengatasi masalah ini, ada konsep yang disebut *proof of work* (POW), yang membutuhkan komputasi yang kuat untuk menghitung hash yang unik untuk setiap blok.

Dengan demikian, teknologi blockchain menawarkan implikasi keamanan yang lebih kuat melalui transparansi, audit yang baik, dan keamanan data terjamin. Namun, masih ada beberapa tantangan yang perlu diatasi, termasuk kurangnya edukasi, implementasi yang kompleks, biaya implementasi yang tinggi, serta tantangan khusus dalam implementasi jaringan hibrida dan konsorsium (Universitas Islam Indonesia, 2021).

6.6 Robotika dan Otomasi

Teknologi robotika telah berkembang dengan pesat, berkat kemajuan dalam AI dan sensor. Robotika dan otomasi adalah dua konsep yang saling terkait dalam meningkatkan

efisiensi dan akurasi dalam berbagai industri. Berikut adalah perbedaan dan peran masing-masing (UNMAHA, 2023b; Wicaksana, 2022):

1. Robotika

Robotika adalah bidang ilmu yang mempelajari tentang desain, konstruksi, pemrograman, dan pengoperasian robot. Robot adalah mesin yang dapat dikendalikan secara otomatis atau dengan perintah manusia untuk melakukan berbagai tugas, seperti mengangkat bahan, membersihkan, atau menyelesaikan proses produksi di industri. Robot memiliki berbagai set sensor yang dipasang di dalamnya untuk menangkap dampak audio, visual, serta sentuhan. Mereka menggunakan artificial intelligence (AI) dan machine learning untuk melaksanakan kegiatan fitur lunak yang sudah ditetapkan dan bekerja persis semacam manusia. Robot dapat melakukan tugas yang berulang secara otomatis dengan tingkat kecermatan yang tinggi, meningkatkan efisiensi produksi. Mereka juga dapat bekerja di area yang beresiko dan tidak dapat ditempuh oleh manusia, seperti lingkungan yang berbahaya atau tempat yang sulit dijangkau.

2. Otomasi

Otomasi adalah proses penggunaan teknologi untuk mengendalikan dan mengoperasikan mesin, peralatan, dan proses produksi secara otomatis. Otomasi dapat menggunakan mikroprosesor tingkat lanjut dan fitur seperti PC untuk mengontrol proses produksi. Otomasi digunakan dalam berbagai proses industri seperti pengelasan, pengecatan, perakitan, pengemasan, pelabelan, dan pengangkutan. Sistem otomatis memberikan dukungan dalam produksi mobil, pemrosesan logam dan plastik, serta menjalankan proses

dalam kondisi higienis untuk produksi produk makanan, farmasi, atau kosmetik[2]. Otomasi meningkatkan efisiensi produksi dengan memastikan bahwa proses berjalan lancar dan mengurangi kemungkinan kesalahan yang dapat menurunkan akurasi produk. Otomasi juga memungkinkan pekerja manusia untuk fokus pada tugas-tugas yang memerlukan kecerdasan, kreativitas, dan pemecahan masalah.

3. Perbandingan

Robotika menggunakan robot yang dapat dikendalikan secara otomatis atau dengan perintah manusia, sedangkan otomasi menggunakan mikroprosesor tingkat lanjut dan fitur seperti PC untuk mengontrol proses produksi. Robotika lebih fokus pada tugas yang beresiko dan berulang, sementara otomasi lebih luas dalam penggunaannya dalam berbagai proses industri. Keduanya dapat digunakan secara bersamaan dalam industri untuk meningkatkan efisiensi dan akurasi produksi. Robotika dapat digunakan untuk tugas-tugas yang spesifik dan beresiko, sementara otomasi mengontrol proses produksi secara keseluruhan.

Dengan demikian, robotika dan otomasi adalah dua konsep yang saling melengkapi dalam meningkatkan efisiensi dan akurasi dalam berbagai industri. Robotika fokus pada tugas-tugas yang spesifik dan beresiko, sedangkan otomasi mengontrol proses produksi secara keseluruhan.

Robot digunakan dalam berbagai bidang, termasuk manufaktur, kesehatan, dan eksplorasi ruang angkasa, dengan cara yang berbeda-beda tetapi saling melengkapi. Berikut adalah detail penggunaan robot dalam masing-masing bidang tersebut (Misel Editor, 2023; Sulaiman, 2017):

1. Manufaktur

Robot digunakan secara luas dalam industri manufaktur untuk meningkatkan efisiensi dan akurasi produksi. Mereka dapat melakukan tugas-tugas yang berulang, seperti pengelasan, pengecatan, dan pengangkutan bahan. Robot membantu dalam optimasi proses produksi dengan menganalisis data real-time dari berbagai sumber, seperti sensor dan perangkat lainnya. Hal ini memungkinkan perusahaan untuk mengidentifikasi masalah sebelum mereka terjadi dan mengambil tindakan yang tepat untuk meningkatkan efisiensi.

Selain itu, robot juga digunakan dalam pengelolaan logistik, seperti pengawasan stok, pengiriman, dan penggunaan ruang penyimpanan. Hal ini memungkinkan perusahaan untuk mengoptimalkan rantai pasokan dan mengurangi biaya logistik.

2. Kesehatan

Robot bedah seperti Da Vinci Surgical System digunakan dalam prosedur bedah yang rumit. Keakuratan tinggi dan kontrol yang lebih baik dari sistem ini membantu meminimalkan risiko kesalahan dan mengurangi waktu pemulihan pasien. Robot untuk rehabilitasi digunakan untuk membantu pasien yang membutuhkan terapi fisik atau terapi rehabilitasi. Robot tersebut dirancang untuk memberikan gerakan terapeutik yang tepat dan membantu pasien dalam proses pemulihan. Robot pendukung seperti Pepper dapat digunakan untuk menghibur atau memberikan dukungan emosional kepada pasien, terutama anak-anak atau pasien yang mengalami isolasi sosial.

3. Eksplorasi Ruang Angkasa

Robot digunakan dalam eksplorasi ruang angkasa untuk melakukan tugas-tugas yang berbahaya atau tidak dapat ditempuh oleh manusia. Contohnya, NASA menggunakan robot untuk menginspeksi dan melakukan perawatan pada wahana antariksa, serta untuk melakukan penelitian di permukaan Mars. Robot modular dapat disesuaikan dengan berbagai tugas, seperti pengiriman obat otomatis atau pemantauan jarak jauh. Hal ini memungkinkan robot untuk berperan dalam berbagai misi ruang angkasa dengan fleksibilitas yang tinggi.

Dengan demikian, penggunaan robot dalam manufaktur, kesehatan, dan eksplorasi ruang angkasa membawa manfaat yang signifikan, termasuk meningkatkan efisiensi, akurasi, dan keamanan dalam berbagai bidang.

Otomatisasi memiliki dampak yang signifikan terhadap pasar tenaga kerja dan keterampilan yang dibutuhkan. Berikut adalah beberapa poin penting yang menjelaskan dampak tersebut (Kautsar, 2024; Mc Kinsey & Company, 2019):

1. Penggantian Pekerjaan

Otomatisasi dapat menggantikan beberapa pekerjaan yang berulang dan dapat diprediksi, seperti pengumpulan dan pemrosesan data, pengelasan, dan pengecatan. Namun, data dari McKinsey menunjukkan bahwa jumlah pekerjaan yang hilang karena otomatisasi lebih sedikit daripada pekerjaan baru yang muncul.

2. Pekerjaan Baru

Otomatisasi juga menciptakan lapangan pekerjaan baru yang tidak dapat digantikan oleh mesin. Contohnya, dalam industri logistik, otomatisasi membantu dalam pengelolaan stok, pengiriman, dan penggunaan ruang penyimpanan. Selain itu, perusahaan juga membutuhkan

tenaga kerja untuk mengelola dan memperbarui sistem otomatisasi cerdas.

3. Keterampilan yang Dibutuhkan

Dengan adanya otomatisasi, keterampilan teknologi seperti penggunaan perangkat lunak, kecerdasan buatan (AI), dan machine learning menjadi sangat penting. Perusahaan membutuhkan tenaga kerja yang dapat mengoperasikan dan memelihara sistem otomatisasi cerdas. Keterampilan Sosial dan Emosional : Selain keterampilan teknologi, keterampilan sosial dan emosional juga sangat dibutuhkan. Contohnya, dalam industri kesehatan, robot medis membutuhkan keterampilan untuk berinteraksi dengan pasien dan keterampilan komunikasi yang baik untuk menjelaskan hasil diagnosis dan perawatan.

Keterampilan Adaptasi: Dalam era yang semakin dinamis, keterampilan adaptasi menjadi sangat penting. Tenaga kerja harus dapat beradaptasi dengan perubahan teknologi dan situasi baru, serta memiliki kemampuan untuk memecahkan masalah yang kompleks.

4. Strategi Pendidikan

Pendidikan yang Sesuai: Untuk menghadapi tantangan yang dihadapi oleh otomatisasi, pendidikan harus disesuaikan dengan kebutuhan pasar tenaga kerja. Pendidikan harus meliputi keterampilan teknologi, keterampilan sosial, dan keterampilan adaptasi. Selain itu, pelatihan dan kursus online dapat membantu meningkatkan keterampilan digital dan kolaborasi.

Dengan demikian, otomatisasi tidak hanya menggantikan pekerjaan tertentu, tetapi juga menciptakan lapangan pekerjaan baru yang membutuhkan keterampilan teknologi, sosial, dan emosional. Perusahaan dan individu harus beradaptasi

dengan perubahan ini dan memastikan bahwa mereka memiliki keterampilan yang dibutuhkan untuk sukses di era digital.

Inovasi terbaru dalam pengembangan robot yang semakin cerdas dan otonom melibatkan berbagai teknologi canggih yang meningkatkan kemampuan dan fleksibilitas robot. Berikut adalah beberapa inovasi terbaru dalam pengembangan robot (Yuniar, 2022):

1. Teknologi Kecerdasan Buatan (AI)

Robot modern menggunakan kecerdasan buatan untuk mempelajari dan menyesuaikan diri dengan lingkungan sekitar. AI membantu robot dalam mengenali objek, menginterpretasi data sensor, dan membuat keputusan berdasarkan informasi yang diterima[6]. Teknologi machine learning digunakan untuk melatih robot dalam mengenali objek-objek tertentu, seperti boat aluminium di pabrik. Robot dapat memprediksi nama dan lokasi objek tersebut dengan akurasi tinggi.

2. Robot Otonom

Robot otonom dirancang untuk melakukan tugas-tugas tanpa kontrol manusia secara langsung. Contohnya, robot penyedot debu yang dapat beroperasi secara mandiri untuk membersihkan ruangan[2]. Robot mobil beroda dapat dikendalikan secara otonom dengan menggunakan kerangka kerja kontrol berbasis lokasi. Robot ini dapat bergerak menuju titik-titik lokasi yang telah ditentukan dengan presisi tinggi.

3. Teknologi Sensor dan Interpretasi Data

Robot menggunakan sensor time-of-flight kamera untuk merekam objek dalam 2 dan 3 dimensi. Data sensor ini diolah dan dimanfaatkan untuk mengklasifikasi dan menentukan posisi objek. Robot dapat menginterpretasi

lingkungan sekitar dengan menggunakan teknologi sensor yang canggih. Hal ini memungkinkan robot untuk berinteraksi dengan lingkungan secara lebih akurat dan efektif.

4. Penggunaan dalam Berbagai Sektor

Robot digunakan dalam sektor industri untuk memasang komponen secara otomatis dan mendistribusikan barang. Hal ini meningkatkan efisiensi produksi dan mengurangi kesalahan manusia. Robot digunakan dalam sektor kesehatan untuk membantu tenaga medis dalam berbagai tugas, seperti menyiapkan kamar pasien, mengangkut seprai ke fasilitas cuci, dan menangani logistik harian di rumah sakit.

Dengan demikian, inovasi terbaru dalam pengembangan robot yang semakin cerdas dan otonom melibatkan penggunaan teknologi AI, machine learning, sensor canggih, dan aplikasi dalam berbagai sektor. Hal ini meningkatkan kemampuan dan fleksibilitas robot untuk melakukan tugas-tugas yang lebih kompleks dan akurat.

6.7 Keberlanjutan dalam Teknologi Komputer

Dengan meningkatnya kesadaran lingkungan, ada dorongan untuk menciptakan teknologi yang lebih ramah lingkungan. Penggunaan energi terbarukan dalam pusat data dan perangkat komputasi yang lebih efisien telah menjadi topik penting dalam upaya mengurangi dampak lingkungan dan meningkatkan efisiensi operasional. Berikut adalah beberapa cara dan manfaat penggunaan energi terbarukan dalam pusat data dan perangkat komputasi (Mirza, 2024):

1. Penggunaan Sumber Energi Terbarukan

Panel Surya dan Energi Angin: Pusat data ramah lingkungan seperti yang dijelaskan dalam dan

menggunakan sumber energi terbarukan seperti panel surya, energi angin, dan hidroelektrik. Hal ini membantu mengurangi ketergantungan pada bahan bakar fosil yang menghasilkan emisi karbon tinggi. Evaluasi Kebutuhan Energi: Sebelum mengintegrasikan energi terbarukan, perlu melakukan analisis kebutuhan energi untuk mengidentifikasi apakah implementasi energi terbarukan memungkinkan dan berapa besar energi yang terintegrasi, seperti yang disebutkan dalam.

2. Teknologi Hemat Energi

Peralatan seperti server, perangkat penyimpanan, dan peralatan jaringan yang dirancang untuk mengonsumsi lebih sedikit daya tanpa mengorbankan kinerja merupakan komponen utama dari pusat data yang berkelanjutan. Teknologi virtualisasi juga memungkinkan beberapa server digabungkan menjadi satu perangkat, mengurangi jumlah perangkat fisik yang diperlukan dan, dengan demikian, mengurangi konsumsi energi, seperti yang dijelaskan dalam.

3. Sistem Pendingin Hemat Energi

Pusat data ramah lingkungan menggunakan sistem pendingin hemat energi seperti pendinginan cairan dan penggunaan udara luar (*free cooling*) untuk menjaga suhu optimal peralatan. Hal ini mengurangi konsumsi energi yang biasanya dibutuhkan oleh sistem pendingin tradisional, seperti yang disebutkan dalam.

4. Manajemen Daya Cerdas

Implementasi manajemen daya yang cerdas adalah kunci lain untuk efisiensi energi di pusat data ramah lingkungan. Sistem manajemen daya cerdas dapat memantau dan mengelola penggunaan energi secara real-time, memungkinkan penyesuaian otomatis untuk mengoptimalkan konsumsi energi. Ini termasuk

penjadwalan penggunaan perangkat saat kebutuhan daya rendah dan menonaktifkan perangkat yang tidak digunakan, seperti yang dijelaskan dalam.

5. Desain Arsitektur yang Efisien

Desain arsitektur pusat data ramah lingkungan juga berkontribusi terhadap efisiensi energi. Desain yang berkelanjutan, seperti menggunakan bangunan yang dirancang untuk memaksimalkan pencahayaan alami dan mengurangi kebutuhan pendinginan, dapat membantu mengurangi konsumsi energi secara signifikan, seperti yang disebutkan dalam.

Dengan demikian, penggunaan energi terbarukan dan teknologi hemat energi dalam pusat data dan perangkat komputasi yang lebih efisien membantu mengurangi dampak lingkungan dan meningkatkan efisiensi operasional. Hal ini termasuk penggunaan sumber energi terbarukan, teknologi hemat energi, sistem pendingin hemat energi, manajemen daya cerdas, serta desain arsitektur yang efisien.

Upaya untuk mengurangi jejak karbon dari produksi perangkat komputer melibatkan beberapa strategi yang berfokus pada efisiensi energi, penggunaan bahan ramah lingkungan, dan praktik penggunaan yang ramah lingkungan. Berikut adalah beberapa langkah yang dapat diambil (Solarkita, 2024):

1. Pemilihan Perangkat yang Efisien Energi

Memilih perangkat elektronik yang telah mendapatkan sertifikasi energi seperti Energy Star atau label efisiensi energi lainnya. Sertifikasi ini menunjukkan bahwa perangkat tersebut telah memenuhi standar efisiensi tertentu, sehingga dapat mengurangi konsumsi listrik dan emisi karbon secara signifikan.

2. **Pemilihan Perangkat yang Tahan Lama**
Pertimbangkan juga umur pakai perangkat. Perangkat yang dirancang untuk bertahan lama mengurangi frekuensi penggantian dan limbah elektronik. Hal ini dapat membantu mengurangi dampak lingkungan dengan mengurangi jumlah limbah elektronik yang dihasilkan.
3. **Optimalkan Pengaturan Energi**
Banyak perangkat elektronik memiliki pengaturan energi yang dapat dioptimalkan untuk mengurangi konsumsi listrik. Misalnya, mengatur komputer agar masuk ke mode tidur saat tidak digunakan dalam waktu lama atau menyesuaikan kecerahan layar dapat mengurangi konsumsi energi. Memanfaatkan fitur penghematan energi pada perangkat elektronik dapat membantu mengurangi jejak karbon.
4. **Penggunaan Energi Terbarukan**
Menggunakan sumber energi terbarukan seperti energi surya atau angin untuk menyuplai listrik ke perangkat elektronik juga dapat mengurangi jejak karbon. Beralih ke penyedia energi terbarukan atau memasang sistem panel surya dapat mengurangi ketergantungan pada sumber energi fosil dan emisi karbon yang dihasilkan.
5. **Edukasi dan Kesadaran**
Pendidikan dan kesadaran tentang jejak karbon dari konsumsi elektronik penting untuk mendorong perubahan perilaku. Kampanye kesadaran dapat membantu konsumen memahami dampak lingkungan dari penggunaan perangkat elektronik dan mendorong mereka untuk mengambil tindakan proaktif. Program edukasi yang baik dapat memotivasi individu untuk memilih perangkat yang lebih ramah lingkungan dan menerapkan praktik pengelolaan energi yang efisien.

Dengan demikian, upaya untuk mengurangi jejak karbon dari produksi perangkat komputer melibatkan pemilihan perangkat yang efisien energi, pemilihan perangkat yang tahan lama, optimalkan pengaturan energi, penggunaan energi terbarukan, serta edukasi dan kesadaran tentang dampak lingkungan dari konsumsi elektronik.

Inisiatif dalam mengurangi limbah elektronik dan memperpanjang umur perangkat komputer melibatkan beberapa strategi yang berfokus pada daur ulang, pengelolaan limbah elektronik yang tepat, dan praktik penggunaan yang ramah lingkungan. Berikut adalah beberapa inisiatif yang dapat diambil (Defitri, 2022):

1. Daur Ulang Perangkat Elektronik

Komponen perangkat elektronik seperti logam, plastik, dan kaca dapat didaur ulang. Daur ulang ini membantu mengurangi jumlah limbah elektronik yang dihasilkan dan menghemat sumber daya alam. Banyak kota di Indonesia kini memiliki fasilitas daur ulang elektronik yang dapat membantu mengurangi dampak lingkungan dari limbah elektronik.

2. Pengelolaan Limbah Elektronik yang Tepat

Limbah elektronik harus dijempit dan dipilah dengan benar untuk memastikan bahwa bahan-bahan berbahaya seperti merkuri, timbal, dan kadmium tidak mencemari lingkungan. Proses ini dimulai dari penjemputan di rumah warga, pemilahan sampah di tempat penampungan sementara, hingga akhirnya sampah sampai ke tempat penampungan tingkat provinsi.

3. Menggunakan Program Trade-In

Beberapa produsen elektronik menawarkan program trade-in di mana perangkat lama dapat ditukar dengan diskon untuk pembelian perangkat baru. Program ini tidak hanya membantu mengurangi limbah elektronik

tetapi juga mendorong penggunaan teknologi yang lebih efisien.

4. Menjaga Keawetan Perangkat Elektronik

Menjaga keawetan perangkat elektronik merupakan salah satu cara untuk meningkatkan masa pakai dan mengurangi limbah elektronik. Perawatan yang tepat dapat memperpanjang umur perangkat dan mengurangi kebutuhan untuk membeli perangkat baru.

5. Membeli Perangkat yang Ramah Lingkungan

Membeli perangkat elektronik yang ramah lingkungan seperti yang berlabel Energy Star atau disertifikasi oleh *Electronic Product Environmental Assessment Tool* (EPEAT) dapat membantu mengurangi dampak lingkungan. Perangkat ini dirancang untuk memiliki desain yang ramah lingkungan, proses produksi yang lebih baik, konsumsi energi yang lebih rendah, dan daur ulang yang lebih efektif.

6. Mendonasikan atau Menyumbangkan Perangkat Lama

Mendonasikan atau menyumbangkan perangkat elektronik yang masih berfungsi kepada teman atau saudara terdekat dapat membantu mengurangi limbah elektronik. Hal ini juga dapat memberikan manfaat kepada orang lain yang mungkin membutuhkannya.

Dengan demikian, inisiatif dalam mengurangi limbah elektronik dan memperpanjang umur perangkat komputer melibatkan daur ulang, pengelolaan limbah elektronik yang tepat, menggunakan program trade-in, menjaga keawetan perangkat, membeli perangkat yang ramah lingkungan, serta mendonasikan atau menyumbangkan perangkat lama.

6.8 Kesimpulan

Teknologi komputer terus berkembang dengan cepat, membawa berbagai inovasi yang mengubah cara manusia berinteraksi, bekerja, dan hidup. Tren masa depan seperti AI, komputasi kuantum, IoT, dan blockchain menawarkan peluang besar, namun juga tantangan yang harus diatasi. Mempersiapkan diri untuk memahami dan memanfaatkan teknologi masa depan sangat penting bagi individu dan organisasi.

DAFTAR PUSTAKA

- Al Berlant Ghulam. (2023, June 26). Kontrak Pintar (Smart Contract): Definisi, Konsep, Manfaat dan Kaitannya dengan Blockchain Halaman 1 - Kompasiana.com. KOMPASIANA; Kompasiana.com. <https://www.kompasiana.com/alberlant/649914f64adde0dd248be04/kontrak-pintar-smart-contract-definisi-konsep-manfaat-dan-kaitannya-dengan-blockchain>
- Amazon. (2022). Apa itu Debugging? Penjelasan tentang Debugging - AWS. Amazon Web Services, Inc. <https://aws.amazon.com/id/what-is/debugging/>
- Andhika R. (2024, March 7). Blockchain dan Keamanan Siber: Aplikasi dan Implikasinya - Fourtrezz. Fourtrezz. <https://fourtrezz.co.id/blockchain-dan-keamanan-siber-aplikasi-dan-implikasinya/>
- Binus University. (2023a, June 19). Jenis-Jenis Bahasa Pemrograman dan Penjelasan. BINUS UNIVERSITY MALANG | Pilihan Universitas Terbaik Di Malang. <https://binus.ac.id/malang/2023/06/jenis-jenis-bahasa-pemrograman-dan-penjelasan/>
- Binus University. (2023b, August 17). Menilik Masa Depan Karier dan Pendidikan dengan Otomatisasi - BINUS Business School. BINUS Business School. <https://bbs.binus.ac.id/2023/08/menilik-masa-depan-karier-dan-pendidikan-dengan-otomatisasi/>
- Bravo, S. (2024, June 21). Apa itu Komputasi Quantum? - Azure Quantum. Microsoft.com. <https://learn.microsoft.com/id-id/azure/quantum/overview-understanding-quantum-computing>
- Chairunnisa, T. (2024, June 21). Perbedaan antara Supervised dan Unsupervised Learning | D3 Teknologi

- Telekomunikasi. D3 Teknologi Telekomunikasi.
<https://dte.telkomuniversity.ac.id/perbedaan-antara-supervised-dan-unsupervised-learning/>
- Citra Pramesti, V. C. (2023, December 12). Apakah Dunia Sudah Siap untuk Kedatangan Era Komputer Kuantum? Pendidikan Teknik Informatika Dan Komputer.
<https://ptik.fkip.uns.ac.id/apakah-dunia-sudah-siap-untuk-kedatangan-era-komputer-kuantum/>
- CoinMarketCap. (2021, August 21). Smart Contract (Kontrak Pintar) Definition. CoinMarketCap Academy; CoinMarketCap.
<https://coinmarketcap.com/academy/id/glossary/smart-contract>
- Darmawan, I. S. (2023, July 6). Memahami Dasar-dasar Komputasi Kuantum: Konsep, Prinsip, dan Keunggulannya Halaman 1 - Kompasiana.com. KOMPASIANA; Kompasiana.com.
<https://www.kompasiana.com/ronybrothers/64a6554608a8b554ed78bb32/memahami-dasar-dasar-komputasi-kuantum-konsep-prinsip-dan-keunggulannya>
- Defitri, M. (2022, October 12). Pengelolaan Sampah Elektronik dan Peraturannya di Indonesia. Waste4Change.
<https://waste4change.com/blog/pengelolaan-sampah-elektronik-dan-peraturannya-di-indonesia/>
- dewan studio. (2024, June 28). Blockchain dalam Pemerintahan: Transparansi dan Efisiensi Digital. DEWANSTUDIO Digital Agency.
<https://www.dewanstudio.com/blockchain-dalam-pemerintahan-transparansi-dan-efisiensi-digital/>
- Educa Studio. (2023b). 15 Bahasa Pemrograman Populer, Pemula Wajib Tahu! | Berita | Gamelab Indonesia. Gamelab.ID. <https://www.gamelab.id/news/2640-15-bahasa-pemrograman-populer--pemula-wajib-tahu>

- Educa Studio. (2023c). Memahami Dasar-Dasar Pemrograman untuk Karier IT yang Sukses | Berita | Gamelab Indonesia. Gamelab.ID. <https://www.gamelab.id/news/2653-memahami-dasar-dasar-pemrograman-untuk-karier-it-yang-sukses>
- Fuad, M. (2024, May 21). Quantum Computing: Pengertian, Sejarah, dan Manfaat. Program Studi Sarjana Informatika. <https://bif.telkomuniversity.ac.id/quantum-computing-pengertian-sejarah-dan-manfaat/>
- Gillis, A. (2023). What Is IoT (Internet of Things) and How Does It Work? TechTarget. <https://www.techtarget.com/iotagenda/definition/Internet-of-Things-IoT>
- IBM. (2023). Apa yang dimaksud dengan Kecerdasan Buatan (AI)? | IBM. Wwww.ibm.com. <https://www.ibm.com/id-id/topics/artificial-intelligence>
- Ismail, B. (2022, December 5). 9 Contoh Penerapan IoT dalam Kehidupan Sehari-hari! Smartven. <https://www.smartven.biz/blog/contoh-penerapan-iotIyan>
- Ramadhani. (2023, February 20). Perbedaan Bahasa Pemrograman Tingkat Tinggi & Tingkat Rendah. El Samara. <https://elsamara.id/perbedaan-bahasa-pemrograman-tingkat-tinggi-tingkat-rendah/>
- Kautsar, A. (2024, August). Keterampilan yang Dibutuhkan Generasi Z, Future Ready! | Future Skills. Future Skills. <https://futureskills.id/blog/keterampilan-yang-dibutuhkan-generasi-z/>
- Laraswati, B. D. (2022, September 8). Perbedaan Supervised vs Unsupervised Learning. Algoritma Data Science School; Algoritma Data Science School. <https://blog.algoritma.com/supervised-vs-unsupervised-learning/>
- Mc Kinsey & Company. (2019). Otomasi dan masa depan

- pekerjaan di Indonesia.
<https://www.mckinsey.com/~media/mckinsey/featured%20insights/asia%20pacific/automation%20and%20the%20future%20of%20work%20in%20indonesia/automation-and-the-future-of-work-in-indonesia-indonesian.pdf>
- meilinaeka. (2023, May 10). Internet of Things : Pengertian, Sejarah, Kelebihan dan Kekurangannya. Direktorat Pusat Teknologi Informasi.
<https://it.telkomuniversity.ac.id/internet-of-things-pengertian-sejarah-kelebihan-dan-kekurangannya/>
- Metaverse. (2024). Metaverse, Realitas Virtual, dan Realitas Tertambah: Rencana Apple dan Meta. Ar-Code.com.
<https://ar-code.com/id/blog/metaverse-virtual-reality-dan-augmented-reality-rencana-apple-dan-meta>
- Mirza, F. (2024, July 30). Apa yang Dimaksud Green Data Center dan Mengapa itu Penting? UltraHost Blog; UltraHost. <https://ulthost.com/blog/id/apa-yang-dimaksud-green-data-center-dan-mengapa-itu-penting/>
- Misel Editor. (2023, October 16). Robot Medis di Dunia Kesehatan: Tujuan Hingga Etika dalam Penggunaannya - PT Mitrainti Sejahtera Eletrindo. PT Mitrainti Sejahtera Eletrindo. <https://misel.co.id/robot-medis-di-dunia-kesehatan-tujuan-hingga-etika-dalam-penggunaannya/>
- Patria, R. (2023, April 12). Memahami Cara Kerja Machine Learning - DomaiNesia. DomaiNesia.
<https://www.domainesia.com/berita/machine-learning/>
- Plisio. (2024, September 14). Kontrak Cerdas: Peran dan Operasinya di Blockchain. Plisio.
<https://plisio.net/id/blog/smart-contracts-their-role-and-operation-in-blockchain>

- PuskoMedia Indonesia. (2024a, September 9). Keamanan Identitas Digital dalam Teknologi Blockchain | PuskoMedia Indonesia. PuskoMedia Indonesia. <https://www.puskomedia.id/blog/keamanan-identitas-digital-dalam-teknologi-blockchain/>
- PuskoMedia Indonesia. (2024b, September 9). Mengenal Realitas Virtual (VR): Pengantar dan Panduan Lengkap | PuskoMedia Indonesia. PuskoMedia Indonesia. <https://www.puskomedia.id/blog/mengenal-realitas-virtual-vr-pengantar-dan-panduan-lengkap/>
- Revou. (2024). Bootcamp Digital Marketing. Revou.co. <https://revou.co/kosakata/supervised-learning>
- Rezhkita, D. (2023, November 1). Apa Itu Blockchain? Teknologi Dibalik Semua Aset Crypto - Pintu Academy. Pintu.co.id. <https://pintu.co.id/academy/post/bagaimana-cara-kerja-blockchain>
- Sofyan, A. (2019). Internet Of Things. Smkn4tangsel.sch.id. <https://smkn4tangsel.sch.id/read/8/internet-of-things>
- Solarkita. (2024, August 6). Mengurangi Jejak Karbon dari Konsumsi Elektronik. Solarkita.com; Solarkita. <https://www.solarkita.com/blog/mengurangi-jejak-karbon-dari-konsumsi-elektronik>
- Sulaiman, M. R. (2017, April 26). 5 Robot Canggih di Bidang Kesehatan. DetikHealth; detikcom. <https://health.detik.com/berita-detikhealth/d-3483890/5-robot-canggih-di-bidang-kesehatan>
- Surti. (2023, July 31). 3 Perbedaan Virtual Reality dan Augmented Reality. Fortuneidn.com; fortuneidn.com. <https://www.fortuneidn.com/tech/surti/perbedaan-virtual-reality-dan-augmented-reality>
- Trivusi. (2022, June 21). Ketahui Perbedaan Supervised dan Unsupervised Learning. Trivusi.web.id; Blogger.

- <https://www.trivusi.web.id/2022/06/perbedaan-supervised-dan-unsupervised-learning.html>
- Universitas Islam Indonesia. (2021, June 7). Blockchain Tingkatkan Keamanan Data Dari Peretasan. Universitas Islam Indonesia. <https://www.uui.ac.id/blockchain-tingkatkan-keamanan-data-dari-peretasan/>
- UNMAHA. (2023b, December 15). Masa Depan Otomasi: Peran Robotika dalam Industri Manufaktur. Blog | Universitas Mahakarya Asia | UNMAHA. <https://blog.unmaha.ac.id/masa-depan-otomasi-peran-robotika-dalam-industri-manufaktur/>
- Verihubs. (2023, September 22). Apa Itu Machine Learning? Ini Contoh & Cara Kerjanya! Verihubs.com; Verihubs. <https://verihubs.com/blog/machine-learning-adalah>
- Verihubs. (2024, July 19). 6 Contoh Aplikasi Penerapan IoT di Berbagai Bidang. Verihubs.com; Verihubs. <https://verihubs.com/blog/aplikasi-penerapan-iotversha>.
- ivertsha. (2023, September 11). 7 Contoh Penerapan IoT yang Sering Digunakan di Kehidupan Sehari-hari. Berca Hardayaperkasa. <https://www.berca.co.id/2023/09/11/7-contoh-penerapan-iot-yang-sering-digunakan-didalam-kehidupan-sehari-hari/>
- Wicaksana, A. M. (2022, April 12). Robotika dan Otomasi - Robotika Indonesia. Sari Teknologi - Robotik. <https://sariteknologi.com/perbedaan-robotika-dan-otomasi/Wikipedia>.
- Wikipedia. (2004, March). cabang linguistika terhadap makna bahasa. Wikipedia.org; Wikimedia Foundation, Inc. <https://id.wikipedia.org/wiki/Semantik>
- Wikipedia. (2006, December 13). Realitas virtual. Wikipedia.org; Wikimedia Foundation, Inc. https://id.wikipedia.org/wiki/Realitas_virtual
- Wikipedia. (2019, March 16). Internet of things. Wikipedia;

- Wikimedia Foundation.
https://en.wikipedia.org/wiki/Internet_of_Things
- Wikipedia. (2022, December 20). Komputasi kuantum. Wikipedia.org; Wikimedia Foundation, Inc.
https://id.wikipedia.org/wiki/Komputasi_kuantum
- Wikipedia. (2023, May 10). Kecerdasan buatan. Wikipedia.
https://id.wikipedia.org/wiki/Kecerdasan_buatan
- Yuniar, R. (2022, July 3). Robot Otonom Akan Gantikan Manusia Tangani Pekerjaan Berbahaya. Suarasurabaya.net; Suara Surabaya.
<https://www.suarasurabaya.net/kelanakota/2022/robot-otonom-akan-gantikan-manusia-tangani-pekerjaan-berbahaya/>

BIODATA PENULIS



Rafika Sari, S.Si. M.Si.

Program Studi Informatika

Fakultas Ilmu Komputer

Universitas Bhayangkara Jakarta Raya

Penulis lahir di Jakarta tanggal 29 September 1989. Menyelesaikan Pendidikan S1 Fisika FMIPA UGM pada tahun 2012 dan S2 Sains Komputasi FMIPA ITB pada tahun 2017. Saat ini penulis mengabdikan diri sebagai Dosen di program studi Informatika Fakultas Ilmu Komputer Universitas Bhayangkara Jakarta Raya. Ketertarikan penulis pada bidang Data Science dan Artificial Intelligence sehingga telah banyak menghasilkan karya publikasi penelitian dibidang tersebut. Beberapa penelitian yang telah dilakukan didanai oleh internal perguruan tinggi dan juga Kemdikbudristek DIKTI. Selain peneliti, penulis juga aktif menulis buku dengan harapan dapat memberikan kontribusi positif bagi bangsa dan negara. Saat ini penulis juga aktif sebagai Editor in Chief pada Journal of Computer Science Contributions (JUCOSCO). Penulis dapat dihubungi melalui e-mail: rafik45sari@gmail.com

BIODATA PENULIS



Dr. Rakhmi Khalida, ST., M.M.S.I

Dosen Program Studi Informatika, Fakultas Ilmu Komputer
Universitas Bhayangkara Jakarta Raya

Rakhmi Khalida resmi bergabung dengan Universitas Bhayangkara Jakarta Raya tahun 2016, sebelumnya pernah menjadi junior programmer dan dosen luar biasa di Universitas Gunadarma. Ia lahir di Jakarta pada 4 September 1992. Ia menempuh Strata 1 jurusan teknik Informatika pada tahun 2010, melanjutkan Strata 2 jurusan magister Sistem Informasi dengan pembiayaan beasiswa unggulan pada tahun 2014, dan meraih gelar Doktor pada Program Teknologi Informasi di Universitas Gunadarma dengan pembiayaan beasiswa LPDP. Riset dan penelitiannya pada bidang ilmu komputer dan peminatan meliputi software development, kecerdasan buatan, pembelajaran mesin, augmented reality, interaksi manusia-mesin, dan multimedia. Penulis dapat dihubungi melalui email: rakhmikhhalida7@gmail.com

BIODATA PENULIS



Aida Fitriyani S.Kom., MMSI.
Program Studi Informatika
Fakultas Ilmu Komputer
Universitas Bhayangkara Jakarta Raya
Penulis dapat dihubungi melalui
e-mail: aida.fitriyani@gmail.com

BIODATA PENULIS



Achmad Ardiansyah, S.Kom, M.Kom

Dosen Program Studi Teknik Informatika
Fakultas Teknologi Informasi
Universitas Budi Luhur

Riwayat Pendidikan S1 Teknik Informatika (2007) Universitas Budi Luhur dan S2 Ilmu Komputer (2013) Universitas Budi Luhur. Saat ini bertugas sebagai dosen dan praktisi. Penelitian di bidang Integration System , Augmented Reality dan Internet of Things.

Penulis dapat dihubungi melalui e-mail:

ahd.ardiansyah@gmail.com

BIODATA PENULIS



Mukrodin, M.Kom

Program Studi Sistem Informas
Fakultas Sains dan Teknologi
Universitas Peradaban

Penulis lahir di Brebes tanggal 10 Maret 1980. Menyelesaikan Pendidikan S-1 pada program studi Sistem Informasi Universitas STIKUBANK Semarang pada tahun 2006, kemudian melanjutkan studi di program studi Magister Sistem Informasi Universitas Diponegoro Semarang. Penulis pernah bekerja dikonsultan IT PT. MEP (Muti Empati Profesional) di Purwokerto pada tahun 2006, tahun 2007 bekerja sebagai guru di SMA Negeri 1 Brebes, tahu 2014 memulai karir sebagai dosen di Universitas Peradaban Jawa Tengah sampai saat ini. Penulis dapat dihubngi melalui email: mukrodins@gmail.com

BIODATA PENULIS



Mepa Kurniasih, S.Kom, M.Kom

Dosen Program Studi Teknik Informatika
Fakultas Teknologi Informasi
Universitas Budi Luhur

Riwayat Pendidikan S1 Teknik Informatika (2007) Universitas Budi Luhur dan S2 Ilmu Komputer (2013) Universitas Budi Luhur. Saat ini bertugas sebagai dosen dan praktisi. Menguasai aspek di bidang Integration System dan Digital Marketing mengembangkan strategi pemasaran yang efektif melalui berbagai platform digital. Penulis dapat dihubungi melalui e-mail: mepa.kurnia@gmail.com