

AI yang Proaktif: Memahami Agentic AI dengan Contoh Sederhana

Onno W. Purbo
onno@indo.net.id
onno@itts.ac.id
twitter onnowpurbo

Institut Teknologi Tangerang Selatan (ITTS)
OnnoCenter
2025

Lisensi

Buku ini dilisensikan di bawah **Creative Commons Attribution 4.0 International License (CC BY 4.0)**.

Anda bebas untuk:

- **Berbagi** — menyalin dan menyebarkan materi ini dalam bentuk dan media apa pun.
- **Mengadaptasi** — mengubah, mengubah, dan membuat turunan dari materi ini untuk tujuan apa pun, termasuk tujuan komersial.



Dengan ketentuan:

- **Atribusi** — Anda harus memberikan kredit yang sesuai kepada penulis (*Onno W. Purbo*), menyertakan tautan lisensi, dan menunjukkan jika ada perubahan yang dilakukan. Anda dapat melakukannya dengan cara yang wajar, namun tidak dengan cara yang menyiratkan bahwa pemberi lisensi mendukung Anda atau penggunaan Anda.

Lisensi lengkap dapat dilihat di:

<https://creativecommons.org/licenses/by/4.0/>

© 2025 Onno W. Purbo.

Disclaimer

Buku ini disusun untuk tujuan **pendidikan dan informasi umum** mengenai *Agentic AI*; penulis tidak menjamin seluruh isi, ilustrasi, maupun kode Python selalu akurat atau bebas kesalahan, dan segala penerapan praktis menjadi **tanggung jawab pembaca** sepenuhnya; buku ini bukan panduan resmi untuk sistem kritis (misalnya keuangan, kesehatan, atau keamanan), dan penulis serta penerbit tidak bertanggung jawab atas risiko atau kerugian apa pun yang timbul; seluruh materi dilisensikan di bawah **Creative Commons Attribution 4.0 International (CC BY 4.0)** dengan kewajiban atribusi yang sesuai kepada penulis.

Daftar Isi

Introduction to Agentic AI.....	1
Daftar Isi.....	3
BAB 1: Pendahuluan.....	5
Apa itu AI?.....	5
Ilustrasi: Mesin Hitung vs. Manusia Menghitung.....	5
Contoh Konkret yang Mudah Dipahami.....	5
Script Python Sederhana.....	6
Perbedaan AI Tradisional & Agentic AI.....	7
Tabel Perbandingan.....	7
Contoh Skenario Sederhana.....	7
Ilustrasi dengan Python.....	7
Ringkasan Penting.....	8
BAB 2: Konsep Dasar Agentic AI.....	9
Tiga Pilar Utama Agentic AI.....	9
Ilustrasi Praktis.....	9
BAB 3: Perbedaan “Reactive AI” vs. “Agentic AI”.....	11
Ilustrasi Perbandingan.....	11
Contoh Nyata dalam Kehidupan.....	11
Ilustrasi Sederhana dengan Python.....	12
Penutup.....	12
BAB 4: Komponen Utama Agentic AI.....	13
Memory (Ingatan Kontekstual).....	13
Planning (Perencanaan Strategis).....	13
Action (Eksekusi Tindakan).....	13
Feedback Loop (Pembelajaran dari Hasil).....	13
Ringkasan dalam Tabel.....	14
Ilustrasi dengan Python.....	14
Penutup.....	15
BAB 5: Contoh Skenario Agentic AI.....	16
Asisten Belajar (Learning Assistant).....	16
Personal Finance Agent.....	16
Smart Home Agent.....	17
Ilustrasi Visual Konseptual.....	18
Penutup.....	18
BAB 6: Tools & Ekosistem Pendukung.....	19
Agent Frameworks.....	19
Keterhubungan dengan API.....	19
Ringkasan dalam Tabel.....	20
Ilustrasi Sederhana dengan Python.....	20
Penutup.....	20
BAB 7: Tantangan & Batasan.....	22
Bias (Bias dalam Penilaian).....	22

Keamanan (Safety & Security).....	22
Kontrol Manusia (Human-in-the-Loop).....	22
Ringkasan Tantangan dalam Tabel.....	22
Ilustrasi dengan Python.....	23
Penutup.....	23
BAB 8: Masa Depan Agentic AI.....	24
Dari Alat ke Kolaborator.....	24
Kemampuan Mengelola Tugas Sehari-hari.....	24
Ilustrasi Imajinatif: AI sebagai Co-pilot.....	25
Simulasi dengan Python.....	25
Penutup.....	25
BAB 9: Kesimpulan & Refleksi.....	27
Ringkasan.....	27
Refleksi Praktis.....	28
Penutup.....	28

Kata Pengantar

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa, karena atas rahmat dan karunia-Nya buku *AI yang Proaktif: Memahami Agentic AI dengan Contoh Sederhana* ini dapat tersusun. Buku ini lahir dari kegelisahan melihat bagaimana **Artificial Intelligence (AI) tradisional yang pasif** sudah tidak lagi memadai untuk menjawab kebutuhan zaman, sehingga perlu dipahami paradigma baru yaitu **Agentic AI**—*AI yang proaktif, adaptif, dan mampu bertindak layaknya kolaborator manusia*.

Melalui buku ini, penulis berusaha menghadirkan penjelasan yang **sederhana, praktis, dan aplikatif**, sehingga dapat diakses tidak hanya oleh akademisi atau praktisi teknologi, tetapi juga oleh masyarakat umum yang ingin memahami bagaimana AI dapat menjadi bagian dari kehidupan sehari-hari. Contoh-contoh nyata, ilustrasi dengan *script Python*, serta pembahasan tantangan dan masa depan Agentic AI disajikan untuk memberikan gambaran utuh mengenai potensi dan risiko yang ada.

Penulis menyadari sepenuhnya bahwa buku ini masih jauh dari sempurna. Oleh karena itu, kritik, saran, maupun masukan dari pembaca sangat penulis harapkan demi penyempurnaan karya ini di masa mendatang. Harapannya, buku ini dapat menjadi pijakan awal bagi siapa pun yang ingin mengenal, mempelajari, dan mengimplementasikan Agentic AI secara bijak, sehingga teknologi ini dapat menjadi **mitra cerdas yang mendukung kreativitas, produktivitas, serta pengambilan keputusan manusia—bukan menggantikannya**.

Jakarta, Agustus 2025

Onno W. Purbo

Executive Summary

Buku "**AI yang Proaktif: Memahami Agentic AI dengan Contoh Sederhana**" karya Onno W. Purbo menjelaskan pergeseran penting dari **AI tradisional yang pasif** menjadi **Agentic AI yang proaktif**—bukan sekadar *responder*, tetapi **aktor cerdas** yang dapat mengingat konteks (*memory*), merencanakan strategi (*planning*), mengeksekusi aksi nyata (*action*), dan belajar dari pengalaman (*feedback loop*). Konsep ini diperkaya dengan ilustrasi praktis menggunakan Python serta skenario nyata seperti **asisten belajar, personal finance agent, dan smart home agent**. Buku ini juga menyoroti **tantangan utama** seperti *bias*, keamanan, dan pentingnya **kontrol manusia (*human-in-the-loop*)** agar AI tetap aman dan bermanfaat. Ke depan, **Agentic AI diproyeksikan sebagai kolaborator atau *co-pilot*** dalam kehidupan sehari-hari, membantu manusia mengambil keputusan strategis sekaligus mengelola rutinitas. Pesan utama buku ini jelas: **AI seharusnya menjadi mitra cerdas yang mendukung kreativitas, produktivitas, dan pengambilan keputusan manusia**—bukan menggantikannya.

BAB 1: Pendahuluan

Apa itu AI?

Artificial Intelligence (AI) atau *Kecerdasan Buatan* adalah cabang ilmu komputer yang berfokus pada kemampuan mesin untuk melakukan tugas yang biasanya membutuhkan kecerdasan manusia. Misalnya, mengenali wajah, memahami bahasa, atau mengambil keputusan berdasarkan data. Secara sederhana, AI berusaha menjawab pertanyaan: *“Bisakah mesin berpikir dan bertindak layaknya manusia?”*

Bayangkan perbandingan sederhana:

- **Mesin hitung** hanya bisa menjumlahkan angka ketika kita memintanya.
- **Manusia** tidak hanya menghitung, tetapi juga menafsirkan konteks (misalnya, “apakah jumlah ini masuk akal dalam situasi tertentu?”).

Dengan kata lain, **AI adalah usaha membawa kemampuan manusia dalam berpikir ke dunia mesin.**

Ilustrasi: Mesin Hitung vs. Manusia Menghitung

Untuk memberikan gambaran yang mudah dipahami, mari kita lihat perbandingan berikut:

Aspek	Mesin Hitung (Calculator)	Manusia Menghitung
Cara bekerja	Menjawab sesuai input	Menjawab + menafsir konteks
Fleksibilitas	Terbatas pada operasi matematis	Bisa menyesuaikan strategi
Inisiatif	Pasif (hanya jika diminta)	Aktif, bisa mencari cara sendiri
Contoh	1234 + 5678 = 6912	“Jumlah ini masuk akal untuk laporan bulanan?”

Dari tabel ini terlihat jelas: mesin hitung hanyalah **alat pasif**, sementara manusia memiliki **kemampuan interpretasi dan pengambilan keputusan**. Nah, AI berusaha menjembatani jurang tersebut—agar mesin tidak hanya pasif, tetapi juga lebih “cerdas” dalam memahami dan bertindak.

Contoh Konkret yang Mudah Dipahami

1. **AI Pasif (Tradisional):**
 - Chatbot FAQ yang hanya menjawab jika ditanya.
 - Google Translate yang menerjemahkan kata demi kata.

2. **Agentic AI (AI yang lebih aktif):**

- Asisten digital yang **mengingat jadwal rapat**, lalu otomatis menjadwalkan ulang bila ada konflik.
- Aplikasi belanja yang **secara otomatis memesan ulang kopi** saat stok hampir habis.

Perbedaan ini akan lebih dibahas dalam bagian berikutnya, namun penting untuk dicatat bahwa **Agentic AI adalah langkah maju dari AI tradisional**.

Script Python Sederhana

Untuk memberi gambaran praktis, berikut contoh perbedaan antara *mesin hitung* biasa dengan *AI sederhana*:

```
# Contoh mesin hitung (pasif)
def kalkulator(a, b):
    return a + b

print("Hasil penjumlahan:", kalkulator(10, 5))
```

Output:

Hasil penjumlahan: 15

Di sini, program hanya menambah angka ketika diperintahkan. Sekarang, mari tambahkan sedikit “kecerdasan”:

```
# Contoh AI sederhana (aktif)
def asisten_keuangan(pengeluaran, batas):
    total = sum(pengeluaran)
    if total > batas:
        return f"⚠️ Pengeluaran {total} melebihi batas {batas}.  
Kurangi belanja!"
    else:
        return f"✅ Pengeluaran {total} masih aman."

print(asisten_keuangan([50, 200, 100], 300))
```

Output:

⚠️ Pengeluaran 350 melebihi batas 300. Kurangi belanja!

Contoh ini masih sangat sederhana, tetapi sudah menunjukkan perbedaan: mesin tidak hanya menghitung, tetapi juga **memberi rekomendasi berdasarkan konteks**—sebuah ciri awal dari *agentic behavior*.

Perbedaan AI Tradisional & Agentic AI

AI Tradisional (sering disebut *Reactive AI*) pada dasarnya bersifat **pasif**. Ia hanya merespons ketika diberi input. Misalnya, sebuah chatbot FAQ hanya akan menjawab pertanyaan jika kita mengetikkannya.

Sebaliknya, **Agentic AI** bersifat **proaktif**. Ia tidak sekadar menunggu instruksi, tetapi bisa *mengambil inisiatif*, mengantisipasi kebutuhan, serta melakukan aksi nyata. Contohnya, sebuah *digital assistant* yang secara otomatis memesan ulang kopi favorit Anda ketika stok di rumah hampir habis.

Tabel Perbandingan

Aspek	AI Tradisional (Pasif)	Agentic AI (Aktif)
Pola Interaksi	Menjawab bila ditanya	Bisa memulai interaksi, mengingatkan, atau memberi saran
Peran	Responder (<i>jawaban atas input</i>)	Kolaborator (<i>bekerja bersama manusia</i>)
Contoh Sederhana	Chatbot FAQ	Asisten pribadi digital yang menjadwalkan ulang rapat otomatis
Kemampuan Belajar	Terbatas, lebih banyak “rule-based”	Adaptif, belajar dari pengalaman & umpan balik
Orientasi	Reaktif	Proaktif & inisiatif

Contoh Skenario Sederhana

- AI Tradisional (Pasif):**
Anda bertanya ke chatbot: “Kapan toko buka?” → AI menjawab: “Pukul 9 pagi.”
- Agentic AI (Aktif):**
AI mendeteksi bahwa Anda sering mencari jam buka toko, lalu otomatis mengingatkan:
“Besok toko buka pukul 9 pagi, saya sudah siapkan pengingat di kalender Anda.”

Ilustrasi dengan Python

Untuk memperjelas, bayangkan dua potongan kode Python berikut:

AI Tradisional (Pasif) – hanya merespons input:

```
def chatbot_faq(question):  
    faq = {
```

```

        "jam buka": "Toko buka pukul 9 pagi",
        "alamat": "Jl. Merdeka No. 10"
    }
    return faq.get(question.lower(), "Maaf, saya tidak tahu")

# Contoh interaksi
print(chatbot_faq("jam buka"))
# Output: Toko buka pukul 9 pagi

```

Agentic AI (Aktif) – bisa mengambil inisiatif:

```

import datetime

def agentic_ai(user_history):
    if "cari jam buka toko" in user_history:
        return f"Besok toko buka pukul 9 pagi. Saya sudah tambahkan ke kalender: {datetime.date.today() + datetime.timedelta(days=1)}"
    else:
        return "Ada yang bisa saya bantu?"

# Contoh interaksi
print(agentic_ai(["cari jam buka toko"]))
# Output: Besok toko buka pukul 9 pagi. Saya sudah tambahkan ke kalender: 2025-08-25

```

Ringkasan Penting

- **AI tradisional = pasif** → menunggu instruksi.
- **Agentic AI = aktif** → bisa *mengantisipasi kebutuhan* dan *bertindak*.
- Bagi praktisi bisnis, perbedaan ini berarti: **AI pasif hanya efisien**, sedangkan **Agentic AI bisa menciptakan nilai baru** dengan membantu pengambilan keputusan dan otomatisasi.

BAB 2: Konsep Dasar Agentic AI

Agentic AI dapat didefinisikan sebagai **kecerdasan buatan yang tidak hanya menerima perintah, tetapi juga mampu mengambil keputusan, bertindak, dan belajar dari interaksi dengan lingkungannya**. Dengan kata lain, *agentic AI* memiliki kemampuan untuk menjadi lebih dari sekadar sistem pasif; ia bisa menjadi “aktor” yang aktif, mirip dengan manusia yang memiliki persepsi, penalaran, dan aksi dalam kehidupan sehari-hari.

Konsep ini penting karena AI tidak lagi sebatas mesin yang menunggu input, melainkan dapat **berinisiatif dan beradaptasi**. Misalnya, bukan hanya sekedar menjawab pertanyaan pengguna, melainkan mampu memprediksi kebutuhan, mengambil langkah pencegahan, bahkan mengeksekusi tindakan tanpa harus diminta secara eksplisit.

Tiga Pilar Utama Agentic AI

Untuk memahami bagaimana *agentic AI* bekerja, terdapat **tiga pilar utama** yang menjadi fondasinya:

- 1. **Persepsi (Perception)** → kemampuan AI **memahami lingkungannya**, baik melalui data sensor, teks, suara, maupun gambar.
- 2. **Penalaran (Reasoning)** → kemampuan AI **merencanakan langkah** berdasarkan informasi yang dimiliki, termasuk membuat prioritas atau strategi.
- 3. **Aksi (Action)** → kemampuan AI untuk **melakukan sesuatu** secara nyata, baik dalam bentuk interaksi digital maupun fisik.

Tabel berikut menggambarkan hubungan ketiga pilar ini:

Pilar	Fungsi Utama	Contoh dalam Kehidupan Sehari-hari
Persepsi	Mengumpulkan & memahami informasi	Kamera robot vacuum mendeteksi kotoran
Penalaran	Membuat keputusan & strategi	Robot vacuum memilih rute pembersihan
Aksi	Melaksanakan keputusan	Robot vacuum bergerak menyedot debu

Ilustrasi Praktis

Salah satu contoh sederhana adalah **robot vacuum**. Saat robot melihat kotoran di lantai (*persepsi*), ia kemudian memutuskan jalur pembersihan yang paling efisien (*penalaran*), lalu mulai bergerak untuk membersihkan area tersebut (*aksi*). Proses ini mirip dengan bagaimana manusia bertindak: melihat masalah → berpikir solusi → mengeksekusi tindakan.

Contoh lainnya dapat ditemukan dalam dunia digital. Misalnya:

- **AI personal finance** yang bukan hanya mencatat pengeluaran, tetapi juga *merekomendasikan strategi menabung* dan *otomatis mengalihkan dana* ke tabungan jika saldo mencukupi.
- **Asisten belajar berbasis AI** yang tidak hanya menjawab pertanyaan siswa, tetapi juga *membuat jadwal belajar*, *memberi pengingat*, dan *menyesuaikan materi* jika siswa mengalami kesulitan.

Dengan demikian, **agentic AI membuka kemungkinan AI menjadi kolaborator aktif**, bukan sekadar alat yang menunggu instruksi.

BAB 3: Perbedaan “Reactive AI” vs. “Agentic AI”

Dalam perkembangan kecerdasan buatan, terdapat dua paradigma besar yang membedakan cara kerja sistem: **Reactive AI** dan **Agentic AI**. Perbedaan keduanya dapat dilihat dari tingkat kemandirian dan proaktivitas yang dimiliki.

Reactive AI adalah jenis AI yang *hanya merespons input* dari pengguna atau lingkungan, tanpa memiliki kemampuan untuk memprediksi, merencanakan, ataupun mengambil inisiatif. Ia berfungsi mirip dengan **kalkulator**: ketika kita memberikan angka dan operasi, sistem hanya menampilkan hasil tanpa “memikirkan” apa kebutuhan kita di luar instruksi tersebut.

Sebaliknya, **Agentic AI** bersifat **proaktif dan adaptif**. Artinya, sistem ini mampu *mengambil inisiatif*, menyesuaikan diri dengan konteks, serta melakukan tindakan meskipun tidak secara langsung diperintahkan. Fungsinya lebih mirip dengan **sekretaris pribadi** yang tidak hanya menuliskan jadwal, tetapi juga *mengatur ulang pertemuan jika ada konflik* dan *memberikan pengingat sebelum agenda berlangsung*.

Ilustrasi Perbandingan

Untuk mempermudah pemahaman, berikut perbedaan keduanya dalam bentuk tabel:

Aspek	Reactive AI	Agentic AI
Fungsi Utama	Merespons input sesuai perintah	Proaktif, adaptif, mampu mengambil keputusan
Contoh	Kalkulator: hanya menghitung bila diminta	Asisten digital: otomatis mengingatkan dan menjadwalkan ulang
Keterbatasan	Tidak bisa belajar dari konteks	Masih bergantung pada data & batasan etika
Ilustrasi Sederhana	Kamu minta jadwal → AI hanya menampilkan	AI melihat konflik jadwal → otomatis melakukan penyesuaian

Contoh Nyata dalam Kehidupan

- Reactive AI:** Chatbot FAQ di situs web yang hanya memberikan jawaban sesuai pertanyaan yang kita ajukan. Jika pertanyaannya berbeda dari pola yang dipahami, sistem gagal menjawab.
- Agentic AI:** Asisten virtual (misalnya di smartphone) yang bukan hanya menampilkan jadwal rapat, tetapi juga *mengirim email otomatis* jika kamu terlambat, atau *mencarikan waktu alternatif* untuk menghindari bentrokan jadwal.

Dengan demikian, **perbedaan kunci terletak pada kemampuan untuk mengambil inisiatif**. Reactive AI berhenti pada “jawaban”, sementara Agentic AI melangkah lebih jauh ke “tindakan”.

Ilustrasi Sederhana dengan Python

Berikut adalah contoh sederhana bagaimana **Reactive AI** dan **Agentic AI** bisa dimodelkan secara konseptual menggunakan Python:

```
# Reactive AI: hanya merespons jika diberi input
def reactive_ai(command):
    if command == "tampilkan_jadwal":
        return "Jadwal hari ini: 09.00 Meeting, 13.00 Presentasi"
    else:
        return "Perintah tidak dikenali."

# Agentic AI: proaktif, bisa mendeteksi konflik dan bertindak
def agentic_ai(schedule, new_event):
    if new_event in schedule:
        return f"Konflik terdeteksi dengan {new_event}. Saya akan menjadwalkan ulang."
    else:
        schedule.append(new_event)
        return f"Event {new_event} berhasil ditambahkan tanpa konflik."

# Contoh penggunaan
print(reactive_ai("tampilkan_jadwal"))
print(agentic_ai(["09.00 Meeting", "13.00 Presentasi"], "09.00 Meeting"))
```

Output yang dihasilkan:

```
Jadwal hari ini: 09.00 Meeting, 13.00 Presentasi
Konflik terdeteksi dengan 09.00 Meeting. Saya akan menjadwalkan ulang.
```

Contoh kode di atas memperlihatkan bahwa **Reactive AI hanya menampilkan data** sesuai permintaan, sementara **Agentic AI mencoba memahami situasi (konflik jadwal) dan mengambil inisiatif untuk menyelesaikan masalah**.

Penutup

Dari uraian ini, jelas bahwa **Reactive AI berfungsi sebagai “responden”**, sedangkan **Agentic AI berperan sebagai “aktor” yang dapat menginisiasi tindakan**. Bagi praktisi, pemahaman perbedaan ini krusial karena menentukan strategi implementasi AI di berbagai bidang, baik itu di pendidikan, kesehatan, bisnis, maupun teknologi rumah pintar.

BAB 4: Komponen Utama Agentic AI

Agar dapat berfungsi secara efektif, **Agentic AI** dibangun di atas beberapa komponen inti yang membuatnya mampu memahami konteks, membuat keputusan, melakukan tindakan nyata, serta belajar dari hasil tindakannya. Empat komponen ini adalah: **Memory**, **Planning**, **Action**, dan **Feedback Loop**.

Komponen-komponen ini saling berhubungan, membentuk siklus cerdas yang memungkinkan AI untuk tidak hanya menjadi *responder*, tetapi juga *aktor* aktif yang beradaptasi dengan lingkungan.

Memory (*Ingatan Kontekstual*)

Memory memungkinkan AI untuk *mengingat pengalaman atau informasi sebelumnya*. Tanpa memori, AI hanya akan merespons secara statis tanpa mempertimbangkan riwayat interaksi.

- **Contoh sederhana:** AI yang mengingat pesanan kopi favorit pengguna, sehingga bisa otomatis menawarkan ulang tanpa perlu ditanya.
- **Nilai praktis:** Dalam bisnis, AI dapat mengingat preferensi pelanggan, sehingga meningkatkan *customer experience*.

Planning (Perencanaan Strategis)

Planning adalah kemampuan AI untuk *merancang langkah-langkah atau strategi* dalam mencapai tujuan. Ini adalah perbedaan kunci antara AI pasif dan AI yang bersifat agentic.

- **Contoh sederhana:** AI menyiapkan rencana belajar selama 30 hari menjelang ujian.
- **Nilai praktis:** Di dunia industri, AI bisa menyusun jadwal pemeliharaan mesin agar tidak terjadi downtime.

Action (Eksekusi Tindakan)

Action merepresentasikan kemampuan AI untuk *bertindak di dunia nyata atau digital*. Tindakan ini bisa berupa pengiriman pesan, mengeksekusi perintah di aplikasi, hingga mengendalikan perangkat fisik.

- **Contoh sederhana:** AI yang otomatis mengirim email *follow-up* setelah rapat.
- **Nilai praktis:** Dalam bisnis, AI dapat mengeksekusi transaksi stok barang secara otomatis ketika persediaan menipis.

Feedback Loop (Pembelajaran dari Hasil)

Feedback Loop adalah mekanisme AI untuk *mengevaluasi hasil tindakannya* dan belajar dari kesalahan maupun keberhasilan. Komponen ini yang membuat AI bisa menjadi semakin adaptif seiring waktu.

- **Contoh sederhana:** Jika email *follow-up* tidak direspon, AI mencoba metode lain seperti mengirim pengingat WhatsApp.
- **Nilai praktis:** Dalam pendidikan, AI bisa menyesuaikan gaya belajar jika siswa tampak kesulitan memahami materi tertentu.

Ringkasan dalam Tabel

Komponen	Fungsi Utama	Contoh Sederhana	Contoh Praktis
Memory	Mengingat konteks	Ingat pesanan kopi	Riwayat preferensi pelanggan
Planning	Membuat strategi	Rencana belajar 30 hari	Jadwal pemeliharaan mesin
Action	Melakukan sesuatu	Kirim email otomatis	Transaksi stok barang
Feedback Loop	Belajar dari hasil	Ubah cara follow-up	AI guru yang adaptif

Ilustrasi dengan Python

Untuk memberikan gambaran sederhana, mari kita lihat implementasi konsep ini dalam bentuk kode Python:

```
class AgenticAI:
    def __init__(self):
        self.memory = []

    # Memory: menyimpan pengalaman
    def remember(self, info):
        self.memory.append(info)
        return f"Mengingat: {info}"

    # Planning: membuat strategi sederhana
    def plan(self, goal):
        return f"Membuat rencana untuk: {goal}"

    # Action: melakukan eksekusi
    def act(self, action):
        return f"Melakukan aksi: {action}"

    # Feedback Loop: evaluasi hasil
    def feedback(self, result):
        if result == "gagal":
            return "Mengubah strategi, mencoba pendekatan lain."
        return "Tindakan berhasil, lanjutkan strategi."

# Contoh penggunaan
```

```
ai = AgenticAI()
print(ai.remember("Pesanan kopi latte"))
print(ai.plan("Belajar 30 hari untuk ujian"))
print(ai.act("Mengirim email follow-up"))
print(ai.feedback("gagal"))
```

Output yang dihasilkan:

```
Mengingat: Pesanan kopi latte
Membuat rencana untuk: Belajar 30 hari untuk ujian
Melakukan aksi: Mengirim email follow-up
Mengubah strategi, mencoba pendekatan lain.
```

Kode sederhana ini memperlihatkan bagaimana **empat komponen utama Agentic AI** saling melengkapi: mulai dari mengingat konteks, membuat rencana, melakukan tindakan, hingga mengevaluasi hasil untuk meningkatkan kinerja.

Penutup

Dengan adanya **Memory, Planning, Action, dan Feedback Loop**, AI tidak lagi menjadi sekadar mesin pasif, tetapi **sistem cerdas yang mampu beradaptasi, berinisiatif, dan terus belajar**. Bagi praktisi, memahami komponen ini penting untuk merancang solusi AI yang benar-benar memberi nilai tambah, baik di sektor pendidikan, industri, maupun layanan publik.

BAB 5: Contoh Skenario Agentic AI

Setelah memahami komponen utama, kini saatnya melihat bagaimana **Agentic AI** bekerja dalam skenario nyata. Skenario ini membantu menjelaskan bagaimana AI tidak sekadar menunggu perintah, tetapi **aktif mengambil inisiatif, beradaptasi, dan bertindak**. Tiga contoh populer yang bisa menggambarkan hal ini adalah **Asisten Belajar, Personal Finance Agent, dan Smart Home Agent**.

Asisten Belajar (Learning Assistant)

Asisten Belajar adalah contoh sederhana namun sangat relevan. AI tidak hanya menjawab pertanyaan siswa, tetapi juga **membuat rencana belajar jangka panjang, menyesuaikan materi, dan memberikan pengingat otomatis**.

- **Contoh konkrit:** Seorang mahasiswa yang mempersiapkan ujian 30 hari ke depan. AI bisa menyusun jadwal harian (misalnya 2 jam belajar tiap malam), mengingatkan jika ada jadwal yang terlewat, bahkan mengubah strategi jika mahasiswa kesulitan pada topik tertentu.

Script Python sederhana:

```
import datetime

def generate_study_plan(days, subject):
    today = datetime.date.today()
    plan = []
    for i in range(days):
        date = today + datetime.timedelta(days=i)
        plan.append(f"{date}: Belajar {subject} - Hari ke-{i+1}")
    return plan

# Contoh penggunaan
study_plan = generate_study_plan(5, "Matematika")
for p in study_plan:
    print(p)
```

Output (contoh):

```
2025-08-24: Belajar Matematika - Hari ke-1
2025-08-25: Belajar Matematika - Hari ke-2
```

Personal Finance Agent

Personal Finance Agent membantu mengatur pengeluaran bulanan dengan cara **memantau transaksi, memberi peringatan jika pengeluaran berlebihan, dan menyarankan alokasi dana**. Tidak berhenti di sana, Agentic AI bahkan bisa *otomatis memindahkan dana* ke tabungan atau investasi sesuai kebijakan pengguna.

- **Contoh konkret:** Jika pengeluaran makan melebihi 30% dari total pendapatan, AI bisa memberikan notifikasi dan rekomendasi strategi.

Tabel sederhana perbandingan:

Fungsi	Reactive AI	Agentic AI
Catat pengeluaran	Hanya menampilkan laporan	Memberi saran & otomatis alokasi dana
Monitoring	Pasif, menunggu input	Aktif, mendeteksi pola boros
Intervensi	Tidak ada	Mengirim pengingat atau memindahkan dana

Script Python sederhana:

```
def finance_agent(income, expenses):
    limit = 0.3 * income
    if expenses > limit:
        return "Pengeluaran berlebihan! Kurangi belanja makan di luar."
    else:
        return "Keuangan aman, sisihkan sebagian untuk tabungan."

print(finance_agent(5000000, 2000000)) # 5 juta income, 2 juta untuk makan
```

Output:

```
Pengeluaran berlebihan! Kurangi belanja makan di luar.
```

Smart Home Agent

Smart Home Agent bekerja di ranah rumah pintar, di mana AI **mengendalikan perangkat secara otomatis**. Berbeda dengan sekadar *otomasi berbasis jadwal*, Agentic AI bisa mengambil keputusan berdasarkan konteks.

- **Contoh konkret:** Ketika pengguna pulang kerja, AI mendeteksi lokasi GPS di smartphone, menyalakan lampu ruang tamu, serta menyalakan AC sebelum pengguna masuk rumah.

Script Python sederhana (simulasi logika):

```
def smart_home_agent(user_location, home_status):
    if user_location == "dekat rumah" and home_status == "kosong":
        return "Lampu & AC dinyalakan."
    return "Rumah tetap standby."

print(smart_home_agent("dekat rumah", "kosong"))
```

Output:

Lampu & AC dinyalakan.

Ilustrasi Visual Konseptual

Skenario-skenario di atas bisa digambarkan dengan alur sederhana:

AI melihat → memutuskan → bertindak

Contohnya:

- AI *melihat* pola pengeluaran berlebih → AI *memutuskan* memberi peringatan → AI *bertindak* dengan rekomendasi alokasi.
- AI *melihat* jadwal belajar → AI *memutuskan* strategi → AI *bertindak* dengan pengingat otomatis.

Penutup

Contoh-contoh ini menunjukkan bahwa **Agentic AI tidak berhenti pada “jawaban”, melainkan melangkah ke “aksi nyata”**. Baik sebagai asisten belajar, pengatur keuangan pribadi, maupun pengendali rumah pintar, kehadirannya bisa menjadi *co-pilot* dalam kehidupan sehari-hari.

BAB 6: Tools & Ekosistem Pendukung

Agar **Agentic AI** dapat berfungsi optimal, dibutuhkan **tools** dan **ekosistem pendukung** yang memungkinkan AI untuk *mengingat, merencanakan, dan bertindak* secara terintegrasi. Dalam praktiknya, ekosistem ini terdiri dari dua elemen utama: **frameworks AI** dan **keterhubungan dengan API**. Kedua hal ini saling melengkapi, sehingga AI bisa bekerja tidak hanya di ruang sempit (isolated system), tetapi juga terhubung dengan aplikasi dunia nyata.

Agent Frameworks

Agent Frameworks adalah *kerangka kerja* yang membantu membangun dan mengelola perilaku AI yang bersifat agentic. Framework ini menyediakan *alat bantu* agar AI bisa memiliki **memori, perencanaan, dan kemampuan eksekusi** tanpa harus membangun semuanya dari nol.

Beberapa framework populer:

- **LangChain** → memudahkan integrasi AI dengan data, API, maupun aplikasi eksternal.
- **AutoGen** → memungkinkan AI *berkolaborasi* dengan AI lain atau manusia untuk menyelesaikan tugas kompleks.

Contoh sederhana: Dengan LangChain, kita bisa membuat AI yang dapat membaca dokumen penelitian, menyimpan ringkasannya ke dalam memori, lalu menjawab pertanyaan pengguna berdasarkan konteks tersebut.

Keterhubungan dengan API

Selain framework, **API (Application Programming Interface)** adalah *jembatan* yang memungkinkan AI berinteraksi dengan aplikasi lain. Tanpa API, AI hanya sebatas “pintar dalam dirinya sendiri”, tetapi tidak bisa *bertindak* di dunia nyata.

- **Contoh praktis:** AI menggabungkan data dari *kalender* dan *email*. Jika mendeteksi rapat baru, AI bisa otomatis mengirim pengingat ke peserta.
- **Nilai tambah untuk praktisi:** API membuat AI lebih bermanfaat karena bisa mengakses sistem bisnis (ERP, CRM), layanan publik (peta, cuaca), maupun layanan pribadi (banking, e-commerce).

Ringkasan dalam Tabel

Elemen	Fungsi Utama	Contoh Penggunaan
LangChain	Integrasi AI dengan data & aplikasi	AI membaca dokumen + menjawab pertanyaan
AutoGen	Kolaborasi multi-agent (AI ↔ AI / AI ↔ manusia)	AI riset + AI penulis menyiapkan laporan
API	Penghubung AI dengan aplikasi eksternal	AI menggabungkan kalender & email untuk reminder otomatis

Ilustrasi Sederhana dengan Python

Berikut contoh Python sederhana yang mensimulasikan **keterhubungan AI dengan API kalender** dan email:

```
# Simulasi API Kalender & Email
def get_calendar_events():
    return ["Meeting 09.00", "Presentasi 13.00"]

def send_email(reminder):
    return f"Email dikirim: {reminder}"

# Agentic AI yang memanfaatkan API
def ai_reminder_agent():
    events = get_calendar_events()
    for e in events:
        reminder = f"Jangan lupa: {e}"
        print(send_email(reminder))

# Contoh penggunaan
ai_reminder_agent()
```

Output yang dihasilkan:

```
Email dikirim: Jangan lupa: Meeting 09.00
Email dikirim: Jangan lupa: Presentasi 13.00
```

Kode ini sederhana, tetapi menggambarkan bagaimana **Agentic AI menggunakan API** untuk *mengambil data (kalender)* dan *bertindak (mengirim email)*. Dalam implementasi nyata, integrasi ini bisa diperluas ke aplikasi bisnis atau sistem IoT.

Penutup

Dari uraian di atas, jelas bahwa **frameworks dan API adalah tulang punggung Agentic AI**. Framework memberikan *struktur* agar AI dapat berpikir dan belajar, sementara API

membuka jalan bagi AI untuk *bertindak* di dunia nyata. Kombinasi keduanya menjadikan Agentic AI bukan sekadar konsep, melainkan solusi praktis yang dapat diterapkan di berbagai bidang, mulai dari pendidikan, keuangan, hingga smart home dan industri.

BAB 7: Tantangan & Batasan

Meskipun **Agentic AI** menjanjikan potensi besar, ada sejumlah **tantangan dan batasan** yang harus diperhatikan. Sama halnya seperti “anak kecil pintar” yang masih membutuhkan arahan, AI jenis ini perlu **kontrol manusia** agar tidak salah langkah. Tiga aspek penting yang paling sering dibahas adalah **bias**, **keamanan**, dan **kontrol manusia**.

Bias (Bias dalam Penilaian)

AI belajar dari data, dan **data sering kali tidak netral**. Jika data yang digunakan memiliki bias, maka keputusan AI pun bisa keliru.

- **Contoh:** Sistem rekrutmen berbasis AI yang lebih sering merekomendasikan kandidat pria dibanding wanita karena data historis perusahaan yang bias.
- **Risiko:** Keputusan yang tidak adil, diskriminatif, atau salah arah.

Keamanan (Safety & Security)

Keamanan menjadi tantangan besar karena Agentic AI dapat melakukan *aksi nyata*. Jika tidak diawasi, AI bisa salah mengeksekusi perintah yang membahayakan pengguna.

- **Contoh:** Smart home agent yang salah menyalakan kompor listrik karena mendeteksi sinyal sensor yang keliru.
- **Risiko:** Potensi bahaya fisik maupun kerugian finansial.

Kontrol Manusia (Human-in-the-Loop)

Meski AI semakin cerdas, **kontrol manusia tetap penting**. Agentic AI sebaiknya dirancang dengan *human-in-the-loop*, yaitu mekanisme yang memastikan manusia bisa **memberi arahan, membatasi, atau menghentikan tindakan AI** jika diperlukan.

- **Contoh:** AI yang mengatur investasi otomatis harus tetap menunggu konfirmasi pengguna untuk transaksi besar.
- **Analogi:** Sama seperti “anak kecil pintar” yang butuh bimbingan, AI memerlukan *rule* dan *guardrails*.

Ringkasan Tantangan dalam Tabel

Tantangan	Risiko Utama	Contoh Kasus
Bias	Keputusan tidak adil	AI rekrutmen pilih kandidat pria lebih banyak
Keamanan	Aksi berbahaya atau merugikan	Smart home salah menyalakan kompor

Kontrol manusia	AI bertindak tanpa pengawasan	Investasi otomatis tanpa persetujuan
------------------------	-------------------------------	--------------------------------------

Ilustrasi dengan Python

Berikut contoh Python sederhana untuk memperlihatkan bagaimana *kontrol manusia* bisa menjadi filter sebelum AI melakukan aksi:

```
# Simulasi Agentic AI dengan Human-in-the-Loop

def ai_decision(task, confidence):
    if confidence < 0.7:
        return f"Keputusan untuk '{task}' ditunda, perlu verifikasi manusia."
    return f"AI mengeksekusi tugas: {task}"

# Contoh penggunaan
print(ai_decision("Menyalakan lampu rumah", 0.9))    # AI yakin
print(ai_decision("Mengirim transfer bank", 0.6))    # AI ragu
```

Output yang dihasilkan:

```
AI mengeksekusi tugas: Menyalakan lampu rumah
Keputusan untuk 'Mengirim transfer bank' ditunda, perlu verifikasi manusia.
```

Kode sederhana ini menggambarkan prinsip bahwa **AI tidak boleh bertindak penuh tanpa verifikasi manusia**, terutama dalam konteks berisiko tinggi.

Penutup

Dengan potensi besarnya, **Agentic AI tetap memiliki keterbatasan**. **Bias** dapat menyesatkan, **keamanan** harus dijaga agar tidak terjadi aksi berbahaya, dan **kontrol manusia** mutlak diperlukan untuk memastikan AI tetap berada dalam koridor yang benar. Oleh karena itu, pengembangan Agentic AI perlu menyeimbangkan antara *otonomi sistem* dengan *kendali manusia*, agar AI bisa menjadi kolaborator yang aman dan dapat dipercaya.

BAB 8: Masa Depan Agentic AI

Perkembangan **Agentic AI** mengarah pada sebuah paradigma baru: **AI bukan hanya alat, terapi kolaborator**. Jika sebelumnya AI dipandang sebagai “mesin penjawab” yang pasif, di masa depan AI akan berperan sebagai *partner cerdas* yang mampu bekerja berdampingan dengan manusia. Dengan kata lain, AI bukan lagi sekadar eksekutor, melainkan menjadi **“co-pilot” dalam pekerjaan dan kehidupan sehari-hari**.

Dari Alat ke Kolaborator

Di masa lalu, AI terbatas pada fungsi reaktif—sekadar memproses perintah yang diberikan. Namun, dengan kemampuan *persepsi, penalaran, aksi, dan feedback loop*, Agentic AI berkembang menjadi **entitas proaktif yang mampu berbagi tanggung jawab dengan manusia**.

- **Contoh di dunia kerja:** AI dapat menjadi asisten riset yang bukan hanya mencari literatur, tetapi juga *menyusun ringkasan kritis, serta memberikan saran metodologi*.
- **Contoh di kehidupan pribadi:** AI dapat membantu mengatur jadwal harian, mengingatkan aktivitas kesehatan, bahkan *mengantisipasi kebutuhan* sebelum pengguna menyadarinya.

Kemampuan Mengelola Tugas Sehari-hari

Masa depan Agentic AI juga ditandai dengan **kemampuan mengelola tugas rutin secara otomatis**, sehingga manusia bisa fokus pada pekerjaan yang lebih strategis dan kreatif.

- **Rumah tangga** → AI mengatur smart home: lampu, suhu ruangan, hingga belanja bulanan.
- **Pendidikan** → AI mempersonalisasi gaya belajar tiap siswa.
- **Bisnis** → AI bertindak sebagai *digital manager* yang mengatur jadwal tim, memantau KPI, bahkan memprediksi risiko.

Tabel berikut menggambarkan pergeseran peran AI:

Era	Peran AI	Contoh
AI Tradisional	Alat pasif	Kalkulator, chatbot FAQ
Agentic AI Saat Ini	Asisten proaktif	Reminder otomatis, email follow-up
Agentic AI Masa Depan	Kolaborator strategis	<i>Co-pilot</i> kerja, manajer digital, guru adaptif

Ilustrasi Imajinatif: AI sebagai Co-pilot

Bayangkan seorang profesional yang sibuk dengan berbagai proyek. Di masa depan:

- AI **melihat** jadwal rapat tumpang tindih,
- AI **memutuskan** memindahkan salah satunya ke slot waktu yang kosong, AI **bertindak** dengan mengirim email otomatis ke semua peserta.

Lebih jauh lagi, AI bisa menjadi *co-pilot* pekerjaan: menyarankan strategi bisnis, mengingatkan risiko hukum, hingga membantu mengambil keputusan berbasis data real-time.

Simulasi dengan Python

Berikut simulasi sederhana bagaimana **AI masa depan dapat berperan sebagai co-pilot** dalam pekerjaan:

```
# Simulasi AI Co-pilot
def ai_copilot(task, context):
    if task == "meeting":
        if "conflict" in context:
            return "Konflik jadwal terdeteksi. Rapat dipindahkan ke pukul 15.00."
        else:
            return "Semua rapat berjalan sesuai jadwal."
    elif task == "finance":
        if context["spending"] > context["budget"]:
            return "Pengeluaran melebihi anggaran! Saran: kurangi biaya promosi."
        else:
            return "Keuangan stabil. Bisa dialokasikan ke R&D."
    return "Tugas tidak dikenali."

# Contoh penggunaan
print(ai_copilot("meeting", "conflict"))
print(ai_copilot("finance", {"spending": 120, "budget": 100}))
```

Output yang dihasilkan:

```
Konflik jadwal terdeteksi. Rapat dipindahkan ke pukul 15.00.
Pengeluaran melebihi anggaran! Saran: kurangi biaya promosi.
```

Walaupun sederhana, kode ini mencerminkan ide bahwa **AI tidak hanya mendukung, tetapi juga mengambil inisiatif dalam pengambilan keputusan.**

Penutup

Masa depan Agentic AI bukan lagi tentang “alat pintar” yang sekadar membantu, melainkan tentang **kolaborasi cerdas antara manusia dan mesin**. Dengan kemampuan adaptif dan

proaktif, AI akan menjadi **co-pilot** yang mendampingi manusia dalam pekerjaan, pembelajaran, hingga kehidupan sehari-hari. Tantangan etika dan keamanan tetap ada, tetapi jika diarahkan dengan benar, Agentic AI bisa menjadi salah satu inovasi paling transformatif dalam sejarah teknologi.

BAB 9: Kesimpulan & Refleksi

Setelah menelusuri perjalanan dari konsep dasar hingga skenario penerapan, jelas bahwa **Agentic AI adalah langkah berikutnya setelah chatbot tradisional**. Jika chatbot hanya sebatas *responder pasif*—memberi jawaban ketika ditanya—maka Agentic AI mampu **mengambil keputusan, bertindak, dan beradaptasi secara proaktif**. Transformasi ini mengubah AI dari sekadar *alat bantu* menjadi **partner kerja** yang dapat berkolaborasi dengan manusia dalam berbagai aspek kehidupan.

Bagi pemula, memahami Agentic AI tidak harus dimulai dari aspek teknis yang rumit. Cukup dengan menguasai **konsep inti, ilustrasi sehari-hari, dan contoh penerapan sederhana**, seseorang sudah bisa melihat potensinya. Misalnya, perbedaan mendasar antara chatbot FAQ (AI pasif) dengan AI yang otomatis penjadwalan ulang rapat jika ada konflik (Agentic AI) sudah cukup untuk menggambarkan pergeseran paradigma ini.

Lebih penting lagi, pesan utama yang perlu ditanamkan adalah: **“AI bisa jadi teman kerja, bukan pengganti manusia.”** Dengan pendekatan ini, AI diposisikan sebagai kolaborator yang mendukung kreativitas, produktivitas, dan pengambilan keputusan, bukan sebagai ancaman bagi eksistensi manusia.

Ringkasan

Bab	Inti Utama	Contoh Praktis
1. Pendahuluan	AI berkembang dari alat pasif ke Agentic AI	Chatbot FAQ vs AI yang memesan ulang kopi otomatis
2. Konsep Dasar	AI dengan persepsi, penalaran, dan aksi	Robot vacuum: melihat → memutuskan → bertindak
3. Perbedaan	Reactive AI pasif, Agentic AI proaktif	Chatbot jadwal vs AI yang menjadwalkan ulang
4. Komponen	Memory, Planning, Action, Feedback Loop	AI ingat kopi favorit, belajar dari hasil
5. Skenario	Asisten belajar, keuangan, smart home	AI guru, personal finance agent, lampu otomatis
6. Tools	Framework (LangChain, AutoGen) & API	Integrasi email + kalender untuk reminder
7. Tantangan	Bias, keamanan, kontrol manusia	AI rekrutmen bias, smart home salah perintah
8. Masa Depan	AI sebagai kolaborator, co-pilot	AI manajer digital atau guru adaptif

Refleksi Praktis

Melihat keseluruhan perjalanan ini, ada beberapa refleksi penting:

1. **Perubahan peran AI** → dari sekadar *alat hitung* menuju *co-pilot* kehidupan.
2. **Keseimbangan penting** → AI harus proaktif namun tetap diawasi (*human-in-the-loop*).
Relevansi luas → dari rumah, pendidikan, hingga bisnis, Agentic AI membawa dampak nyata.

Penutup

Kesimpulannya, **Agentic AI adalah tonggak evolusi AI yang membawa kita menuju kolaborasi manusia–mesin yang lebih harmonis**. Dengan pemahaman yang sederhana, pemula dapat langsung melihat potensinya, sementara praktisi bisa mulai mengeksplorasi penerapan nyata di berbagai sektor. Refleksi utama dari seluruh bab ini adalah bahwa **AI seharusnya menjadi mitra cerdas yang mendampingi manusia, bukan menggantikannya**.

Referensi

Acharya, D. B., & Karunanayake, K. (2025). Agentic AI: Autonomous intelligence for complex goals—A comprehensive survey. **IEEE Access**, **13**, 18912–18936. <https://doi.org/10.1109/ACCESS.2025.3532853>

Biswas, A., Talukdar, W., Scott, M. R., & Acero, A. (2025). **Building agentic AI systems: Create intelligent, autonomous AI agents that can reason, plan, and adapt**. Packt Publishing.

Bornet, P., Wirtz, J., Davenport, T. H., De Cremer, D., Evergreen, B., Fersht, P., Gohel, R., Khiyara, S., Barkin, I., Bouchard, P. L., Cravino, N., Daher, D., Ellis, S., Fanning, A., Gomez, O., Gilmurray, K., Khan, M., Kozyrkov, C., Ioffe, M., ... Sund, P. (2025). **Agentic artificial intelligence: Harnessing AI agents to reinvent business, work, and life**. World Scientific Publishing. <https://www.researchgate.net/publication/389845606>

Dhamani, N. (2025, May 29). The agentic AI handbook: A beginner's guide to autonomous intelligent agents. freeCodeCamp. <https://www.freecodecamp.org/news/the-agentic-ai-handbook/>

Hosseini, S., & Seilani, H. (2025). The role of agentic AI in shaping a smart future: A systematic review. **Array**, 100399. <https://doi.org/10.1016/j.array.2025.100399>

Hughes, L., Dwivedi, Y. K., Malik, T., Shawosh, M., Albashrawi, M. A., Jeon, I., Dutot, V., Appanderanda, M., Crick, T., De', R., & Walton, P. (2025). AI agents and agentic systems: Redefining global IT management. **Journal of Global Information Technology Management**, **28**(3), 181–186. <https://doi.org/10.1080/1097198X.2025.2524286>

Kamal, A., Ansari, M. T. J., & Chapaneri, K. (2024). **Agentic AI – the new frontier in GenAI: An executive playbook**. PwC. <https://www.pwc.com/m1/en/publications/documents/2024/agentic-ai-the-new-frontier-in-gen-ai-an-executive-playbook.pdf>

Karunanayake, N. (2025). Next-generation agentic AI for transforming healthcare. **Intelligence-Based Medicine**, 100162. <https://doi.org/10.1016/j.ibmed.2025.100162>

Lukose, D. (2025, January 13). Agentic and multi-agentic AI. Medium. <https://medium.com/@dickson.lukose/agentic-and-multi-agentic-ai-ec81cd65bacc>

Murugesan, S. (2025). The rise of agentic AI: Implications, concerns, and the path forward. **IEEE Intelligent Systems**, **40**(2), 8–14. <https://doi.org/10.1109/MIS.2025.3544940>

Olujimi, P. A., Owolawi, P. A., Mogase, R. C., & Van Wyk, E. (2025). Agentic AI frameworks in SMMEs: A systematic literature review of ecosystemic interconnected agents. **AI**, **6**(6), 123. <https://doi.org/10.3390/ai6060123>

Sapkota, R. (2025). *AI agents vs. agentic AI: A conceptual taxonomy, applications and challenges* (arXiv:2505.10468). arXiv. <https://arxiv.org/abs/2505.10468>

Shavit, Y., Agarwal, S., Brundage, M., Adler, S., O’Keefe, C., Campbell, R., Lee, T., Mishkin, P., Eloundou, T., Hickey, A., Slama, K., Ahmad, L., McMillan, P., Beutel, A., Passos, A., & Robinson, D. G. (2023). *Practices for governing agentic AI systems* [White paper]. OpenAI. <https://cdn.openai.com/papers/practices-for-governing-agentic-ai-systems.pdf>

Simms, K. (2025, January 13). What agentic AI actually is: A deeply researched and definitive explanation. Kane Simms. <https://kanesimms.substack.com/p/what-agentic-ai-actually-is-a-deeply>

Thomson Reuters. (2025, August 11). How to use agentic AI workflows in professional services. Thomson Reuters. <https://www.thomsonreuters.com/en/insights/articles/how-to-use-agentic-ai-workflows-in-professional-services>

Tiwari, A. (2025). Conceptualising the emergence of Agentic Urban AI: From automation to agency. *Urban Informatics*, *4*(3), 1–22. <https://doi.org/10.1007/s44212-025-00079-7>

Viswanathan, P. S. (2025). Agentic AI: A comprehensive framework for autonomous decision-making systems in artificial intelligence. *International Journal of Computer Engineering and Technology*, 16*(1), 862–880.

Glosarium (A–Z)

- **Action (Eksekusi Tindakan)**
Kemampuan AI untuk melakukan tindakan nyata, baik dalam bentuk digital (misalnya mengirim email) maupun fisik (misalnya menyalakan lampu pintar).
- **Agent Frameworks**
Kerangka kerja untuk membangun Agentic AI, seperti **LangChain** (integrasi AI dengan data & aplikasi) dan **AutoGen** (kolaborasi multi-agent).
- **Agentic AI**
AI **proaktif** yang mampu mengingat (*memory*), merencanakan (*planning*), bertindak (*action*), dan belajar dari pengalaman (*feedback loop*), sehingga berperan sebagai kolaborator, bukan sekadar responder.
- **API (Application Programming Interface)**
Jembatan yang menghubungkan AI dengan aplikasi eksternal, sehingga AI bisa bertindak di dunia nyata, misalnya sinkronisasi kalender dan email.
- **Artificial Intelligence (AI)**
Cabang ilmu komputer yang berfokus pada kemampuan mesin melakukan tugas yang biasanya membutuhkan kecerdasan manusia, seperti pengenalan wajah, bahasa, dan pengambilan keputusan.
- **Bias**
Distorsi dalam pengambilan keputusan AI akibat data yang tidak netral, dapat menyebabkan diskriminasi atau hasil tidak adil.
- **Co-pilot AI**
Konsep masa depan Agentic AI sebagai kolaborator strategis yang mendampingi manusia dalam pekerjaan, pembelajaran, hingga kehidupan sehari-hari.
- **Feedback Loop (Pembelajaran dari Hasil)**
Mekanisme evaluasi hasil tindakan AI yang memungkinkan sistem beradaptasi dan memperbaiki strategi di masa depan.
- **Human-in-the-Loop (Kontrol Manusia)**
Mekanisme di mana manusia tetap terlibat dalam pengambilan keputusan AI, terutama untuk mencegah risiko atau kesalahan fatal.
- **Learning Assistant (Asisten Belajar)**
Contoh penerapan Agentic AI yang dapat menyusun jadwal belajar, memberi pengingat, serta menyesuaikan materi sesuai kebutuhan siswa.
- **Memory (Ingatan Kontekstual)**
Kemampuan AI untuk menyimpan dan mengingat informasi atau pengalaman

sebelumnya sebagai dasar pengambilan keputusan.

- **Personal Finance Agent**

Agentic AI yang membantu mengatur keuangan dengan memantau pengeluaran, memberi rekomendasi, hingga melakukan alokasi dana otomatis.

- **Planning (Perencanaan Strategis)**

Proses AI menyusun langkah-langkah atau strategi untuk mencapai tujuan tertentu.

- **Reactive AI (AI Tradisional)**

Jenis AI yang **pasif**, hanya merespons jika diberi input, tanpa inisiatif atau kemampuan beradaptasi.

- **Smart Home Agent**

Agentic AI yang mengendalikan perangkat rumah pintar secara kontekstual, misalnya menyalakan lampu atau AC berdasarkan lokasi pengguna.

Tentang Penulis



Onno W. Purbo, saat ini bertugas sebagai rektor di Institut Teknologi Tangerang Selatan (ITTS). Onno memperoleh gelar Ph.D bidang Electrical Engineering dari University of Waterloo, Canada, adalah seorang copyleft, educator dan ICT evangelist. Dia sudah mempublikasikan 50++ buku, termasuk free ICT ebook untuk sekolah tahun 2008. Beberapa buku terakhirnya adalah "Internet-TCP/IP: Konsep Dan Implementasi", 2018; "Sistem Operasi, Konsep Dan Membuat Linux OpenWRT Dan ROM Android", 2019; "IPv6 Untuk Mendukung Operasi Jaringan Dan Domain Name System", 2019; "Kubernetes untuk Pemula", 2024 dan "Membuat Operator Seluler 5G Sendiri", 2024. Dia memimpin sambungan pertama Internet di Institut Teknologi Bandung, tahun 1993-2000, dan menggunakannya untuk membuat jaringan Internet pendidikan yang pertama di

Indonesia. Dia membebaskan frekuensi WiFi, memperkenalkan RT/RW-net, antenna Wajanbolic dan jaringan selular OpenBTS dan private 5G sendiri. Dia memimpin jaringan telepon pertama di atas Internet, VoIP Merdeka, yang kemudian hari dikenal sebagai VoIP Rakyat berbasis SIP dan menggunakan kode area +62520 dan +62521. Dia saat ini aktif memperkenalkan e-Learning, dan menjalankan server e-Learning di <http://lms.onnocenter.or.id/moodle/> 66,000++ siswa / mahasiswa dan <https://opencourse.itts.ac.id> 36.000+ mahasiswa secara gratis. Email: onno@indo.net.id