ARTIFICIAL INTELLIGENCE

UNTUK SMA KELAS XII SEMESTER 2



Taufiq Iqbal Ramdhani
Onno W. Purbo

AI UNTUK MASA DEPANMU:
BANGUN PROYEK, KERJA TIM, DAN SIMULASI DUNIA KERJA

InTech press

INSTITUT TEKNOLOGI TANGERANG SELATAN itts

.id Identitas Digital Indonesia

ASTRA

Buku AI SMA Kelas 12 Semester 2

AI untuk Masa Depanmu:
Bangun Proyek, Kerja Tim, dan Simulasi Dunia Kerja

Taufik Iqbal Ramdhani
Onno W. Purbo

Institut Teknologi Tangerang Selatan (ITTS)
2025

Daftar Isi

Daftar Isi **3**
Lisensi & Catatan Karya **7**
Disclaimer **7**
Kata Pengantar **8**
BAB 1: Penentuan Topik & Tools untuk Proyek Akhir 9
Tujuan Pembelajaran 9
Peta Konsep 9
Apersepsi 10
Penjelasan Konsep (teori) 10
Brainstorming Ide Proyek: Membuka Pintu Imajinasi 10
Identifikasi Masalah: Temukan Tantangan di Sekitar 11
Menentukan Tujuan Proyek: Jadikan Impian Lebih Terarah 11
Manfaat Proyek: Apa Dampaknya? 11
Menilai Kelayakan dan Kualitas Proyek 11
Pemilihan Tools (Alat) yang Sesuai: Jangan Salah Pilih Senjata! 12
Menentukan Skala Proyek: Jangan Terlalu Besar atau Terlalu Kecil 12
Kesimpulan 13
Ilustrasi: "Mimpi Besar dari Ide Sederhana" 13
Contoh Kasus atau Ilustrasi 13
Contoh Soal dan Pembahasan 15
Fakta Menarik / Fun Facts 18
Tips Belajar atau Tips Cepat Menghafal 18
Aktivitas Siswa 19
Rangkuman 21
Latihan Soal 21
Tugas Proyek 24
BAB 2: Pengembangan Aplikasi / Sistem AI Mini 27
Tujuan Pembelajaran 27
Peta Konsep 27
Apersepsi 27
Penjelasan Konsep (teori) 28
Tahapan Pengembangan Aplikasi / Sistem AI Mini 28
Dasar Pemrograman Python (Jika Relevan) 29
Struktur Dasar Program Python 29
Fungsi 30
Library 30
Pengenalan App Inventor 2 31
Desainer UI (User Interface Designer) 31
Editor Blok (Blocks Editor) 34
Ekstensi AI 36
Personal Image Classifier 37
Personal Audio Classifier 39

Contoh Blok Program App Inventor 2....40
Eksperimen dengan Google Teachable Machine....43
Experimen Integrasi Google Teachable Machine dengan Python....43
Pengujian Aplikasi AI Mini....46
Penutup: Belajar AI itu Keren!....46
Ilustrasi: “Dari Bingung Jadi Bisa!”....47
Contoh Kasus atau Ilustrasi....47
Contoh Soal dan Pembahasan....49
Fakta Menarik / Fun Facts....51
Tips Belajar atau Tips Cepat Menghafal....51
Aktivitas Siswa....52
Rangkuman....54
Latihan Soal....55
Tugas Proyek....58
BAB 3: Kolaborasi Tim: Peran Programmer, Data Analyst, Editor....61
Tujuan Pembelajaran....61
Peta Konsep....61
Apersepsi....61
Penjelasan Konsep (teori)....62
Mengapa Kolaborasi dalam Proyek AI itu Penting?....62
Siapa Saja yang Terlibat dalam Tim Proyek AI?....63
Komunikasi Efektif: Bukan Sekadar Chatting Biasa....64
Tools Kolaborasi: Senjata Rahasia Tim Sukses....64
Kesimpulan: Satu Tim, Satu Visi....64
Ilustrasi: Perjuangan Mengatasi Masalah Kolaborasi....65
Contoh Kasus atau Ilustrasi....65
Contoh Soal dan Pembahasan....66
Fakta Menarik / Fun Facts....68
Tips Belajar atau Tips Cepat Menghafal....68
Aktivitas Siswa....69
Rangkuman....71
Latihan Soal....72
Tugas Proyek....75
BAB 4: Pengembangan Model AI (NLP, ML Sederhana)....78
Tujuan Pembelajaran....78
Peta Konsep....78
Apersepsi....78
Penjelasan Konsep (teori)....79
Apa itu Model AI?....79
Tahapan Pengembangan Model AI....79
Contoh Sederhana Natural Language Processing (NLP)....81
Contoh Sederhana Machine Learning (ML)....81
Eksperimen: Google Teachable Machine....81
Ilustrasi....81

Contoh Kasus atau Ilustrasi..... 82
Contoh Soal dan Pembahasan..... 84
Fakta Menarik / Fun Facts.....86
Tips Belajar atau Tips Cepat Menghafal..... 86
Aktivitas Siswa..... 87
Rangkuman.....89
Latihan Soal..... 89
Tugas Proyek..... 92
BAB 5: Buat Pitch Deck dengan Bantuan ChatGPT.....95
Tujuan Pembelajaran..... 95
Peta Konsep.....95
Apersepsi..... 95
Penjelasan Konsep (teori).....96
Apa itu Pitch Deck?.....96
Komponen Wajib dalam Sebuah Pitch Deck.....96
Struktur Slide yang Efektif.....97
Gunakan ChatGPT Sebagai Partner Kreatif..... 97
Prinsip Desain Slide..... 98
Elevator Pitch: Ringkasan Cepat yang Mengesankan..... 98
Ilustrasi: Ketika Ide Hebat Sulit Didengar.....99
Contoh Kasus atau Ilustrasi..... 99
Contoh Soal dan Pembahasan..... 101
Fakta Menarik / Fun Facts.....102
Tips Belajar atau Tips Cepat Menghafal..... 103
Aktivitas Siswa..... 104
Rangkuman.....106
Latihan Soal..... 106
Tugas Proyek..... 109
BAB 6: Simulasi Interview Kerja dengan AI (Mock Interview)..... 112
Tujuan Pembelajaran.....112
Peta Konsep..... 112
Apersepsi.....112
Penjelasan Konsep (teori)..... 113
Apa Tujuan dari Wawancara Kerja?..... 113
Jenis-Jenis Pertanyaan Wawancara..... 113
Tips Jitu Menghadapi Wawancara.....114
Menggunakan AI (seperti ChatGPT) Sebagai Simulasi Wawancara.....115
Penutup: Belajar Interview Itu Seperti Latihan Bela Diri!.....115
Ilustrasi..... 115
Contoh Kasus atau Ilustrasi.....116
Contoh Soal dan Pembahasan.....118
Fakta Menarik / Fun Facts.....120
Tips Belajar atau Tips Cepat Menghafal..... 120
Aktivitas Siswa..... 121

Rangkuman....123
Latihan Soal....123
Tugas Proyek....126
BAB 7: Portofolio Digital AI Siswa....129
Tujuan Pembelajaran....129
Peta Konsep....129
Apersepsi....129
Penjelasan Konsep (teori)....130
Mengapa Portofolio Digital Itu Penting?....130
Apa Saja Manfaat Membuat Portofolio Digital?....130
Platform Populer untuk Portofolio AI....130
Cara Menjelaskan Proyek dalam Portofolio....131
Tips untuk Portofolio AI yang Keren dan Profesional....132
Siap Unjuk Diri?....132
Ilustrasi: Perjuangan Membangun Portofolio yang Keren....132
Contoh Kasus atau Ilustrasi....133
Contoh Soal dan Pembahasan....135
Fakta Menarik / Fun Facts....137
Tips Belajar atau Tips Cepat Menghafal....138
Aktivitas Siswa....138
Rangkuman....140
Latihan Soal....141
Tugas Proyek....144
BAB: Mengenal Domain .id....147
Apa Itu Domain?....148
Apa Itu .id?....148
Keunggulan domain .id....148
Siapa Itu PANDI?....149
Kenapa Kamu Harus Tahu Ini?....149
Lebih dalam tentang PANDI....149
Tata Kelola Internet....151
Institut Teknologi Tangerang Selatan (ITTS) dan Penulis....152

Lisensi & Catatan Karya

Karya ini dilisensikan di bawah lisensi **Creative Commons Attribution-ShareAlike 4.0 International (CC BY-SA 4.0)**.
Artinya, **siapapun bebas untuk menggunakan, menyalin, membagikan, dan mengadaptasi** materi ini, **dengan syarat**:

- **Memberikan atribusi yang sesuai** (menyebut sumber asli),
- **Menyertakan lisensi yang sama** jika dimodifikasi atau dikembangkan lebih lanjut.

Lihat detail lisensi di sini https://creativecommons.org/licenses/by-nd/4.0/legalcode.id

Desain Cover: Irwan Siswanto

Disclaimer

Materi pembelajaran ini dibuat dengan **dana swadaya masyarakat Indonesia** dan **kontribusi sukarela dari para dosen di Institut Teknologi Tangerang Selatan(ITTS)**. Karya ini didistribusikan secara bebas untuk mendukung pendidikan digital dan kecerdasan buatan di kalangan pelajar Indonesia.

Kami **memohon maaf** jika terdapat kekurangan dalam isi maupun penyajian materi ini. Kritik dan saran sangat kami harapkan untuk terus menyempurnakan karya serupa di masa depan.

Mohon doa agar amal ibadah mereka yang terlibat di kegiatan ini diterima dan dilipatgandakan.

Kata Pengantar

Buku ini disusun sebagai bentuk partisipasi sederhana dari sivitas akademika Institut Teknologi Tangerang Selatan (ITTS) dalam mendukung kemajuan pendidikan di Indonesia. Di tengah pesatnya perkembangan teknologi, kami terdorong untuk menghadirkan materi pembelajaran yang relevan dan dapat diakses secara luas oleh masyarakat, khususnya generasi muda.

Dengan semangat berbagi dan memberdayakan, kami berharap buku ini dapat menjadi langkah awal yang bermakna dalam memperkenalkan konsep dasar kecerdasan buatan (AI) kepada pelajar tingkat SMP dan SMA. Materi disusun secara sistematis dan aplikatif, mencakup antara lain:

- Dasar-dasar *Computational Thinking*
- Pembuatan aplikasi Android menggunakan App Inventor 2
- Pemrograman visual dengan Scratch
- Pengenalan AI melalui Google Teachable Machine
- Eksplorasi AI generatif seperti ChatGPT, Gemini, dan Grok
- Pembelajaran AI berbasis data dengan Orange Data Mining, disertai contoh yang sederhana

Penyusunan dan penerbitan buku ini sepenuhnya dibiayai secara mandiri berkat dukungan berbagai pihak yang memiliki kepedulian terhadap kemajuan ilmu pengetahuan di tanah air. Untuk itu, kami mengucapkan terima kasih yang tulus atas segala bantuan dan dorongan yang telah diberikan.

Kami menyadari bahwa karya ini masih jauh dari sempurna. Karena itu, segala bentuk kritik, saran, maupun masukan yang membangun sangat kami hargai dan nantikan. Silahkan sampaikan melalui kontak resmi ITTS atau langsung kepada tim penulis. Pembaruan dan perbaikan akan terus kami lakukan demi meningkatkan kualitas isi dan manfaat buku ini.

Semoga buku ini dapat menjadi pijakan awal bagi para pelajar dalam memahami dunia kecerdasan buatan, serta turut menginspirasi langkah kecil menuju masa depan Indonesia yang lebih cerah.

Dengan rendah hati, kami memohon doa dan dukungan dari para pembaca agar upaya sederhana ini dapat membawa manfaat yang luas dan menjadi amal jariyah bagi semua penulis yang terlibat.

Tangerang Selatan, Agustus 2025
Penulis

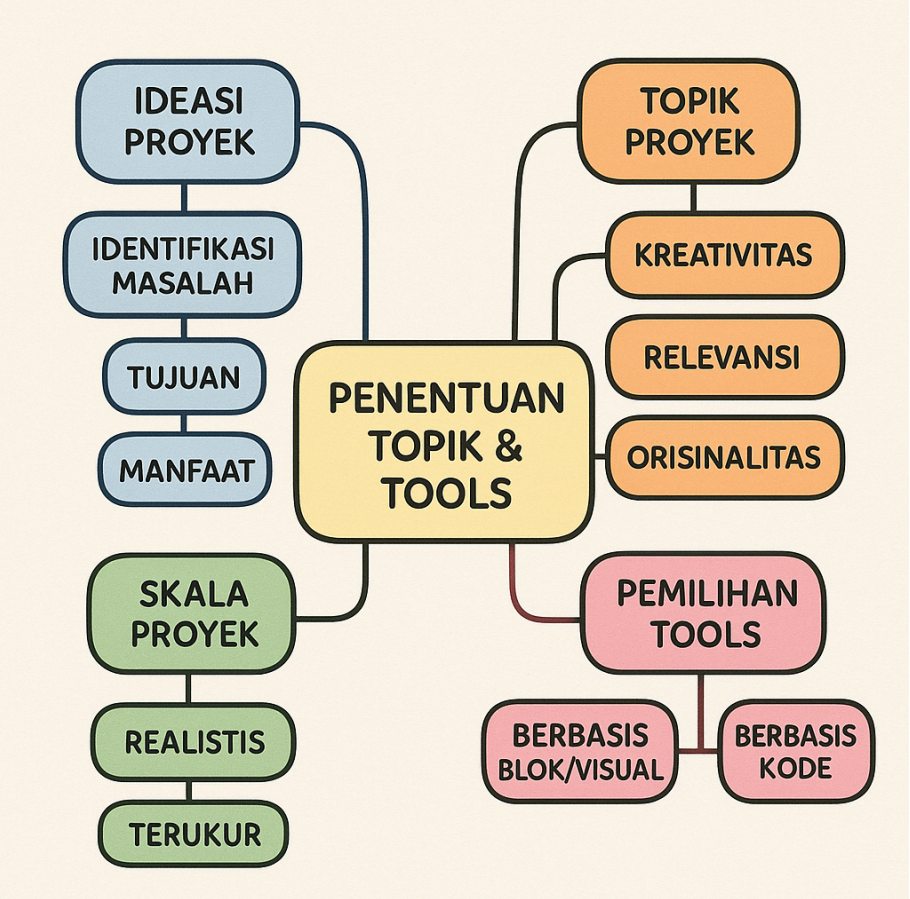
BAB 1: Penentuan Topik & Tools untuk Proyek Akhir

Tujuan Pembelajaran

Setelah mengikuti kegiatan pembelajaran ini, siswa mampu:

1. Menentukan topik proyek akhir berbasis AI yang relevan dan menarik.
2. Mengidentifikasi masalah nyata di sekitar yang bisa diselesaikan dengan AI.
3. Mengembangkan ide proyek menggunakan teknik kreatif seperti mind mapping dan **SCAMPER.**
4. Menyusun tujuan proyek menggunakan prinsip **SMART.**
5. Memilih tools AI yang sesuai dengan kebutuhan dan kemampuan tim.
6. Menilai kelayakan dan potensi dampak proyek secara kritis.

Peta Konsep



Apersepsi

Pernah nggak, kalian melihat masalah di sekitar—di sekolah, rumah, atau lingkungan—lalu berpikir, *“Andai ada teknologi canggih yang bisa bantu menyelesaikan ini…”*? Nah, hari ini kita akan mulai petualangan seru dengan menjelajahi bagaimana kecerdasan buatan (AI) bisa menjadi solusi nyata! Sebelum kita terlalu jauh, ayo kita tonton dulu video singkat tentang proyek-proyek AI sederhana tapi berdampak besar. Contohnya, ada aplikasi yang bisa mendeteksi penyakit pada tanaman, chatbot untuk membantu siswa bertanya ke sekolah, hingga sistem pintar yang merekomendasikan buku di perpustakaan. Keren, kan?

Setelah itu, saatnya kalian berpikir lebih dekat ke diri sendiri. Coba amati lingkungan sekitar—apakah ada masalah yang bikin kamu merasa, “Ini bisa dibantu AI, nih!”? Yuk, sebutkan satu saja ide yang muncul di kepala kalian saat ini. Bisa tentang belajar, lingkungan, atau bahkan masalah sehari-hari.

Dan inilah tantangan kalian hari ini: **Temukan satu masalah nyata di sekitarmu dan pikirkan satu tools AI yang bisa kamu gunakan untuk mulai membangun solusinya.** Siap? Let’s create something impactful!

Penjelasan Konsep (teori)

Merancang proyek akhir berbasis Kecerdasan Buatan (AI) bukanlah sekadar tugas, tetapi sebuah petualangan seru di mana kalian bisa menjelajahi dunia teknologi sekaligus menyelesaikan masalah nyata di sekitar kalian. Di bagian ini, kita akan mempelajari langkah-langkah awal yang penting dalam merancang proyek AI yang relevan, inovatif, dan dapat dilaksanakan. Yuk, kita mulai dari proses menggali ide hingga memilih alat yang paling cocok!

Brainstorming Ide Proyek: Membuka Pintu Imajinasi

Langkah pertama adalah membiarkan ide-ide mengalir bebas. Dua teknik kreatif berikut bisa kalian gunakan untuk mengembangkan gagasan:

- **Mind Mapping**
  Teknik ini membantu kalian memecah satu ide besar menjadi cabang-cabang ide kecil. Misalnya, jika topik awalnya adalah “kesehatan,” kalian bisa menjabarkan menjadi “gizi,” “deteksi penyakit,” atau “mental health.”
  SCAMPER
  Teknik berpikir kreatif ini mendorong kalian untuk:
  - **S**ubstitute (Mengganti sesuatu dalam ide),
  - **C**ombine (Menggabungkan dua ide),
  - **A**dapt (Menyesuaikan ide dari hal yang sudah ada),
  - **M**odify (Mengubah sebagian aspek),

    - **P**ut to other use (Mengalihkan penggunaan untuk tujuan lain),
    - **E**liminate (Menghapus bagian yang tidak perlu),
    - **R**everse (Membalik alur atau fungsi).

Misalnya: Mengembangkan sistem absensi sekolah menjadi sistem pendeteksi kehadiran otomatis berbasis wajah.

Identifikasi Masalah: Temukan Tantangan di Sekitar

Kunci dari proyek AI yang baik adalah **berangkat dari masalah nyata**. Amati lingkungan sekitar kalian:

- Apakah ada masalah sampah yang belum tertangani?
- Adakah kesulitan siswa memahami pelajaran tertentu?
- Apakah masyarakat di sekitarmu memiliki tantangan dalam hal kesehatan atau transportasi?

Tanyakan pada diri sendiri: *“Apakah teknologi bisa membantu menyelesaikan ini?”*

Menentukan Tujuan Proyek: Jadikan Impian Lebih Terarah

Setelah menemukan masalah, saatnya merumuskan **tujuan proyek** secara jelas. Gunakan prinsip **SMART Goals**, yaitu:

- **S**pecific (Spesifik)
- **M**easurable (Terukur)
- **A**chievable (Dapat dicapai)
- **R**elevant (Relevan dengan masalah)
- **T**ime-bound (Terikat waktu)

Contoh: *“Membuat aplikasi AI yang dapat mendeteksi kondisi daun tanaman tomat dalam waktu 2 bulan sebagai upaya membantu petani lokal.”*

Manfaat Proyek: Apa Dampaknya?

Sebuah proyek yang baik bukan hanya keren secara teknis, tapi juga **bermanfaat**. Coba pikirkan:

- Apakah proyek bisa membantu orang lain?
- Apakah dapat menghemat waktu, biaya, atau tenaga?
- Apakah bisa meningkatkan kualitas hidup seseorang?

Semakin besar dampaknya, semakin bernilai proyekmu!

Menilai Kelayakan dan Kualitas Proyek

Sebelum lanjut ke pembuatan, uji dulu apakah proyek kalian benar-benar layak dilakukan. Gunakan kriteria ini:

- **Relevansi:** Masalah yang diselesaikan penting dan dekat dengan kehidupan.
- **Orisinalitas/Inovasi:** Ide unik atau pendekatan yang kreatif.
- **Kelayakan (Feasibility):** Bisa dikerjakan dengan waktu, alat, dan kemampuan yang ada.
- **Terukur:** Ada indikator keberhasilan yang jelas (misalnya akurasi AI, jumlah pengguna, tingkat kepuasan).

Pemilihan Tools (Alat) yang Sesuai: Jangan Salah Pilih Senjata!

Setiap ide hebat butuh alat yang tepat untuk mewujudkannya. Pilih tools berdasarkan kebutuhan dan tingkat kemampuan kalian:

Tools Berbasis Blok/Visual (Contoh: App Inventor 2)

- **Kelebihan:**
  Cocok untuk pemula, tidak perlu menghafal sintaks, fokus ke logika dan desain.
- **Kekurangan:**
  Fungsinya terbatas untuk proyek kompleks.
- **Contoh Penggunaan:**
  Membuat aplikasi deteksi sampah menggunakan AI sederhana.
- *Catatan:* Bahasa pemrogramannya tetap Bahasa Inggris. Siswa perlu membiasakan diri dengan istilah seperti “if”, “then”, “image classifier”.

Tools Drag-and-Drop (Contoh: Google Teachable Machine, Orange)

- **Kelebihan:**
  Super gampang! Tanpa coding, tinggal tarik dan lepas.
- **Kekurangan:**
  Kurang fleksibel dan terbatas untuk model AI tertentu.
- **Contoh Penggunaan:**
  Melatih model untuk mengenali suara, gambar, atau gerakan tubuh.
- *Tips:* Gunakan screenshot antarmuka untuk panduan visual langkah per langkah.

Tools Berbasis Kode (Contoh: Python dengan TensorFlow Lite atau Scikit-learn)

- **Kelebihan:**
  Sangat fleksibel dan bisa membuat sistem AI yang canggih.
- **Kekurangan:**
  Butuh pemahaman dasar pemrograman. Tidak cocok untuk yang benar-benar baru belajar.
- **Contoh Penggunaan:**
  Membuat chatbot pintar, atau sistem rekomendasi berbasis data.

Menentukan Skala Proyek: Jangan Terlalu Besar atau Terlalu Kecil

Agar proyek bisa selesai tepat waktu dan tetap berkualitas, tentukan skala yang realistis dengan mempertimbangkan:

- Durasi semester
- Jumlah anggota tim
- Tingkat kesulitan fitur yang akan dibuat
- Ketersediaan tools dan data

Kesimpulan

Merancang proyek AI bukan sekadar memilih teknologi, tapi juga tentang **memahami masalah, berpikir kreatif, dan membuat sesuatu yang bermanfaat**. Dengan proses yang terstruktur – mulai dari brainstorming ide, identifikasi masalah, hingga pemilihan tools – kalian bisa menghasilkan proyek akhir yang tak hanya keren, tapi juga berdampak nyata.

Ilustrasi: "Mimpi Besar dari Ide Sederhana"

Bayangkan seorang siswa bernama Dita. Di sekolahnya, banyak teman yang sering lupa jadwal ulangan atau tugas. Dita merasa ini masalah kecil, tapi nyata. Ia pun punya ide: bagaimana kalau membuat aplikasi pengingat jadwal belajar berbasis AI yang bisa memberi notifikasi otomatis dan bahkan menyemangati pengguna?

Awalnya Dita ragu, karena ia bukan anak jurusan IT. Tapi ia mencoba **Google Teachable Machine** untuk mengenali ekspresi wajah pengguna (apakah terlihat lelah atau fokus). Ia pakai **App Inventor 2** untuk merancang aplikasinya. Setelah dua bulan, jadilah aplikasi mini yang bisa mengenali ekspresi dan mengingatkan jadwal sesuai mood pengguna. Proyek itu bahkan menang lomba tingkat kota!

Apa yang Dita pelajari? **Bukan soal seberapa rumit proyekmu, tapi seberapa relevan dan bermanfaat ide sederhana yang kamu buat.**

Contoh Kasus atau Ilustrasi

Kasus 1: Drone dan Sungai yang Kotor

> **Judul:** *"Detektif Sampah: Misi Udara Menyelamatkan Sungai"*
>
> Bayangkan kamu dan timmu diberi misi menyelamatkan sungai di kota kalian dari pencemaran. Kalian menciptakan sistem yang bisa mendeteksi tumpukan sampah di sungai menggunakan gambar dari drone. Sistem ini bisa memberi peringatan otomatis ke petugas kebersihan.
>
> **Diskusikan**:
>
> - Bagaimana sistem mengenali mana yang sampah dan mana yang bukan?
> - Mengapa proyek ini bisa sangat berguna untuk kota kalian?
> - Apa tujuan, metode, dan manfaat jangka panjangnya?

Kasus 2: Chatbot Kantin Pintar

Judul: *"Si Koki Virtual: Pemesanan Makanan Bebas Antre!"*

Di sekolah kamu, antrian kantin sering panjang. Kamu dan teman-temanmu punya ide: buat chatbot yang bisa menerima pesanan makanan sebelum jam istirahat dimulai. Murid tinggal kirim pesan ke chatbot, dan makanannya sudah siap saat istirahat.

Diskusikan:

- Bagaimana chatbot ini tahu makanan apa saja yang tersedia hari ini?
- Apa manfaat untuk petugas kantin dan siswa?
- Mengapa chatbot ini lebih menarik dibanding sistem pemesanan biasa?

Kasus 3: Inovasi Kecil, Dampak Besar

Judul: *"Reminder Cerdas vs. Robot Luar Angkasa"*

Dua kelompok membuat proyek. Satu membuat aplikasi sederhana pengingat belajar. Satunya lagi membuat sistem AI rumit yang menganalisis pola tidur dan emosi. Tapi ternyata, aplikasi pengingat lebih sering dipakai dan disukai.

Diskusikan:

- Mengapa proyek yang lebih sederhana bisa lebih berguna?
- Apa kriteria proyek yang berdampak untuk kehidupan nyata?
- Bagaimana inovasi kecil bisa mengubah kebiasaan banyak orang?

Kasus 4: Kelas Kotor, Siapa Bertanggung Jawab?

Judul: *"Kelas Detektor: Mata-Mata Kebersihan Digital"*

Setiap hari ada yang lupa piket. Jadilah kelas selalu kotor. Tim kamu membuat alat AI yang memantau kebersihan kelas menggunakan kamera, lalu memberi peringatan jika terlihat kotor.

Diskusikan:

- Bagaimana sistem mengenali kondisi "bersih" dan "kotor"?
- Apa tantangan dalam mengajari AI mengenali kekacauan?
- Apakah alat ini bisa mendorong tanggung jawab bersama?

Kasus 5: Sering Telat karena Hujan

Judul: *"Alarm Pintar: Bangun Sebelum Terlambat"*

Kamu sering telat ke sekolah, apalagi saat hujan. Kamu buat sistem yang memberi alarm lebih awal jika cuaca buruk atau jarak rumah jauh. AI ini menghitung waktu ideal untuk berangkat berdasarkan info cuaca online.

Diskusikan:

- Bagaimana sistem ini tahu kapan harus membunyikan alarm?
- Apa manfaatnya dibanding alarm biasa?
- Apakah sistem ini bisa membuatmu lebih disiplin?

Kasus 6: Si Jenius Matematika Virtual

Judul: *"Chatbot Tutor: Belajar Matematika Nggak Lagi Menyeramkan"*

Teman kamu kesulitan memahami soal matematika dasar. Kamu membuat chatbot sederhana yang bisa menjawab soal-soal dasar dengan penjelasan mudah.

Diskusikan:

- Bagaimana chatbot bisa memahami pertanyaan matematika?
- Apakah AI ini bisa menggantikan guru? Kenapa tidak?
- Apa manfaatnya untuk siswa yang belajar mandiri?

Contoh Soal dan Pembahasan

Soal:

Sebuah tim siswa ingin mengembangkan aplikasi berbasis kecerdasan buatan (AI) untuk memprediksi harga saham. Menurut pendapat Anda, apakah proyek ini sesuai dengan ruang lingkup tugas akhir siswa SMA? Jelaskan berdasarkan aspek relevansi, tingkat kesulitan, dan ketersediaan sumber daya.

Pembahasan:
Proyek prediksi harga saham terlalu kompleks untuk level SMA karena:

- Memerlukan data real-time dan besar.
- Butuh algoritma machine learning tingkat lanjut.
- Waktu dan sumber daya siswa terbatas.

Alternatif: Analisis sentimen berita ekonomi untuk melihat arah sentimen pasar (positif/negatif) bisa lebih realistis.

Langkah-langkah pengerjaan alternatif:

1. **Riset dasar** tentang AI dan sentiment analysis.
2. **Kumpulkan data** berita ekonomi (misalnya 50 artikel).
3. **Labeli data** secara manual: berita positif, netral, atau negatif.

4. **Gunakan model sederhana** (misalnya Naive Bayes) untuk melatih AI.
5. **Uji model** dan lihat akurasi prediksinya.
6. **Tampilkan hasil** di aplikasi sederhana (bisa berbasis web/mobile).

Soal:

Seseorang mengusulkan proyek aplikasi mobile untuk memantau harga saham harian, lengkap dengan fitur prediksi menggunakan AI. Apa potensi tantangan utama dari proyek ini bila dikerjakan oleh siswa SMA, dan bagaimana solusinya?

Pembahasan:
Tantangan utama:

- Akses data saham legal dan real-time sulit.
- Model prediksi butuh banyak pelatihan dan tuning.
- Risiko error tinggi karena data pasar sangat dinamis.

Solusi: Ubah fokus menjadi **analisis berita ekonomi berbasis AI**.

Langkah-langkah pengerjaan solusi:

1. **Definisikan ulang tujuan proyek** agar lebih sederhana dan realistis.
2. **Ambil berita ekonomi** dari situs berita resmi (bisa disimpan sebagai dokumen).
3. **Gunakan Python + library NLP** seperti `scikit-learn` atau `TextBlob`.
4. **Latih model klasifikasi sederhana** berdasarkan dataset yang disiapkan.
5. **Sajikan hasilnya dalam bentuk antarmuka** sederhana (misalnya: hasil prediksi sentimen dari berita yang dimasukkan).
6. **Buat laporan hasil dan evaluasi akurasi**.

Soal:

Mengapa penting bagi siswa untuk merancang tujuan proyek dengan prinsip SMART (Specific, Measurable, Achievable, Relevant, Time-bound)? Jelaskan dan berikan contohnya.

Pembahasan:
Prinsip SMART membantu:

- Menyusun tujuan yang tidak mengambang.
- Memudahkan penjadwalan dan evaluasi.
- Memastikan proyek selesai tepat waktu.

Contoh tujuan SMART:
"Membuat chatbot AI berbasis teks yang bisa menjawab 10 pertanyaan seputar kegiatan sekolah dalam 30 hari."

Langkah-langkah penerapan SMART:

1. **Specific:** Tetapkan fungsi jelas dari proyek (contoh: chatbot menjawab pertanyaan).
2. **Measurable:** Tentukan ukuran kesuksesan (jawaban chatbot akurat 80%).
3. **Achievable:** Pastikan sesuai dengan kemampuan dan sumber daya.
4. **Relevant:** Hubungkan dengan kebutuhan nyata di lingkungan sekolah.
5. **Time-bound:** Buat jadwal dan deadline per bagian (misalnya: minggu ke-2 chatbot harus bisa menjawab 3 pertanyaan).

Soal:

Apa kelebihan platform seperti App Inventor 2 dalam pengembangan aplikasi oleh pemula, terutama siswa SMA?

Pembahasan:
Kelebihan:

- Tidak perlu coding kompleks.
- Antarmuka visual (drag-and-drop) memudahkan pemahaman logika program.
- Cepat diuji di smartphone Android.

Langkah-langkah penggunaan App Inventor 2:

1. **Masuk ke platform App Inventor 2** (di web).
2. **Buat akun dan proyek baru.**
3. **Gunakan antarmuka visual** untuk mendesain tampilan aplikasi (UI).
4. **Tambahkan logika** menggunakan blok-blok perintah yang tersedia.
5. **Tes langsung aplikasi** di perangkat Android menggunakan kode QR.
6. **Refleksi hasil dan perbaikan** berdasarkan uji coba.

Soal:

Sebutkan dan jelaskan contoh proyek AI sederhana yang bisa langsung diterapkan di lingkungan sekolah dan relevan untuk siswa.

Pembahasan:
Contoh: **Detektor kebisingan di perpustakaan.**
Fungsi: Menggunakan sensor mikrofon untuk mendeteksi suara keras dan memberi peringatan jika melampaui ambang batas.

Langkah-langkah pengerjaan:

1. **Pelajari dasar suara dan sensor suara** (misalnya menggunakan Arduino + sound sensor).
2. **Rancang sistem** untuk mendeteksi level desibel suara.
3. **Set ambang batas suara**, misalnya > 60 dB dianggap bising.
4. **Kembangkan sistem peringatan otomatis**, seperti suara bip atau tampilan peringatan di layar.
5. **Lakukan uji coba** di perpustakaan dalam waktu-waktu berbeda.
6. **Tulis laporan proyek** termasuk manfaat dan kemungkinan pengembangan.

Fakta Menarik / Fun Facts

- "Tahukah kamu, banyak startup AI sukses dimulai dari ide sederhana yang memecahkan masalah sehari-hari? Contohnya, aplikasi Grammarly yang membantu jutaan orang menulis lebih baik, berawal dari ide untuk memperbaiki kesalahan penulisan!"
- **Grammarly** awalnya hanya proyek kecil untuk memperbaiki grammar, tapi sekarang digunakan oleh jutaan orang di seluruh dunia!
- Ada siswa SMA dari India yang membuat AI untuk mendeteksi kelainan mata hanya dengan kamera ponsel!
- Tools seperti Teachable Machine bisa melatih AI hanya dalam beberapa menit tanpa satu baris kode pun!
- Beberapa proyek AI SMA sudah menang kompetisi nasional bahkan internasional. Jadi... **kenapa bukan kamu berikutnya?**

Tips Belajar atau Tips Cepat Menghafal

Checklist Memilih Proyek AI

- Apakah ini memecahkan masalah nyata?
- Apakah datanya tersedia?
- Apakah bisa dibuat dengan tools yang tersedia?
- Apakah saya paham masalahnya?
- Apakah saya suka topiknya?

Mind Mapping Contoh



Teknik "5 Why"

Masalah: Siswa sering telat.
Kenapa? Karena bangun kesiangan.
Kenapa? Karena tidur larut.
Kenapa? Karena tugas tidak selesai.
Kenapa? Karena tidak tahu deadline.
Kenapa? Karena tidak ada pengingat.
→ Solusi: AI pengingat tugas dan waktu tidur!

Aktivitas Siswa

Aktivitas 1: Ideathon AI: Menemukan Topik Proyek Akhir

Tujuan: Menyusun beberapa ide proyek AI yang inovatif dan relevan.

Langkah-langkah saran:

- Bentuk kelompok kecil atau kerjakan secara individu.
- Mulailah dengan diskusi santai: teknologi apa yang kalian gunakan sehari-hari? Apa tantangan yang bisa dibantu oleh AI?
- Catat 3–5 ide proyek dengan singkat (misal: deteksi suasana hati dari suara, pengingat belajar otomatis, AI pendeteksi sampah).
- Fokus pada ide yang *menarik*, *bermanfaat*, dan *realistis untuk dikembangkan*.

Tips: Gunakan papan tulis atau sticky notes untuk menuangkan semua ide secara bebas!

Aktivitas 2: Presentasi dan Umpan Balik: Menyaring Ide Terbaik

Tujuan: Memilih satu ide terbaik untuk dikembangkan lebih lanjut.

Langkah-langkah saran:

- Siapkan presentasi singkat (2–3 menit) tentang ide-ide yang telah disusun.
- Sajikan ke teman-teman dan guru secara menarik. Ceritakan alasan di balik ide tersebut.
- Terima masukan dengan terbuka. Catat saran yang bisa memperkuat atau menyederhanakan ide.
- Akhiri dengan memilih satu ide yang paling menjanjikan.

Tips: Gunakan voting atau diskusi cepat untuk menentukan pilihan terbaik.

Aktivitas 3: AI Scavenger Hunt: Menjelajah Teknologi AI di Sekolah

Tujuan: Mengenali teknologi AI yang sudah ada di lingkungan sekitar.

Langkah-langkah saran:

- Kelilingi sekolah dan amati perangkat atau sistem yang mungkin menggunakan AI.
- Buat daftar dengan bukti atau foto: misalnya CCTV pintar, sistem absensi otomatis, papan interaktif, aplikasi belajar adaptif.
- Diskusikan di kelas: mana yang paling mengejutkan? Apakah semua siswa pernah sadar itu memakai AI?

Tips: Tantang diri kalian untuk menemukan minimal 5 contoh unik yang berbeda dari kelompok lain!

Aktivitas 4: Mini Hackathon: Ciptakan AI Mini dengan Teachable Machine

Tujuan: Menguji langsung cara kerja model AI sederhana.

Langkah-langkah saran:

- Buka situs Teachable Machine.
- Pilih proyek: klasifikasi gambar, suara, atau pose. Misalnya, AI yang bisa membedakan ekspresi wajah (senang vs bosan).
- Kumpulkan data kalian sendiri dan latih modelnya.
- Uji dan lihat apakah AI kalian bisa mengenali input baru dengan tepat.

Tips: Waktu hanya 2 jam—bagi tugas dalam tim agar efisien (data, pelatihan, pengujian).

Aktivitas 5: Pitch Deck Challenge: Menjual Ide Proyek AI

Tujuan: Membuat presentasi proyek yang meyakinkan dan profesional.

Langkah-langkah saran:

- Gunakan Canva atau Google Slides untuk mendesain presentasi.
- Isi presentasi mencakup:
    1. Judul proyek
    2. Masalah yang ingin diselesaikan
    3. Solusi berbasis AI
    4. Tools atau platform yang akan digunakan
    5. Target pengguna
- Sajikan seolah-olah kalian "menjual" ide ini ke investor atau juri.

Tips: Desain menarik dan ringkas akan membuat ide kalian lebih mudah diingat!

Aktivitas 6: Prototyping App: Buat Pengingat Belajar dalam Sekali Coding

Tujuan: Membuat mockup aplikasi edukasi berbasis AI atau logika sederhana.

Langkah-langkah saran:

- Gunakan App Inventor untuk membuat tampilan awal aplikasi.
- Fokus pada fitur dasar, misalnya: menyapa pengguna saat aplikasi dibuka, memberikan pengingat belajar harian.

    Tambahkan *pseudocode* seperti:

    ```
    when Screen1.Initialize
    call Notifier.ShowAlert("Selamat Datang! Siap belajar?")
    ```

- Eksplorasi fitur lain: tombol untuk memulai belajar, grafik progres belajar, atau suara motivasi.

Tips: Prototipe tidak perlu sempurna—yang penting ide dan fungsinya bisa dipahami!

Rangkuman

Keberhasilan sebuah proyek sangat ditentukan oleh perencanaan yang terstruktur dan matang. Proses ini dimulai dari **ideasi yang kuat**, **pemilihan topik yang relevan**, hingga **pemilihan *tools* yang sesuai** dengan kompleksitas dan kebutuhan proyek. Dalam konteks proyek *Artificial Intelligence (AI)*, **penentuan topik harus terkait dengan permasalahan nyata** yang dihadapi siswa di lingkungan mereka.

Melalui pendekatan seperti *brainstorming*, *mind mapping*, dan teknik *SCAMPER*, siswa bisa mengeksplorasi berbagai ide proyek yang **kreatif sekaligus realistis**. **Tujuan proyek perlu dirumuskan secara *SMART* (Specific, Measurable, Achievable, Relevant, Time-bound)** agar arah kerja jelas dan terukur. Selain itu, penting juga untuk memahami **manfaat serta dampak proyek** bagi pengguna atau lingkungan sekitar.

Pemilihan *tools* harus mempertimbangkan kemampuan dan kebutuhan siswa. Untuk pemula, *tools* visual seperti *App Inventor* atau *Teachable Machine* sangat disarankan karena lebih intuitif. Sebaliknya, *tools* berbasis kode memberikan **fleksibilitas yang lebih tinggi**, cocok bagi siswa yang sudah lebih mahir. **Proyek yang efektif bukanlah yang paling rumit, tetapi yang paling tepat guna dan bisa diselesaikan dengan sumber daya yang tersedia.**

Latihan Soal

A. Benar atau Salah (5 Soal)

1. Brainstorming adalah proses yang dilakukan di akhir proyek untuk menilai hasil kerja. **Salah**

2. Proyek AI yang baik harus menyelesaikan masalah nyata di lingkungan sekitar.
   Benar

3. App Inventor 2 termasuk tools berbasis kode dan cocok untuk proyek AI kompleks.
   Salah

4. Kriteria SMART digunakan untuk merancang tujuan proyek.
   Benar

5. Semakin rumit proyek AI, semakin besar kemungkinan keberhasilannya di tingkat SMA.
   Salah

B. Pilihan Ganda (10 Soal)

6. Yang bukan termasuk kriteria proyek AI yang baik adalah:
   a. Relevansi
   b. Orisinalitas
   c. Kompleksitas tanpa batas
   d. Kelayakan
   Jawaban: c

7. Tujuan dari teknik SCAMPER dalam brainstorming adalah untuk:
   a. Menguji hasil proyek
   b. Menganalisis data
   c. Mengembangkan ide dari ide yang sudah ada
   d. Membuat laporan akhir
   Jawaban: c

8. Alat yang cocok untuk melatih model AI berbasis suara tanpa kode adalah:
   a. Python
   b. Google Teachable Machine
   c. Visual Studio Code
   d. Scratch
   Jawaban: b

9. Salah satu kelebihan tools berbasis visual seperti App Inventor 2 adalah:
   a. Membutuhkan coding tingkat lanjut
   b. Hanya bisa digunakan oleh profesional
   c. Mudah dipelajari dan cocok untuk pemula
   d. Digunakan untuk proyek robotik
   Jawaban: c

10. Contoh proyek AI yang sederhana namun berdampak adalah:
    a. Aplikasi deteksi penyakit tanaman
    b. Mobil otonom multi sensor
    c. Jaringan neural generatif untuk seni abstrak
    d. Mesin penerjemah berbasis transformer skala besar

Jawaban: a

11. Teknik mind mapping bermanfaat untuk:
    a. Mempersingkat waktu coding
    b. Membuat laporan akhir
    c. Memecah ide besar menjadi komponen kecil
    d. Membuat grafik AI
    Jawaban: c

12. Tools drag-and-drop seperti Orange AI cocok untuk:
    a. Pembuatan animasi
    b. Visualisasi dan pelatihan model AI dasar
    c. Membuat game 3D
    d. Pemrograman sistem operasi
    Jawaban: b

13. Apa tujuan merancang proyek dengan SMART Goals?
    a. Agar proyek cepat selesai tanpa rencana
    b. Agar proyek menarik secara estetika
    c. Agar tujuan jelas, terukur, dan realistis
    d. Agar proyek kompleks dan rumit
    Jawaban: c

14. Manfaat eksplorasi masalah sekitar dalam perencanaan proyek adalah:
    a. Menciptakan solusi yang tidak relevan
    b. Meningkatkan tingkat kesulitan proyek
    c. Menyesuaikan proyek dengan kebutuhan nyata
    d. Menghindari keterlibatan masyarakat
    Jawaban: c

15. Salah satu kekurangan Google Teachable Machine adalah:
    a. Terlalu sulit dipahami pemula
    b. Tidak dapat melatih model suara
    c. Kontrol terhadap model sangat terbatas
    d. Tidak dapat digunakan tanpa internet
    Jawaban: c

C. Isian Singkat (5 Soal)

16. Teknik untuk mengeksplorasi pengembangan ide yang sudah ada disebut _____.
    Jawaban: SCAMPER

17. Proyek AI yang baik harus memiliki dampak _____ terhadap pengguna atau masyarakat.
    Jawaban: positif

18. Tools seperti App Inventor 2 menggunakan pendekatan _____ untuk membuat aplikasi.

Jawaban: visual/blok

19. Model klasifikasi sederhana bisa dilatih tanpa coding menggunakan _____.
    Jawaban: Google Teachable Machine

20. Tujuan proyek harus dirumuskan dengan metode _____ agar jelas dan terukur.
    Jawaban: SMART

D. Soal Eksplorasi

Soal:

Bayangkan kamu tinggal di daerah yang sering mengalami banjir. Jelaskan bagaimana kamu bisa menggunakan AI untuk membantu masyarakat di sekitarmu menghadapi masalah ini. Sertakan ide alat atau aplikasi yang mungkin kamu buat dan tools apa yang akan kamu pilih.

Soal:

Kamu dan tim mu ingin membuat chatbot untuk membantu siswa menjawab pertanyaan pelajaran. Tuliskan langkah awal yang harus dilakukan mulai dari identifikasi masalah hingga pemilihan tools yang cocok.

Soal:

Jelaskan alasan mengapa proyek “AI untuk Memprediksi Harga Saham” kurang cocok untuk proyek akhir di tingkat SMA. Bandingkan dengan ide yang lebih sederhana namun tetap bermanfaat.

Tugas Proyek

Proyek 1: Tantangan 1 Halaman: Rancang Proposal Proyek AI Mini

Deskripsi Proyek:
Siswa diminta membuat proposal singkat (maks. 1 halaman) yang menjelaskan ide proyek AI pilihan mereka. Proposal harus mencakup: judul, masalah yang diangkat, solusi berbasis AI, tujuan utama proyek, dan alat (tools) yang akan digunakan.

Langkah-langkah:

- Pilih topik yang kamu minati (misalnya: edukasi, kesehatan mental, keamanan sekolah).
- Identifikasi satu masalah nyata yang bisa diselesaikan dengan AI.
- Rancang solusi kreatif yang memanfaatkan teknologi AI.
- Tulis proposal dalam format ringkas namun jelas.
- Gunakan template untuk memudahkan struktur penulisan.

Proyek 2: AI untuk Sekolah Lebih Baik: Proyek Pemecahan Masalah

Deskripsi Proyek:
Dalam kelompok kecil, siswa memilih satu permasalahan nyata di lingkungan sekolah (misalnya: keterlambatan, sampah, kebisingan) dan merancang solusi berbasis AI. Hasil proyek dipresentasikan di depan kelas.

Langkah-langkah:

- Brainstorm masalah yang sering terjadi di sekolah.
- Pilih satu masalah dan lakukan riset singkat (wawancara, observasi, dll).
- Kembangkan ide solusi menggunakan AI (misalnya: pengenalan suara, klasifikasi gambar, chatbot).
- Buat presentasi menarik dengan visualisasi ide dan manfaatnya.
- Presentasikan dengan percaya diri dan buka sesi tanya-jawab.

Proyek 3: Peta Ide AI + SCAMPER Kreatif

Deskripsi Proyek:
Siswa diminta membuat mind map berisi 3 ide proyek AI yang menurut mereka menarik. Dari salah satu ide tersebut, gunakan teknik SCAMPER (Substitute, Combine, Adapt, Modify, Put to another use, Eliminate, Reverse) untuk mengembangkan proyek lebih lanjut.

Langkah-langkah:

- Brainstorm 3 ide awal proyek AI dan buat dalam bentuk mind map.
- Pilih 1 ide yang paling potensial.
- Gunakan setiap langkah SCAMPER untuk menantang kreativitas dan menyempurnakan ide tersebut.
- Catat pengembangan ide hasil SCAMPER.
- Tulis deskripsi singkat dari versi final ide yang telah dikembangkan.

Proyek 4: Eksperimen AI Interaktif: Google Teachable Machine

Deskripsi Proyek:
Siswa mengenal dan mencoba Google Teachable Machine untuk membuat model klasifikasi sederhana, misalnya mengenali ekspresi wajah senang vs sedih. Setelah itu, siswa membuat laporan proses eksperimen dan hasil yang diperoleh.

Langkah-langkah:

- Akses situs Teachable Machine.
- Pilih tipe model (image, sound, atau pose), lalu kumpulkan data sendiri.
- Latih model dengan beberapa contoh per kategori (minimal 2 kelas).
- Uji hasil dan catat akurasi serta tantangan yang muncul.
- Buat laporan singkat yang mencakup proses, kesulitan, hasil, dan pembelajaran.

Proyek 5: Jadwal Misi AI: Buat Timeline Proyek Akhirmu

Deskripsi Proyek:
Siswa menyusun timeline proyek AI dari awal hingga presentasi akhir. Proyek bisa dilakukan individu atau kelompok, dan timeline harus mencakup minimal 5 tahap kegiatan.

Langkah-langkah:

- Tentukan ide atau topik proyek terlebih dahulu.
- Bagi seluruh proses menjadi beberapa tahap (misalnya: perencanaan, riset, pengumpulan data, pelatihan model, evaluasi).
- Susun timeline berdasarkan minggu/bulan yang tersedia di semester ini.
- Tambahkan milestone penting dan waktu cadangan untuk revisi.
- Presentasikan timeline ke guru atau kelompok lain untuk mendapat masukan.

Proyek 6: Proposal Proyek Akhir AI: Menuju Proyek Nyata

Deskripsi Proyek:
Siswa secara individu atau kelompok menulis proposal lengkap untuk proyek AI akhir mereka. Proposal ini mencakup komponen penting seperti latar belakang, tujuan, metodologi, dan anggaran (jika perlu).

Langkah-langkah:

- Tentukan topik dan masalah yang ingin diselesaikan.
- Rancang tujuan proyek yang terukur dan realistis.
- Buat garis besar solusi dan metodologi (tools, tahapan, jenis data).
- Susun jadwal proyek dan prediksi kebutuhan (waktu, alat, biaya).
- Tulis proposal sesuai format yang ditentukan dan siap untuk diseminasi.

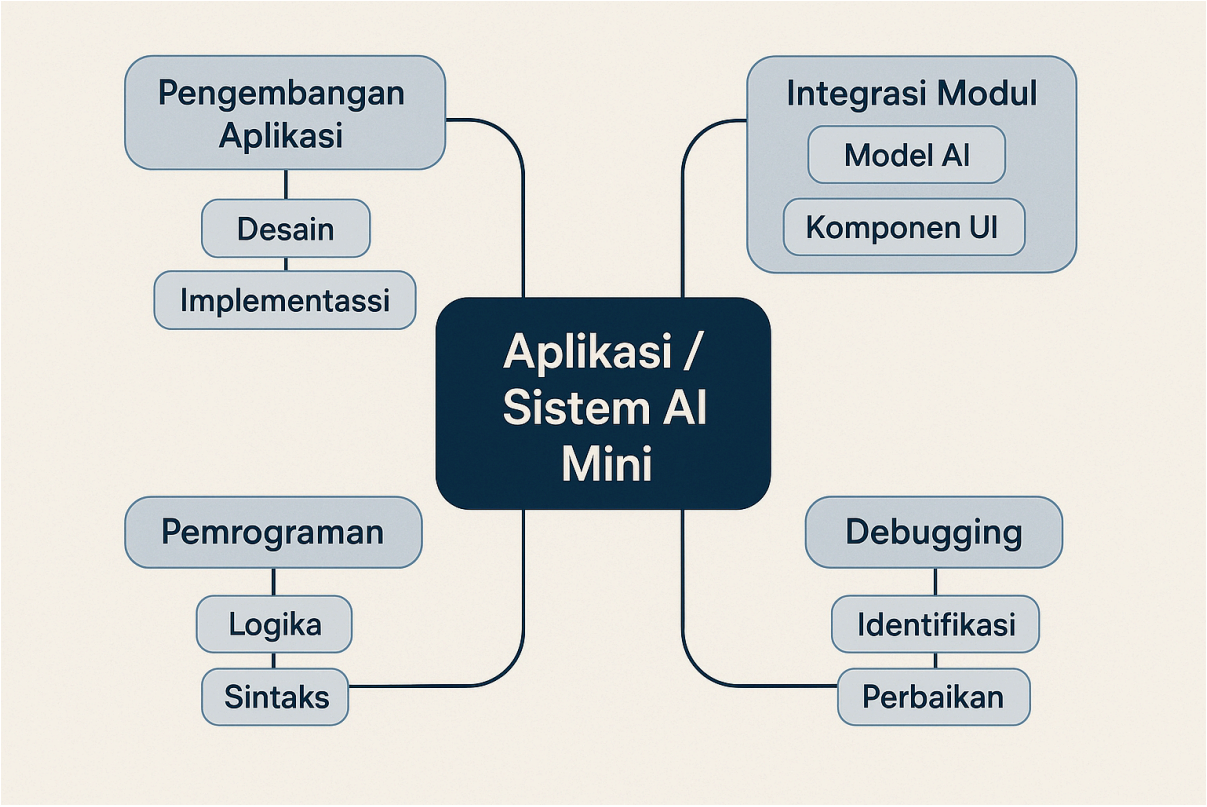
BAB 2: Pengembangan Aplikasi / Sistem AI Mini

Tujuan Pembelajaran

Setelah belajar bab ini, kalian diharapkan mampu:

- Merancang dan membuat aplikasi mini berbasis AI secara mandiri atau berkelompok.
- Memahami langkah-langkah utama pengembangan aplikasi: dari ide, desain, coding, hingga pengujian.
- Menggunakan tools seru seperti **App Inventor 2** atau **Teachable Machine** untuk membuat aplikasi yang pintar dan berguna.
- Menguji aplikasi dan memperbaikinya agar bisa digunakan oleh orang lain.

Peta Konsep



Apersepsi

Sebelum kita mulai membuat aplikasi AI mini sendiri, yuk kita coba lihat dulu contoh aplikasi sederhana berbasis AI! Misalnya, aplikasi yang bisa mengenali benda di sekitar kita lewat kamera HP—seru, kan? Demo ini bisa dibuat dengan alat bantu seperti **App Inventor 2** atau **Google Teachable Machine**, jadi kalian tidak perlu jago coding dulu untuk menikmatinya. Setelah menonton demonya, kita akan diskusi singkat: Apa sih sebenarnya yang dilakukan aplikasi itu? Bagian mana yang disebut AI-nya? Dan kenapa menurut kalian aplikasi ini keren atau menarik? Nah, setelah itu, tantangan dimulai! Bayangkan kalian jadi

pembuat aplikasinya—aplikasi AI seperti apa yang akan kalian buat, dan siapa yang akan menggunakannya? Yuk, buktikan ide kreatif kalian!

Penjelasan Konsep (teori)

Dalam pengembangan aplikasi atau sistem berbasis AI (Artificial Intelligence/Kecerdasan Buatan), kita tidak bisa langsung lompat ke bagian coding tanpa peta jalan yang jelas. Ibarat membangun rumah, kita perlu merancang dari pondasi hingga atap. Maka dari itu, penting untuk memahami **tahapan proses rekayasa perangkat lunak (software engineering process)** serta komponen teknis yang mendukungnya.

Mari kita kupas satu per satu, sambil tetap seru!

Tahapan Pengembangan Aplikasi / Sistem AI Mini

Pengembangan aplikasi AI mengikuti alur kerja sistematis. Berikut adalah tahapan-tahapan utama yang harus dilalui:

1. **Perencanaan (Planning)**
    - Merumuskan ide: aplikasi AI apa yang ingin dibuat? Untuk apa dan siapa penggunanya?
    - Menentukan fitur: apa saja kemampuan yang harus dimiliki aplikasi?
    - Memetakan batasan dan kebutuhan: hardware apa yang digunakan? Koneksi internet?
    - Membuat kerangka dasar (arsitektur) dari aplikasi.

2. **Desain (Design)**
    - Merancang **antarmuka pengguna (UI/UX)**: bagaimana tampilan aplikasi? Apakah mudah digunakan?
    - Menentukan **alur logika**: bagaimana pengguna akan berinteraksi dengan aplikasi?

3. **Implementasi (Implementation/Coding)**
    - Menulis kode program atau menyusun blok logika. Bisa menggunakan bahasa pemrograman seperti Python, atau platform visual seperti **App Inventor 2**.

4. **Pengujian (Testing)**
    - Mengecek apakah aplikasi berjalan sesuai rencana.
    - Menemukan dan memperbaiki bug/error.

5. **Penyebaran (Deployment)**
    - Membuat aplikasi bisa digunakan oleh orang lain (misalnya, menginstal di HP atau dibagikan secara online).

6. **Pemeliharaan (Maintenance)**

- Setelah rilis, aplikasi mungkin butuh perbaikan atau penambahan fitur baru. Nah, disinilah proses pemeliharaan bekerja.

Dasar Pemrograman Python (Jika Relevan)

Jika kamu menggunakan Python dalam proyek AI mini-mu, berikut beberapa dasar yang wajib dikuasai:

Struktur Dasar Program Python

1. Variabel

Variabel adalah nama yang digunakan untuk menyimpan **data** dalam program.

```
nama = "Aulia"
umur = 30
```

- `nama` menyimpan teks (string).
- `umur` menyimpan angka (integer).

2. Tipe Data

Python memiliki berbagai tipe data. Berikut contohnya:

```
# Tipe dasar
angka = 10                  # int
desimal = 3.14              # float
teks = "Halo"               # str
boolean = True              # bool
# Struktur data
daftar = [1, 2, 3]          # list
kamus = {"nama": "Aulia"}   # dict
```

3. Operator

Operator digunakan untuk melakukan operasi pada nilai/variabel.

```
# Aritmatika
a = 5 + 3   # Penjumlahan
b = 10 - 2  # Pengurangan
c = 4 * 2   # Perkalian
d = 8 / 2   # Pembagian

# Logika

x = True and False   # False
y = True or False    # True
z = not True         # False
```

4. Kontrol Alur (Flow Control)

Python menggunakan `if`, `for`, dan `while` untuk mengatur alur eksekusi program.

```
# if-else
umur = 20
if umur >= 17:
    print("Dewasa")
else:
    print("Anak-anak")

# for loop
for i in range(3):
    print("Ulang:", i)
# while loop
count = 0
while count < 3:
    print("Hitung:", count)
    count += 1
```

Fungsi

Fungsi adalah blok kode yang dapat dipanggil berulang kali.

```
def sapa(nama):
    print("Halo,", nama)
sapa("Aulia")
sapa("Budi")
```

- Fungsi `sapa` menerima parameter `nama` dan mencetak pesan.

Library

Library adalah kumpulan kode tambahan yang dapat digunakan dengan `import`.

```
import random
import math
import numpy as np

# random
angka_acak = random.randint(1, 10)

# math
akar = math.sqrt(16)

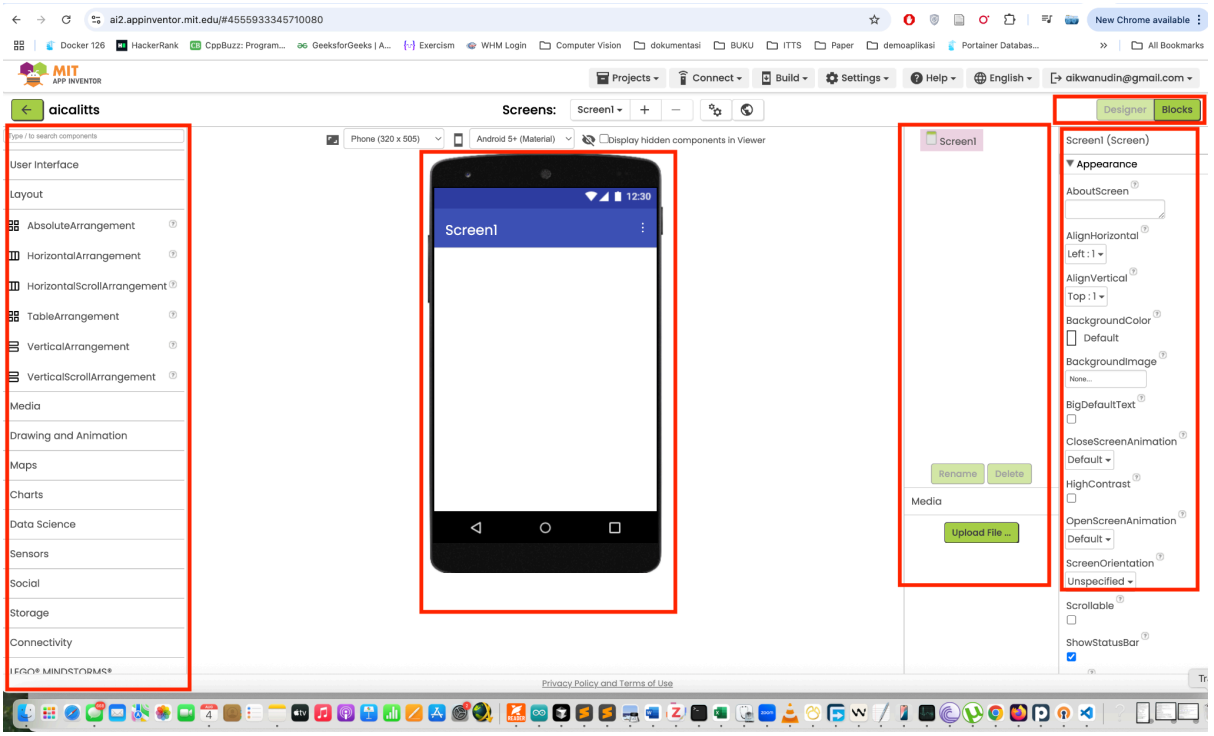
# numpy
array = np.array([1, 2, 3])
```

- `random`: untuk angka acak.
- `math`: untuk fungsi matematika.
- `numpy`: untuk pengolahan array numerik.

Pengenalan App Inventor 2

App Inventor 2 adalah platform visual yang sangat cocok untuk pemula, terutama untuk membuat aplikasi Android. Kamu tidak perlu menulis kode, cukup menyusun **blok logika** layaknya puzzle!

Desainer UI (User Interface Designer)



Gambar di atas menunjukkan antarmuka utama "Designer" pada MIT App Inventor.

1. Area Proyek dan Tampilan Utama Aplikasi (Tengah)

Menampilkan simulasi smartphone Android.
Di sini siswa bisa melihat preview dari aplikasi yang sedang mereka buat.
Sekarang hanya ada satu Button dengan label “Screen1”.

2. Palette (Kiri)

Berisi komponen user interface yang bisa ditambahkan ke aplikasi:
Button, Label, TextBox, Image, ListView, dll.

Cukup drag and drop ke area tengah untuk digunakan.

3. Viewer (Tengah)

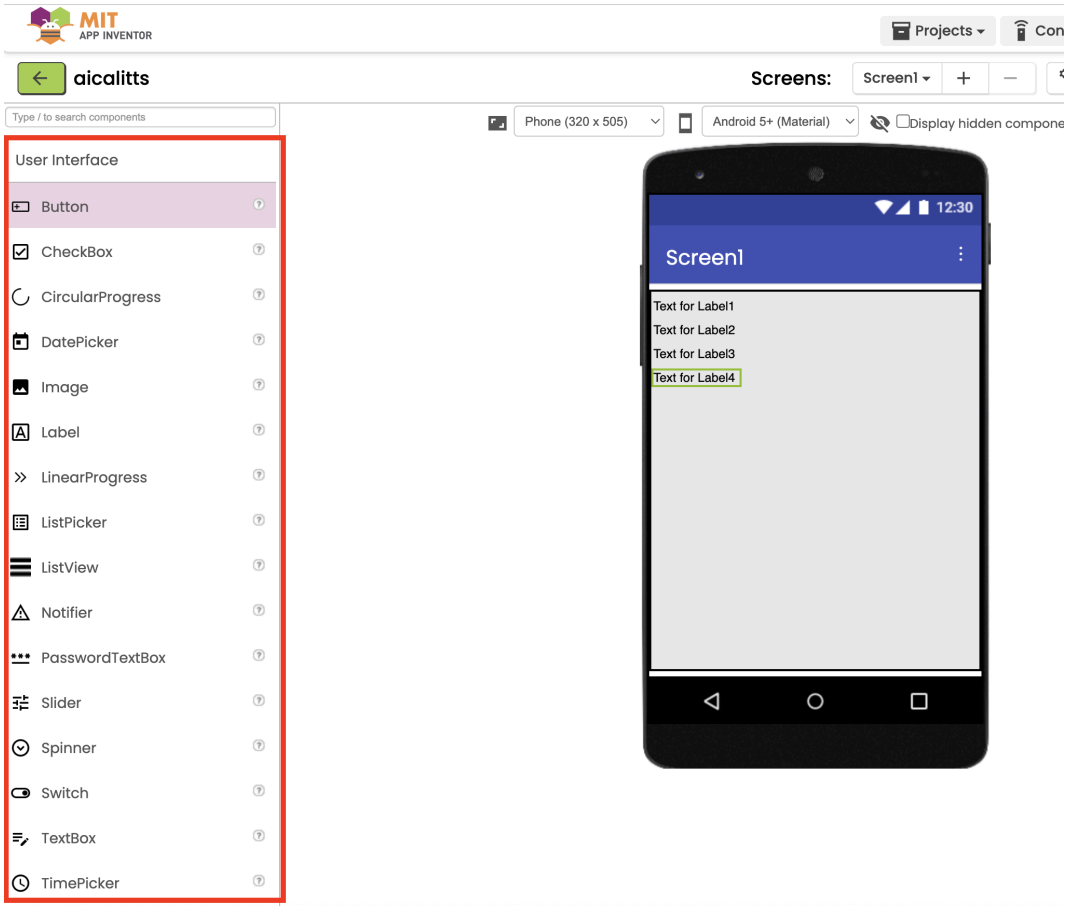
Tempat menyusun tampilan aplikasi Android.
Seperti meletakkan komponen-komponen ke layar ponsel.
Siswa dapat mengatur posisi dan layout-nya.

4. Komponen Properties (Kanan)

Setelah memilih satu komponen (misalnya Button), di sini bisa:
Mengubah teks,
Mengatur posisi horizontal/vertikal (AlignHorizontal, AlignVertical),
Mengganti warna latar belakang (BackgroundColor),
Menentukan orientasi layar (ScreenOrientation), dll.
Ini adalah tempat siswa menyesuaikan tampilan dan perilaku dari elemen yang dipilih.

5. Tabs: Designer dan Blocks

Designer = untuk tampilan aplikasi (UI).
Blocks = untuk menyusun logika pemrograman dengan blok visual seperti puzzle (misalnya: ketika tombol diklik, lakukan sesuatu).



Button

Komponen tombol.
Jika ditekan, bisa memicu aksi tertentu (misalnya: tampilkan pesan, pindah layar, dll).

CheckBox

Komponen kotak centang.
Cocok digunakan untuk pilihan ya/tidak atau lebih dari satu pilihan.

CircularProgress

Menampilkan indikator proses berbentuk lingkaran (biasanya untuk loading).
Cocok untuk menandakan aplikasi sedang memuat data.

DatePicker

Komponen untuk memilih tanggal dari kalender.
Cocok untuk aplikasi yang butuh input tanggal (misal: booking, jadwal).

Image

Untuk menampilkan gambar dari file atau URL.
Bisa digunakan untuk menampilkan logo, foto, dll.

Label

Komponen untuk menampilkan teks statis di layar.
Contoh: “Selamat datang”, “Nama Anda:”.

LinearProgress

Indikator proses yang berbentuk garis lurus.
Bisa digunakan untuk menunjukkan progress tugas atau loading.

ListPicker

Membuka daftar pilihan dari daftar data.
Mirip menu dropdown atau combobox.

ListView

Menampilkan daftar data secara vertikal.
Cocok untuk aplikasi daftar item, seperti daftar nama, produk, dll.

Notifier

Untuk menampilkan pesan (pop-up), dialog konfirmasi, atau alert.
Contoh: "Data berhasil disimpan!", "Yakin ingin keluar?".

PasswordTextBox

Seperti TextBox biasa, tapi isiannya disamarkan (●●●).
Cocok untuk input password atau PIN.

Slider

Komponen batang geser untuk memilih nilai dalam rentang tertentu.
Contoh: mengatur volume atau tingkat kecerahan.

Spinner

Dropdown list (mirip ListPicker) tapi langsung terlihat tanpa membuka layar baru.

Switch

Tombol on/off (seperti saklar).
Cocok untuk pilihan biner: aktif/tidak aktif, nyala/mati.

TextBox

Tempat pengguna mengetikkan teks.
Bisa digunakan untuk mengisi nama, komentar, dll.

TimePicker

Untuk memilih waktu (jam dan menit).
Berguna untuk aplikasi pengingat, alarm, atau jadwal.

Editor Blok (Blocks Editor)

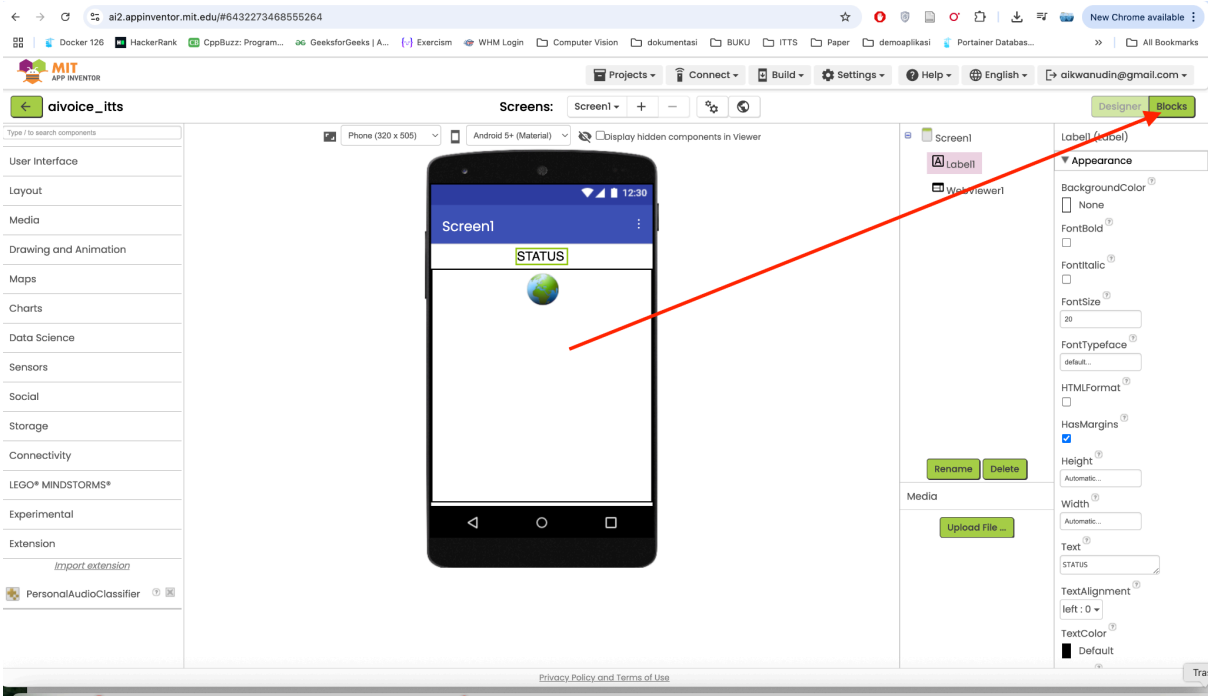
Di MIT App Inventor 2, membuat aplikasi dibagi menjadi dua bagian: **desain** dan **coding**.

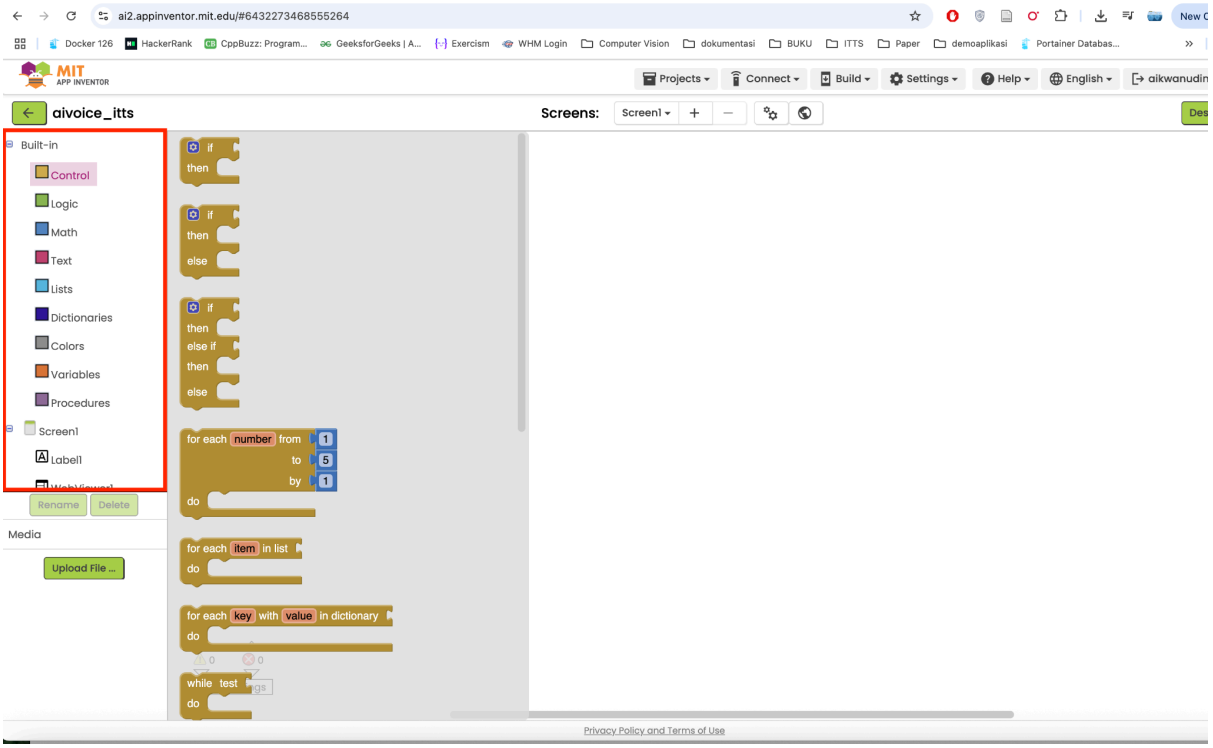
Bagian **desain** (Designer) digunakan untuk mengatur tampilan aplikasi, misalnya menambahkan tombol, teks, gambar, atau komponen lainnya di layar.

Supaya aplikasi bisa bekerja dan punya fungsi, kita perlu masuk ke bagian **coding**. Di App Inventor 2, coding dilakukan dengan cara menyusun **blok-blok** seperti mainan puzzle, jadi tidak perlu mengetik kode yang rumit.

Caranya mudah: cukup klik tombol **"Blocks"** di pojok kanan atas layar. Setelah itu, kita bisa mulai menyusun blok-blok logika agar aplikasi bisa melakukan apa yang kita inginkan.

Dengan metode ini, membuat aplikasi Android jadi lebih gampang dan seru, bahkan untuk pemula.





Gambar di atas menunjukkan tampilan **MIT App Inventor 2** di bagian **Blocks Editor**, yaitu tempat kita membuat logika program dengan cara menyusun blok-blok seperti puzzle.

Di sisi kiri (kotak merah) ada berbagai kategori blok bawaan (**Built-in**) yang punya fungsi berbeda-beda:

1. **Control** – Blok untuk mengatur jalannya program, misalnya *jika-maka* (if-then), *jika-maka-sebaliknya* (if-then-else), atau perulangan (for, while).
2. **Logic** – Blok untuk logika seperti benar/salah (true/false) dan perbandingan angka atau teks.
3. **Math** – Blok untuk menghitung, misalnya tambah, kurang, kali, bagi, dan operasi matematika lainnya.
4. **Text** – Blok untuk bekerja dengan teks, misalnya menggabungkan kata atau mencari kata tertentu.
5. **Lists** – Blok untuk membuat dan mengatur daftar (list) berisi banyak data.
6. **Dictionaries** – Blok untuk menyimpan data dengan pasangan *nama* dan *nilai* (key-value).
7. **Colors** – Blok untuk mengatur warna.
8. **Variables** – Blok untuk membuat dan menyimpan data dalam variabel.
9. **Procedures** – Blok untuk membuat "fungsi" atau "perintah khusus" yang bisa digunakan berulang kali.

Di bawah kategori ini, ada blok untuk **komponen di layar (Screen1)**, seperti `Label1`, yang punya perintah dan pengaturan sendiri.

Cara kerja coding di App Inventor:

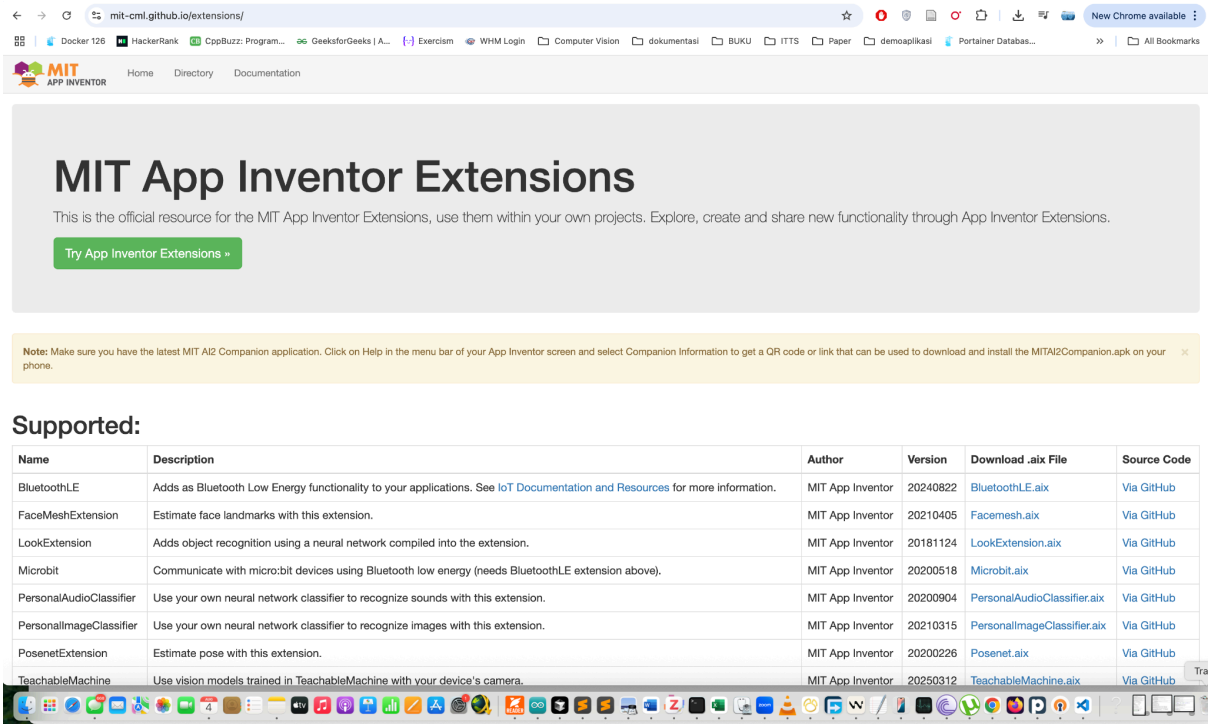
Kita tinggal drag and drop (seret dan lepas) blok ke area kerja di kanan, lalu menyusunnya seperti puzzle. Blok-blok ini saling terhubung, jadi kita tidak perlu khawatir salah penulisan kode. Dengan cara ini, kita bisa membuat logika aplikasi dengan mudah, bahkan tanpa pengalaman coding sebelumnya.

Ekstensi AI

Ada beberapa extensi AI untuk App Inventor 2. Extensi App Inventor 2 dapat di akses di web:

https://mit-cml.github.io/extensions/

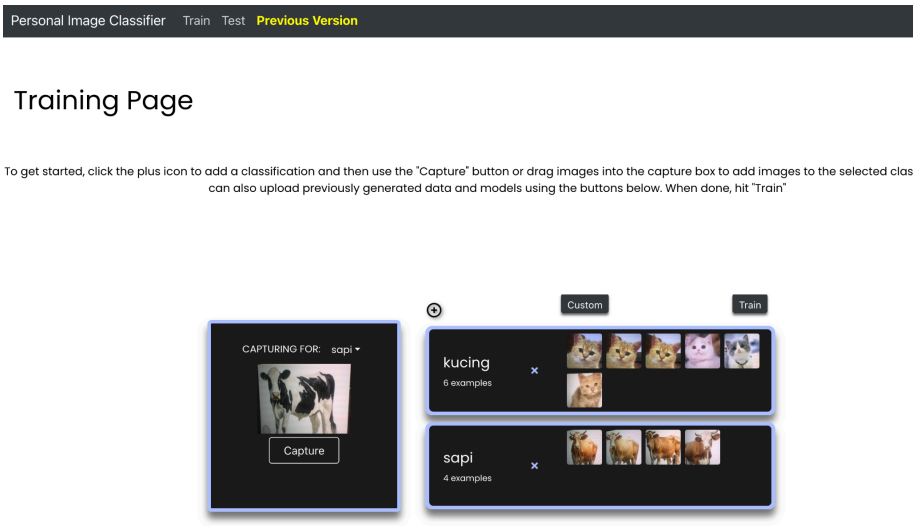
Menambahkan komponen AI, seperti:



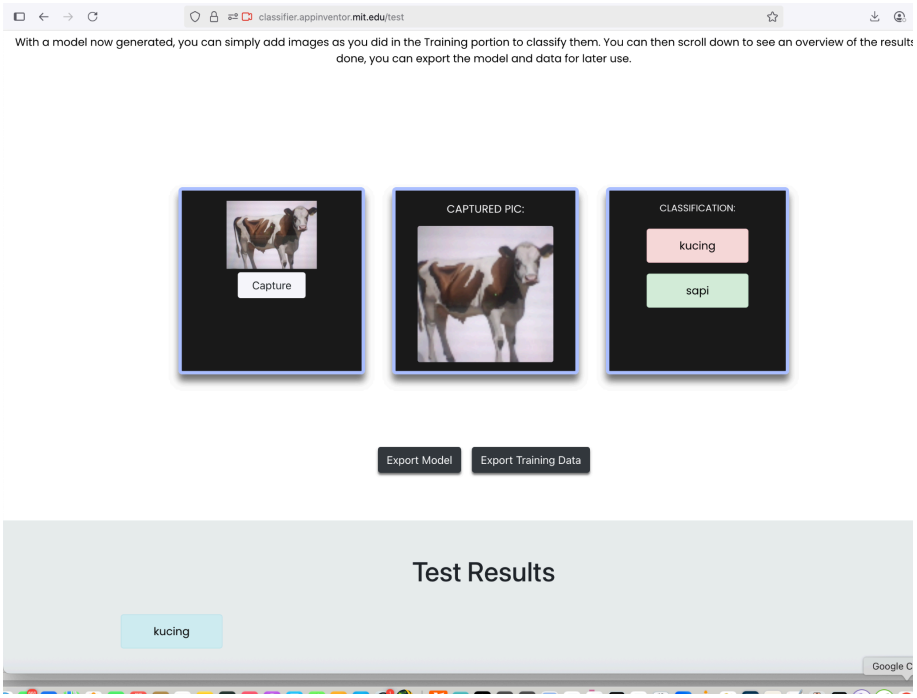
Personal Image Classifier

Personal Image Classifier (PIC) adalah alat yang bisa melatih komputer untuk mengenali dan membedakan gambar sesuai kategori yang kita tentukan. Kita mengunggah beberapa contoh gambar untuk setiap kategori, lalu sistem mempelajarinya. Setelah itu, kita bisa menguji apakah komputer dapat menebak kategori dari gambar baru yang belum pernah dilihatnya. URL PIC adalah:

https://classifier.appinventor.mit.edu/



Gambar di atas menunjukkan halaman pelatihan untuk membuat program pengenal gambar sederhana. Ada dua kategori yang dilatih, yaitu “kucing” (kucing) dan “sapi” (sapi), masing-masing berisi beberapa contoh foto. Pengguna bisa menambahkan kategori baru dengan tombol plus, lalu mengambil gambar lewat tombol “Capture” atau mengunggah foto untuk dimasukkan ke kategori yang dipilih. Setelah semua contoh terkumpul, tombol “Train” digunakan untuk melatih komputer agar bisa mengenali gambar sesuai kategori yang sudah dibuat.

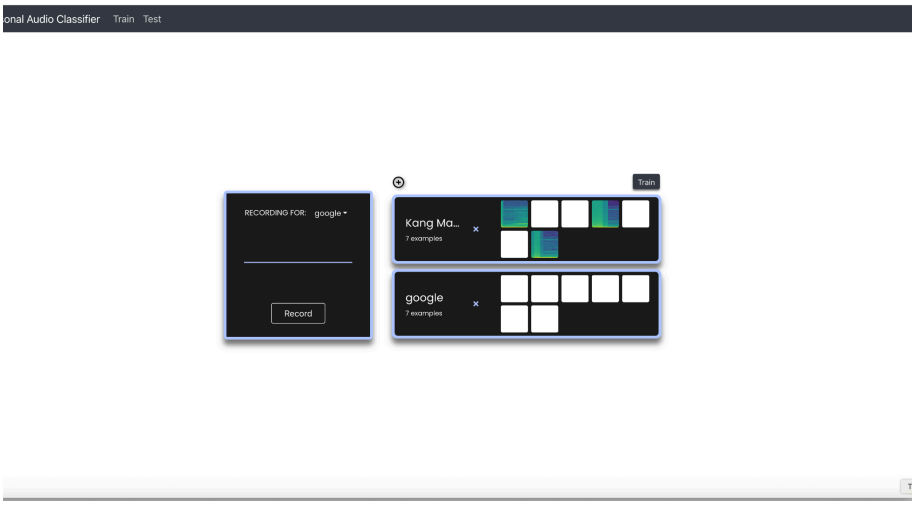


Gambar diatas menunjukkan sebuah percobaan menggunakan aplikasi klasifikasi gambar di MIT App Inventor. Pada percobaan ini, gambar seekor sapi diambil melalui fitur “Capture”, lalu sistem mencoba mengenali objek pada gambar. Hasil klasifikasi menunjukkan dua pilihan: “kucing” dan “sapi”. Sistem secara baik mendeteksi bahwa gambar adalah Sapi, dengan kode warna hijau.

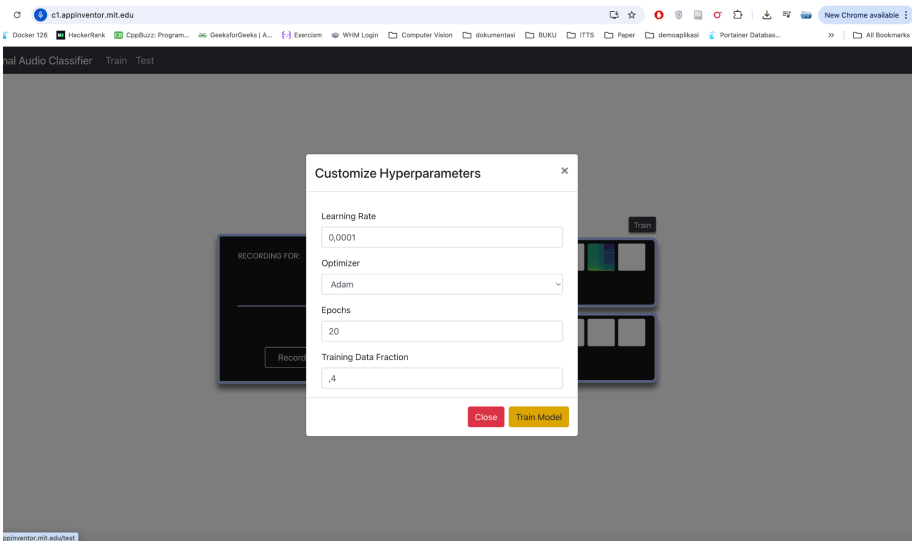
Personal Audio Classifier

Personal Audio Classifier adalah tool berbasis web yang memungkinkan pengguna membuat, melatih, dan menerapkan model klasifikasi suara khusus secara langsung di browser—kemudian digunakan dalam aplikasi seluler lewat ekstensi MIT App Inventor.

https://c1.appinventor.mit.edu/

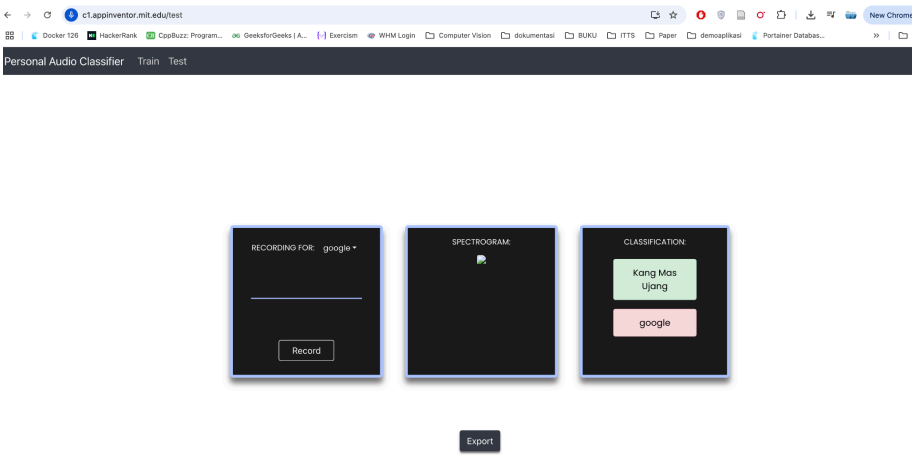


Gambar tersebut menampilkan antarmuka **Personal Audio Classifier** yang digunakan untuk merekam, mengelompokkan, dan melatih model pengenalan suara berdasarkan contoh audio yang dibuat pengguna.



Gambar di atas menampilkan antarmuka aplikasi *Personal Audio Classifier* pada platform MIT App Inventor, khususnya pada jendela pop-up “Customize Hyperparameters” yang digunakan untuk mengatur parameter pelatihan model kecerdasan buatan. Di dalamnya terdapat beberapa kolom isian seperti *Learning Rate* yang diatur ke 0,0001, pilihan *Optimizer* yang menggunakan Adam, jumlah *Epochs* sebanyak 20, serta *Training Data Fraction* sebesar 0,4. Tombol “Close” berwarna merah digunakan untuk menutup jendela, sedangkan tombol “Train Model” berwarna kuning digunakan untuk memulai proses

pelatihan model sesuai parameter yang telah diatur. Tampilan ini menunjukkan tahap persiapan sebelum melatih model untuk mengklasifikasikan audio secara personal.



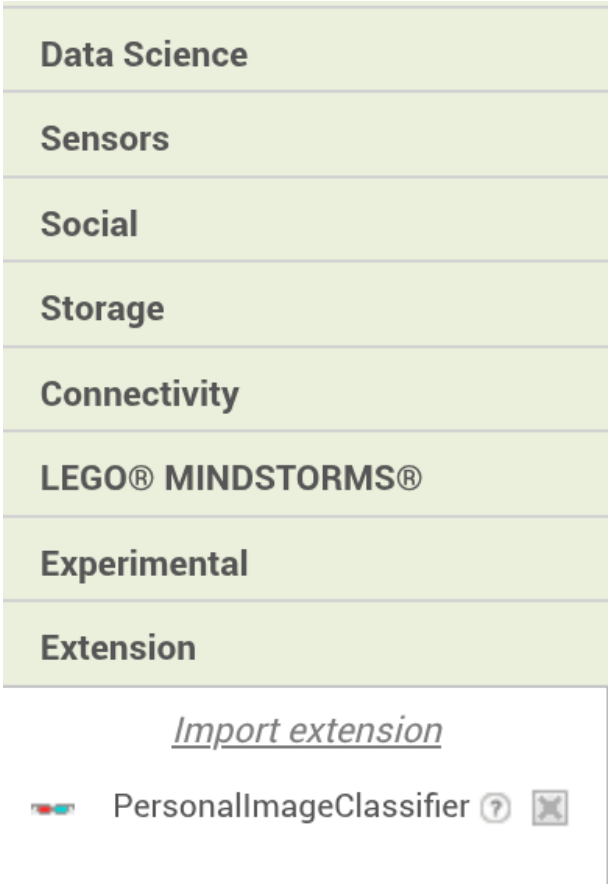
Gambar di atas menampilkan antarmuka aplikasi web *Personal Audio Classifier* yang digunakan untuk melatih dan menguji pengklasifikasian suara secara personal. Pada tampilan ini, terdapat tiga panel utama: panel pertama di sebelah kiri digunakan untuk merekam audio berdasarkan label yang dipilih (misalnya "google"), panel kedua di tengah seharusnya menampilkan hasil *spectrogram* dari rekaman namun terlihat tidak muncul dengan benar, dan panel ketiga di sebelah kanan menampilkan hasil klasifikasi suara ke dalam dua kemungkinan kategori, yaitu “Kang Mas Ujang” dan “google” dengan indikator warna berbeda. Di bagian bawah, terdapat tombol “Export” yang kemungkinan digunakan untuk menyimpan atau mengekspor model atau data hasil pelatihan.

Contoh Blok Program App Inventor 2

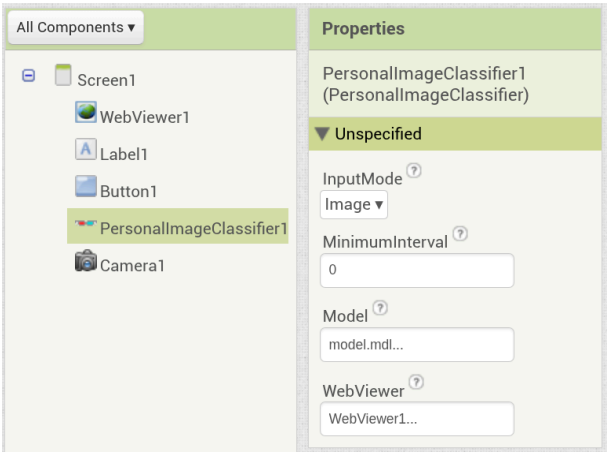
Kita bisa mencoba membuat aplikasi sederhana di App Inventor 2 dengan memanfaatkan **Personal Image Classifier** sebagai contoh penggunaan kecerdasan buatan (AI). Misalnya, kita membuat aplikasi pendeteksi jenis sampah—organik dan anorganik—dengan cara mengumpulkan foto-foto sampah dari ponsel, lalu melatih model AI di Personal Image Classifier agar bisa mengenali perbedaannya. Setelah model dilatih, kita menghubungkannya ke App Inventor 2 sehingga aplikasi yang kita buat dapat langsung memprediksi kategori sampah dari gambar yang diambil kamera. Proyek ini tidak hanya melatih keterampilan coding visual, tetapi juga memberi pemahaman nyata tentang cara AI belajar dari data dan digunakan untuk membantu memecahkan masalah di kehidupan sehari-hari.

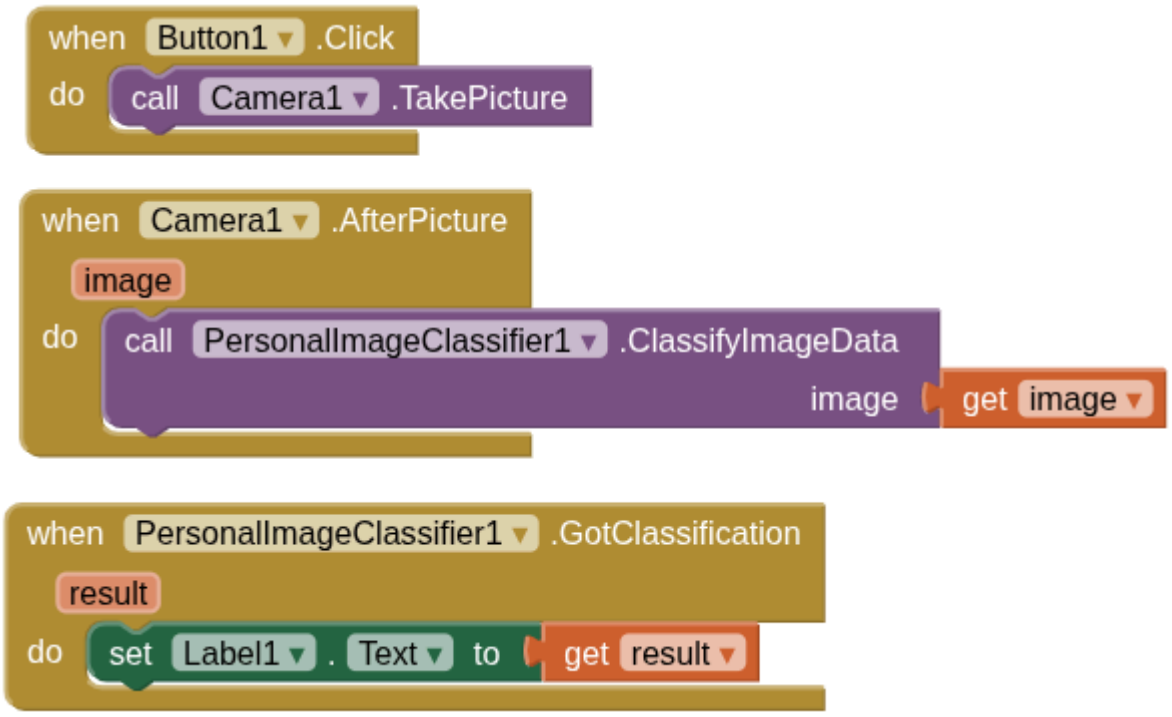
Untuk mengunggah (upload) ekstensi **PersonalImageClassifier** di MIT App Inventor, pertama buka tab **Extension** di panel komponen, lalu klik tautan *Import extension*. Setelah itu, pilih opsi "*Import from my computer*" dan unggah file ekstensi **.aix** yang sudah kamu miliki, misalnya, `PersonalImageClassifier.aix`.

Jika proses berhasil, nama **PersonalImageClassifier** akan muncul di daftar ekstensi dan siap digunakan di proyek sebagai komponen non-visual untuk membuat aplikasi yang bisa mengenali dan mengklasifikasikan gambar.

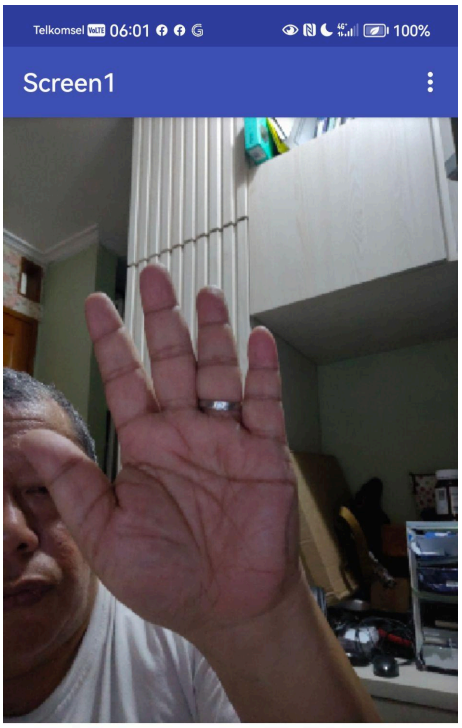


Gambar berikutnya menunjukkan tampilan App Inventor 2 dengan beberapa komponen seperti WebViewer, Label, Button, Camera, dan yang paling penting **PersonalImageClassifier1**. Komponen PersonalImageClassifier ini digunakan untuk membuat aplikasi yang bisa mengenali gambar menggunakan kecerdasan buatan (AI). Pada bagian **Model** terdapat parameter `model.mdl` yang sangat penting karena file inilah yang berisi "otak" AI—model pembelajaran mesin yang sudah dilatih untuk mengenali objek atau gambar tertentu. Tanpa file model ini, komponen tidak bisa melakukan klasifikasi gambar karena tidak memiliki pengetahuan yang diperlukan. Jadi, jika diibaratkan, PersonalImageClassifier adalah tubuhnya, sedangkan `model.mdl` adalah pikirannya yang membuatnya bisa "mengerti" gambar yang diberikan.





Blok program App Inventor 2 tersebut berfungsi untuk mengambil gambar menggunakan kamera, kemudian menganalisisnya dengan AI melalui komponen **PersonalImageClassifier** dan menampilkan hasil klasifikasi ke layar. Saat **Button1** ditekan, aplikasi memanggil **Camera1.TakePicture** untuk mengambil foto. Setelah foto diambil (**Camera1.AfterPicture**), gambar tersebut dikirim ke **PersonalImageClassifier1.ClassifyImageData** untuk diproses. Ketika hasil klasifikasi sudah tersedia (**GotClassification**), nilai hasil analisis tersebut ditampilkan pada **Label1.Text** sehingga pengguna dapat melihat kategori atau klasifikasi gambar yang telah dianalisis AI.



Gambar selanjutnya menampilkan sebuah aplikasi yang dibuat dengan App Inventor 2 dan dijalankan pada perangkat HP. Aplikasi ini memanfaatkan fitur **Personal Image Classifier** untuk mengenali gesture tangan. Pada tampilan terlihat kamera menangkap gambar seorang pengguna yang mengangkat telapak tangan terbuka ke arah kamera. Hasil klasifikasi ditampilkan di bagian bawah layar dalam format JSON:

{"Bye":0.86182,"Siap":0.13879}



Artinya, sistem mengidentifikasi gesture tersebut sebagai **kelas "Bye"** dengan tingkat keyakinan sebesar **0.86182** atau sekitar **86%**, sedangkan kemungkinan gesture tersebut sebagai "Siap" hanya **13.8%**. Hal ini menunjukkan model berhasil mengenali gerakan tangan melambaikan salam perpisahan secara cukup akurat.

Eksperimen dengan Google Teachable Machine

Google Teachable Machine adalah platform berbasis web yang memungkinkan kamu **melatih model AI tanpa coding**. Berikut langkah-langkahnya:
https://teachablemachine.withgoogle.com/

1. **Mengumpulkan Data**
    - Ambil banyak contoh gambar/suara/pose dari berbagai variasi untuk tiap kategori.
    - Semakin banyak dan beragam data, semakin pintar modelnya!

2. **Melatih Model (Training)**
    - Klik tombol “Train Model” dan biarkan Teachable Machine belajar dari data yang kamu berikan.

3. **Evaluasi Model**
    - Uji apakah model bisa mengenali data baru dengan akurat.

4. **Mengekspor Model**
    - Setelah puas dengan hasilnya, ekspor model ke format yang bisa digunakan di aplikasi kamu, misalnya:
        - Untuk App Inventor (.tflite)
        - Untuk web (TensorFlow.js)

Sertakan tangkapan layar (screenshot) setiap langkah agar dokumentasi makin keren dan mudah diikuti!

Experimen Integrasi Google Teachable Machine dengan Python

Langkah 1: Export Model dari Teachable Machine

1. Buka proyek di Teachable Machine.
2. Klik **Export Model**.
3. Pilih **TensorFlow → Download (keras model)**.
4. Akan diunduh file ZIP yang berisi model Keras (`.h5`) dan labelnya. Biasanya bernama converted_keras.zip
5. Copykan ke folder tempat programming python akan dilakukan misalnya pythonenv
6. unzip file converted_keras.zip tersebut.

Langkah 2: Buat Script Python untuk Google Colab

Langkah yang perlu dilakukan

1. Login ke Google Colab https://colab.research.google.com/ menggunakan account Google.
2. Klik tombol New Notebook
3. Paste Source code di bawah ini.
4. Klik Run All untuk menjalankan.
5. Upload model Google Teachable Machine keras_model.h5 jika diminta.
6. Upload labels.txt jika diminta
7. Upload gambar yang akan diklasifikasi dari hasil tangkapan kamera ketika diminta.

Source code Python untuk Google Colab.

```
# @title ▶️ Jalankan Sel Ini untuk Memuat Model & Melakukan Prediksi
# @markdown ### tensorflow version 2.x

# 1. Menginstal pustaka yang diperlukan
import tensorflow as tf
import jax

import tensorflow as tf
import numpy as np
from keras.models import load_model
from PIL import Image, ImageOps
from google.colab import files
import io

# Menonaktifkan notasi ilmiah untuk kejelasan
np.set_printoptions(suppress=True)

print("✅ Pustaka berhasil diinstal dan diimpor.")

# 2. Fungsi untuk Mengunggah dan Memuat Model & Label
def upload_and_load_model():
    """
    Meminta pengguna untuk mengunggah file model Keras (.h5) dan file label (.txt)
    dari Teachable Machine, lalu memuatnya.
    """
    print("\n⬆️ Silakan unggah file 'model_keras.h5' Anda...")
```

```
    uploaded_model = files.upload()
    model_path = next(iter(uploaded_model))
    model = load_model(model_path, compile=False)
    print("✅ Model berhasil dimuat.")

    print("\n⬆️ Sekarang, silakan unggah file 'labels.txt'
Anda...")
    uploaded_labels = files.upload()
    labels_path = next(iter(uploaded_labels))
    class_names = open(labels_path, "r").readlines()
    print("✅ Label berhasil dimuat.")

    return model, class_names

# Memuat model dan nama kelas
try:
    model, class_names = upload_and_load_model()

    # 3. Mengunggah dan Memproses Gambar Pengguna
    print("\n🖼️ Silakan unggah gambar yang ingin Anda
klasifikasikan...")
    uploaded_image = files.upload()
    image_path = next(iter(uploaded_image))

    # Membuat array dengan bentuk yang tepat untuk diumpankan ke
model Keras
    # Dimensi array adalah: (1, 224, 224, 3)
    data = np.ndarray(shape=(1, 224, 224, 3), dtype=np.float32)

    # Membuka dan mengubah ukuran gambar
    image = Image.open(image_path).convert("RGB")
    size = (224, 224)
    image = ImageOps.fit(image, size, Image.Resampling.LANCZOS)

    # Mengubah gambar menjadi array numpy
    image_array = np.asarray(image)

    # Normalisasi gambar
    normalized_image_array = (image_array.astype(np.float32) /
127.5) - 1

    # Memuat gambar ke dalam array
    data[0] = normalized_image_array
```

```
    # 4. Melakukan Prediksi
    prediction = model.predict(data)
    index = np.argmax(prediction)
    class_name = class_names[index]
    confidence_score = prediction[0][index]

    # 5. Menampilkan Hasil
    print("\n\n--- HASIL PREDIKSI ---")
    print("🤖 Kelas:", class_name[2:]) # Menghilangkan nomor
indeks dari nama kelas
    print("Confidence Score: {:.2f}%".format(confidence_score *
100))
    print("----------------------")

except Exception as e:
    print(f"\n Terjadi kesalahan: {e}")
    print("Pastikan Anda telah mengunggah file yang benar
('model_keras.h5' dan 'labels.txt') dan formatnya tidak rusak.")
```

Pengujian Aplikasi AI Mini

Pastikan aplikasimu bukan hanya "jalan", tapi **berjalan dengan baik**. Beberapa jenis pengujian yang bisa dilakukan:

- **Uji Fungsional**
  - Apakah setiap tombol dan fitur berfungsi sesuai rencana?
- **Uji Integrasi**
  - Apakah semua bagian aplikasi (UI, AI, sensor, dst.) saling terhubung dan bekerja sama dengan baik?
- **Uji Pengguna (User Testing)**
  - Ajak teman atau keluarga mencoba aplikasimu. Minta masukan: apakah aplikasi mudah digunakan? Apakah AI-nya bekerja dengan baik?

Penutup: Belajar AI itu Keren!

Dengan mempelajari tahapan pengembangan aplikasi mini berbasis AI, kamu tidak hanya memahami cara membuat aplikasi, tapi juga belajar berpikir logis, kreatif, dan kolaboratif. Siap menjadi pencipta teknologi masa depan? Tantang dirimu, mulai dari ide kecil dan jadikan nyata!

Ilustrasi: “Dari Bingung Jadi Bisa!”

Bayangkan kamu adalah siswa SMA yang baru pertama kali mendengar istilah **AI (Artificial Intelligence)**. Lalu guru menunjukkan demo aplikasi sederhana: cukup arahkan kamera ke wajah temanmu, dan aplikasi bisa bilang “kamu sedang tersenyum!”. Wah, keren ya?

Awalnya kamu bingung, “Kok bisa sih? Aku belum ngerti coding!” Tapi setelah tahu bahwa kita bisa membuat aplikasi seperti itu **tanpa harus jago ngoding**—cukup pakai **Google Teachable Machine** dan **App Inventor 2**, semangatmu pun bangkit! Kamu mulai dengan hal sederhana: mengumpulkan gambar, melatih model, lalu menghubungkannya ke aplikasi buatanmu. Meskipun sering salah deteksi dan tampilan masih polos, kamu belajar memperbaikinya satu per satu.

Perjuanganmu menghasilkan aplikasi nyata yang bisa dikenalkan ke teman dan keluarga. Kamu bangga: “Ini aku yang buat!”

Contoh Kasus atau Ilustrasi

Kasus 1: Mood Detector: Mengenali Ekspresi Wajah dengan AI

Ilustrasi Kasus:
Bayangkan kamu dan temanmu membuat aplikasi Android yang bisa membaca *mood* hanya dari wajah kalian! Kamera ponsel digunakan untuk mengenali apakah seseorang sedang **senang, sedih, atau marah**.

Cara kerja singkat:

- Gunakan kamera di aplikasi untuk mengambil gambar wajah.
- Aplikasi menganalisis ekspresi dengan AI dan menampilkan hasilnya di layar.

Tantangan seru:
Kamu harus **melatih AI dengan ekspresi wajahmu sendiri dan teman-temanmu**. Semakin banyak variasi, semakin pintar aplikasimu!

Tools yang digunakan: Teachable Machine + App Inventor 2

Kasus 2: Smart Plant Identifier: Mengenali Jenis Tanaman dari Foto

Ilustrasi Kasus:
Pernah lihat tanaman unik dan ingin tahu namanya? Sekarang bayangkan kamu membuat **aplikasi web** yang bisa mengenali jenis tanaman dari **foto daun**.

Cara kerja singkat:

- Kamu unggah gambar daun ke halaman web.
- AI menganalisis bentuk daun dan memberikan nama tanaman.

Tantangan seru:
Pastikan gambar yang kamu unggah **jelas dan bebas dari latar belakang yang membingungkan**, supaya AI tidak salah nebak!

Tools yang digunakan: Teachable Machine + HTML/CSS + JavaScript

Kasus 3: Smart Trash Classifier: Klasifikasi Sampah Pintar

Ilustrasi Kasus:
Bayangkan sebuah sistem sederhana yang bisa mengenali apakah sampah termasuk **organik, plastik, atau kertas** hanya dengan melihat bentuk dan warnanya!

Cara kerja singkat:

- Gambar jenis sampah diunggah ke sistem.
- AI mengenali dan menampilkan jenisnya.

Tantangan seru:
Kamu harus berpikir: **apa sih ciri visual yang membedakan plastik dan kertas?** Tantangan ini melatih logika dan kejelianmu dalam mengelola data visual.

Tools yang digunakan: Teachable Machine + Python (simulasi)

Kasus 4: Smart Voice Calculator: Kalkulator Berbasis Suara

Ilustrasi Kasus:
Bayangkan kamu bisa berkata "dua tambah tiga" dan aplikasimu langsung menjawab "lima"! Seru, kan?

Cara kerja singkat:

- Aplikasi mendengar perintahmu lewat suara.
- Suara diubah menjadi teks, kemudian diproses menjadi perhitungan.
- Hasil muncul di layar.

Tantangan seru:
Apa yang terjadi jika pengguna mengatakan "duaa tambah tigaaa"? Sistem harus bisa mengatasi kesalahan pelafalan!

Tools yang digunakan: App Inventor 2 + Speech Recognition + Text-to-Speech

Kasus 5: EcoSort Challenge: Game Edukasi Pemilah Sampah

Ilustrasi Kasus:
Kamu membuat **game edukatif berbasis web** tempat pemain harus menarik gambar sampah ke tempat yang tepat — organik, kertas, atau plastik.

Cara kerja singkat:

- Gambar-gambar sampah muncul di layar.
- Pemain drag-drop ke tempat sampah yang sesuai.
- Skor ditampilkan sesuai ketepatan.

Tantangan seru:
Desain antarmuka yang **menarik dan intuitif**, serta sistem skor yang membuat pemain semangat terus bermain sambil belajar.

Tools yang digunakan: HTML/CSS/JavaScript (opsional AI untuk klasifikasi otomatis)

Kasus 6: Ekspresi AI: Tebak Mood Temanmu

Ilustrasi Kasus:
Aplikasi yang **menyimpan data ekspresi wajah banyak orang**, lalu saat kamu foto wajah temanmu, aplikasi menebak ekspresinya berdasarkan pola yang dikenali.

Cara kerja singkat:

- Model AI dilatih dengan berbagai ekspresi dari banyak wajah.
- Saat diuji, aplikasi menebak apakah orang dalam foto sedang bahagia, terkejut, atau netral.

Tantangan seru:
Kamu harus membuat model AI yang bisa **membedakan ekspresi halus** — misalnya, senyum tipis vs senyum lebar.

Tools yang digunakan: Teachable Machine + Web Interface sederhana

Contoh Soal dan Pembahasan

Soal: Tantangan Validasi Kalkulator

Bayangkan kamu membuat aplikasi kalkulator sederhana. Suatu saat, pengguna malah memasukkan huruf, bukan angka. Aplikasi pun langsung error!
Pertanyaannya: Jenis bug apa yang terjadi di sini? Dan, sebutkan dua cara cerdas untuk mencegahnya agar aplikasi tetap berjalan mulus!

Pembahasan:
Ini adalah contoh **bug validasi input**. Bug terjadi karena sistem menerima data yang

tidak sesuai harapan (huruf padahal seharusnya angka).
Dua metode pencegahan:

1. **Validasi Input** – Cek tipe data sebelum digunakan, misalnya dengan `isnumeric()` atau konversi yang aman.
2. **Error Handling** – Gunakan `try-except` untuk menangani kesalahan saat proses konversi atau perhitungan.

Soal: Siapa Kamu – Desainer atau Programmer?

Dalam App Inventor 2, kamu bisa melihat dua sisi dunia: Desainer dan Blocks Editor. Kapan kamu perlu memakai masing-masing? Apa sih sebenarnya perbedaan fungsi dari keduanya?

Pembahasan:

- **Desainer** digunakan untuk mengatur tampilan antarmuka pengguna (UI), seperti menambahkan tombol, gambar, atau label.
- **Blocks Editor** digunakan untuk menulis logika aplikasi, seperti apa yang terjadi jika tombol ditekan, atau bagaimana aplikasi merespons input.

Gunakan **Desainer** saat ingin menyusun tampilan, dan **Blocks Editor** saat ingin menulis "otak" dari aplikasi.

Soal: Uji Coba Aplikasi, Tapi Bagaimana?

Kamu sudah selesai membuat aplikasi deteksi gambar. Tapi, bagaimana cara memastikan aplikasimu benar-benar bekerja dengan baik dan disukai pengguna? Jelaskan dua jenis pengujian yang perlu kamu lakukan!

Pembahasan:

1. **Testing Fungsional** – Uji setiap fitur aplikasi, apakah bekerja sesuai harapan. Contoh: Saat tombol ditekan, apakah benar hasil klasifikasi muncul?
2. **Testing Pengguna** – Ajak pengguna mencoba aplikasimu, lalu amati apakah mereka merasa nyaman, bingung, atau senang saat menggunakannya.

Soal: AI-nya Pintar atau Bingung?

Kamu melatih model AI menggunakan Teachable Machine. Tapi ternyata, semua data latihmu tampak serupa. Apa risiko yang terjadi, dan mengapa penting menggunakan data yang beragam?

Pembahasan:
AI belajar dari pola data. Jika data terlalu seragam, model tidak belajar menghadapi variasi nyata di dunia. Akibatnya, saat diuji dengan data baru yang sedikit berbeda, AI bisa gagal mengenalinya. Jadi, **keberagaman data = kecerdasan model yang lebih baik!**

Soal: Menyusun Fungsi Python – Siap Tantangan Kategori Angka?

Buatlah sebuah fungsi Python yang bisa mengelompokkan angka ke dalam tiga kategori berikut:

- "kecil" jika kurang dari 10
- "sedang" jika 10 hingga 20
- "besar" jika lebih dari 20

Pembahasan (kode fungsi):

```
def kategori_angka(nilai):
    if nilai < 10:
        return "kecil"
    elif nilai <= 20:
        return "sedang"
    else:
        return "besar"
```

Fungsi ini menggunakan percabangan `if-elif-else` untuk menentukan kategori berdasarkan nilai input.

Fakta Menarik / Fun Facts

- **Bug pertama dalam sejarah komputer** bukanlah kesalahan kode—tapi **seekor ngengat** yang terjebak dalam mesin! Dari situlah istilah 'debugging' muncul.
- Proses pengembangan aplikasi tidak selalu mulus! Bahkan raksasa teknologi seperti Google dan Apple pun menghadapi bug dan harus terus memperbarui aplikasi mereka.
- **AI bisa dilatih hanya dengan kamera laptop dan beberapa klik**, tidak perlu jago coding!
- Beberapa aplikasi pintar yang kamu pakai sehari-hari (seperti TikTok dan Google Lens) juga menggunakan model AI mirip dengan yang kamu pelajari di sekolah.

Tips Belajar atau Tips Cepat Menghafal

- **Teknik Debugging:**
    1. **Print/Log:** Menampilkan nilai variabel atau pesan di titik-titik tertentu untuk melacak alur program.
    2. **Breakpoints:** Menghentikan eksekusi program sementara untuk memeriksa status.
    3. **Divide and Conquer:** Memecah masalah besar menjadi bagian-bagian kecil yang lebih mudah diuji.

4. **Baca Pesan Error:** Pesan error seringkali memberikan petunjuk lokasi dan jenis kesalahan.

- Selalu mulai dengan bagian terkecil yang berfungsi, lalu tambahkan fitur satu per satu.
- Rutin melakukan 'commit' atau menyimpan progres pekerjaan Anda!

Mind Mapping



Cerita Membantu Hafalan

"Suatu hari, Adit buat aplikasi pendeteksi wajah. Tapi saat menguji, AI bilang kucingnya sedang ‘sedih’ padahal kucingnya tidur! Ternyata AI-nya belum diajari gambar wajah manusia yang jelas. Dari situ, Adit belajar pentingnya **melatih model dengan data yang tepat**!"

Aktivitas Siswa

Aktivitas 1: Mulai Mewujudkan Proyek Akhir

Deskripsi Aktivitas:
Siswa mulai mengimplementasikan bagian awal dari proyek akhir sesuai dengan

proposal yang telah disetujui. Fokus pada pembuatan kerangka dasar sistem atau aplikasi.

Langkah yang Disarankan:

- Tinjau kembali proposal dan rencana kerja.
- Tentukan bagian proyek yang paling logis untuk dimulai (misalnya UI, sistem login, atau input data).
- Gunakan papan tulis atau papan digital untuk membuat alur logika dasar sebelum coding.
- Konsultasikan progres awal dengan guru pembimbing untuk memastikan arah kerja sudah tepat.

Aktivitas 2: Membangun Fondasi Program (Coding Time!)

Deskripsi Aktivitas:
Siswa mulai menulis kode atau menyusun blok program sebagai dasar dari sistem atau AI mini yang akan dikembangkan.

Langkah yang Disarankan:

- Tentukan fitur inti yang ingin dibangun terlebih dahulu.
- Gunakan teknik clean coding: beri nama variabel yang jelas dan tambahkan komentar.
- Bekerja dalam pasangan (pair programming) untuk saling memberi masukan.
- Lakukan uji coba kecil setiap kali menambahkan bagian kode baru.

Aktivitas 3: Eksplorasi Logika & Fungsi: Uji Fungsionalitas Dasar

Deskripsi Aktivitas:
Siswa menguji apakah fungsi-fungsi utama dari aplikasi atau model AI mereka berjalan sesuai harapan.

Langkah yang Disarankan:

- Buat daftar fungsionalitas minimum yang perlu diuji.
- Lakukan pengujian dengan data input yang bervariasi.
- Dokumentasikan hasil pengujian: mana yang berhasil, mana yang perlu diperbaiki.
- Diskusikan hambatan teknis dengan teman kelompok atau guru.

Aktivitas 4: Tantangan Kilat: Siapa Cepat Latih Model!

Deskripsi Aktivitas:
Kompetisi seru antar kelompok: siapa yang dapat melatih model klasifikasi wajah dengan akurasi terbaik hanya dalam 15 menit?

Langkah yang Disarankan:

- Persiapkan dataset sederhana terlebih dahulu.
- Tentukan pendekatan pelatihan yang akan digunakan (misalnya KNN, CNN, atau lainnya).
- Gunakan waktu secara strategis: 5 menit untuk setup, 10 menit untuk pelatihan dan evaluasi.
- Setelah selesai, presentasikan hasil dan refleksi pendek tentang strategi yang digunakan.

Aktivitas 5: Hackathon Mini: Bangun MVP dalam 1 Jam!

Deskripsi Aktivitas:
Dalam waktu 60 menit, setiap kelompok membuat **Minimum Viable Product** (MVP) dari aplikasi yang dirancang, fokus pada fungsi inti saja.

Langkah yang Disarankan:

- Tentukan satu fitur utama yang wajib berjalan (misalnya klasifikasi gambar atau deteksi suara).
- Gunakan library atau template sederhana untuk mempercepat pengembangan.
- Bagikan tugas antar anggota kelompok secara efisien (UI, logika, pengujian).
- Akhiri sesi dengan demo singkat kepada kelompok lain atau guru.

Aktivitas 6: Simulasi Uji Coba Pengguna & Debugging Challenge

Deskripsi Aktivitas:
Gabungan dua aktivitas: siswa mencoba aplikasi kelompok lain sebagai "user" lalu dilanjutkan dengan tantangan debugging aplikasi dengan error tersembunyi.

Langkah yang Disarankan:

- Lakukan pertukaran aplikasi antar kelompok dan isi form umpan balik singkat.
- Beri nilai pada aspek kenyamanan antarmuka, kejelasan instruksi, dan respons sistem.
- Lanjutkan dengan Debugging Challenge: setiap kelompok mendapat potongan kode bermasalah yang harus dianalisis dan diperbaiki dalam waktu terbatas.
- Guru memberikan apresiasi bagi yang menyelesaikan dengan cepat dan tepat.

Rangkuman

Belajar membuat **aplikasi AI mini** adalah petualangan seru yang dimulai dari perencanaan hingga pengujian. Di tahap awal, siswa diajak untuk menuangkan ide, membuat desain aplikasi, dan memahami bagaimana *logika* dalam aplikasi bekerja. Tak hanya teori, mereka

juga langsung praktik menggunakan **Google Teachable Machine** dan **App Inventor 2**, dua alat keren yang memungkinkan pembuatan aplikasi AI tanpa harus jago ngoding!

Proyek ini mengajak siswa SMA untuk mengalami langsung proses membangun aplikasi, dari menyusun data, menyisipkan *model AI*, hingga menguji apakah aplikasinya bekerja dengan baik. Ini bukan hanya soal membuat program, tapi juga tentang bagaimana cara berpikir kritis, menyusun strategi, dan mencoba solusi kreatif untuk berbagai tantangan. Proses pengujian menjadi bagian penting, karena disinilah mereka melihat apakah aplikasi mereka benar-benar bermanfaat dan akurat.

Yang membuat proses ini menarik adalah pendekatannya yang berbasis proyek nyata. Siswa tidak hanya belajar teori, tetapi juga langsung *hands-on*—menyusun, membangun, dan mencoba sendiri aplikasinya. Ini membuat pengalaman belajar jadi lebih hidup, **menantang**, dan tentu saja, **menyenangkan**. Mereka bukan cuma pengguna teknologi, tapi pencipta yang punya kesempatan untuk membuat solusi nyata lewat AI!

Latihan Soal

A. Benar atau Salah (5 Soal)

1. Proses pengujian dilakukan setelah aplikasi dirilis ke publik.
   Salah

2. App Inventor 2 memungkinkan pengguna membuat aplikasi Android tanpa menulis kode.
   Benar

3. Fungsi utama Google Teachable Machine adalah untuk membuat model AI berbasis teks.
   Salah

4. API digunakan untuk menghubungkan aplikasi dengan model AI yang sudah dilatih.
   Benar

5. Salah satu tujuan perencanaan adalah merancang antarmuka pengguna.
   Salah

B. Pilihan Ganda (10 Soal)

1. Tahap pertama dalam proses pengembangan aplikasi adalah:
   a. Pengujian
   b. Implementasi
   c. Perencanaan
   d. Desain
   Jawaban: c

2. Dalam App Inventor 2, "Blocks Editor" digunakan untuk:
   a. Membuat gambar
   b. Menulis logika aplikasi
   c. Mengatur warna latar
   d. Memasang ekstensi
   Jawaban: b

3. Library dalam Python berfungsi untuk:
   a. Menghapus kode
   b. Mempercepat debugging
   c. Menyediakan fungsi siap pakai
   d. Menghentikan program
   Jawaban: c

4. Untuk mengenali gambar tanaman menggunakan AI, kita dapat menggunakan:
   a. HTML
   b. Notepad
   c. Google Teachable Machine
   d. Paint
   Jawaban: c

5. Fungsi `if-else` dalam kode digunakan untuk:
   a. Menyimpan data
   b. Menampilkan gambar
   c. Membuat pilihan berdasarkan kondisi
   d. Menyusun layout
   Jawaban: c

6. Fungsi pemeliharaan dalam pengembangan aplikasi berguna untuk:
   a. Menentukan fitur utama
   b. Mencari bug setelah rilis
   c. Membuat desain awal
   d. Menulis dokumentasi
   Jawaban: b

7. Manakah dari berikut ini termasuk komponen UI di App Inventor 2:
   a. Loop
   b. Variabel
   c. Tombol
   d. Fungsi
   Jawaban: c

8. Perintah `import` di Python digunakan untuk:
   a. Menghapus file
   b. Menyimpan program
   c. Menggunakan library eksternal
   d. Mengubah tipe data

Jawaban: c

9. File model dari Teachable Machine dapat digunakan ulang karena:
   a. Sudah dalam format .doc
   b. Tidak bisa diubah
   c. Sudah dilatih dan diekspor
   d. Dihapus otomatis
   Jawaban: c

10. Testing pengguna bertujuan untuk:
    a. Mempercepat waktu loading
    b. Menyesuaikan warna UI
    c. Menguji pengalaman pemakai
    d. Menambah fitur
    Jawaban: c

C. Isian Singkat (5 Soal)

1. Tahapan pengembangan aplikasi yang melibatkan penulisan kode disebut _____.
   Jawaban: Implementasi

2. Untuk menghubungkan aplikasi dengan model AI, kita bisa menggunakan konsep _____.
   Jawaban: API

3. Alat dari Google yang bisa melatih model AI secara visual dan mudah digunakan adalah _____.
   Jawaban: Teachable Machine

4. Dalam proses debugging, salah satu teknik yang sering digunakan untuk melacak kesalahan adalah _____.
   Jawaban: Print/Log

5. Model AI yang telah dilatih di Teachable Machine bisa diekspor dalam format _____.
   Jawaban: .zip atau .json

D. Soal Eksplorasi

Soal:

Bayangkan kamu ingin membuat aplikasi untuk mengenali jenis makanan dari foto menggunakan Google Teachable Machine dan App Inventor 2. Jelaskan langkah-langkah utama yang harus kamu lakukan mulai dari pengumpulan data hingga aplikasi bisa digunakan di HP Android.

Soal:

Kamu bekerja dalam tim membuat aplikasi "Suasana Hati Hari Ini" yang mengenali ekspresi wajah. Apa saja peran yang sebaiknya ada dalam tim tersebut, dan bagaimana pembagian tugasnya agar aplikasi selesai tepat waktu dan berfungsi dengan baik?

Soal:

Setelah membuat aplikasi AI mini, kamu menguji aplikasi ke teman sekelas. Mereka memberi masukan bahwa aplikasi kadang salah mengenali gambar. Apa langkah-langkah evaluasi dan perbaikan yang bisa kamu lakukan berdasarkan masukan tersebut?

Tugas Proyek

Proyek 1: Suara Ajaib – Aplikasi Teks ke Suara

Deskripsi Proyek:
Bangun aplikasi Android sederhana menggunakan **App Inventor 2** yang dapat membaca teks yang diketik oleh pengguna menggunakan fitur **Text-to-Speech (TTS)**. Tambahkan tombol untuk “Ulangi” suara dan “Hapus” teks.

Langkah-Langkah:

1. Merancang antarmuka dengan komponen *TextBox*, tombol “Ulangi”, dan tombol “Hapus”.
2. Tambahkan komponen **Text-to-Speech** di App Inventor.
3. Buat blok logika agar ketika tombol ditekan, teks dibacakan atau dihapus.
4. Uji aplikasi dengan berbagai jenis kalimat.
5. (Opsional) Tambahkan pilihan bahasa atau kecepatan suara.

Tujuan Pembelajaran: Mengenal prinsip antarmuka pengguna, pemrosesan input, dan output suara dalam aplikasi.

Proyek 2: Mood Scanner AI – Mengenali Ekspresi Wajah

Deskripsi Proyek:
Latih model AI di **Google Teachable Machine** untuk mengenali minimal tiga ekspresi wajah (misalnya senang, sedih, dan netral). Integrasikan model ke dalam aplikasi Android menggunakan **App Inventor 2**.

Langkah-Langkah:

1. Kumpulkan data wajah dan latih model di Teachable Machine.
2. Ekspor model sebagai file *TensorFlow.js*.
3. Rancang UI aplikasi dengan tombol “Deteksi Ekspresi” dan tampilan ikon emosi.
4. Gunakan *WebViewer* atau metode integrasi lain untuk menampilkan hasil deteksi.
5. Tambahkan fitur log harian ekspresi pengguna.

Tujuan Pembelajaran: Mengenal klasifikasi citra wajah, integrasi AI ke dalam aplikasi, dan desain UI yang informatif.

Proyek 3: Kalkulator AI Python – Mengkategorikan Angka

Deskripsi Proyek:
Buat modul program sederhana di **Python** yang mengkategorikan input angka sebagai “kecil”, “sedang”, atau “besar” berdasarkan batas tertentu. Tampilkan pesan motivasi sesuai kategori.

Langkah-Langkah:

1. Buat fungsi Python dengan input angka dan struktur `if-elif-else`.
2. Tentukan batas kategori (misalnya: <30 = kecil, 30–70 = sedang, >70 = besar).
3. Hubungkan setiap kategori dengan pesan motivasi yang menarik.
4. Tambahkan opsi agar pengguna bisa memasukkan angka berkali-kali.
5. Tampilkan hasil secara jelas di terminal atau GUI sederhana.

Tujuan Pembelajaran: Mengasah logika pemrograman, kontrol alur, dan pemahaman simulasi AI sederhana.

Proyek 4: AI untuk Lingkungan – Deteksi Jenis Sampah

Deskripsi Proyek:
Buat aplikasi yang dapat mengenali jenis sampah (organik, anorganik, B3) dari gambar menggunakan model buatan di **Teachable Machine**. Sertakan fitur edukasi untuk mendukung kesadaran lingkungan.

Langkah-Langkah:

1. Kumpulkan gambar sampah dari berbagai jenis.
2. Latih model klasifikasi gambar di Teachable Machine.
3. Buat aplikasi Android menggunakan App Inventor 2 dengan *WebViewer* atau API integrasi.
4. Tambahkan fitur edukatif yang menampilkan tips daur ulang berdasarkan jenis sampah.
5. Buat dokumentasi proses dari awal hingga akhir.

Tujuan Pembelajaran: Memahami penerapan AI dalam isu nyata, seperti lingkungan hidup, sekaligus mengasah kemampuan dokumentasi proyek.

Proyek 5: Deteksi Suasana Hati Kelas – Proyek Tim

Deskripsi Proyek:
Kerja kelompok untuk membangun aplikasi yang memungkinkan guru melihat suasana hati siswa sebelum kelas dimulai. Aplikasi berbasis ekspresi wajah atau input manual mood.

Langkah-Langkah:

1. Diskusikan dalam tim: apakah menggunakan kamera (AI ekspresi) atau input manual (misalnya slider emosi).
2. Rancang UI ramah pengguna: ikon emosi, tombol kirim mood, dan log harian.
3. Jika memakai AI: latih model di Teachable Machine untuk ekspresi wajah.
4. Integrasikan model ke aplikasi Android.
5. Tampilkan ringkasan mood kelas kepada guru (misalnya: grafik sederhana atau warna dominan).

Tujuan Pembelajaran: Mendorong kolaborasi tim, pengembangan UI yang empatik, serta eksplorasi penggunaan AI dalam pendidikan.

Proyek 6: MVP Proyek Akhir – Versi Mini Proyek AI

Deskripsi Proyek:
Setiap siswa atau kelompok diminta mengembangkan **satu bagian penting (Minimum Viable Product)** dari proyek akhir berbasis AI. Fokus bisa pada input data, tampilan hasil, atau integrasi model.

Langkah-Langkah:

1. Identifikasi bagian paling penting dari proyek akhir.
2. Pecah tugas menjadi modul kecil (misal: input gambar, deteksi ekspresi, atau menampilkan hasil klasifikasi).
3. Bangun modul tersebut secara fungsional meskipun desain belum sempurna.
4. Uji coba dan dokumentasikan hasil serta kendala teknis.
5. Refleksikan: apakah MVP ini bisa dikembangkan lebih lanjut?

Tujuan Pembelajaran: Mengenalkan prinsip pengembangan produk digital secara bertahap dan iteratif.

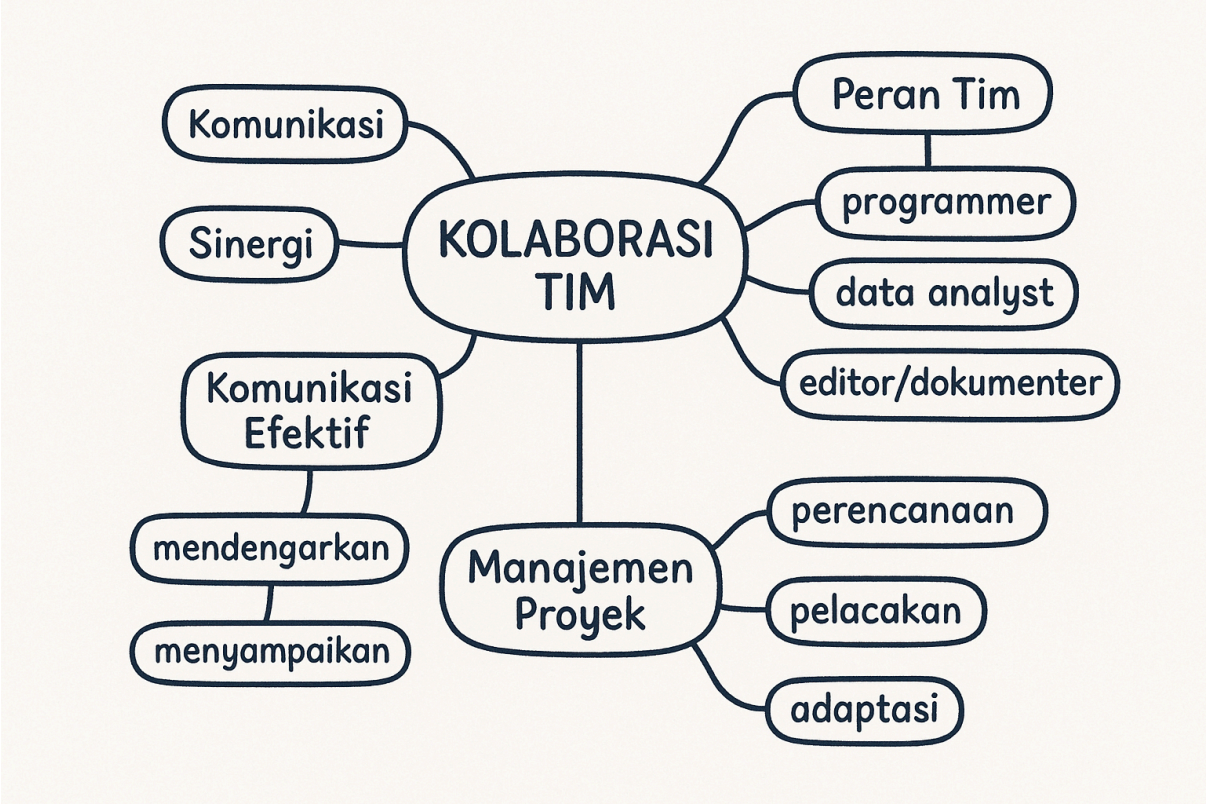
BAB 3: Kolaborasi Tim: Peran Programmer, Data Analyst, Editor

Tujuan Pembelajaran

Setelah belajar bab ini, kamu akan mampu:

- Menjelaskan pentingnya kerja tim dalam proyek AI.
- Mengenal peran-peran utama dalam tim proyek seperti programmer, data analyst, dan editor.
- Menunjukkan cara kerja sama yang efektif dan menyenangkan.
- Menggunakan alat bantu digital untuk mendukung kolaborasi.

Peta Konsep



Apersepsi

Coba ingat-ingat, apakah kamu pernah bermain futsal, tampil di band sekolah, atau mengerjakan tugas kelompok? Semua kegiatan itu pasti membutuhkan kerja sama tim, bukan? Bayangkan kalau dalam satu tim futsal semua pemain hanya mau jadi striker—siapa yang jaga gawang? Atau band tanpa drummer—pasti musiknya terasa hambar. Nah, seperti itulah sebuah proyek Artificial Intelligence (AI). Proyek AI bukan kerjaan satu orang saja, tapi hasil kerja sama tim dengan peran yang berbeda-beda dan saling melengkapi.

Sekarang, mari tonton bersama video singkat tentang tim startup muda yang berhasil menciptakan aplikasi AI. Perhatikan baik-baik: siapa saja yang ada di tim itu, dan bagaimana cara mereka bekerja sama?

Setelah itu, tantangan kecil untukmu: bayangkan kamu diminta membuat aplikasi pendeteksi tanaman langka. Menurutmu, apa saja yang harus disiapkan? Dan siapa saja yang perlu kamu ajak bergabung dalam tim impianmu? Siapkan dirimu untuk menjawabnya sambil belajar seru di bab ini!

Penjelasan Konsep (teori)

Pernahkah kamu bermain dalam tim sepak bola, ikut band sekolah, atau mengerjakan tugas kelompok? Kalau iya, kamu pasti tahu bahwa dalam tim, setiap orang punya tugas dan keahlian masing-masing. Sama seperti itu, dalam sebuah **proyek Artificial Intelligence (AI)**, kerja tim sangat penting agar proyek bisa berjalan lancar dan menghasilkan produk yang keren dan fungsional.

Mengembangkan proyek AI itu tidak seperti mengerjakan PR sendirian di kamar. Ini seperti menggarap pertunjukan besar di mana semua bagian harus menyatu. Butuh banyak orang dengan keahlian berbeda agar semuanya bisa berjalan mulus. Yuk, kita bahas kenapa kerja tim dalam proyek AI itu sangat penting, dan siapa saja yang biasanya terlibat!

Mengapa Kolaborasi dalam Proyek AI itu Penting?

1. **Proyek AI itu Kompleks**
   Proyek AI melibatkan banyak komponen: mulai dari pengumpulan data, pemrograman, hingga desain antarmuka pengguna. Tidak mungkin satu orang saja bisa menangani semuanya dengan maksimal. Maka dari itu, butuh kerja sama banyak orang agar proyek bisa diselesaikan secara efektif.

2. **Tugas Jadi Terbagi, Beban Jadi Ringan**
   Bayangkan jika semua tugas ditumpuk ke satu orang—pasti melelahkan dan tidak efisien. Dengan kolaborasi, tugas bisa dibagi sesuai dengan keahlian tiap anggota tim. Ini seperti dalam band: ada yang pegang gitar, ada yang main drum, ada juga yang jadi vokalis.

3. **Ide yang Beragam, Solusi yang Lebih Kreatif**
   Setiap anggota tim membawa perspektif unik. Diskusi antar anggota bisa memunculkan ide-ide segar yang tidak terpikirkan sebelumnya. Kolaborasi mendorong inovasi!

4. **Proses Lebih Cepat, Hasil Lebih Bagus**
   Dengan bekerja bersama, waktu pengerjaan bisa lebih cepat. Selain itu, ketika satu orang melakukan kesalahan, yang lain bisa membantu mengecek dan memperbaiki, sehingga kualitas akhir lebih baik.

Siapa Saja yang Terlibat dalam Tim Proyek AI?

Berikut ini adalah beberapa peran utama yang biasa ada dalam tim proyek AI (yang bisa disesuaikan untuk level SMA):

1. Programmer / Developer

- **Tugas Utama:**
  Menulis dan menjalankan kode program, mengimplementasikan model AI ke dalam aplikasi, serta mengintegrasikan berbagai komponen teknologi.
- **Keahlian yang Dibutuhkan:**
  - Logika dan struktur pemrograman
  - Kemampuan menulis dan memahami algoritma
  - Keterampilan debugging (mendeteksi dan memperbaiki kesalahan dalam kode)

2. Data Analyst / Data Engineer (Level Siswa SMA)

- **Tugas Utama:**
  Mengumpulkan, membersihkan, dan menyiapkan data yang akan digunakan untuk melatih model AI. Juga bertugas menganalisis hasil kerja model AI.
- **Keahlian yang Dibutuhkan:**
  - Ketelitian dan kesabaran
  - Dasar-dasar statistik
  - Kemampuan menggunakan spreadsheet atau tools seperti Google Sheets

3. Editor / Dokumenter

- **Tugas Utama:**
  Membuat dokumentasi proyek, menulis laporan, membuat presentasi, dan memastikan semua komunikasi dalam tim berjalan lancar.
- **Keahlian yang Dibutuhkan:**
  - Kemampuan menulis dengan jelas
  - Komunikasi yang efektif
  - Perhatian terhadap detail dan struktur informasi

4. Desainer UI/UX (opsional, jika tersedia)

- **Tugas Utama:**
  Membuat tampilan aplikasi yang menarik dan mudah digunakan oleh pengguna.
- **Keahlian yang Dibutuhkan:**
  - Kreativitas
  - Pemahaman prinsip desain (warna, layout, tipografi)
  - Empati terhadap pengalaman pengguna

Komunikasi Efektif: Bukan Sekadar Chatting Biasa

Komunikasi adalah “lem” yang merekatkan semua anggota tim. Tanpa komunikasi yang baik, proyek bisa berantakan walaupun semua anggotanya jago.

Beberapa prinsip komunikasi tim yang efektif:

- **Mendengarkan Aktif**
  Dengarkan ide teman dengan sungguh-sungguh. Jangan hanya menunggu giliran bicara.
- **Penyampaian Ide yang Jelas**
  Bicara atau menulis dengan singkat, padat, dan tidak berbelit-belit.

- **Umpan Balik Konstruktif**
  Kalau memberi kritik, usahakan membangun dan tidak menyakitkan hati. Fokus pada solusi.
- **Gunakan Alat Komunikasi yang Tepat**
  Seperti:
  - **Google Docs:** Untuk menulis dan mengedit laporan bersama
  - **WhatsApp Group:** Untuk komunikasi cepat sehari-hari
  - **Discord:** Untuk diskusi real-time, terutama kalau sedang online meeting

Tools Kolaborasi: Senjata Rahasia Tim Sukses

Agar kerja tim lebih lancar dan terorganisir, kamu bisa menggunakan berbagai tools digital yang mudah diakses dan digunakan:

Tool	Fungsi
Google Drive / Docs / Sheets / Slides	Menyimpan dan mengedit file secara bersamaan secara real-time
WhatsApp Group	Komunikasi cepat dan koordinasi harian
Trello / Asana (opsional)	Mengatur dan melacak progres tugas proyek
GitHub (jika proyek berbasis kode)	Berkolaborasi dalam penulisan kode, serta menyimpan versi-versi proyek dengan aman

Kesimpulan: Satu Tim, Satu Visi

Kolaborasi bukan hanya tentang bekerja bersama-sama, tapi tentang **saling melengkapi**. Dalam tim proyek AI, keberhasilan bukan hanya ditentukan oleh satu orang jenius, tapi oleh **tim yang solid dan terorganisir**. Ketika semua anggota tim memahami peran masing-masing, berkomunikasi dengan baik, dan memanfaatkan tools yang ada—maka proyek apapun bisa diselesaikan dengan hasil yang memuaskan, bahkan luar biasa!

Jika kamu ingin, bagian ini bisa dilengkapi dengan **contoh nyata proyek AI tingkat SMA** yang berhasil karena kerja sama tim. Mau ditambahkan?

Ilustrasi: Perjuangan Mengatasi Masalah Kolaborasi

Bayangkan kamu dan dua temanmu membentuk tim untuk membuat aplikasi AI sederhana yang bisa mengenali suara binatang. Kamu semangat banget di awal, tapi setelah seminggu berjalan, satu temanmu (si “data analyst”) jarang aktif di grup WhatsApp. Data belum dikumpulkan, padahal kamu (si “programmer”) sudah siap membuat programnya.

Kamu bingung: mau ngomong takut menyinggung, tapi kalau dibiarkan proyek bisa gagal. Akhirnya kamu memutuskan untuk **mengajak diskusi baik-baik**, sambil menawarkan bantuan. Ternyata, temanmu kesulitan karena belum paham caranya mencari dataset suara hewan. Setelah ngobrol, kalian akhirnya cari solusi bareng: kamu bantu cari sumber data, dia mengolah dan memberi label. Proyek pun jalan lagi!

Pesan penting: Dalam tim, masalah pasti ada. Tapi dengan komunikasi terbuka dan saling mendukung, semuanya bisa diatasi!

Contoh Kasus atau Ilustrasi

Kasus 1: “Temanmu, Si Raja Konten”

> Bayangkan kamu punya teman sekelas yang viral di media sosial karena sering membagikan video kehidupan pribadi teman-teman tanpa izin. Banyak yang menonton, tapi beberapa orang mulai merasa tidak nyaman.
> **Ilustrasi:** Di suatu hari, video saat kamu menguap lebar di kelas dijadikan meme dan dibagikan tanpa persetujuanmu. Semua orang menertawakannya, bahkan gurumu ikut komentar.
> **Pertanyaan yang timbul:** Apakah semua konten bisa dibagikan hanya karena lucu? Di mana batas antara kreativitas dan privasi?

Kasus 2: “Nilai Bagus, Tapi Bohongan?”

> Di kelasmu, ada isu hangat: seorang siswa diketahui membeli jawaban ujian daring dari luar. Ia mendapatkan nilai sempurna, bahkan mendapat pujian dari guru.
> **Ilustrasi:** Kamu melihat sendiri orang itu membayar melalui aplikasi e-wallet. Sekarang, ia terpilih jadi ketua OSIS karena dianggap "cerdas dan berprestasi."
> **Pertanyaan yang muncul:** Apakah keberhasilan bisa dibenarkan kalau prosesnya curang? Apa pengaruhnya terhadap siswa lain yang jujur tapi nilainya lebih rendah?

Kasus 3: “Debat Digital: ChatGPT Vs Buku Teks”

Di tengah diskusi kelas, seorang siswa berargumen keras bahwa buku pelajaran sudah tidak relevan karena semua bisa dicari lewat AI seperti ChatGPT.
Ilustrasi: Ia bisa menjawab pertanyaan rumit dengan cepat hanya bermodal smartphone, membuat guru dan teman-teman terkagum. Tapi sebagian merasa itu "terlalu instan."
Pertanyaan penting: Apakah menggunakan AI berarti kamu pintar atau sekadar mengandalkan mesin? Bagaimana cara membedakan belajar aktif dan hanya copy-paste jawaban?

Kasus 4: “Kantin VS Kesehatan”

Sekolahmu baru saja membangun kantin baru. Isinya? Makanan cepat saji, minuman manis, dan jajanan viral yang digemari semua siswa.
Ilustrasi: Temanmu mengeluh sering sakit perut dan sulit konsentrasi, tapi tetap makan nugget keju setiap hari karena murah dan enak.
Isu utama: Apakah sekolah bertanggung jawab atas pilihan makanan siswa? Siapa yang harus mengatur: sekolah, orang tua, atau siswa itu sendiri?

Kasus 5: “Proyek Amal atau Ajang Pamer?”

Kelasmu mengadakan kegiatan sosial—membantu panti asuhan. Namun, semua momen dibagikan di media sosial, lengkap dengan filter dan musik dramatis.
Ilustrasi: Kamu melihat seorang teman marah karena wajahnya tidak terlihat di video saat membagikan sembako. Ia merasa "usahanya nggak kelihatan."
Pertanyaan menggelitik: Apakah aksi sosial tetap bermakna kalau niatnya untuk eksis di media? Bagaimana kita bisa tahu mana yang tulus dan mana yang cari perhatian?

Kasus 6: “Gadget, Gaya Hidup, dan Tekanan Sosial”

Di lingkunganmu, siswa berlomba-lomba memiliki gawai terbaru. Yang tidak punya dianggap "ketinggalan zaman" dan sering disindir.
Ilustrasi: Temanmu yang biasanya ceria jadi murung karena hp-nya rusak dan orang tuanya belum bisa membelikan baru.
Pertanyaan menantang: Apa arti "gengsi digital" di zaman sekarang? Apakah kamu bisa tetap eksis tanpa mengikuti standar mahal dari media sosial?

Contoh Soal dan Pembahasan

Soal:

Situasi: Dalam sebuah proyek AI sekolah, salah satu anggota tim mengeluh karena merasa tugasnya terlalu berat dibanding anggota lainnya.
Pertanyaan: Sebagai Project Manager, apa langkah bijak yang harus kamu ambil agar konflik tidak makin besar dan pembagian kerja terasa adil?

Pembahasan:
Langkah terbaik adalah **mengadakan diskusi terbuka**. Ajak semua anggota untuk menjelaskan tugas yang mereka kerjakan. Dengan data ini, kamu bisa mengevaluasi dan **redistribusi beban kerja** secara lebih seimbang. Pastikan semua merasa **didengar dan dihargai**. Proyek yang sukses berasal dari tim yang saling mendukung!

Soal:

Pertanyaan:
Mengapa seorang **programmer** dalam tim AI perlu memiliki sedikit pengetahuan tentang **data**, meskipun bukan tugas utamanya?

Pembahasan:
Karena dalam AI, **data adalah bahan bakar utama**. Programmer perlu tahu bagaimana data harus disiapkan—misalnya, format, kebersihan data, atau nilai-nilai yang hilang. Tanpa pemahaman dasar ini, program bisa error atau hasil AI menjadi tidak akurat. Contohnya, jika data kosong atau salah format, model AI tidak bisa berfungsi dengan baik.

Soal:

Pertanyaan:
Bayangkan komunikasi dalam tim buruk. Semua bekerja sendiri-sendiri. Menurutmu, **apa resiko terbesar** dari kondisi ini?

Pembahasan:
Risikonya antara lain:

- Tugas bisa **tumpang tindih** atau malah ada yang terlupakan.
- Bisa muncul **konflik antaranggota**.
- **Proyek bisa terlambat** atau gagal diselesaikan.

Komunikasi yang baik itu seperti koneksi Wifi: kalau lemah, semua jadi lambat dan membingungkan!

Soal:

Pertanyaan:
Sebutkan **dua tools gratis** yang bisa kamu gunakan untuk kolaborasi dokumen secara **real-time** dalam proyek kelompok!

Pembahasan:
Jawaban yang paling umum dan efektif adalah:

- **Google Docs** untuk menulis bersama.
- **Google Sheets** untuk data dan tabel.
- **Google Slides** untuk presentasi bareng.

Tools ini memungkinkan semua anggota mengedit dokumen secara bersamaan, jadi semua bisa terlibat aktif, kapanpun dan dimanapun.

Soal:

Pertanyaan:
Bagaimana caramu memastikan bahwa setiap anggota tim merasa **dihargai dan punya peran penting**, meskipun tanggung jawabnya berbeda?

Pembahasan:
Mulailah dengan **mengakui kontribusi tiap anggota**, sekecil apa pun. Selalu ajak mereka berdiskusi dan beri ruang untuk berpendapat. Saat membagi tugas, pastikan peran disesuaikan dengan minat dan kekuatan masing-masing. Dengan begitu, mereka akan merasa **terlibat dan bangga** atas hasil tim.

Fakta Menarik / Fun Facts

- Banyak inovasi besar, seperti internet atau sistem operasi Linux, berawal dari kolaborasi terbuka ribuan orang di seluruh dunia!
- Tahukah kamu? **Sistem operasi Linux** diciptakan lewat kolaborasi ribuan orang yang bahkan tidak saling kenal!
- Rata-rata tim proyek AI yang sukses terdiri dari orang-orang dengan keahlian berbeda yang saling melengkapi, bukan hanya kumpulan programmer hebat.
- Banyak **startup sukses dunia** seperti Instagram atau WhatsApp berawal dari tim kecil yang solid dan saling percaya.
- Di Google, setiap proyek AI besar melibatkan **tim lintas disiplin**: ada psikolog, seniman, insinyur, bahkan ahli etika!

Tips Belajar atau Tips Cepat Menghafal

- **Prinsip dasar kerja tim:**
    1. **Komunikasi Terbuka:** Jangan takut bertanya atau memberikan umpan balik.
    2. **Saling Percaya:** Percayakan tugas pada rekan tim Anda.
    3. **Tanggung Jawab Bersama:** Kegagalan atau keberhasilan adalah milik tim.
    4. **Fleksibel:** Siap membantu rekan tim jika diperlukan.
- "Ingat pepatah, 'If you want to go fast, go alone. If you want to go far, go together.'"

Mind Mapping Kerja Tim



Cerita Mudah Diingat

Bayangkan sebuah **band musik**: gitaris tidak akan bisa membuat lagu enak sendiri. Tapi saat digabung dengan vokalis dan drummer — *JENG! Lagu jadi sempurna.* Begitu pula tim AI!

Tips Kilat

- Gunakan **warna berbeda** untuk mencatat.
- Belajar sambil **mengajar teman**, supaya kamu lebih paham.
- Buat **singkatan lucu** atau **lagu pendek** dari poin-poin penting.

Aktivitas Siswa

Aktivitas 1: Simulasi Pembagian Peran Tim Proyek

Tujuan: Mengenalkan pentingnya peran yang seimbang dan saling melengkapi dalam tim.
Langkah-langkah:

- Guru membagi siswa ke dalam kelompok kecil (3–4 orang per tim).
- Berikan gambaran proyek akhir (misalnya: membuat aplikasi klasifikasi makanan sehat vs *junk food*).
- Ajak siswa mendiskusikan peran yang dibutuhkan: **Programmer, Data Analyst, Editor, Project Manager**.
- Siswa menyepakati peran masing-masing berdasarkan minat dan kekuatan pribadi.

- Dorong siswa membuat kontrak kerja tim mini yang berisi tanggung jawab dan kesepakatan kolaborasi.

Aktivitas 2: Presentasi Strategi Kolaborasi Tim

Tujuan: Melatih perencanaan tim dan komunikasi terbuka.
Langkah-langkah:

- Setelah menentukan peran, tiap tim menyusun strategi kolaborasi mereka.
- Bahas: bagaimana mereka akan berkomunikasi (WhatsApp, Trello, dll.), bagaimana menyelesaikan konflik, dan bagaimana progres dipantau.
- Setiap tim mempresentasikan rencana mereka dalam waktu 3–5 menit di depan kelas.
- Berikan umpan balik positif dan saran untuk peningkatan antar tim.

Aktivitas 3: Role-play: Komunikasi dalam Konflik

Tujuan: Melatih soft skill dalam menghadapi masalah di dalam tim.
Langkah-langkah:

- Guru memberikan skenario: “Salah satu anggota tidak menyelesaikan tugas penting.”
- Tiap kelompok membuat dua versi dialog:
    - Versi 1: *Konflik terbuka tanpa solusi*
    - Versi 2: *Kolaboratif dan solutif*
- Siswa memainkan peran mereka di depan kelas. Setelah itu, diskusikan versi mana yang lebih efektif dan kenapa.
- Tekankan pentingnya **empati, mendengarkan aktif, dan kejelasan komunikasi**.

Aktivitas 4: Game Kolaborasi Puzzle Informasi

Tujuan: Mengembangkan kepercayaan dan kerja sama dalam menyelesaikan masalah.
Langkah-langkah:

- Guru menyiapkan soal atau teka-teki yang informasinya dipecah menjadi beberapa bagian.
- Setiap anggota hanya mendapat satu potongan informasi.
- Agar berhasil, mereka **harus saling bertanya dan berbagi informasi** tanpa langsung memperlihatkan catatan mereka.
- Setelah selesai, diskusikan bagaimana strategi komunikasi tim membantu (atau menghambat) penyelesaian.

Aktivitas 5: Challenge: 2 Jam Coding Sprint

Tujuan: Mengasah kemampuan teknis dan koordinasi tim secara cepat.
Langkah-langkah:

- Berikan tantangan membuat prototipe aplikasi menggunakan **Teachable Machine** dan **MIT App Inventor**.
- Proyek: Aplikasi yang bisa mengenali gambar wajah dan menampilkan hasil klasifikasi.
- Berikan pseudocode awal sebagai panduan.
  Contoh:

  ```
  Mulai
  Ambil input gambar wajah
  Kirim ke model AI
  Tampilkan hasil klasifikasi di layar
  Selesai
  ```

- Waktu terbatas: hanya **2 jam** untuk diskusi, eksekusi, dan demo.
- Setelah itu, tiap tim menampilkan hasil dan menceritakan pengalaman kerjanya.

Aktivitas 6: Kolaborasi via Google Docs: Rencana Mini-Proyek

Tujuan: Melatih dokumentasi bersama dan penggunaan *tools* digital.
Langkah-langkah:

- Setiap tim membuat dokumen rencana proyek di Google Docs.
- Harus mencakup: Tujuan proyek, pembagian peran, alur kerja, dan rencana waktu.
- Pastikan setiap anggota ikut menulis dan mengedit dokumen—jejak kontribusi bisa dilihat di *version history*.
- Guru memantau dan memberi masukan langsung melalui komentar di dokumen.

Rangkuman

Bayangkan kamu sedang membangun robot AI bersama teman-temanmu—misi yang *serius*, tapi juga super seru! Nah, untuk sukses menyelesaikan proyek seperti ini, **kolaborasi tim adalah kunci utama**. Kerja bareng bukan cuma soal bagi-bagi tugas, tapi juga soal saling percaya, terbuka saat berdiskusi, dan menyatukan ide-ide keren dari berbagai keahlian. Dalam tim yang solid, kamu bisa mencapai hal-hal luar biasa yang mustahil dilakukan sendirian.

Masing-masing anggota punya peran spesial: **programmer** bertugas membuat sistem dan menulis *code*, **data analyst** menyiapkan dan mengolah data biar bisa diproses AI-nya, sedangkan **editor** menyusun laporan dan dokumentasi agar semuanya rapi dan mudah dipahami. Kalau semua peran dijalankan dengan tanggung jawab dan komunikasi berjalan lancar, kerja jadi lebih cepat, hasilnya pun jauh lebih keren dan profesional.

Untuk menunjang kerja bareng ini, tim biasanya pakai *tools* seperti **Google Docs**, **WhatsApp**, atau **Trello**—semua membantu berbagi info dan ngecek progres secara

real-time. Lewat pengalaman ini, kamu bukan cuma belajar hal teknis soal AI, tapi juga mengasah **soft skill penting** kayak empati, tanggung jawab, dan cara menyelesaikan masalah bareng-bareng. Jadi, proyek AI ini bukan sekadar soal teknologi, tapi juga tantangan seru untuk tumbuh jadi tim yang hebat!

Latihan Soal

A. Benar atau Salah (5 Soal)

Tuliskan "Benar" atau "Salah" untuk setiap pernyataan berikut:

1. Proyek AI sederhana bisa dikerjakan sendirian tanpa perlu kolaborasi tim.
   Jawaban: Salah

2. Seorang Data Analyst hanya bekerja setelah model AI selesai dibuat.
   Jawaban: Salah

3. Komunikasi terbuka membantu menghindari konflik dalam tim.
   Jawaban: Benar

4. Editor hanya bertugas memperbaiki tulisan, tidak perlu memahami proyek AI.
   Jawaban: Salah

5. Tools seperti Google Docs membantu tim bekerja secara bersamaan dalam satu dokumen.
   Jawaban: Benar

B. Pilihan Ganda (10 Soal)

Pilih jawaban yang paling tepat:

1. Apa peran utama seorang programmer dalam proyek AI?
   A. Membuat presentasi proyek
   B. Menyusun dan menjalankan kode
   C. Menganalisis data dan menyimpulkan hasil
   D. Menyusun laporan akhir
   Jawaban: B

2. Berikut ini yang **bukan** alasan pentingnya kolaborasi dalam proyek AI adalah...
   A. Mempercepat proses
   B. Mempermudah kerja sendiri
   C. Meningkatkan kualitas hasil
   D. Mendapatkan banyak perspektif
   Jawaban: B

3. Siapakah yang bertanggung jawab membersihkan data?
   A. Programmer
   B. Editor
   C. Data Analyst
   D. Project Manager
   Jawaban: C

4. Contoh tools yang digunakan untuk berbagi dokumen dan kolaborasi real-time adalah...
   A. WhatsApp
   B. Excel offline
   C. Google Docs
   D. Paint
   Jawaban: C

5. Dalam tim proyek AI, siapa yang bertugas membuat dokumentasi dan pitch deck?
   A. Developer
   B. Data Analyst
   C. Editor
   D. UI Designer
   Jawaban: C

6. Mengapa penting mendengarkan aktif saat berdiskusi dalam tim?
   A. Supaya diskusi cepat selesai
   B. Agar semua pendapat didengar dan dipahami
   C. Supaya bisa menang debat
   D. Karena itu sopan santun
   Jawaban: B

7. Manakah yang merupakan contoh komunikasi tidak efektif dalam tim?
   A. Memberikan saran perbaikan secara sopan
   B. Mendengarkan rekan sebelum menyela
   C. Mengabaikan ide orang lain karena berbeda
   D. Bertanya jika ada yang tidak paham
   Jawaban: C

8. GitHub digunakan terutama untuk...
   A. Menyimpan video presentasi
   B. Mengedit gambar
   C. Kolaborasi kode dan kontrol versi
   D. Menganalisis data
   Jawaban: C

9. Siapa yang biasanya merancang tampilan antarmuka aplikasi?
   A. Data Analyst
   B. Programmer
   C. UI/UX Designer
   D. Editor

Jawaban: C

10. Apa fungsi utama dari Trello atau Asana dalam proyek AI?
    A. Mengedit kode
    B. Memainkan musik
    C. Mengatur dan melacak progres tugas
    D. Menyimpan file video
    Jawaban: C

C. Isian Singkat (5 Soal)

1. Tugas utama seorang Data Analyst adalah mengumpulkan, _____, dan menyiapkan data.
   Jawaban: membersihkan

2. Salah satu skill penting untuk Editor adalah kemampuan _____.
   Jawaban: menulis

3. Tools seperti Google Slides digunakan untuk membuat _____ presentasi.
   Jawaban: visual

4. Dalam proyek AI, komunikasi yang baik bisa mencegah _____ antar anggota tim.
   Jawaban: konflik

5. Jika anggota tim tidak bisa menyelesaikan tugas tepat waktu, maka perlu dilakukan _____ tugas.
   Jawaban: redistribusi

D. Soal Eksplorasi

Jawablah secara naratif pendek (3-5 kalimat per soal):

Soal:

Bayangkan kamu adalah seorang Programmer dalam proyek AI sekolah. Temanmu (sebagai Data Analyst) belum selesai membersihkan datanya padahal kamu sudah menunggu. Apa yang kamu lakukan agar proyek tetap berjalan tanpa membuat konflik?

Jawaban: Saya akan menghubungi teman saya secara sopan dan menanyakan apakah dia membutuhkan bantuan. Jika memungkinkan, saya bisa membantu membersihkan sebagian data agar proyek tetap berjalan. Tujuannya adalah mendukung, bukan menyalahkan.

Soal:

Tim kamu membuat aplikasi AI untuk mengenali ekspresi wajah. Salah satu anggota tidak menyelesaikan tugas dokumentasi dengan baik. Apa langkah yang sebaiknya kamu dan tim lakukan untuk memperbaiki situasi tersebut?

Jawaban: Tim bisa mengadakan pertemuan singkat untuk membahas masalah tersebut secara terbuka. Kami akan mendengarkan alasan anggota tersebut, lalu membagi ulang tugas dokumentasi agar bisa selesai tepat waktu. Semua dilakukan dengan sikap saling menghargai.

Soal:

Jika kamu diberikan peran sebagai Editor dalam proyek, tetapi kamu lebih tertarik pada pemrograman, bagaimana kamu menyikapinya agar tetap berkontribusi positif?

Jawaban: Saya akan tetap menjalankan peran Editor dengan serius karena itu tanggung jawab saya. Namun, saya bisa berdiskusi dengan tim apakah ada ruang kontribusi kecil di bagian pemrograman. Yang terpenting, saya tetap komitmen pada peran awal dan mendukung tim secara keseluruhan.

Tugas Proyek

Proyek 1: Bentuk Tim AI Mini Kalian

Deskripsi:
Bentuk kelompok beranggotakan tiga orang. Diskusikan dan tentukan peran setiap anggota, seperti *Programmer*, *Data Analyst*, dan *Editor Proyek*. Jelaskan alasan pemilihan peran berdasarkan minat dan kemampuan masing-masing.

Langkah-langkah:

- Lakukan sesi perkenalan kekuatan dan minat tiap anggota.
- Tentukan peran yang paling cocok bagi masing-masing.
- Buat catatan singkat (1 paragraf per orang) yang menjelaskan alasan pembagian tugas.

Proyek 2: Rancang dan Susun Rencana Proyek AI Mini

Deskripsi:
Kembangkan dokumen rencana proyek bersama menggunakan Google Docs. Dokumen ini mencakup tujuan proyek, pembagian peran, alat yang digunakan, jadwal kerja, dan strategi komunikasi tim.

Langkah-langkah:

- Tentukan tujuan proyek (misalnya klasifikasi suara burung, deteksi emosi dari teks, dsb).
- Susun jadwal mingguan serta deadline mini.
- Pilih tools yang akan digunakan (misalnya Teachable Machine, Google Colab, Canva).
- Buat sistem komunikasi tim (grup WA, Discord, atau lainnya).

Proyek 3: Simulasi Konflik dan Penyelesaiannya

Deskripsi:
Lakukan simulasi bagaimana tim menghadapi dan menyelesaikan konflik ringan, seperti ketidaksepakatan dalam ide atau bentrok jadwal kerja. Bisa dibuat dalam bentuk naskah percakapan atau video pendek.

Langkah-langkah:

- Tentukan skenario konflik (misalnya: dua anggota ingin membuat proyek berbeda).
- Tulis dialog penyelesaiannya secara dewasa dan kreatif.
- Jika memilih video, cukup rekam percakapan sederhana berdurasi 2–3 menit.

Proyek 4: Catat Log Aktivitas Proyek

Deskripsi:
Gunakan Google Sheets sebagai log tim untuk mendokumentasikan aktivitas mingguan tiap anggota, tantangan yang dihadapi, serta progres yang dicapai.

Langkah-langkah:

- Buat tabel dengan kolom: Tanggal, Nama Anggota, Tugas Mingguan, Progres, Catatan Tambahan.
- Setiap minggu, isi log secara rutin.
- Gunakan warna atau simbol untuk menandai progres (✔️ untuk selesai, ⏳ untuk dalam proses, ❌ untuk belum mulai).

Proyek 5: Presentasi Final – Tunjukkan Kolaborasi Kalian!

Deskripsi:
Persiapkan presentasi proyek akhir menggunakan Google Slides. Sajikan hasil akhir proyek AI mini kalian, serta ceritakan bagaimana tim bekerja sama menghadapi tantangan.

Langkah-langkah:

- Buat slide yang mencakup: tujuan proyek, cara kerja, hasil akhir, tantangan & solusi.
- Latih presentasi bersama. Bisa dibawakan langsung di kelas atau direkam sebagai video.
- Tampilkan log aktivitas sebagai bukti kerja nyata.

Proyek 6: Teka-Teki Tim – Pecahkan Masalah Kolaborasi!

Deskripsi:
Guru memberikan skenario fiktif berupa masalah dalam tim (misalnya, satu anggota tidak aktif, ide terlalu banyak, atau peran tidak seimbang). Tugas kalian: analisis masalah dan buat strategi penyelesaiannya.

Langkah-langkah:

- Diskusikan skenario dalam kelompok.
- Identifikasi akar masalah dan akibatnya bagi tim.
- Tuliskan solusi dalam bentuk strategi komunikasi atau perubahan pembagian tugas.

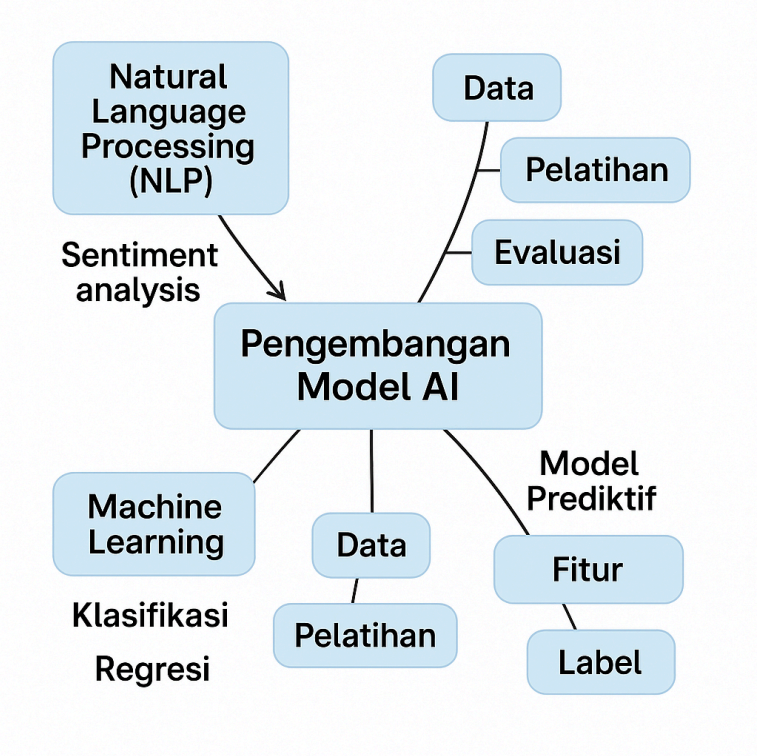
BAB 4: Pengembangan Model AI (NLP, ML Sederhana)

Tujuan Pembelajaran

Setelah mempelajari bab ini, kamu diharapkan dapat:

- Mengembangkan model kecerdasan buatan (AI) sederhana menggunakan Machine Learning (ML) atau Natural Language Processing (NLP).
- Memahami tahapan dalam membangun model AI dari awal hingga bisa digunakan.
- Melakukan eksperimen seru seperti membuat model prediksi atau klasifikasi sendiri.

Peta Konsep



Apersepsi

Pernahkah kamu menggunakan Google Translate atau berbicara dengan asisten suara seperti Siri atau Google Assistant? Tanpa kita sadari, teknologi canggih itu bisa bekerja karena adanya kecerdasan buatan (AI) yang mampu memahami bahasa manusia dan memberikan respons yang sesuai. Nah, coba bayangkan kalau kamu bisa membuat model AI sederhana sendiri! Misalnya, model yang bisa menebak apakah sebuah teks berisi perasaan positif atau negatif, mengenali apakah sebuah gambar menunjukkan kucing atau anjing, bahkan memprediksi nilai ujian hanya dari jumlah jam kamu belajar. Di bab ini, kamu

akan belajar membangun model AI dengan langkah-langkah yang mudah dipahami, seru untuk dicoba, dan menantang kreativitasmu. Yuk, bersiap jadi kreator teknologi masa depan—petualanganmu di dunia AI dimulai dari sini!

Penjelasan Konsep (teori)

Pernahkah kamu bertanya-tanya bagaimana **Google Translate** bisa menerjemahkan bahasa dalam hitungan detik? Atau bagaimana **asisten suara** seperti Siri atau Google Assistant bisa memahami perintah kita? Jawabannya: di balik semua itu ada **model kecerdasan buatan (Artificial Intelligence/AI)** yang sangat canggih.

Nah, di bab ini, kita akan belajar **membangun versi sederhananya**! Tapi sebelum langsung praktik, yuk kita pahami dulu **teori dan tahapan** di balik pengembangan model AI. Jangan khawatir—akan dijelaskan dengan cara yang ringan, jelas, dan menyenangkan!

Apa itu Model AI?

Model AI adalah **program komputer yang dapat belajar dari data dan membuat prediksi atau keputusan sendiri** tanpa harus diprogram secara eksplisit untuk setiap tugas. Dua cabang utama yang akan kita bahas adalah:

- **Machine Learning (ML)**: AI yang belajar dari data untuk memprediksi atau mengklasifikasi sesuatu.
- **Natural Language Processing (NLP)**: AI yang bisa memahami dan mengolah bahasa manusia, seperti membaca teks, menganalisis emosi dalam kalimat, atau menjawab pertanyaan.

Tahapan Pengembangan Model AI

Untuk membangun model AI, kita perlu melalui beberapa tahap, mirip seperti membuat kue: mulai dari memilih bahan (data), mengolahnya, sampai menyajikan hasilnya (model siap pakai). Berikut adalah tahapan umumnya:

Pengumpulan Data

Data adalah bahan bakar utama AI. Tanpa data, model tidak bisa belajar apa-apa.

- **Sumber data** bisa berupa teks (kata-kata), gambar, suara, atau angka.
- Pentingnya **kuantitas, kualitas, dan keragaman**: Semakin banyak dan beragam data, semakin cerdas model kita.
- Cara mendapatkan data:
    - Mencari di internet (dataset publik).
    - Membuat sendiri (misalnya, memotret buah untuk model klasifikasi buah).

Preprocessing Data (Pembersihan & Persiapan Data)

Data mentah biasanya berantakan. Maka dari itu, perlu dipersiapkan sebelum digunakan.

- **Pembersihan**: Menghapus data yang rusak, duplikat, atau kosong.
- **Normalisasi/Standarisasi**: Menyamakan skala nilai agar adil untuk semua fitur.
- **Transformasi data**:
    - Untuk teks: mengubah kata menjadi angka (agar model bisa memahaminya).
    - Contoh: Kata “senang” diubah jadi angka tertentu untuk analisis sentimen.

Pemilihan Model

Setiap masalah punya model yang cocok. Kita perlu memilih model berdasarkan **jenis tugas dan kompleksitas**.

- **Model klasifikasi**: Untuk membedakan kategori (contoh: spam atau bukan).
- **Model regresi**: Untuk memprediksi angka (contoh: harga rumah).
- Sesuaikan dengan **tujuan dan kemampuan komputer** yang digunakan.

Pelatihan Model (Training)

Ini adalah fase di mana model “belajar” dari data.

- **Fitur (features)**: Bagian dari data yang digunakan sebagai input (misalnya: usia, tinggi badan).
- **Label**: Jawaban yang ingin diprediksi (misalnya: sehat/sakit).
- Data dibagi menjadi dua bagian:
    - **Training set**: Untuk melatih model.
    - **Testing set**: Untuk menguji model apakah bisa “menebak” dengan benar.

Evaluasi Model

Setelah model dilatih, kita harus mengukur seberapa baik performanya.

- Gunakan **metrik evaluasi**, misalnya:
    - **Akurasi**: Seberapa sering model benar.
    - **Presisi & Recall**: Seberapa tepat dan lengkap hasil model.
    - **F1-score**: Gabungan antara presisi dan recall.
- Tujuannya: Mengetahui apakah model sudah cukup baik atau masih perlu diperbaiki.

Penyebaran Model (Deployment)

Jika model sudah bagus, langkah terakhir adalah **mengintegrasikannya ke dunia nyata**.

- Contoh: Model klasifikasi buah digunakan di aplikasi smartphone.
- Format model bisa disimpan dalam berbagai bentuk: seperti TensorFlow Lite (untuk ponsel) atau JavaScript (untuk website).

Contoh Sederhana Natural Language Processing (NLP)

Sentiment Analysis (Analisis Sentimen)

- Tujuan: Menentukan apakah sebuah teks bersifat **positif, negatif, atau netral**.
- Contoh: “Makanannya enak banget!” → Positif
- Teknik dasar:
    - **Tokenization**: Memecah teks menjadi kata-kata.
    - **Stop Words Removal**: Menghapus kata umum seperti “yang”, “dan”, “di”.
    - **Stemming/Lemmatization**: Mengubah kata ke bentuk dasarnya, seperti “makan”, “memakan”, “dimakan” → “makan”.

Chatbot Sederhana

- Merespons pertanyaan dengan mencocokkan pola kata.
- Misal: Jika input mengandung kata “cuaca”, maka jawab “Hari ini cerah!”

Contoh Sederhana Machine Learning (ML)

Klasifikasi

- Tujuan: Mengelompokkan data ke dalam **kelas tertentu**.
- Contoh: Mengenali apakah gambar itu **kucing atau anjing**.
- Tools: **Google Teachable Machine** (mudah digunakan, berbasis web).
- Aktivitas: Unggah gambar, beri label, dan latih model.

Regresi

- Tujuan: Memprediksi nilai **angka kontinu**.
- Contoh: Memprediksi **nilai ujian** berdasarkan **jumlah jam belajar**.
- Tools: Bisa menggunakan **Orange** atau spreadsheet sederhana.

Eksperimen: Google Teachable Machine

Langkah-langkah membangun model AI secara sederhana:

1. **Pilih jenis proyek**: Gambar, suara, atau pose tubuh.
2. **Tambahkan kelas dan unggah sampel**: Usahakan bervariasi agar model lebih akurat.
3. **Latih model**: Klik “Train Model” dan tunggu prosesnya.
4. **Evaluasi & Ekspor model**: Bisa diekspor dalam berbagai format.

Dengan memahami konsep-konsep ini, kamu sudah selangkah lebih dekat menjadi *AI creator!* Seru, kan? Yuk, lanjut ke eksperimen berikutnya!

Ilustrasi

Bayangkan kamu sedang mengirim email ke guru, dan secara otomatis masuk ke folder spam. Kenapa bisa begitu? Karena sistem email menggunakan **model AI** yang dilatih untuk mengenali email “mencurigakan.” Tapi tidak selalu sempurna! Kadang email penting pun bisa tersaring.

Dulu, seorang siswa bernama Rani penasaran kenapa email pemberitahuan lombanya tidak masuk. Ia lalu belajar tentang bagaimana AI bekerja. Ia pun mencoba membangun model AI sederhana dengan bantuan Google Teachable Machine. Awalnya, hasilnya buruk – modelnya sering salah mengenali gambar. Tapi setelah mencoba mengumpulkan gambar dari berbagai sudut dan belajar tentang **data preprocessing**, hasilnya membaik! Kini, Rani tahu: **AI bukan sulap, tapi butuh ketelitian dan eksperimen.**

Contoh Kasus atau Ilustrasi

Kasus 1: Menyulap Review Film Jadi AI yang Bisa “Menilai”

Kasus: Klasifikasi Teks Ulasan Film
Ilustrasi:
Bayangkan kamu adalah pengelola sebuah situs ulasan film. Setiap hari, ratusan review masuk—ada yang berkomentar “Film ini keren banget!”, ada juga yang “Bikin ngantuk total!”.
Tugas kamu adalah melatih sistem AI menggunakan *Orange* agar bisa secara otomatis mengklasifikasikan review-review tersebut ke dalam dua kategori: **positif** dan **negatif**.

Bagaimana caranya?

- Kumpulkan data review film yang sudah diberi label (positif/negatif).
- Masukkan ke Orange dan olah menggunakan widget seperti *Bag of Words* dan *Naive Bayes*.
- Lihat performa klasifikasinya secara visual—apakah akurat?
- Uji AI kamu dengan review baru dan lihat apakah ia bisa “menilai” dengan benar!

Kasus 2: Detektif Masker: AI yang Bisa Membedakan Cara Pakai Masker

Kasus: Klasifikasi Gambar dengan Google Teachable Machine
Ilustrasi:
Kamu jadi relawan digital dalam kampanye kesehatan. Tantangannya? Melatih model AI yang bisa membedakan tiga hal:

- Orang yang memakai masker **dengan benar**
- Orang yang **salah memakai** masker
- Orang yang **tidak memakai masker**

Langkahnya seru:

- Kumpulkan foto dari ketiga kategori.
- Latih model di Google Teachable Machine hanya dengan drag-and-drop!
- Uji dengan foto baru: apakah AI bisa menebak dengan tepat?
- Kalau sudah jago, ekspor modelnya dan gunakan di aplikasi buatanmu sendiri seperti App Inventor!

Kasus 3: Robot Rumah: AI Pengenal Barang Sehari-hari

Kasus: Klasifikasi Gambar Barang Rumah Tangga
Ilustrasi:
Bayangkan kamu membuat AI asisten rumah pintar. Tapi sebelum bisa bantu kamu, dia harus tahu bedanya **pensil**, **penghapus**, dan **buku**.
Di sini, kamu melatih model AI agar bisa mengenali tiga benda tersebut dari gambar.

Prosesnya seperti melatih anak kecil mengenal benda:

- Ambil banyak foto dari setiap objek.
- Masukkan ke platform seperti Teachable Machine atau Pictoblox.
- Latih model dan uji dengan benda sungguhan.
- Apakah AI bisa menebak dengan benar saat kamu tunjukkan pensil dari sisi berbeda?

Kasus 4: Mesin Pencicip Rasa: Analisis Ulasan Makanan

Kasus: Analisis Sentimen Ulasan Makanan
Ilustrasi:
Kamu seperti “robot pencicip” yang membaca komentar orang soal makanan.
Kalimat seperti “nasi gorengnya mantap!” atau “ga doyan rasanya, maaf ya” harus bisa diklasifikasikan: ini ulasan **positif** atau **negatif**?

Gunakan Orange atau Excel untuk:

- Memasukkan data ulasan.
- Lakukan klasifikasi teks dengan Naive Bayes.
- Tampilkan grafik yang menunjukkan mana makanan favorit netizen!

Kasus 5: Ramalan Nilai Ulangan Berdasarkan Jam Belajar

Kasus: Regresi Sederhana untuk Prediksi Nilai
Ilustrasi:
Kamu sedang penasaran: “Kalau aku belajar 2 jam sehari, nilainya kira-kira berapa ya?”
Gunakan data nyata dari teman-temanmu untuk membuat model prediksi.

Gunakan Excel/Google Sheets untuk:

- Masukkan data: jam belajar dan nilai ulangan.
- Buat grafik dan jalur regresi.

- Prediksi nilai berdasarkan pola: apakah lebih banyak belajar memang bikin nilai naik?

Kasus 6: Chatbot Jadwal Pelajaran: Asisten Pribadi di Sekolah

Kasus: Chatbot Jadwal Sekolah
Ilustrasi:
Bayangkan kamu punya chatbot di HP yang bisa jawab:
“Kamu tanya: Hari ini pelajaran apa?”
Dan dia langsung jawab: “Senin: Matematika, Biologi, Bahasa Inggris”.

Kamu bisa membuatnya menggunakan platform seperti Dialogflow, Scratch, atau App Inventor.

- Simpan data jadwal per hari.
- Latih chatbot agar bisa merespons sesuai hari.
- Tambahkan fitur: chatbot juga bisa kasih pengingat tugas atau ulangan!

Contoh Soal dan Pembahasan

Soal 1: Tantangan Data Kucing vs Anjing

Bayangkan kamu sedang membuat model AI yang bisa membedakan gambar kucing dan anjing. Wah, seru! Nah, sebutkan dua hal penting yang perlu kamu perhatikan saat mengumpulkan gambar agar AI-mu tidak "asal tebak**".**

Clue: Apa yang bisa bikin model AI pintar membedakan dua makhluk menggemaskan ini?

Pembahasan:

1. **Kualitas dan Variasi** – Gambar harus jelas dan diambil dari berbagai sudut, latar belakang, dan kondisi cahaya.
2. **Keseimbangan Data** – Jumlah gambar kucing dan anjing harus seimbang agar model tidak “berpihak” ke salah satu.

Soal 2: Awas, AI-mu Bisa "Ngafal"!

Model AI-mu punya akurasi 95% saat dilatih, tapi pas dites sama data baru cuma 60%. Waduh, kenapa bisa begitu ya? Coba jelaskan apa yang terjadi dan bagaimana cara mengatasinya.

Clue: Pernah ngafal soal ujian tapi malah bingung pas soal baru? Sama kayak AI!

Pembahasan:

- **Masalahnya adalah overfitting** – model terlalu hafal data latih, sehingga gagal memahami pola umum.
- **Solusinya**: sederhanakan model, tambah data latih yang beragam, atau gunakan regularisasi (cara untuk mencegah model terlalu rumit).

Soal 3: Klasifikasi vs Regresi – Tebak Gaya AI

Kapan AI digunakan untuk klasifikasi, dan kapan untuk regresi? Berikan satu contoh untuk masing-masing!

Clue: Satu soal tebak-tebakan, satu soal nilai!

Pembahasan:

- **Klasifikasi**: Memilah ke dalam kategori, contoh: apakah email itu spam atau tidak.
- **Regresi**: Memprediksi nilai kontinu, contoh: memprediksi nilai ujian dari jam belajar.

Soal 4: Aduh, Data Berat Sebelah!

Kenapa penting banget punya data yang seimbang dalam melatih AI? Misalnya, kalau kamu punya 90% gambar kucing dan 10% gambar anjing, apa yang bisa terjadi?

Clue: Bayangkan kalau kamu hanya dilatih menjawab soal kucing...

Pembahasan:

- Model bisa jadi **terlalu sering menebak "kucing"**, bahkan saat melihat anjing.
- **Data yang seimbang membantu model belajar adil** dan mengenali kedua kelas dengan sama baik.

Soal 5: Teks Berantakan? Yuk, Bersih-bersih Dulu!

Dalam Natural Language Processing (NLP), kenapa perlu membersihkan data teks sebelum diproses oleh AI? Apa yang biasanya dibersihkan?

Clue: Teks mentah itu seperti buku yang belum disunting – susah dipahami AI!

Pembahasan:

- Preprocessing membantu **menghilangkan simbol, angka tak penting, atau kata-kata umum** yang tidak membantu.
- Ini membuat teks **lebih rapi dan siap dipahami** oleh algoritma AI.

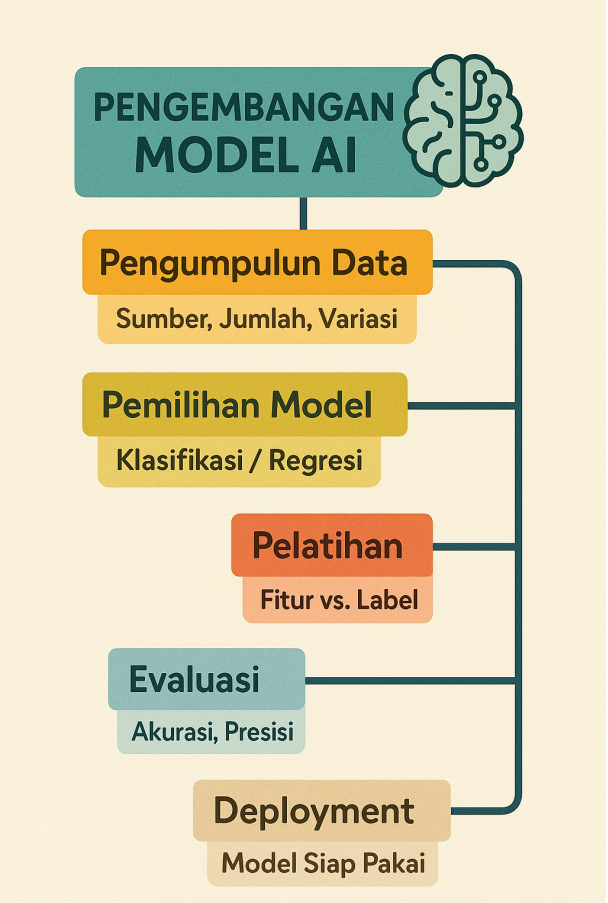
Fakta Menarik / Fun Facts

- Diperlukan jutaan bahkan milyaran data untuk melatih model AI yang canggih seperti yang digunakan di mobil otonom atau pengenal wajah!
- Model AI seperti ChatGPT ini dilatih dari **ratusan miliar kata!**
- Tanpa data yang bagus, AI bisa membuat prediksi lucu: seperti mengira zebra adalah kuda belang!
- AI bisa belajar dari **gambar, suara, teks, dan angka** – seolah-olah belajar dari semua indera manusia.

Tips Belajar atau Tips Cepat Menghafal

- **Langkah-langkah pengembangan model:** "Kumpulkan, Bersihkan, Latih, Uji, Sebarkan!" (C.B.L.U.S)
- "Fokus pada data! Model AI secanggih apapun tidak akan berfungsi baik jika datanya buruk (Garbage In, Garbage Out)."

Mind Mapping Pengembangan Model AI



Cerita Mudah Diingat

Bayangkan kamu ingin melatih robot untuk mengenali kucing dan anjing. Tapi kalau kamu hanya kasih foto kucing hitam dari depan, si robot tidak tahu kalau kucing bisa warna putih, atau dilihat dari samping! Maka kamu perlu data **beragam**, bersih, dan jelas. Dan robot perlu belajar berkali-kali sampai dia **bisa menebak dengan benar.**

Aktivitas Siswa

Aktivitas 1: Menjelajahi Dunia AI: Membangun Model Sendiri

Deskripsi:
Kembangkan model kecerdasan buatan (AI) sederhana sesuai dengan topik proyek akhir kalian, baik dalam bidang pengenalan gambar (ML) maupun bahasa alami (NLP). Gunakan platform seperti Google Teachable Machine, atau eksplorasi tools lain yang sesuai.

Langkah Seru:

- Tentukan tujuan model AI yang ingin dibuat (misalnya: mengenali ekspresi wajah atau memahami pertanyaan sederhana).
- Kumpulkan data yang relevan secara kreatif—bisa dari internet atau buat sendiri.
- Latih model menggunakan platform pilihan.
- Uji keakuratan dan performanya, lalu lakukan perbaikan jika diperlukan.

Aktivitas 2: Uji Coba AI: Bermain dengan Parameter Model

Deskripsi:
Eksperimen dengan parameter pelatihan model AI yang kalian buat. Perhatikan bagaimana hasilnya berubah saat kalian mengatur ulang jumlah data, epoch, atau ukuran batch.

Langkah Seru:

- Ubah satu parameter (misalnya jumlah data atau epoch) lalu latih ulang model.
- Catat hasil prediksi yang muncul dan diskusikan: apakah lebih akurat atau justru menurun?
- Tantang diri kalian: coba temukan kombinasi terbaik agar model lebih cerdas.

Aktivitas 3: Tantangan Teachable Machine: AI Pengenal Objek Rumah

Deskripsi:
Gunakan Google Teachable Machine untuk menciptakan model yang dapat membedakan antara 3 objek di rumah (misalnya: sendok, remote, dan botol).

Langkah Seru:

- Ambil beberapa foto dari masing-masing objek dalam berbagai posisi dan pencahayaan.
- Latih model dengan data tersebut, lalu uji seberapa akurat AI mengenali objeknya.
- Ajak temanmu ikut mencoba modelmu—apakah AI bisa mengenali objek yang mereka tunjukkan?

Aktivitas 4: Detektif Ulasan Makanan: Klasifikasi Teks dengan NLP

Deskripsi:
Kumpulkan beberapa ulasan makanan dari internet (misalnya dari Google Review atau Tokopedia), lalu analisis apakah ulasan tersebut bersifat positif, negatif, atau netral.

Langkah Seru:

- Ambil 10-15 ulasan pendek dari satu jenis makanan.
- Gunakan Python NLTK (atau manual) untuk mengklasifikasikan isi ulasan.
- Coba buat visualisasi hasil klasifikasi (diagram batang atau pie chart).
- Bandingkan hasil analisis otomatis vs manual—apakah AI cukup jago membaca emosi?

Aktivitas 5: Bangun Mini Chatbot: Asisten Sekolah di Genggamanmu

Deskripsi:
Ciptakan chatbot sederhana yang bisa menjawab pertanyaan umum seputar kegiatan sekolah, seperti ekstrakurikuler atau jadwal guru.

Langkah Seru:

- Gunakan dasar logika pemrograman (if-else atau keyword detection).
- Rancang beberapa pertanyaan umum dan jawaban yang sesuai.
- Uji dengan temanmu: apakah chatbot mampu merespons dengan benar?
- Tantangan ekstra: tambahkan lebih banyak pertanyaan atau gunakan tools chatbot visual seperti Dialogflow.

Aktivitas 6: Data Detektif: Prediksi Nilai Temanmu dengan Jam Belajar

Deskripsi:
Gunakan Google Sheets untuk membuat prediksi nilai berdasarkan jumlah jam belajar per minggu.

Langkah Seru:

- Kumpulkan data sederhana dari teman sekelas (jam belajar dan nilai ulangan).
- Masukkan ke Google Sheets, lalu buat grafik scatter plot.

- Tambahkan trendline (garis regresi) untuk memprediksi nilai.
- Diskusikan: apakah jam belajar benar-benar berpengaruh besar?

Ingin tantangan lebih? Gabungkan dua aktivitas menjadi satu proyek besar! Misalnya, chatbot yang bisa juga memprediksi nilai atau merekomendasikan jadwal belajar. Tantang kreativitas dan logika kalian!

Rangkuman

Pengembangan **model AI** melibatkan serangkaian tahapan krusial, mulai dari **pengumpulan data**, *preprocessing*, **pemilihan dan pelatihan model**, hingga **evaluasi dan *deployment***. Proses ini merupakan inti dari **kecerdasan buatan**, yang mengikuti alur sistematis dari awal hingga akhir pengembangan.

Dengan memiliki pemahaman dasar tentang ***Natural Language Processing (NLP)*** dan ***Machine Learning (ML)***, siswa dapat mulai mengembangkan **solusi cerdas** untuk berbagai permasalahan. Mereka bisa membuat **model AI sederhana** seperti **klasifikasi gambar** atau **prediksi nilai**, menggunakan alat seperti **Google Teachable Machine**, **Orange**, atau *library* Python dasar.

Melalui **latihan rutin** dan **proyek-proyek kecil**, siswa SMA bisa mulai memahami cara kerja **teknologi AI** di balik aplikasi sehari-hari seperti **Google Translate** atau sistem **deteksi spam email**. Untuk mengenalkan konsep *NLP*, siswa juga dapat mencoba membuat **chatbot sederhana** atau melakukan **analisis sentimen**.

Pada akhirnya, **pemahaman terhadap data** serta **siklus hidup model AI** sangatlah penting untuk menciptakan solusi yang **efektif** dan **akurat**.

Latihan Soal

A. Benar atau Salah (5 Soal)

1. Model AI hanya membutuhkan sedikit data untuk bisa bekerja dengan baik.
   Jawaban: Salah

2. Preprocessing data diperlukan sebelum melatih model AI.
   Jawaban: Benar

3. Regresi digunakan untuk memprediksi kategori seperti "spam" atau "bukan spam".
   Jawaban: Salah

4. Google Teachable Machine bisa digunakan untuk melatih model AI tanpa menulis kode.

Jawaban: Benar

5. Evaluasi model AI tidak penting jika akurasi data latih sudah tinggi.
   Jawaban: Salah

B. Pilihan Ganda (10 Soal)

6. Apa tujuan dari tahap preprocessing data?
   A. Mengubah data menjadi grafik
   B. Menambah kompleksitas model
   C. Membersihkan dan menyiapkan data untuk pelatihan
   D. Menyebarkan model ke internet
   Jawaban: C

7. Manakah yang merupakan contoh klasifikasi dalam AI?
   A. Mengukur suhu
   B. Menentukan apakah email termasuk spam atau tidak
   C. Menghitung luas bangun datar
   D. Mengukur tinggi badan
   Jawaban: B

8. Apa yang dimaksud dengan fitur dalam pelatihan model AI?
   A. Hasil akhir model
   B. Input atau data yang digunakan oleh model untuk belajar
   C. Bahasa pemrograman yang digunakan
   D. Tampilan aplikasi
   Jawaban: B

9. Apa yang terjadi jika model terlalu menghafal data latih?
   A. Underfitting
   B. Overfitting
   C. Regularisasi
   D. Normalisasi
   Jawaban: B

10. Tools berikut yang bisa digunakan untuk membuat model klasifikasi gambar adalah...
    A. Google Docs
    B. Google Slides
    C. Google Teachable Machine
    D. Google Sheets
    Jawaban: C

11. Model regresi cocok digunakan untuk...
    A. Menghitung luas bangun
    B. Mengkategorikan warna
    C. Memprediksi nilai ujian berdasarkan jam belajar
    D. Membuat logo aplikasi

Jawaban: C

12. Apa kegunaan data uji dalam pelatihan model?
    A. Untuk membuat aplikasi
    B. Untuk menguji performa model terhadap data baru
    C. Untuk menampilkan grafik
    D. Untuk mempersingkat proses pelatihan
    Jawaban: B

13. Proses tokenization digunakan dalam NLP untuk...
    A. Menghapus data duplikat
    B. Mengubah angka menjadi teks
    C. Memecah teks menjadi bagian-bagian kecil
    D. Mempercepat komputer
    Jawaban: C

14. Istilah *stop words* merujuk pada...
    A. Kata-kata yang memperlambat sistem
    B. Kata umum yang sering diabaikan dalam analisis teks
    C. Bahasa pemrograman khusus
    D. Kata yang harus disensor
    Jawaban: B

15. Apa itu akurasi dalam evaluasi model AI?
    A. Lama pelatihan model
    B. Banyaknya data
    C. Persentase prediksi model yang benar
    D. Jumlah algoritma yang digunakan
    Jawaban: C

C. Isian Singkat (5 Soal)

16. Istilah untuk proses membagi data menjadi data latih dan data uji disebut _____.
    Jawaban: split data

17. Dua contoh aplikasi NLP yang umum digunakan adalah _____ dan _____.
    Jawaban: Google Translate, Chatbot

18. Regresi adalah metode AI yang digunakan untuk memprediksi _____.
    Jawaban: nilai kontinu (atau angka)

19. _____ adalah proses membersihkan data sebelum digunakan oleh model AI.
    Jawaban: Preprocessing

20. Dalam model AI, "label" merujuk pada _____ yang ingin diprediksi.
    Jawaban: output

D. Soal Explorasi

Soal:

Jelaskan secara singkat bagaimana kamu akan melatih model AI yang bisa membedakan antara suara tepuk tangan dan suara ketikan keyboard. Apa data yang kamu butuhkan, bagaimana kamu mengumpulkannya, dan tools apa yang kamu gunakan?

Soal:

Bayangkan kamu ingin membuat chatbot sederhana untuk menjawab pertanyaan tentang jadwal pelajaran di sekolahmu. Jelaskan langkah-langkah yang akan kamu lakukan agar chatbot ini bisa merespons dengan benar.

Soal:

Dalam sebuah proyek, kamu diminta memprediksi nilai ulangan berdasarkan jumlah jam belajar siswa. Buatlah rencana sederhana yang mencakup jenis model AI yang digunakan, bentuk data yang diperlukan, dan bagaimana kamu mengevaluasi akurasi prediksinya.

Tugas Proyek

Proyek 1: Klasifikasi Objek Rumah dengan AI

Deskripsi:
Latih model AI menggunakan *Google Teachable Machine* untuk mengenali dan membedakan 2–3 objek sehari-hari, seperti sendok, botol, atau sandal.

Langkah-langkah:

- Pilih 2–3 objek di sekitarmu.
- Ambil 30–50 gambar untuk masing-masing objek dari berbagai sudut.
- Unggah gambar ke Google Teachable Machine dan latih modelnya.
- Uji model menggunakan objek nyata dan catat tingkat akurasi.
- Buat laporan singkat yang menjelaskan proses, hasil uji coba, dan evaluasi akurasi.

Tantangan Seru:
Cobalah mengenali objek dalam kondisi pencahayaan berbeda atau objek serupa (misal: sendok logam vs sendok plastik).

Proyek 2: Chatbot Jadwal Sekolah

Deskripsi:
Rancang chatbot berbasis teks yang dapat menjawab pertanyaan sederhana tentang jadwal pelajaran harian di sekolahmu.

Langkah-langkah:

- Tentukan data yang dibutuhkan (jadwal pelajaran mingguan).
- Gunakan platform seperti Google Sheets dan AppSheet atau pseudocode untuk merancang respons otomatis berdasarkan kata kunci.
- Buat antarmuka sederhana untuk menguji chatbot.
- Uji chatbot dengan berbagai pertanyaan seperti “Hari ini pelajaran apa?” atau “Senin ada pelajaran apa?”

Tantangan Seru:
Tambahkan fitur seperti notifikasi otomatis atau respons untuk pertanyaan seputar ekstrakurikuler.

Proyek 3: Prediksi Nilai Berdasarkan Jam Belajar

Deskripsi:
Gunakan data sederhana (jam belajar dan nilai ujian) untuk membuat model prediksi menggunakan regresi linier di Google Sheets atau Excel.

Langkah-langkah:

- Kumpulkan data dari teman-teman (misal: jam belajar dan nilai ujian).
- Masukkan data ke Google Sheets atau Excel.
- Buat grafik scatter plot menggunakan fitur regresi linier.
- Interpretasikan hubungan antara jam belajar dan hasil nilai.
- Presentasikan hasil prediksi dan kemungkinan akurasinya.

Tantangan Seru:
Coba masukkan variabel tambahan seperti “jumlah tidur” atau “jumlah tugas selesai” untuk melihat apakah prediksi meningkat.

Proyek 4: Diagram Alur Proyek AI

Deskripsi:
Rancang diagram alur proyek AI mulai dari pengumpulan data hingga penyebaran model (deployment).

Langkah-langkah:

- Tentukan proyek AI yang ingin kamu visualisasikan (misalnya proyek klasifikasi gambar).
- Bagi tahap-tahap: pengumpulan data, pelabelan, pelatihan model, evaluasi, dan deployment.
- Gunakan simbol alur standar (proses, keputusan, input/output).

- Gambar di kertas atau pakai tools online seperti Lucidchart, Canva, atau Draw.io.

Tantangan Seru:
Tambahkan kemungkinan kesalahan model dan solusi di setiap tahapan.

Proyek 5: Eksperimen Jumlah Data Pelatihan

Deskripsi:
Lakukan eksperimen untuk melihat pengaruh jumlah data terhadap akurasi model AI yang kamu latih.

Langkah-langkah:

- Pilih 2 kelas objek di Google Teachable Machine.
- Latih dua model: satu dengan data sedikit (misal 10 gambar per kelas), satu dengan data lebih banyak (misal 50 gambar per kelas).
- Bandingkan performa masing-masing model dengan pengujian yang sama.
- Buat laporan yang menyimpulkan bagaimana jumlah data mempengaruhi akurasi.

Tantangan Seru:
Coba eksperimen tambahan seperti mengganti latar belakang atau posisi kamera.

Proyek 6: Rancang Arsitektur Sederhana AI

Deskripsi:
Buat diagram arsitektur sederhana untuk menunjukkan bagaimana data mengalir dalam sistem AI: dari input, melalui proses model, hingga menghasilkan output.

Langkah-langkah:

- Pilih salah satu proyek AI milikmu (misal chatbot atau klasifikasi gambar).
- Identifikasi tahapan: input (misal gambar atau teks), preprocessing, pemrosesan AI, output.
- Buat diagram visual dengan anotasi di setiap tahap.
- Tambahkan keterangan sederhana agar mudah dipahami orang lain.

Tantangan Seru:
Presentasikan arsitektur ini seolah kamu menjadi tim teknis di perusahaan teknologi!

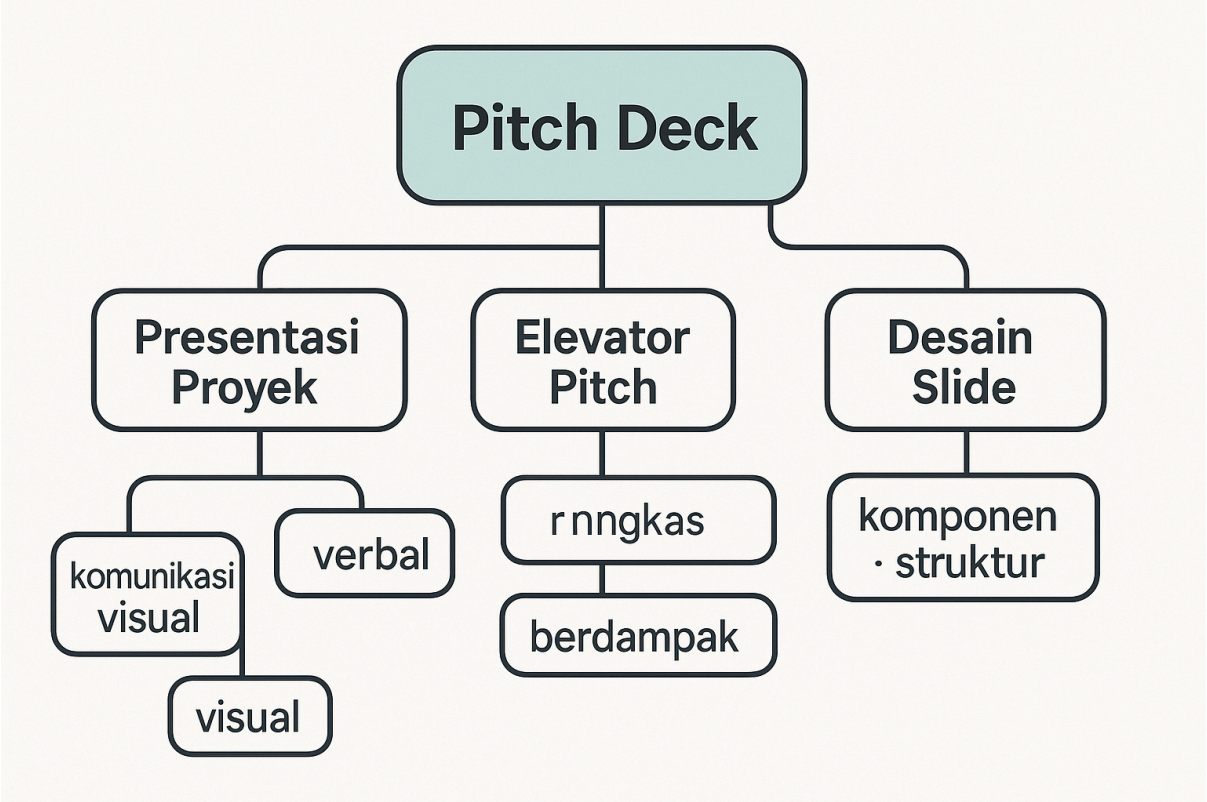
BAB 5: Buat Pitch Deck dengan Bantuan ChatGPT

Tujuan Pembelajaran

Setelah mengikuti pembelajaran ini, siswa mampu:

- Membuat *pitch deck* yang menarik dan meyakinkan untuk proyek AI.
- Menyampaikan ide secara ringkas, jelas, dan visual.
- Menggunakan ChatGPT sebagai alat bantu kreatif dalam menyusun presentasi.
- Melatih kemampuan presentasi dan berbicara di depan umum secara percaya diri.

Peta Konsep



Apersepsi

Pernahkah kamu mencoba meyakinkan orang lain tentang ide yang menurutmu keren banget, tapi ternyata sulit diterima? Mungkin karena cara penyampaiannya belum cukup menarik atau jelas. Nah, hari ini kita akan melihat contoh *pitch deck* dari startup besar seperti Airbnb dan Uber saat mereka baru memulai. Kita akan bedah kenapa presentasi mereka bisa membuat orang percaya dan mau mendukung. Seru, kan? Kabar baiknya, kamu juga bisa melakukannya! Dengan pendekatan yang kreatif dan bantuan teknologi seperti ChatGPT, kamu akan belajar membuat presentasi yang bukan hanya informatif, tapi

juga bikin orang berkata, “Wow, ini keren banget!” Tantang dirimu untuk tampil percaya diri dan tunjukkan bahwa ide kamu layak untuk didengar!

Penjelasan Konsep (teori)

Sebuah ide, meskipun luar biasa, akan sulit diterima jika tidak disampaikan dengan cara yang menarik dan meyakinkan. Di sinilah peran **pitch deck** menjadi sangat penting. Pitch deck adalah serangkaian slide yang dirancang untuk memperkenalkan ide, proyek, atau produk kamu kepada orang lain—bisa investor, mentor, guru, atau bahkan teman sekelas.

Pada bagian ini, kita akan belajar **konsep dasar pembuatan pitch deck** yang kuat, terstruktur, dan persuasif. Bukan hanya itu, kita juga akan melihat bagaimana **ChatGPT** bisa menjadi "asisten digital" kamu dalam menyusun materi presentasi yang keren dan efektif.

Apa itu Pitch Deck?

Pitch deck adalah presentasi visual yang biasanya terdiri dari 10–12 slide. Tujuannya adalah untuk:

- Menjelaskan ide atau proyek secara ringkas dan menarik
- Meyakinkan audiens bahwa idemu layak didukung
- Menunjukkan rencana yang jelas dan terarah
- Menyampaikan pesan secara visual, tidak hanya lewat kata-kata

Komponen Wajib dalam Sebuah Pitch Deck

Agar pitch deck kamu berbobot dan meyakinkan, berikut adalah bagian-bagian penting yang biasanya harus ada:

1. **Judul / Slide Pembuka**
    - Nama proyek atau produk kami.
    - Gunakan kalimat pembuka yang *catchy* dan menggugah rasa penasaran.

2. **Masalah (Problem)**
    - Ceritakan permasalahan nyata yang terjadi di masyarakat atau kelompok tertentu.
    - Tambahkan data, kutipan, atau cerita nyata agar lebih hidup.

3. **Solusi (Solution)**
    - Jelaskan bagaimana proyek AI kamu akan membantu menyelesaikan masalah tersebut.
    - Buat sesederhana mungkin agar mudah dipahami siapa pun.

4. **Demo Produk**
    - Tampilkan video pendek, gambar, atau simulasi cara kerja aplikasi kamu.

    - Ini adalah momen “wow” yang menunjukkan bahwa idemu bukan hanya teori.

5. **Tim (Team)**
    - Perkenalkan anggota tim dan keahlian mereka.
    - Tunjukkan bahwa kalian adalah tim yang solid dan punya peran masing-masing.

6. **Pasar (Market Size)**
    - Siapa yang akan menggunakan produk ini? Seberapa besar dampaknya?
    - Gunakan data atau prediksi jumlah pengguna untuk memperkuat klaim.

7. **Model Bisnis (Business Model)**
    - Jelaskan bagaimana proyekmu akan berkelanjutan secara finansial atau sosial.
    - Apakah akan gratis, berbayar, donasi, atau hibah?

8. **Kompetitor (Competition)**
    - Siapa saja pesaingmu? Apa keunggulan proyek kamu dibanding mereka?
    - Jelaskan keunikan atau nilai tambah dari solusi yang kamu tawarkan.

9. **Timeline / Pencapaian**
    - Ceritakan perjalanan proyekmu hingga sekarang.
    - Sertakan rencana jangka pendek dan panjang untuk menunjukkan arah yang jelas.

10. **Call to Action (CTA)**

    - Akhiri dengan ajakan jelas: “Kami butuh mentor”, “Berikan masukanmu”, atau “Dukung proyek kami!”

Struktur Slide yang Efektif

> “Pitch deck terbaik itu **singkat, jelas, dan visual**.”

Tips praktis dalam menyusun pitch deck:

- Idealnya hanya **10–12 slide**
- Gunakan **bahasa ringkas** dan **visual** yang membantu pemahaman
- Fokus pada esensi, bukan semua detail teknis

Gunakan ChatGPT Sebagai Partner Kreatif

ChatGPT bisa menjadi teman pintar yang membantu berpikir lebih tajam dan menyusun presentasi yang lebih profesional. Caranya?

1. **Brainstorming Ide**

    - Minta ChatGPT untuk bantu mencari sudut pandang unik untuk setiap bagian pitch deck.

2. **Menulis Draft Teks**
    - Tanyakan: “Bisa bantu tuliskan kalimat pembuka yang menarik untuk pitch saya?”

3. **Menyusun Poin Penting**
    - Gunakan ChatGPT untuk mengubah paragraf panjang menjadi bullet points yang ringkas dan jelas.

4. **Memperbaiki Bahasa**
    - Minta saran untuk membuat bahasamu lebih profesional, elegan, atau persuasif.

Catatan penting: Jangan hanya “copy-paste”! Gunakan hasil dari ChatGPT sebagai bahan awal, lalu **tinjau kembali, edit, dan sesuaikan** agar sesuai dengan suara dan karakter kamu.

Prinsip Desain Slide

Pitch deck yang bagus tidak hanya soal isi, tapi juga cara penyampaiannya. Gunakan prinsip desain berikut agar presentasimu nyaman dilihat dan mudah diingat:

- **Sederhana & Rapi:** Jangan terlalu banyak teks atau gambar.
- **Visual Menarik:** Gunakan ilustrasi, grafik, atau ikon yang mendukung isi.
- **Konsisten:** Gunakan font dan warna yang seragam.
- **Hierarki Informasi:** Sorot bagian penting dengan ukuran font, warna, atau posisi.

Elevator Pitch: Ringkasan Cepat yang Mengesankan

Bayangkan kamu hanya punya 30–60 detik untuk menjelaskan proyekmu kepada seorang investor. Apa yang akan kamu katakan?

> Elevator pitch adalah versi “super singkat” dari pitch deck-mu — **menarik, jelas, dan menggugah rasa ingin tahu**.

Gunakan ini sebagai latihan menyampaikan ide kamu dalam situasi terbatas tapi menantang!

Dengan memahami teori dan konsep dasar ini, kamu siap membuat pitch deck yang bukan hanya informatif, tapi juga **menggugah perhatian dan membuka peluang baru**. Jadikan ChatGPT sebagai rekan kreatifmu, dan biarkan ide-idemu bersinar melalui presentasi yang cerdas, menarik, dan berani!

Ilustrasi: Ketika Ide Hebat Sulit Didengar

Bayangkan kamu punya ide hebat: aplikasi AI yang bisa bantu teman-temanmu belajar dengan lebih cepat. Kamu percaya banget ini bisa bantu banyak orang. Tapi... saat kamu menjelaskan ke guru atau teman, mereka malah bingung dan tidak tertarik. Rasanya seperti membawa harta karun, tapi tidak ada yang peduli.

Inilah tantangannya: **bukan hanya punya ide, tapi juga bisa menyampaikannya dengan cara yang menarik, jelas, dan meyakinkan**. Di sinilah peran **pitch deck** sangat penting.

Dengan **ChatGPT sebagai partner**, kamu tidak sendirian. Ia bisa bantu kamu mengubah ide mentah menjadi presentasi keren dan profesional. Tapi kamu tetap harus berpikir kritis, kreatif, dan berlatih presentasi dengan percaya diri.

Contoh Kasus atau Ilustrasi

Kasus 1: "Udara Tak Terlihat, Tapi Membunuh": Aplikasi AI Deteksi Kualitas Udara

Ilustrasi Masalah:
Bayangkan kamu sedang jogging di taman kota. Langit biru, tapi napasmu berat dan tenggorokan terasa gatal. Tanpa kamu sadari, kualitas udara sedang buruk—dan kamu tidak punya cara cepat untuk mengetahuinya.

Solusi AI:
Sebuah aplikasi berbasis **sensor dan AI** memberikan peringatan dini langsung ke ponselmu saat kualitas udara memburuk. Kamu bisa tahu kapan harus pakai masker atau menunda aktivitas di luar.

Kasus 2: Kasus "Sampah Menyapa Kamera": AI Deteksi Sampah Otomatis

Ilustrasi Masalah:
Di kantin sekolah, siswa bingung: gelas plastik ini masuk ke tempat sampah organik atau non-organik? Akhirnya, semua masuk ke tempat yang salah. Hasilnya? Gunungan sampah tercampur yang tak bisa didaur ulang.

Solusi AI:
Bayangkan kamu arahkan kamera HP ke sampah, dan aplikasi langsung memberitahu: “Ini plastik! Masukkan ke tempat non-organik!” Teknologi **computer vision** membantu kita memilah sampah secara cerdas dan cepat.

Kasus 3: "Guru 24 Jam di Saku": Chatbot Bimbingan Belajar

Ilustrasi Masalah:
Kamu ingin tanya soal matematika jam 10 malam, tapi guru privat sudah tidak bisa dihubungi. Buku pelajaran tak membantu. Frustasi meningkat, semangat menurun.

Solusi AI:
Sebuah chatbot bimbingan belajar berbasis AI yang bisa kamu ajak ngobrol kapan saja. Ia bisa menjelaskan rumus, memberikan contoh soal, bahkan menyemangati kamu saat belajar sendirian.

Kasus 4: "Mendeteksi Emosi yang Tak Terucap": AI Pendeteksi Emosi

Ilustrasi Masalah:
Seorang temanmu tiba-tiba sering murung dan menyendiri. Tapi saat ditanya, dia cuma bilang “nggak apa-apa”. Ternyata, ia sedang stres berat tapi sulit mengungkapkannya.

Solusi AI:
Aplikasi AI yang menganalisis nada suara atau cara menulis pesan bisa mendeteksi tanda-tanda stres atau kecemasan. Aplikasi ini bisa menyarankan bantuan: meditasi, konselor, atau sekadar pengingat untuk istirahat.

Kasus 5: Kasus "Cuaca Bisa Menipu": AI Prakiraan Cuaca Mikro untuk Petani

Ilustrasi Masalah:
Seorang petani memutuskan untuk menyiram sawahnya karena ramalan TV bilang hari ini panas. Ternyata siang harinya hujan deras. Tanamannya rusak, hasil panen berkurang.

Solusi AI:
Aplikasi AI dengan sensor cuaca lokal dan model prediksi mikro-cuaca memberikan ramalan **khusus di lokasi** petani. Akurat hingga per desa. Petani bisa ambil keputusan tepat berdasarkan data real-time.

Kasus 6: Bonus Kasus "Slide Bikin atau Bikin Pusing?": Desain Slide AI

Ilustrasi Masalah:
Kamu membuat presentasi AI dengan banyak data, tapi semua tulisan kecil dan penuh teks. Teman-teman malah sibuk main HP saat kamu presentasi. Kenapa?

Solusi:
Dengan memahami prinsip **desain visual yang efektif**, kamu bisa membuat dua versi slide: satu padat dan membingungkan, satu lagi sederhana namun menarik. AI bisa bahkan bantu kamu menyarankan tata letak yang memikat.

Contoh Soal dan Pembahasan

Soal:

Bayangkan kamu hanya punya 30 detik untuk berbicara dengan seorang investor di dalam lift. Buatlah elevator pitch untuk proyek AI “Chatbot Edukasi untuk Pelajar SMA”. Jelaskan mengapa idemu layak didanai.

Pembahasan:

Elevator pitch adalah seni menjelaskan ide secara cepat dan meyakinkan. Gunakan bahasa yang menarik dan langsung ke inti masalah. Contoh:

> *“Setiap hari, jutaan pelajar SMA bingung memahami pelajaran. Chatbot kami hadir sebagai tutor digital 24 jam yang menjawab pertanyaan, menjelaskan materi sesuai gaya belajar, dan membuat belajar jadi seru. Dengan AI ini, pendidikan jadi lebih setara dan personal. Ini masa depan* ***belajar mandiri!”***

Soal:

Apa tujuan utama dari menyertakan ‘Demo Produk’ dalam pitch deck proyek AI kamu? Jelaskan alasannya secara logis.

Pembahasan:

Demo produk bukan sekadar formalitas, tapi bukti bahwa solusi kamu benar-benar bisa diterapkan. Tanpa demo, ide hanya terlihat bagus di atas kertas. Saat demo berjalan lancar, kepercayaan investor meningkat karena mereka bisa **melihat bukti nyata** bagaimana produk bekerja.

Soal:

Bayangkan kamu membuat pitch deck untuk proyek “AI Pemantau Kualitas Udara”. Tuliskan tiga poin penting yang harus ada di slide “Masalah” dan jelaskan mengapa itu penting bagi masyarakat.

Pembahasan:

Slide “Masalah” harus menyentuh isu yang nyata dan dirasakan. Tiga poin yang dapat dituliskan:

1. **Warga tidak mengetahui kondisi udara di sekitar mereka.** → Kesadaran rendah, risiko tinggi.
2. **Data kualitas udara sulit diakses secara real-time.** → Masyarakat tidak bisa mengambil keputusan cepat.
3. **Polusi udara diam-diam berdampak pada kesehatan jangka panjang.** → Masalah serius yang sering diabaikan.

Semua poin ini membangun urgensi akan perlunya solusi dari proyek AI kamu.

Soal:

Menurutmu, berapa jumlah slide ideal dalam sebuah pitch deck proyek AI dan mengapa jumlah tersebut dianggap efektif?

Pembahasan:

Idealnya, **10–12 slide** cukup untuk menjelaskan ide secara komprehensif namun ringkas. Mengapa?

- Terlalu banyak slide → audiens lelah dan kehilangan fokus.
- Terlalu sedikit → informasi penting bisa terlewat.
  Struktur seperti ini (Masalah – Solusi – Produk – Pasar – Tim – Demo – Penutup) membuat presentasi padat, jelas, dan tetap menarik.

Soal:

Kamu diminta menilai dua pitch: satu menjelaskan teknis AI dengan sangat detail, satu lagi fokus pada dampak AI untuk pengguna. Pitch mana yang lebih baik untuk investor awam, dan kenapa?

Pembahasan:

Pitch yang **fokus pada dampak pengguna** biasanya lebih menarik bagi investor awam. Investor ingin tahu **manfaat**, bukan algoritma. Mereka bertanya: *"Apa dampaknya bagi masyarakat? Seberapa besar pasar potensialnya?"* Detail teknis penting, tapi sebaiknya disampaikan setelah mereka tertarik secara emosional dan logis pada **masalah dan solusinya**.

Fakta Menarik / Fun Facts

- Steve Jobs terkenal dengan presentasi produknya yang minimalis dan fokus pada cerita. Ia sering hanya menggunakan beberapa kata kunci atau gambar besar di slide-nya!
- Steve Jobs hanya menggunakan gambar besar dan 1-2 kata per slide saat memperkenalkan iPhone pertama.
- Beberapa startup bahkan mendapatkan investasi jutaan dollar hanya dari satu pitch deck yang meyakinkan!
- Pitch deck Airbnb di awal hanya 14 slide—tapi bisa membuat investor percaya dan memberi dana jutaan dolar!
- Beberapa startup sukses hanya karena bisa menjelaskan ide mereka dalam **30 detik** dengan kuat dan jelas.

Tips Belajar atau Tips Cepat Menghafal

Tips Cepat Menghafal

- Alternatif Alur Pitch yang kuat:
    - **Masalah → Solusi → Cara Kerja → Tim → Dampak → Ajakan**
    - **Masalah → Solusi → Bagaimana (demo) → Tim → Mengapa Sekarang → Apa yang kami butuhkan**
- Bikin cerita pendek seolah kamu sedang ngobrol dengan teman.
- Fokus pada cerita, bukan hanya fakta. Emosi lebih mudah diingat.
- Gunakan warna dan gambar untuk membantu mengingat poin penting.
- Berlatih, berlatih, berlatih! Semakin sering Anda mempresentasikan, semakin lancar dan percaya diri Anda.

Mind Map Pitch Deck



Cerita untuk Mengingat Struktur Pitch

Bayangkan kamu ingin bantu nenek kamu yang tinggal di kota penuh polusi. Kamu ceritakan masalahnya (udara kotor), lalu solusi (alat AI yang kasih info polusi), cara kerjanya (pakai sensor dan notifikasi), siapa yang bantu kamu (tim), dan apa yang kamu butuh (dukungan untuk produksi). Cerita ini akan mudah diingat dan jadi dasar pitch kamu.

Aktivitas Siswa

Aktivitas 1: Kreasi Pitch Deck Bareng ChatGPT

Ayo jadi tim kreatif startup!
Bentuk kelompok dan mulailah menyusun pitch deck untuk proyek akhir kalian. Gunakan bantuan ChatGPT sebagai *partner brainstorming*.

Langkah seru yang bisa dicoba:

- Tentukan struktur pitch deck kalian (Masalah, Solusi, Target Pasar, dll.).
- Diskusikan isi setiap bagian dengan tim, lalu tanyakan pada ChatGPT untuk menyempurnakan narasinya.
- Eksplor ide visual dan cara menyampaikan masalah secara menarik.

Goal: Kalian akan punya pitch deck yang solid dan siap diuji ke publik!

Aktivitas 2: Simulasi Presentasi Internal – “Tunjukkan Giginya!”

Saatnya gladi resik jadi presenter handal.
Setiap tim mempresentasikan pitch deck mereka di depan guru dan tim lain. Ini bukan ujian—tapi sesi latihan terbuka penuh insight.

Cara menjalankan:

- Latih pembagian peran: siapa pembuka, siapa penyampai solusi, siapa penutup.
- Presentasikan di depan kelas secara profesional namun santai.
- Minta masukan dari audiens dan guru—catat semuanya!

Goal: Pitch deck kalian akan makin matang berkat masukan jujur dari “investor palsu”.

Aktivitas 3: 30-Detik Elevator Pitch Challenge

Bisa jual ide dalam setengah menit? Tantang dirimu!
Buat video pitch berdurasi maksimal 30 detik yang menjelaskan proyek kalian dengan gaya kreatif.

Langkah-langkahnya:

- Rangkai skrip pendek dan padat.
- Pilih gaya penyampaian: bisa naratif, pakai analogi unik, atau sentuhan humor.
- Rekam videonya dan putar bersama di kelas!

Goal: Mengasah kejelasan berpikir dan kemampuan meyakinkan audiens dalam waktu super singkat.

Aktivitas 4: Slide Duel: “Desain Membosankan vs Desain Keren”

Buktikan kekuatan visual!
Buat dua versi slide tentang bagian *Masalah* dalam proyek kalian—satu dibuat dengan desain seadanya, satu lagi dengan visual yang menarik.

Langkahnya:

- Rancang kedua versi slide.
- Presentasikan keduanya kepada teman-teman.
- Diskusikan: versi mana yang lebih memikat? Kenapa?

Goal: Belajar bahwa penyampaian visual bisa menentukan seberapa kuat pesan kalian diterima.

Aktivitas 5: AI Partner Coding Challenge (Versi Pseudocode)

Kalian = Penemu AI yang solutif!
Tulis pseudocode sederhana untuk fitur AI dalam proyek kalian. Tak perlu koding asli—cukup logika alurnya.

Langkah yang bisa dijalani:

- Tentukan fitur cerdas yang kalian buat (misalnya, notifikasi kualitas udara).
- Tulis pseudocode bersama tim.
- Gunakan ChatGPT untuk membantu menyusun logika atau menyempurnakan formatnya.

Goal: Pahami dasar logika pemrograman sebagai pondasi fitur AI kalian.

Aktivitas 6: Roleplay: Founder vs Investor

Uji daya jual ide kalian!
Siswa dibagi menjadi dua kelompok: tim *founder* dan tim *investor*. Founder mempresentasikan ide, investor bertanya dan memberi tantangan.

Langkah seru:

- Siapkan pitch pendek dari proyek masing-masing tim.
- Rotasi peran jika perlu agar semua mengalami jadi investor dan founder.
- Setelah sesi, diskusikan apa yang membuat pitch itu meyakinkan atau kurang kuat.

Goal: Kalian belajar berpikir kritis, tangguh menerima kritik, dan makin percaya diri menjual ide.

Rangkuman

Pitch deck merupakan alat yang sangat penting untuk menyampaikan ide proyek *AI* secara **ringkas**, **menarik**, dan **meyakinkan** kepada berbagai audiens, seperti **investor**, **pendidik**, atau **mitra potensial**. Kemampuan untuk mengkomunikasikan proyek *AI* secara efektif melalui pitch deck yang kuat adalah **keterampilan krusial di dunia nyata**.

Dengan memanfaatkan *tools AI* seperti **ChatGPT**, siswa dapat menyusun presentasi yang **persuasif** dan memberikan **kesan mendalam** kepada audiens. Pitch deck yang efektif umumnya terdiri dari beberapa komponen utama, yakni: **identifikasi masalah**, **solusi yang ditawarkan**, **demo produk**, **profil tim**, **analisis pasar**, dan **ajakan bertindak (CTA)**.

Dalam proses pembuatan, **ChatGPT** bisa berperan sebagai alat bantu dalam **menyusun teks**, **mengembangkan ide**, serta **meningkatkan kualitas bahasa** pada setiap slide. Dengan desain yang **sederhana namun menarik**, narasi yang **jelas**, dan struktur yang **terorganisasi**, siswa dapat menyampaikan ide mereka dengan lebih **percaya diri**.

Selain itu, **latihan presentasi** secara berulang dan eksplorasi ide secara mendalam akan membantu siswa **mengasah kemampuan komunikasi dan kolaborasi**, yang sangat dibutuhkan dalam dunia profesional saat ini.

Latihan Soal

A. Benar atau Salah (5 Soal)

1. Pitch deck yang baik harus berisi teks sebanyak mungkin agar informasi lengkap.
   Jawaban: Salah

2. ChatGPT bisa digunakan untuk membuat draft teks persuasif untuk pitch deck.
   Jawaban: Benar

3. Slide "Tim" tidak terlalu penting jika proyek dilakukan sendiri.
   Jawaban: Salah

4. Elevator pitch sebaiknya berdurasi maksimal 1 menit.
   Jawaban: Benar

5. Desain slide yang menarik tidak memengaruhi keberhasilan pitch deck.
   Jawaban: Salah

B. Pilihan Ganda (10 Soal)

6. Tujuan utama pitch deck adalah untuk:
   a. Menjelaskan sejarah perusahaan
   b. Menarik perhatian dan meyakinkan audiens
   c. Menyajikan semua data secara detail

d. Menunjukkan keunggulan teknologi
Jawaban: b

7. Fungsi utama dari slide “Masalah” adalah:
   a. Menyampaikan permintaan pendanaan
   b. Menjelaskan cara kerja teknologi
   c. Menyampaikan urgensi yang ingin diselesaikan
   d. Menjelaskan ukuran pasar
   Jawaban: c

8. Komponen pitch deck yang berisi informasi tentang siapa pengguna dan seberapa besar pasar adalah:
   a. Tim
   b. Pasar (Market Size)
   c. Demo Produk
   d. Call to Action
   Jawaban: b

9. Yang bukan termasuk prinsip desain pitch deck adalah:
   a. Sederhana dan bersih
   b. Banyak teks agar informatif
   c. Konsisten dalam warna dan font
   d. Gunakan visual menarik
   Jawaban: b

10. ChatGPT paling efektif digunakan dalam pembuatan pitch deck untuk:
    a. Mendesain slide
    b. Menyusun poin-poin utama dan narasi teks
    c. Menyusun grafik pasar
    d. Membuat presentasi animasi
    Jawaban: b

11. Fungsi slide “Call to Action” adalah:
    a. Menjelaskan alur kerja produk
    b. Menyampaikan permintaan atau ajakan kepada audiens
    c. Menyampaikan masalah
    d. Menjelaskan model bisnis
    Jawaban: b

12. Salah satu keuntungan menggunakan ChatGPT adalah:
    a. Menyajikan gambar 3D
    b. Memfilter pendapat investor
    c. Membantu merumuskan kalimat persuasif
    d. Menyusun presentasi dengan PowerPoint
    Jawaban: c

13. Elevator pitch sebaiknya mencakup:
    a. Semua bagian proyek secara mendalam

b. Penjelasan teknis model AI
c. Masalah, solusi, dan dampak secara singkat
d. Timeline proyek
Jawaban: c

14. Slide “Kompetisi” digunakan untuk menunjukkan:
a. Siapa yang akan membeli produk
b. Apa masalah utama di masyarakat
c. Siapa pesaing dan keunggulan proyek kita
d. Estimasi biaya produksi
Jawaban: c

15. Jika sebuah pitch deck terlalu panjang dan penuh teks, maka:
a. Akan lebih mudah dimengerti
b. Investor akan lebih tertarik
c. Kemungkinan audiens kehilangan fokus meningkat
d. Slide menjadi lebih interaktif
Jawaban: c

C. Isian Singkat (5 Soal)

16. Ringkasan proyek yang disampaikan dalam waktu singkat disebut _____.
Jawaban: Elevator pitch

17. Slide yang berisi daftar anggota dan peran dalam tim disebut slide _____.
Jawaban: Tim

18. Pitch deck yang baik biasanya terdiri dari _____ hingga _____ slide.
Jawaban: 10, 12

19. Alat bantu berbasis AI yang dapat digunakan untuk brainstorming ide pitch deck adalah _____.
Jawaban: ChatGPT

20. Desain pitch deck yang menarik sebaiknya menggunakan _____ dan _____ yang seragam.
Jawaban: font, warna

D. Soal Explorasi

Soal:

Tuliskan Elevator Pitch proyek AI kamu dalam maksimal 4 kalimat. Gunakan struktur *Masalah → Solusi → Dampak.*
Jawaban bervariasi tergantung siswa. **Contoh:**
"Banyak pelajar kesulitan belajar mandiri di rumah. Chatbot Edukasi kami hadir sebagai guru digital yang menjawab pertanyaan kapan saja. Dengan teknologi AI, kami memberi penjelasan yang mudah dipahami dan dipersonalisasi. Solusi ini

membantu siswa belajar lebih efektif."

Soal:

Buat 3 poin isi slide 'Masalah' dan 'Solusi' untuk proyek AI yang kamu minati (misalnya: aplikasi prediksi cuaca, deteksi sampah, atau lainnya).
Jawaban bervariasi. **Contoh:**
Masalah:

- Warga tidak mendapatkan informasi kualitas udara secara real-time.
- Polusi udara menyebabkan gangguan kesehatan.
- Data lingkungan sulit diakses masyarakat awam.

Solusi:

- Aplikasi berbasis AI yang membaca data sensor udara.
- Memberi peringatan dini melalui notifikasi.
- Dashboard visual mudah dipahami semua kalangan.

Soal:

Bayangkan kamu investor. Kamu melihat pitch deck dengan desain buruk dan terlalu banyak teks. Tuliskan 3 hal yang ingin kamu perbaiki agar lebih efektif.
Jawaban contoh:

- Kurangi jumlah teks di setiap slide, gunakan poin-poin.
- Tambahkan gambar/ikon untuk memperjelas isi.
- Gunakan kombinasi warna dan font yang konsisten agar lebih profesional.

Tugas Proyek

Proyek 1: Tantangan Pitch Deck: Susun Ide Proyek AI dalam 10 Slide

Deskripsi Proyek:
Pilihlah satu ide proyek AI yang menarik minatmu, lalu susunlah pitch deck (presentasi) dalam 10 slide. Gunakan bantuan ChatGPT untuk menyusun teks tiap bagian.

Langkah-langkah:

- Tentukan ide proyek AI yang realistis dan relevan (misalnya: deteksi emosi, chatbot pembelajaran, dsb).
- Gunakan struktur 10 slide: Judul, Masalah, Solusi, Demo, Tim, Pasar, Kompetitor, Model Bisnis, Timeline, dan CTA (ajak bertindak).
- Manfaatkan ChatGPT untuk menyusun isi tiap slide agar ringkas dan persuasif.
- Pastikan tiap slide punya satu ide utama yang kuat.

Proyek 2: Desain Kreatif: Visualisasi 3 Slide Utama

Deskripsi Proyek:
Rancang tampilan visual yang menarik untuk 3 slide utama dari pitch deck-mu:

Judul, Masalah, dan Solusi. Fokus pada konsistensi desain dan kemudahan dipahami.

Langkah-langkah:

- Gunakan tools seperti Canva, Google Slides, atau PowerPoint.
- Pilih warna, ikon, dan layout yang seragam.
- Pastikan teks mudah dibaca, tidak terlalu padat.
- Buat desain yang mencerminkan karakter proyekmu.

Proyek 3: Elevator Pitch: Naskah Presentasi 1 Menit

Deskripsi Proyek:
Tulis dan latih naskah presentasi singkat (maksimal 100 kata) yang bisa kamu ucapkan dalam 1 menit untuk menjelaskan proyekmu secara padat dan menarik.

Langkah-langkah:

- Fokus pada masalah, solusi, dan keunikan proyek.
- Uji durasi naskahmu – pastikan tidak lebih dari 1 menit saat dibaca lantang.
- Latih cara penyampaian: intonasi, ekspresi, dan kepercayaan diri.
- Rekam dan evaluasi hasilnya untuk peningkatan.

Proyek 4: Simulasi Presentasi & Evaluasi Diri

Deskripsi Proyek:
Lakukan presentasi singkat di depan teman atau guru, lalu kumpulkan 2 masukan utama. Evaluasi bagaimana kamu akan memperbaiki presentasimu.

Langkah-langkah:

- Latihan di depan 1–2 orang dengan pitch dan slide kamu.
- Minta mereka mencatat kelebihan dan kekurangan presentasimu.
- Tulis dua saran yang kamu terima dan rencanakan perbaikannya (misalnya: lebih percaya diri, kurangi teks di slide, dsb).
- Simulasikan kembali setelah revisi untuk melihat perbedaannya.

Proyek 5: Kolaborasi Kreatif: Bandingkan Pitch Deck bersama Teman

Deskripsi Proyek:
Bekerjalah dalam kelompok kecil (2–3 orang) untuk membuat pitch deck dari ide proyek yang sama, lalu bandingkan hasil masing-masing. Diskusikan versi mana yang paling meyakinkan dan mengapa.

Langkah-langkah:

- Sepakati satu topik proyek bersama teman.
- Buat pitch deck masing-masing secara mandiri.
- Lakukan presentasi satu sama lain.

- Diskusikan: Apa yang paling menarik? Slide mana yang paling kuat? Mana yang paling persuasif?
- Gunakan hasil diskusi untuk menyempurnakan pitch deck final.

Proyek 6: Finalisasi Pitch Deck & Persiapan Presentasi Akhir

Deskripsi Proyek:
Perbaiki dan lengkapi pitch deck akhir kamu berdasarkan semua umpan balik sebelumnya. Pastikan semua elemen penting sudah ada, desainnya menarik, dan narasi presentasimu siap.

Langkah-langkah:

- Revisi isi dan desain berdasarkan masukan dari simulasi dan kolaborasi.
- Pastikan semua 10 slide sudah ada dan saling terhubung.
- Latihan presentasi dari awal hingga akhir dengan waktu maksimal 5 menit.
- Siapkan file dan alat presentasi (PDF, pointer, dll.) untuk ujian akhir atau demo day.

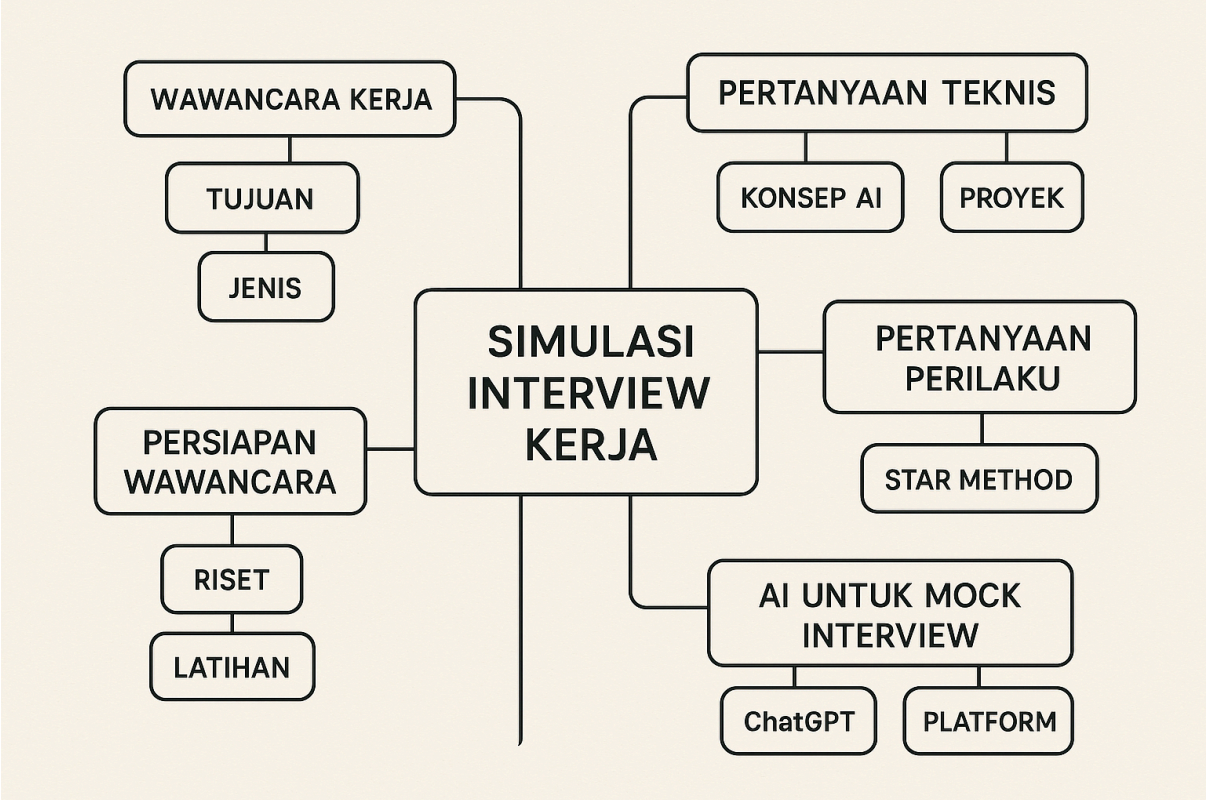
BAB 6: Simulasi Interview Kerja dengan AI (Mock Interview)

Tujuan Pembelajaran

Siswa mampu:

- Siswa mampu mempersiapkan diri menghadapi wawancara kerja di bidang AI melalui simulasi wawancara.
- Siswa dapat mengenali kelebihan dan kekurangan diri saat menjawab pertanyaan interview.
- Siswa memahami jenis-jenis pertanyaan yang umum muncul dalam wawancara kerja.
- Siswa bisa menggunakan AI sebagai alat bantu latihan interview secara mandiri.

Peta Konsep



Apersepsi

Teman-teman, pernahkah kalian membayangkan bagaimana rasanya mengikuti wawancara kerja sungguhan? Apa sih yang membuat kalian merasa tegang atau bahkan takut saat membayangkannya? Mungkin karena tidak tahu harus menjawab apa, atau khawatir terlihat gugup? Nah, siapa tahu ada di antara kalian yang sudah pernah mencoba latihan interview sendiri di rumah — keren banget, lho, kalau sudah!

Sebelum kita mulai belajar lebih jauh, yuk kita cari dan tonton dulu video singkat dan seru tentang hal-hal yang boleh dan tidak boleh dilakukan saat wawancara kerja. Videonya lucu tapi penuh pelajaran penting!

Setelah itu, kita akan berdiskusi bersama: Menurut kalian, mana yang lebih penting dalam wawancara—jawaban yang cerdas atau percaya diri saat berbicara? Dan... bayangkan kalau kalian bisa latihan interview dengan bantuan AI seperti ChatGPT. Seru nggak tuh? Siapkah kalian mencoba tantangan ini?

Penjelasan Konsep (teori)

Dalam dunia kerja, wawancara bukan hanya sekadar sesi tanya jawab. Ini adalah **momen penting yang bisa membuka jalan menuju karir impianmu**. Terutama di bidang yang sangat dinamis seperti Artificial Intelligence (AI), wawancara menjadi ajang untuk menunjukkan kemampuan, karakter, dan kesiapanmu berkontribusi di dunia profesional.

Mari kita belajar bersama secara menyenangkan dan menantang apa saja yang perlu dipahami untuk menghadapi wawancara kerja, khususnya di bidang AI.

Apa Tujuan dari Wawancara Kerja?

Wawancara kerja punya beberapa tujuan utama, di antaranya:

1. **Mengenal Kandidat Lebih Dalam**
   HRD atau rekruter ingin tahu lebih dari sekadar apa yang tertulis di CV. Mereka ingin melihat *siapa kamu sebenarnya*.
2. **Menilai Kecocokan Keterampilan**
   Apakah kamu benar-benar punya skill teknis (hard skill) dan keterampilan sosial (soft skill) yang dibutuhkan?
3. **Melihat Kesesuaian Budaya**
   Setiap perusahaan punya nilai dan budaya kerja sendiri. Recruiter ingin tahu: cocokkah kamu dengan tim mereka?

Jenis-Jenis Pertanyaan Wawancara

Tidak semua pertanyaan itu sulit, tapi banyak yang *mengecoh*. Yuk kenali jenis-jenis pertanyaan yang sering muncul!

Pertanyaan Teknis

- Digunakan untuk menguji seberapa dalam pemahamanmu tentang:
  - Konsep dasar AI (seperti machine learning, neural networks)
  - Tools yang sering dipakai (contohnya: Python, TensorFlow, scikit-learn)
  - Algoritma atau pendekatan yang kamu gunakan dalam proyek AI

Pertanyaan Perilaku (Behavioral Questions)

- Fokus pada pengalaman masa lalu:
    - Bagaimana kamu bekerja dalam tim?
    - Pernah gagal? Bagaimana menghadapinya?
- Jawaban biasanya menggunakan metode **STAR**:
    - **S**ituation (Situasi)
    - **T**ask (Tugas)
    - **A**ction (Tindakan)
    - **R**esult (Hasil)

Pertanyaan Studi Kasus

- Diberikan sebuah skenario, misalnya: “Jika kamu diminta membuat sistem deteksi wajah, langkah apa yang kamu ambil?”
- Tujuannya: menilai cara berpikir, kreativitas, dan kemampuan problem-solving.

Pertanyaan Umum

- Pertanyaan yang tampak mudah, tapi butuh persiapan matang!
    - "Ceritakan tentang diri Anda."
    - "Apa kekuatan dan kelemahan Anda?"
    - "Kenapa kamu ingin bekerja di perusahaan ini?"

Tips Jitu Menghadapi Wawancara

Agar tidak grogi dan bisa tampil maksimal, simak strategi berikut:

- **Riset Perusahaan & Posisi**
    - Kenali visi, misi, dan produk perusahaan. Tunjukkan bahwa kamu benar-benar tertarik!

- **Pahami Job Description (Deskripsi Pekerjaan)**
    - Baca dengan teliti dan temukan *kata kunci penting*. Sesuaikan jawabanmu dengan yang dicari oleh perusahaan.

- **Siapkan Jawaban untuk Pertanyaan Umum**
    - Latih jawabanmu di depan cermin atau dengan teman.

- **Kuasai Proyek Sendiri**
    - Ceritakan proyek AI yang pernah kamu buat. Apa tantangannya? Apa hasilnya? Apa yang kamu pelajari?

- **Ajukan Pertanyaan Balik**
    - Ini menunjukkan bahwa kamu aktif, kritis, dan punya keinginan belajar.
        - Contoh: "Seperti apa budaya kerja di tim AI perusahaan ini?"

- **Penampilan dan Bahasa Tubuh**
    - Berpakaian rapi dan profesional.
    - Kontak mata, postur tubuh tegap, dan jangan lupa senyum!

Menggunakan AI (seperti ChatGPT) Sebagai Simulasi Wawancara

Wawancara kerja bisa dilatih, dan salah satu cara seru adalah... dengan bantuan AI!

Peran AI dalam Simulasi:

- **Menjadi Pewawancara Virtual**
  AI seperti ChatGPT bisa berperan sebagai interviewer yang:
    - Mengajukan pertanyaan teknis dan perilaku
    - Memberikan umpan balik langsung setelah kamu menjawab
    - Membantu menilai kekuatan dan kelemahanmu

Cara Menggunakan ChatGPT untuk Simulasi Interview:

1. **Berikan Konteks yang Jelas**

   "Saya ingin Anda menjadi pewawancara untuk posisi Junior AI Developer."

2. **Mulai Sesi dengan Sopan dan Siap**

   "Halo, saya siap memulai mock interview."

3. **Tentukan Fokus Latihan**

   "Tolong tanyakan tentang proyek AI yang pernah saya kerjakan."

4. **Minta Feedback yang Mendalam**

   "Beri saya umpan balik tentang jawaban saya. Apa yang sudah baik? Apa yang perlu ditingkatkan?"

Penutup: Belajar Interview Itu Seperti Latihan Bela Diri!

Kamu tidak akan menjadi master dalam semalam. Tapi dengan latihan yang konsisten, kamu bisa menguasai “jurus-jurus” wawancara dengan percaya diri dan senyuman.

> “Practice doesn’t make perfect. **Perfect practice** makes perfect.” – Vince Lombardi

Ayo, latih dirimu dengan simulasi interview berbasis AI, dan jadikan wawancara pertamamu sebagai batu loncatan menuju karier hebat di dunia AI!

Ilustrasi

Bayangkan seorang siswa bernama **Dina**, kelas 12 IPA, yang bercita-cita menjadi seorang **AI Engineer**. Ketika ia mendengar bahwa ada lowongan magang untuk "Junior AI

Developer", hatinya berdebar. Namun, begitu membaca bahwa ada **interview kerja** yang harus ia jalani, ia langsung merasa gugup dan tidak percaya diri.

“Bagaimana jika aku tidak tahu jawabannya?” “Bagaimana jika aku terlalu gugup untuk bicara?” pikirnya.

Tapi Dina tidak menyerah. Ia memutuskan untuk **latihan menggunakan ChatGPT**. Setiap malam, ia melakukan mock interview, mencatat pertanyaan-pertanyaan, dan memperbaiki jawabannya berdasarkan **feedback AI**. Awalnya sulit, namun sedikit demi sedikit, Dina mulai menjawab dengan percaya diri, bisa menjelaskan proyek AI-nya dengan baik, dan akhirnya berhasil **lolos wawancara**!

Contoh Kasus atau Ilustrasi

Kasus 1: “Halo, Nama Saya dan Ini Alasan Saya!”

Ilustrasi Kasus:
Bayangkan kamu duduk di depan layar laptop, siap menjalani wawancara virtual untuk posisi *Junior Data Scientist*. Pewawancara (AI seperti ChatGPT) berkata, “Ceritakan sedikit tentang dirimu dan mengapa kamu tertarik dengan posisi ini.”

Tantangannya:
Kamu menjawab dengan penuh semangat, “Saya suka teknologi dan angka!” Tapi... pewawancara ingin lebih. Di mana buktinya?

Solusi:
Gunakan pengalaman nyata. Misalnya:
“Saya suka data karena saat proyek di kelas X, saya menganalisis data penjualan kantin sekolah menggunakan Excel dan menemukan pola belanja siswa.”

Kasus 2: Rian Bingung Saat Ditanya Kelebihan Dirinya

Ilustrasi Kasus:
Wawancara berjalan lancar, sampai kamu ditanya: “Apa kelebihan kamu?”
Rian, seperti banyak siswa lain, terdiam. “Saya... uhh... rajin, mungkin?”

Tantangannya:
Jawaban terlalu umum tidak membedakan dari kandidat lain.

Solusi:
Tunjukkan dengan **contoh konkret**.
Contoh: “Saya teliti dalam memeriksa data. Saat lomba statistik sekolah, saya menemukan kesalahan data yang nyaris membuat tim salah mengambil kesimpulan.”

Kasus 3: Sarah Bingung Tentang Konsep AI

Ilustrasi Kasus:
Sarah antusias ingin masuk dunia AI, tapi bingung saat ditanya, “Apa perbedaan supervised dan unsupervised learning?”

Tantangannya:
Konsep terlalu teknis, tidak tahu cara menjelaskannya.

Solusi:
Gunakan **analogi kreatif**!
Contoh:
“Supervised learning itu seperti ujian dengan kunci jawaban – kita tahu jawaban yang benar. Unsupervised seperti menjelajahi hutan tanpa peta – kita mencari pola sendiri.”

Kasus 4: Budi Gugup Berat Hingga Lupa Jawaban

Ilustrasi Kasus:
Budi sudah belajar keras. Tapi saat wawancara dimulai, tangannya dingin, lidahnya kaku, dan pikirannya kosong.

Tantangannya:
Rasa gugup menghambat performa terbaik.

Solusi:

- Latihan berulang dengan AI seperti ChatGPT.
- Rekam diri sendiri lalu tonton ulang.
- Perbaiki tiap kali.
  Dengan cara ini, kamu bisa mengenali pola bicara, kata-kata pengisi ("emm", "eh"), dan memperbaikinya.

Kasus 5: Ayu Tidak Bisa Menjelaskan Proyek ke Orang Awam

Ilustrasi Kasus:
Ayu membuat sistem canggih dengan Python dan TensorFlow. Tapi saat ditanya oleh pewawancara non-teknis: “Bisa dijelaskan proyek kamu?” Ayu malah bicara tentang “hyperparameter tuning” dan “relu activation” yang membingungkan.

Tantangannya:
Komunikasi teknis ke audiens awam.

Solusi:
Sampaikan dengan **cerita sederhana**.
Contoh:
“Proyek saya mirip seperti Google Lens, tapi khusus untuk buah-buahan. Saya membuat sistem yang bisa mengenali jenis buah hanya dari fotonya.”

Kasus 6: Jawaban STAR untuk Tantangan Proyek

Ilustrasi Kasus:
Kamu ditanya: “Ceritakan tantangan yang kamu hadapi dalam proyek tim.”
Alih-alih menjawab asal, kamu pakai metode STAR:

- **Situation:** “Kami membuat proyek pendeteksi kualitas udara.”
- **Task:** “Sensor utama tidak berfungsi dan saya harus memastikan data tetap akurat.”
- **Action:** “Saya cek kabel, konsultasi guru, dan menulis skrip filter data.”
- **Result:** “Model berjalan baik dan kami dapat nilai A.”

Tantangannya:
Menceritakan pengalaman secara **terstruktur dan meyakinkan**.

Solusi:
Latih metode **STAR** – ini kunci jawaban sukses wawancara!

Contoh Soal dan Pembahasan

Soal:

Kamu baru saja menyelesaikan sesi simulasi wawancara dan mendapat kritik bahwa jawabanmu tentang kelebihan dan kekurangan belum meyakinkan. Nah, kalau kamu dapat pertanyaan itu lagi, strategi apa yang akan kamu gunakan agar jawabanmu lebih kuat dan profesional?

Pembahasan:
Untuk menjawab bagian *kelebihan*, kaitkan langsung dengan kebutuhan posisi yang kamu lamar. Misalnya: “Saya memiliki kemampuan komunikasi yang baik, yang saya gunakan saat menjadi ketua panitia acara sekolah.”
Untuk *kelemahan*, pilih yang tidak fatal tapi menunjukkan kesadaran diri, lalu beri solusi: “Saya dulu terlalu perfeksionis, tapi sekarang saya belajar menyusun prioritas agar lebih efisien.”

Soal:

Bayangkan kamu melamar sebagai Junior AI Developer. Tiba-tiba pewawancara bertanya soal teknis: ‘Apa perbedaan klasifikasi dan regresi dalam Machine Learning?’ Bagaimana kamu akan menjelaskan dengan bahasa yang jelas tapi tetap profesional?

Pembahasan:
Klasifikasi = memprediksi *kategori* (contoh: spam atau bukan spam).
Regresi = memprediksi *angka atau nilai kontinu* (contoh: suhu, harga rumah).
Gunakan contoh kehidupan sehari-hari agar lebih mudah dipahami pewawancara.

Soal:

Menurutmu, kenapa banyak orang menyarankan untuk melakukan ***mock interview*** sebelum wawancara kerja yang sebenarnya? Apa keuntungannya?

Pembahasan:
Mock interview membantu mengenali kelemahan diri, meningkatkan rasa percaya diri, dan membiasakan diri dengan tekanan wawancara. Dengan latihan, kamu bisa menjawab lebih tenang dan terstruktur saat momen sesungguhnya datang.

Soal:

Pewawancara memberi pertanyaan pengalaman: 'Ceritakan saat kamu menghadapi tantangan besar.' Agar jawabannya jelas dan memikat, kamu bisa pakai **STAR Method**. Apa itu STAR dan bagaimana cara kerjanya?

Pembahasan:

- **S**ituation: Apa kondisinya?
- **T**ask: Apa tanggung jawabmu?
- **A**ction: Apa yang kamu lakukan?
- **R**esult: Hasilnya apa?

Contoh jawaban dengan STAR akan menunjukkan bahwa kamu bisa berpikir runtut dan reflektif.

Soal:

Pertanyaan klasik dalam wawancara: “Ceritakan tentang dirimu.” Bagaimana caranya menjawab agar tidak terdengar seperti baca biodata, tapi tetap menarik dan profesional?

Pembahasan:
Jangan ceritakan seluruh hidupmu. Cukup 3 hal: latar belakang singkat, pengalaman relevan, dan motivasimu untuk posisi itu. Misalnya: “Saya siswa jurusan IPA yang suka teknologi. Saya aktif dalam klub robotik, dan ingin mengembangkan diri di bidang AI karena merasa tertantang memecahkan masalah nyata.”

Soal:

Tiba-tiba kamu diberi pertanyaan yang tidak kamu tahu jawabannya. Apa yang harus kamu lakukan agar tetap terlihat tenang dan profesional?

Pembahasan:
Jangan panik! Akui dengan sopan: “Saya belum yakin dengan jawabannya, tapi saya akan mencoba mendekatinya dengan logika...”
Menunjukkan cara berpikir lebih penting daripada jawaban sempurna. Ini mencerminkan kemampuan berpikir kritis dan keinginan belajar.

Fakta Menarik / Fun Facts

- "Beberapa perusahaan besar seperti Unilever dan HireVue bahkan menggunakan AI untuk melakukan wawancara awal atau menganalisis ekspresi wajah kandidat!"
- "Rata-rata wawancara kerja hanya berlangsung sekitar 30-45 menit, jadi setiap jawaban Anda sangat berharga!"
- Beberapa perusahaan besar **sudah menggunakan AI** sebagai pewawancara tahap awal!
- Pewawancara suka jika kamu **bertanya balik** tentang perusahaan. Itu menunjukkan kamu serius.

Tips Belajar atau Tips Cepat Menghafal

Mind Mapping "Persiapan Wawancara"



- **Pertanyaan wawancara umum:** "Ceritakan tentang diri Anda, Kenapa tertarik posisi ini, Kelebihan/Kelemahan, Ceritakan Proyek Terbaik, Kenapa Anda adalah yang terbaik untuk posisi ini?"
- Melatih menjawab pertanyaan teknis dengan menjelaskan konsep secara sederhana seolah Anda sedang mengajarkan kepada teman.
- Siapkan 2-3 pertanyaan untuk diajukan kepada pewawancara di akhir.

Tips Cepat Menghafal

- Buat singkatan lucu: **"CeKaPro-KeKaAI"** → Ceritakan diri, Kekuatan, Proyek, Kelemahan, Kenapa AI?

- Gunakan metode **mind mapping**.

Cerita

"Saya pernah membuat chatbot untuk membantu siswa mencari soal matematika.
Saya belajar cara kerjanya, mencoba banyak kode, gagal beberapa kali, lalu berhasil.
Itu pengalaman yang bikin saya ingin terus belajar AI."

Aktivitas Siswa

Aktivitas 1: Latihan Wawancara Simulatif dengan AI (Mock Interview with ChatGPT)

Deskripsi Aktivitas:
Siswa diajak untuk melakukan simulasi wawancara kerja secara individu menggunakan ChatGPT atau AI sejenis. Latihan ini bertujuan untuk melatih kepercayaan diri dan kemampuan menjawab pertanyaan dengan tepat.

Langkah-langkah Saran:

- Buka ChatGPT dan ketik prompt:
  "Halo ChatGPT, saya ingin latihan wawancara untuk posisi Junior AI Developer."
- Jawablah pertanyaan yang diberikan seolah-olah sedang berada di wawancara sungguhan.
- Catat pertanyaan-pertanyaan penting dan refleksikan jawaban yang sudah diberikan.
- Jika memungkinkan, rekam layar atau suara agar bisa ditinjau ulang untuk evaluasi diri.

Aktivitas 2: Interview Roleplay Seru Bersama Teman

Deskripsi Aktivitas:
Ajak temanmu untuk bermain peran sebagai pewawancara dan kandidat. Ini bukan sekadar bermain—ini latihan yang menyenangkan dan menantang!

Langkah-langkah Saran:

- Tentukan peran: siapa pewawancara, siapa kandidat.
- Gunakan pertanyaan wawancara yang nyata dan relevan dengan dunia kerja (contoh: "Apa tantangan terbesar dalam proyek AI yang pernah kamu kerjakan?").
- Lakukan simulasi secara serius namun tetap santai.
- Setelah selesai, tukar peran agar keduanya mendapatkan pengalaman dari dua sisi.

Aktivitas 3: Berbagi dan Menerima Umpan Balik Konstruktif (Peer Feedback Session)

Deskripsi Aktivitas:
Setelah melakukan latihan wawancara, siswa didorong untuk saling memberi dan menerima masukan dari teman sekelas agar dapat berkembang bersama.

Langkah-langkah Saran:

- Bentuk kelompok kecil (2–4 orang) dan diskusikan pengalaman wawancara.
- Ceritakan tantangan yang dihadapi, jawaban yang dirasa berhasil, dan hal yang perlu ditingkatkan.
- Berikan komentar yang membangun—gunakan kalimat seperti, *“Akan lebih meyakinkan kalau kamu tambahkan contoh nyata.”*
- Tujuannya adalah belajar dari satu sama lain, bukan menilai.

Aktivitas 4: Tantangan Video: Jawaban Interview dalam 2 Menit

Deskripsi Aktivitas:
Buat video pendek berdurasi maksimal 2 menit yang menjawab satu pertanyaan wawancara secara menarik dan jelas.

Langkah-langkah Saran:

- Pilih satu pertanyaan, misalnya: *“Ceritakan proyek AI yang pernah kamu buat atau pelajari.”*
- Rancang skrip sederhana: pembuka yang percaya diri, isi yang informatif, dan penutup yang meyakinkan.
- Rekam dengan kamera ponsel—tidak harus sempurna, yang penting jelas dan antusias.
- Tonton ulang untuk melihat ekspresi, kejelasan suara, dan isi jawaban.

Aktivitas 5: Game Cepat: Kuis Interaktif STAR Method

Deskripsi Aktivitas:
Uji pemahaman tentang teknik menjawab wawancara menggunakan STAR (Situation, Task, Action, Result) dengan game coding sederhana.

Langkah-langkah Saran:

- Jalankan kode kuis mini ini di Python:

```
def quiz_star():
    q = input("Apa huruf pertama dari STAR Method dan artinya? ")
    if q.lower() == "situation":
        print("Betul! STAR dimulai dari Situation.")
    else:
        print("Coba lagi! Huruf pertama adalah 'S'.")
```

```
quiz_star()
```

- Setelah menjawab, diskusikan dengan teman bagaimana menggunakan metode STAR untuk menjawab pertanyaan berbasis pengalaman.
- Tambahkan pertanyaan lanjutan untuk huruf T, A, dan R jika ingin menjadikan ini tantangan kelompok!

Rangkuman

Menggunakan **simulasi wawancara kerja berbasis AI** seperti *ChatGPT* merupakan **strategi yang sangat efektif** dalam mempersiapkan siswa menghadapi dunia kerja, terutama di bidang **teknologi dan kecerdasan buatan (AI)**. Melalui simulasi ini, siswa dapat **melatih kemampuan menjawab pertanyaan teknis, perilaku, dan umum**, sekaligus **meningkatkan kepercayaan diri serta kemampuan komunikasi**.

Latihan ini diperkuat dengan penerapan metode ***STAR*** (*Situation, Task, Action, Result*), yang membantu siswa menyusun jawaban dengan **struktur yang jelas dan meyakinkan**. Selain itu, siswa juga didorong untuk **memahami pentingnya riset perusahaan**, **menjelaskan proyek-proyek mereka secara efektif**, serta **menerima dan belajar dari umpan balik** yang diberikan selama simulasi.

Dengan pendekatan menyeluruh ini, siswa tidak hanya siap dari sisi **kompetensi teknis**, tetapi juga dari aspek **mental dan profesional**, sehingga mereka lebih percaya diri dan siap memasuki dunia kerja yang sesungguhnya.

Latihan Soal

A. Benar atau Salah (5 Soal)

1. Tujuan utama wawancara kerja adalah menguji kemampuan mengetik kandidat.
   Jawaban: Salah

2. STAR Method digunakan untuk menjawab pertanyaan teknis dalam wawancara.
   Jawaban: Salah

3. Salah satu jenis pertanyaan dalam wawancara adalah studi kasus.
   Jawaban: Benar

4. Riset tentang perusahaan tidak terlalu penting sebelum wawancara.
   Jawaban: Salah

5. ChatGPT bisa digunakan untuk melakukan simulasi wawancara kerja.
   Jawaban: Benar

B. Pilihan Ganda (10 Soal)

1. Apa tujuan utama wawancara kerja?
   A. Menguji nilai rapor
   B. Menilai kesesuaian dengan perusahaan
   C. Menilai kemampuan bahasa Inggris
   D. Menentukan nilai ujian akhir
   Jawaban: B

2. Berikut ini yang **bukan** termasuk jenis pertanyaan wawancara:
   A. Pertanyaan teknis
   B. Pertanyaan perilaku
   C. Pertanyaan lisan
   D. Pertanyaan umum
   Jawaban: C

3. Apa fungsi dari STAR Method?
   A. Membuat CV
   B. Menjawab pertanyaan tentang gaji
   C. Menyusun jawaban perilaku
   D. Mendesain proyek AI
   Jawaban: C

4. Komponen pertama dari STAR Method adalah:
   A. Situation
   B. Task
   C. Action
   D. Result
   Jawaban: A

5. Contoh pertanyaan umum dalam wawancara adalah:
   A. Bagaimana cara kerja backpropagation?
   B. Ceritakan tentang diri Anda
   C. Buat model klasifikasi
   D. Hitung akurasi model
   Jawaban: B

6. Mengapa penting menyiapkan contoh proyek saat wawancara AI?
   A. Untuk menyesuaikan gaji
   B. Menunjukkan kemampuan konkret
   C. Untuk menyenangkan pewawancara
   D. Agar tidak perlu wawancara
   Jawaban: B

7. Sikap tubuh yang baik saat wawancara meliputi:
   A. Menunduk terus
   B. Menghindari tatapan
   C. Duduk tegak dan senyum
   D. Mengetuk-ngetuk meja

Jawaban: C

8. Salah satu keuntungan menggunakan AI dalam simulasi wawancara adalah:
   A. Tidak perlu belajar
   B. Bisa berbohong lebih mudah
   C. Mendapatkan umpan balik langsung
   D. Menyingkat CV
   Jawaban: C

9. Dalam wawancara teknis, pewawancara biasanya ingin tahu:
   A. Nilai rapor
   B. Cerita masa kecil
   C. Pemahaman konsep AI
   D. Jenis makanan favorit
   Jawaban: C

10. Yang **harus dihindari** saat wawancara kerja:
    A. Tersenyum
    B. Memberi contoh proyek
    C. Menghindari kontak mata
    D. Menjawab dengan STAR
    Jawaban: C

C. Isian Singkat (5 Soal)

1. Apa arti singkatan dari STAR dalam wawancara?
   Jawaban: Situation, Task, Action, Result

2. Sebutkan dua jenis pertanyaan dalam wawancara kerja selain pertanyaan umum!
   Jawaban: Pertanyaan teknis, pertanyaan perilaku

3. Mengapa penting melakukan riset tentang perusahaan sebelum wawancara?
   Jawaban: Agar memahami budaya dan kebutuhan perusahaan

4. Apa salah satu manfaat melakukan mock interview menggunakan AI?
   Jawaban: Mendapatkan umpan balik dan latihan yang realistis

5. Sebutkan satu contoh pertanyaan teknis dalam wawancara AI!
   Jawaban: Apa perbedaan supervised dan unsupervised learning?

D. Soal Explorasi

Soal:

Bayangkan kamu sedang menghadapi pertanyaan: “Ceritakan tentang saat kamu mengalami kegagalan dalam proyek dan bagaimana kamu menghadapinya.” Tulis jawabanmu dengan menggunakan STAR Method dalam 1 paragraf.

Soal:

Kamu telah menyelesaikan sebuah proyek klasifikasi gambar menggunakan model CNN sederhana. Jelaskan bagaimana kamu akan mempresentasikan proyek ini kepada pewawancara yang mungkin tidak terlalu teknis, agar tetap mudah dipahami.

Soal:

Setelah simulasi interview dengan ChatGPT, kamu mendapat kritik bahwa jawabanmu terlalu umum dan tidak spesifik. Tulis satu strategi yang bisa kamu terapkan untuk membuat jawabanmu lebih kuat dan meyakinkan.

Tugas Proyek

Proyek 1: Simulasi Wawancara Bersama AI

Tujuan: Melatih refleksi diri dan keterampilan wawancara.
Tantangan: Menggunakan AI sebagai cermin kekuatan dan area perbaikan.

Langkah-langkah:

- Gunakan ChatGPT atau platform AI lainnya untuk melakukan simulasi wawancara.
- Jawablah minimal 5 pertanyaan umum wawancara (misalnya: “Apa kelebihanmu?”, “Ceritakan tentang dirimu.”).
- Catat dan analisis respons AI terhadap jawabanmu.
- Tuliskan ringkasan berisi **3 kekuatan utama** dan **3 area yang perlu diperbaiki** berdasarkan masukan dari AI.
- Buat rencana singkat untuk mengembangkan area yang perlu ditingkatkan.

Proyek 2: Video Presentasi AI (3 Menit Tantangan!)

Tujuan: Melatih kemampuan presentasi dan komunikasi yang meyakinkan.
Tantangan: Menjelaskan topik kompleks dengan cara sederhana dan menarik.

Langkah-langkah:

- Pilih proyek AI yang pernah kamu pelajari, buat, atau minati.
- Rancang skrip singkat (maks. 3 menit) dengan struktur: pengenalan, inti, dan penutup.
- Gunakan bahasa yang mudah dipahami dan ekspresi percaya diri.
- Rekam video menggunakan ponsel atau laptop, pastikan suara dan gambar jelas.
- Tampilkan videomu di kelas atau unggah ke platform yang disepakati.

Proyek 3: Tulisan Wawancara STAR Style

Tujuan: Mengasah kemampuan menjawab wawancara secara terstruktur.
Tantangan: Menggunakan metode STAR (Situation, Task, Action, Result) secara efektif.

Langkah-langkah:

- Tulis jawabanmu untuk 3 pertanyaan ini:
    1. Ceritakan tentang dirimu.
    2. Apa kelebihan dan kekuranganmu?
    3. Mengapa kamu tertarik pada bidang AI?
- Gunakan format STAR bila memungkinkan, terutama untuk menggambarkan pengalamanmu.
- Tinjau tulisanmu bersama guru atau teman untuk mendapatkan masukan.

Proyek 4: Roleplay Interview Bersama Teman

Tujuan: Melatih simulasi wawancara dan menerima umpan balik dari sesama.
Tantangan: Meningkatkan respons langsung dan ekspresi non-verbal.

Langkah-langkah:

- Buat daftar minimal **5 pertanyaan wawancara** untuk posisi “Junior AI Developer”.
- Lakukan simulasi wawancara secara berpasangan. Bergiliran sebagai pewawancara dan pelamar.
- Temanmu memberikan umpan balik tertulis terhadap kekuatan dan hal yang bisa kamu tingkatkan.
- Refleksikan hasilnya dan tentukan langkah perbaikan pribadi.

Proyek 5: Poster/Slide: “5 Hal Penting dalam Wawancara AI”

Tujuan: Menyusun panduan visual yang membantu pelamar AI.
Tantangan: Mengubah informasi penting menjadi desain yang menarik dan mudah diingat.

Langkah-langkah:

- Riset topik: Apa saja tips penting saat wawancara kerja di bidang AI?
- Pilih **5 poin utama** (misalnya: riset perusahaan, tunjukkan portofolio AI, latihan STAR Method, dsb.).
- Buat dalam bentuk **poster digital/manual** atau **slide presentasi**.
- Gunakan visual menarik, ikon, atau warna untuk memperjelas pesan.
- Presentasikan di kelas atau pajang hasil karya di dinding kelas.

Proyek 6: Rencana Perbaikan Pribadi dari Mock Interview

Tujuan: Menumbuhkan kesadaran diri dan kemampuan evaluasi.
Tantangan: Menyusun rencana konkret berdasarkan hasil wawancara.

Langkah-langkah:

- Lakukan **mock interview** (simulasi wawancara) dengan AI atau guru.
- Catat jawaban dan observasi selama proses wawancara.

- Buat laporan berisi:
    - 3 kekuatan yang ditunjukkan.
    - 3 aspek yang perlu ditingkatkan.
    - Rencana tindak lanjut (misalnya: latihan menjawab, memperbaiki kontak mata, dsb.).

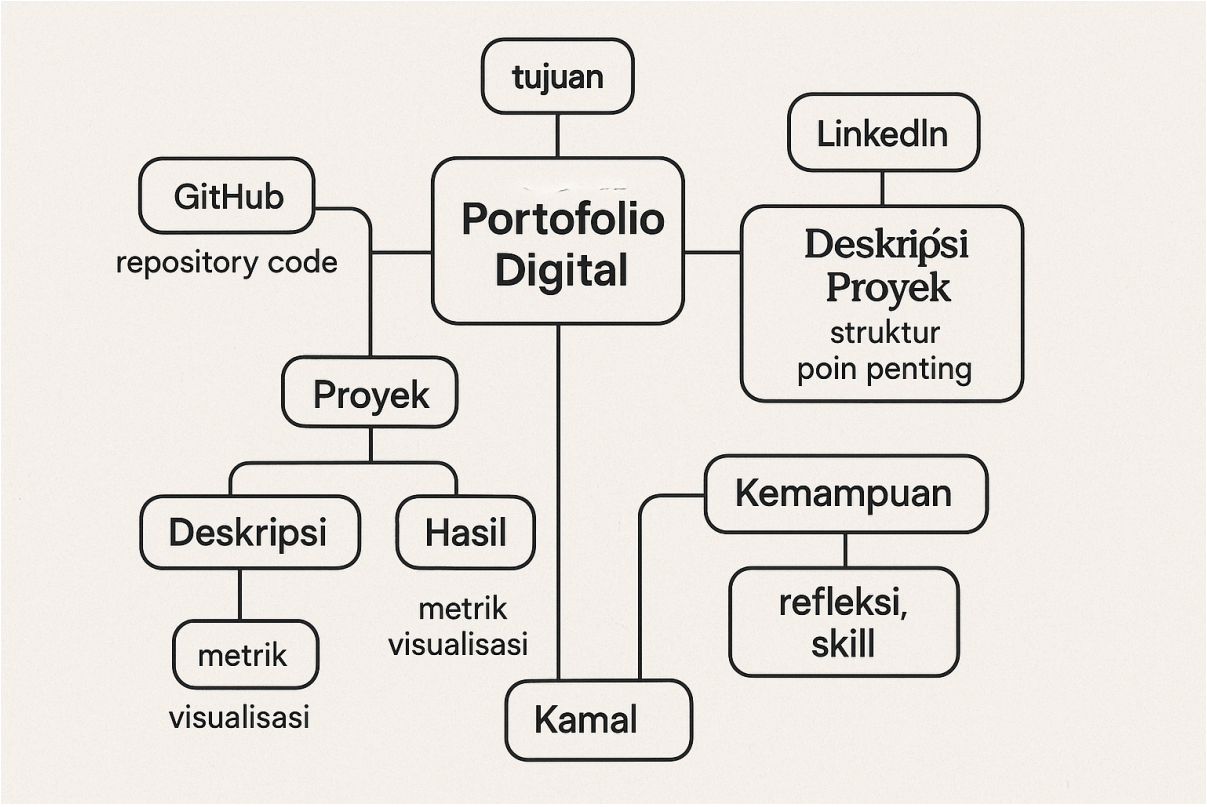
BAB 7: Portofolio Digital AI Siswa

Tujuan Pembelajaran

Setelah mempelajari bab ini, siswa mampu:

- Menyusun portofolio digital AI yang menarik, rapi, dan profesional.
- Menampilkan proyek AI sebagai bukti keterampilan nyata.
- Menggunakan platform digital seperti GitHub dan LinkedIn untuk personal branding.
- Menjelaskan proyek secara jelas, runtut, dan kreatif.
- Memahami pentingnya membangun jejak digital sejak dini untuk masa depan.

Peta Konsep



Apersepsi

Sebelum kita mulai, coba bayangkan kamu sedang melihat portofolio digital milik mahasiswa atau profesional muda di bidang AI. Keren banget! Mereka tidak hanya menampilkan hasil proyek, tapi juga menceritakan proses belajarnya dengan visual yang menarik dan jelas. Nah, sekarang bayangkan kalau kamu sudah membuat proyek AI hebat—misalnya chatbot, analisis data, atau sistem rekomendasi—tapi tidak ada yang tahu. Sayang banget, kan? Maka dari itu, penting banget punya portofolio sebagai bukti nyata dari apa yang sudah kamu pelajari dan kerjakan, bukan hanya nilai di rapor. Sekarang, tantangannya adalah:

berani tunjukkan hasil karyamu? Ayo, mulai bangun portofolio digitalmu hari ini! Ini bukan sekadar tugas sekolah—ini adalah langkah awal agar dunia tahu potensi dan kemampuan hebat yang kamu miliki!

Penjelasan Konsep (teori)

Mengapa Portofolio Digital Itu Penting?

Bayangkan kamu baru saja menyelesaikan sebuah proyek AI keren—misalnya, membuat chatbot, menganalisis data cuaca, atau membangun sistem rekomendasi film. Tapi… siapa yang tahu kamu pernah melakukannya? Di sinilah **portofolio digital** berperan penting: menjadi **etalase karya dan kemampuanmu** di dunia nyata.

Portofolio digital bukan sekadar tempat menyimpan proyek, tapi juga alat untuk **menunjukkan siapa dirimu sebagai pembelajar dan calon profesional**. Nilai di rapor itu penting, tapi **'bukti nyata' dalam bentuk proyek akan lebih berbicara banyak**!

Apa Saja Manfaat Membuat Portofolio Digital?

Berikut alasan mengapa kamu perlu mulai membangun portofolio dari sekarang:

- **Bukti Keterampilan Nyata**
  Tidak hanya teori, portofolio menunjukkan bahwa kamu *benar-benar bisa melakukannya*.
- **Meningkatkan Peluang di Masa Depan**
  Portofolio yang rapi dan menarik bisa membuat kamu dilirik oleh universitas, beasiswa, magang, bahkan perusahaan!
- **Personal Branding**
  Bangun identitasmu sebagai pelajar yang kreatif, cerdas, dan serius menekuni AI.
- **Refleksi Diri**
  Kamu bisa melihat sejauh mana kamu sudah belajar, tumbuh, dan berkembang.

Platform Populer untuk Portofolio AI

Kamu bisa memilih platform yang paling cocok dengan gaya dan kebutuhanmu. Berikut beberapa pilihan:

GitHub – Untuk Proyek Berbasis Kode

> GitHub seperti “lemari digital” tempat kamu menyimpan dan menunjukkan kode-kodemu.

- **Repository**: Folder proyekmu.
- **Commit**: Riwayat perubahan dalam kode.
- **Pull Request**: Kolaborasi dan kontribusi ke proyek orang lain.

- **README.md**: Halaman utama repositori yang menjelaskan isi proyekmu. Harus jelas, singkat, dan informatif!

Tips: Buat struktur folder yang rapi: `kode`, `data`, `model`, dan file `README.md`.

LinkedIn – Untuk Profil Profesional

Platform ini seperti “CV digital” tempat kamu bisa:

- Menuliskan pengalaman belajar dan proyek-proyek AI.
- Terhubung dengan para profesional di bidang teknologi.
- Menambahkan proyek ke dalam profil sebagai “highlight”.

Tips: Gunakan bahasa yang profesional tapi tetap jelas. Fokus pada dampak proyek dan keterampilan yang kamu gunakan.

Situs Pribadi (Opsional tapi Keren!)

Bisa dibangun di platform seperti **Google Sites, Wix, atau WordPress.com**.

- **Kelebihan:** Kamu punya kontrol penuh atas desain dan isinya.
- **Tampilan:** Buat sederhana tapi profesional. Gunakan gambar, ikon, dan tautan dengan baik.

Cara Menjelaskan Proyek dalam Portofolio

Sebuah proyek AI akan lebih *berbicara* jika kamu bisa menjelaskannya dengan **narasi yang menarik dan jelas**. Gunakan struktur berikut:

1. **Judul Proyek**
   Singkat, tapi langsung menarik perhatian.
2. **Tujuan Proyek (Problem Statement)**
   Masalah apa yang kamu coba selesaikan?
3. **Metode / Pendekatan**
   Tools, algoritma, atau model AI apa yang kamu gunakan?
4. **Data yang Digunakan**
   Dari mana sumber datanya? Seberapa besar? Apa karakternya?
5. **Hasil dan Analisis**
   - Tampilkan **grafik, diagram, atau screenshot**.
   - Jelaskan metrik sederhana seperti *akurasi* atau *presisi*.
   - Apa *insight* atau kesimpulan utama dari proyekmu?
6. **Peranmu dalam Tim (jika bekerja kelompok)**
   Ceritakan dengan jujur kontribusimu—apakah kamu mengurus coding, data, desain, atau dokumentasi?
7. **Tantangan & Solusi**
   Masalah apa yang kamu hadapi dan bagaimana kamu menyelesaikannya?
8. **Pembelajaran**
   Apa yang kamu pelajari? Bisa keterampilan teknis, komunikasi, atau kerja tim.

9. **Langkah Selanjutnya**
   Jika kamu mengembangkan proyek ini lagi, apa yang ingin kamu tambahkan?

Tips untuk Portofolio AI yang Keren dan Profesional

- **Kualitas Lebih Penting dari Kuantitas**
  Pilih beberapa proyek terbaikmu dan tampilkan dengan detail.
- **Update Secara Berkala**
  Tambahkan proyek baru dan perbaiki yang lama. Dunia AI terus berkembang!
- **Pastikan Portofolio Mudah Diakses**
  Tautan jangan rusak. Pastikan portofolio bisa dibuka di laptop dan HP.
- **Ceritakan Kisahmu**
  Jangan hanya tampilkan teknis. Ceritakan *perjalananmu* dalam proyek tersebut!

Siap Unjuk Diri?

Membangun portofolio digital adalah **petualangan kreatif sekaligus investasi masa depanmu**. Jangan tunggu “nanti” atau “kalau sudah jago”. Justru dengan memulai sekarang, kamu akan makin cepat berkembang.

Yuk, mulai dokumentasikan perjalanan AI-mu hari ini. Dunia diluar sana ingin tahu apa yang bisa kamu buat!

Ilustrasi: Perjuangan Membangun Portofolio yang Keren

Bayangkan kamu adalah siswa bernama Fajar. Sejak kelas 10, kamu tertarik dengan AI. Kamu sudah membuat beberapa proyek: chatbot sederhana, sistem rekomendasi lagu, dan klasifikasi emosi. Tapi… semua itu hanya tersimpan di laptop.

Ketika Fajar ingin mendaftar program AI untuk remaja internasional, dia diminta **portofolio digital**. Panik? Pasti. Tapi itu awal dari perjuangan!

Fajar mulai belajar GitHub, membuat akun LinkedIn, dan mengumpulkan semua proyeknya. Awalnya bingung menulis deskripsi proyek: harus tulis apa sih? Tapi dengan panduan dan masukan guru, akhirnya portofolionya rampung. Ia menulis judul menarik, deskripsi singkat, data yang digunakan, hasil visual, hingga tantangan yang dihadapi.

Beberapa bulan kemudian, Fajar diterima di program AI impiannya. Semua karena **portofolio yang menceritakan siapa dirinya** — bukan hanya nilainya, tapi juga perjalanannya.

Contoh Kasus atau Ilustrasi

Kasus 1: Petualangan Alya – Dari Nol Jadi Developer Publik

Ilustrasi:
Alya punya proyek AI keren: sistem pengenal wajah! Tapi... semua kodenya masih diam di laptop, tak pernah melihat dunia luar. Suatu hari, dia ingin menunjukkan proyeknya ke kampus impian—lalu sadar, dia belum punya akun GitHub!

Tantangan Alya:

- Membuat akun GitHub
- Belajar cara membuat repository
- Upload file proyeknya
- Menulis README pertamanya

Pesan Penting:
Alya belajar bahwa membagikan karyanya secara publik bukan hanya soal "upload kode", tapi soal menunjukkan kualitas, niat, dan kemampuan berkomunikasi sebagai developer.

Kasus 2: Proyek Budi – Judul yang Menggugah atau Nggak Jelas Sama Sekali?

Ilustrasi:
Budi membuat proyek klasifikasi gambar menggunakan AI. Judul awalnya: “Proyek AI Klasifikasi Gambar.” Simpel, tapi... sangat umum. Seperti judul tugas mingguan.

Setelah belajar cara menulis deskripsi proyek yang baik, Budi ubah menjadi:

> *"Mengembangkan model CNN sederhana untuk klasifikasi buah-buahan lokal (apel, mangga, jeruk) menggunakan dataset dari Kaggle dengan akurasi 87%."*

Tantangan Budi:

- Membuat deskripsi yang **spesifik**, **menarik**, dan **informatif**
- Menunjukkan apa yang membuat proyeknya *unik*

Pelajaran Utama:
Judul dan deskripsi yang kuat membuat proyekmu mudah ditemukan, mudah dipahami, dan lebih *stand out*!

Kasus 3: Citra Si Kolaborator – Tapi Kontribusinya Mana?

Ilustrasi:
Citra kerja kelompok bikin chatbot AI. Proyeknya sukses, tapi di GitHub dia cuma

tulis: “Proyek chatbot AI bersama teman.” Akhirnya gurunya tanya, “Kamu ngapain, Cit?”

Setelah berpikir, Citra menulis ulang:

> *"Saya mengembangkan algoritma intent matching dan menguji chatbot menggunakan data percakapan remaja Indonesia."*

Tantangan Citra:

- Menjelaskan **kontribusi pribadi** dalam proyek kelompok
- Menunjukkan keahlian spesifik yang ia kuasai

Nilai Plus:
Portofolio bukan cuma soal hasil akhir, tapi **cerita tentang proses dan peranmu** di dalamnya.

Kasus 4: Misteri Proyek Dito – Bagus, Tapi Sepi Bukti

Ilustrasi:
Dito bikin proyek AI yang hasilnya keren, tapi semua hanya ditulis dalam teks. Tidak ada screenshot, tidak ada grafik, tidak ada contoh output. Seperti novel tanpa gambar—membosankan!

Setelah diberi masukan, ia tambahan:

- Gambar grafik hasil klasifikasi
- Link ke demo aplikasi
- Contoh input-output

Tantangan Dito:

- Menambahkan **visualisasi**, **grafik**, atau **demo**
- Membuat proyeknya *bisa dirasakan* meskipun hanya dari README

Kesimpulan:
Visualisasi membuat proyekmu **lebih hidup**. Orang lain tak hanya membaca, tapi **merasakan** dampaknya.

Kasus 5: Seni Menulis README – Contoh Lengkap Proyek “Sistem Rekomendasi Film”

Ilustrasi:
Bayangkan kamu sedang mengunjungi repositori GitHub untuk melihat proyek "Sistem Rekomendasi Film Sederhana". Kamu menemukan README seperti ini:

Isi README-nya:

- Judul proyek

- Deskripsi singkat: *"Sistem rekomendasi menggunakan content-based filtering sederhana untuk film."*
- Fitur: Pencarian film, rekomendasi berdasarkan genre, simpan favorit
- Instalasi & cara menjalankan
- Contoh penggunaan (dengan cuplikan kode)
- Struktur file & folder
- Teknologi: Python, Pandas, Scikit-learn
- Kontributor
- Lisensi

Tantangan:

- Merancang README seperti ini sendiri
- Menyusun dengan struktur yang jelas dan profesional
- Menyisipkan bagian “Apa yang Saya Pelajari”

Kasus 6: Refleksi Diri dalam Proyek – Belajar dari Pengalaman

Ilustrasi:
Setelah menyelesaikan proyek AI, kamu menulis bagian refleksi:

> *"Dari proyek ini, saya belajar pentingnya membersihkan data sebelum melatih model. Saya juga meningkatkan kemampuan debugging dan belajar kerja tim."*

Tantangan untuk Kamu:

- Menuliskan pelajaran yang kamu dapat, bukan cuma hasil akhir
- Menunjukkan proses berkembangnya kemampuanmu

Kenapa Penting?
Orang lain tidak hanya ingin tahu apa yang kamu buat, tapi **apa yang kamu pelajari** dan **bagaimana kamu berkembang**.

Ingin mengembangkan proyekmu jadi portofolio yang *keren, profesional, dan meyakinkan*? Mulailah dari membuat README yang bukan hanya informatif, tapi juga **mencerminkan kamu sebagai pembelajar aktif**.

Contoh Soal dan Pembahasan

Soal: "Deskripsi Kilat, Dampak Dahsyat!"

Bayangkan kamu baru menyelesaikan proyek AI keren, tapi hanya punya 5 kalimat untuk menjelaskannya di portofolio. Informasi apa saja yang wajib kamu sampaikan agar pembaca langsung paham nilai dari proyekmu?

Pembahasan:
Fokuslah pada inti dari proyekmu:

1. **Masalah** apa yang ingin kamu pecahkan.
2. **Solusi AI** atau pendekatan teknis yang kamu terapkan.
3. **Hasil utama** – bisa berupa akurasi, performa, atau visualisasi.
4. **Pelajaran penting** yang kamu dapatkan selama proses.
5. Tambahkan **tautan ke kode atau demo**, agar pembaca bisa langsung mencoba.

Soal: "Tantangan = Nilai Tambah?"

Mengapa menyebutkan tantangan dan solusi dalam proyek bisa menjadi hal yang membuat portofolio lebih menonjol?

Pembahasan:
Menceritakan tantangan menunjukkan bahwa kamu bukan hanya menyalin kode dari tutorial, tapi juga **menghadapi hambatan nyata**. Ini menampilkan **kemampuan problem-solving, berpikir kritis, dan daya juangmu**. Perusahaan atau kampus akan lebih menghargai orang yang bisa mengatasi masalah daripada yang hanya mengikuti langkah tanpa hambatan.

Soal: "Data Bukan Sekadar Angka"

Saat menulis deskripsi proyek AI, mengapa menyebutkan **sumber dan karakteristik data** itu penting?

Pembahasan:
Karena **data adalah bahan bakar utama** dalam proyek AI. Tanpa informasi jelas tentang data—dari mana asalnya, apakah bersih, beragam, atau seimbang—maka **validitas hasil jadi sulit dipercaya**. Ini juga menunjukkan bahwa kamu memahami prinsip etis dan teknis dalam pengolahan data.

Soal: "LinkedIn vs GitHub: Mana Lebih Penting?"

Kamu ingin memamerkan proyek AI-mu. Haruskah kamu memilih GitHub atau LinkedIn sebagai platform utama? Jelaskan bedanya!

Pembahasan:
GitHub dan LinkedIn **berfungsi berbeda tapi saling melengkapi**:

- **GitHub** cocok untuk memamerkan kode, dokumentasi, dan kolaborasi teknis.
- **LinkedIn** lebih untuk membangun **personal branding profesional**, berbagi insight, dan membangun jaringan.

Jadi, tampilkan proyekmu di GitHub, lalu **bagikan highlight-nya di LinkedIn** untuk menjangkau audiens yang lebih luas.

Soal: "Uji Cepat Deskripsi Proyek"

Kamu membuat proyek yang mendeteksi jenis burung dari suara kicauannya. Tulislah **deskripsi singkat maksimal 5 kalimat** yang bisa masuk ke portofolio AI.

Pembahasan (Contoh Jawaban):
Saya membuat model machine learning yang dapat mengenali jenis burung dari rekaman suara. Dataset yang digunakan adalah **Xeno-Canto** yang berisi ribuan sampel kicauan. Model CNN yang saya latih mencapai akurasi **78%**. Saya belajar cara memproses data audio dan menyusun pipeline pelatihan model. Proyek ini saya publikasikan di GitHub lengkap dengan demo.

Soal: "Siapakah Juru Bicara Proyekmu?"

README di GitHub ibarat penyambut tamu di pameran. Apa saja informasi penting yang harus selalu ada di dalamnya?

Pembahasan:
README yang baik harus mencakup:

1. **Judul proyek**
2. **Deskripsi singkat**
3. **Cara instalasi dan menjalankan**
4. **Struktur folder/kode**
5. **Teknologi/alat yang digunakan**
6. **Contoh penggunaan dan hasil**
7. **Apa yang kamu pelajari**

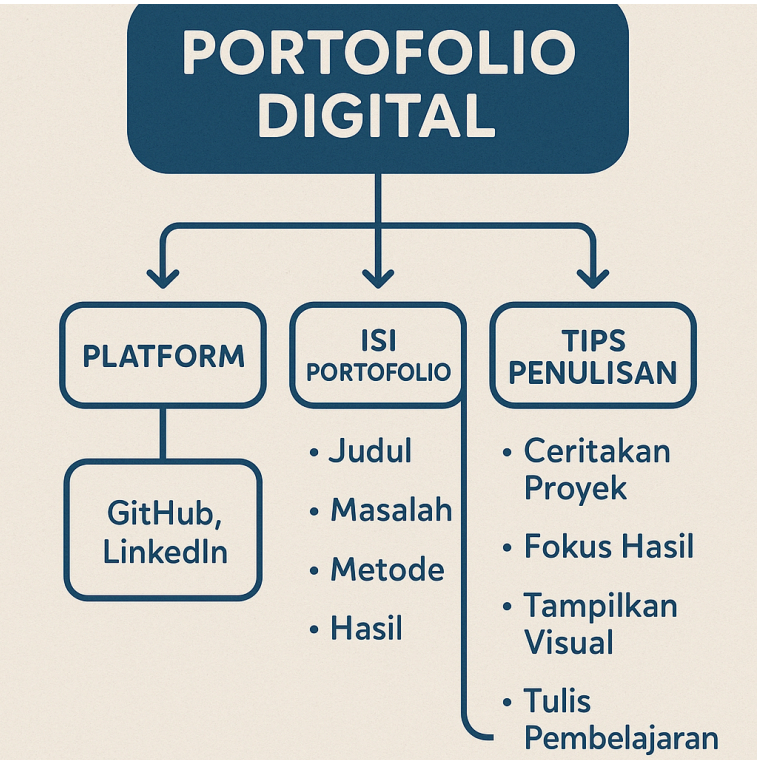
README adalah alat utama untuk **menjelaskan dan mempromosikan proyekmu**, jadi buatlah sejelas dan semenarik mungkin!

Fakta Menarik / Fun Facts

- Banyak recruiter di industri teknologi lebih suka melihat portofolio proyek langsung daripada hanya membaca CV! Portofolio adalah 'bukti' nyata dari keterampilan Anda.
- Banyak beasiswa luar negeri meminta **tautan GitHub atau LinkedIn**, bukan hanya CV!
- Beberapa perusahaan teknologi **tidak mewajibkan ijazah**, asal kamu bisa menunjukkan karya nyata lewat portofolio.
- Elon Musk pernah mengatakan bahwa ia lebih menghargai kemampuan seseorang membuat hal nyata daripada gelar formal, dan portofolio adalah cara terbaik untuk menunjukkan itu.
- Elon Musk berkata: *“Show me what you’ve built.”* Ini berlaku juga buat kalian yang masih SMA!

Tips Belajar atau Tips Cepat Menghafal

Mind Map Portofolio Digital



- **Elemen kunci portofolio:** "Proyek yang bagus, Deskripsi yang jelas, Bukti visual, Pembelajaran yang relevan."
- Anggap portofolio Anda sebagai cerita perjalanan Anda di dunia AI!
- Minta teman atau guru untuk meninjau portofolio Anda sebelum mempublikasikannya.

Cerita Mudah Diingat

Bayangkan proyekmu seperti **film dokumenter pendek**. Ada **judul menarik**, **masalah yang kamu pecahkan**, **alat yang kamu pakai**, dan **ending** berupa hasil dan pembelajaran. Portofolio = cerita hidup proyekmu.

Aktivitas Siswa

Aktivitas 1: Susun Ulang Karya Jadi Portofolio Digital!

Deskripsi Aktivitas:
Mari refleksikan semua karya hebat kalian dari Bab 1–6! Kumpulkan proposal, kode, model, pitch deck, hingga laporan mock interview, lalu susun semuanya dalam bentuk portofolio digital pribadi.

Langkah-langkah yang bisa dilakukan:

- Telusuri kembali semua hasil proyek dari awal.
- Pilih 2–3 proyek terbaik untuk ditampilkan.
- Susun folder rapi (berisi file dan dokumentasi) di laptop atau cloud drive.
- Buat daftar isi sebagai langkah awal membentuk portofolio digital.

Aktivitas 2: Bangun Jejak Profesional: Bikin Akun GitHub & LinkedIn

Deskripsi Aktivitas:
Mulailah membangun identitas profesional secara online! GitHub dan LinkedIn adalah tempat ideal untuk menunjukkan kemampuan dan portofolio.

Langkah-langkah yang bisa dilakukan:

- Jika belum punya, buat akun GitHub dan/atau LinkedIn.
- Tambahkan bio singkat yang mencerminkan minatmu di bidang AI atau teknologi.
- Upload satu proyek ke GitHub dan deskripsikan dengan README sederhana.
- Di LinkedIn, coba buat posting pendek tentang pengalaman saat mengerjakan proyek AI.

Aktivitas 3: Proyek Review Challenge: Tinjau Portofolio Teman

Deskripsi Aktivitas:
Lihat dan bandingkan dua portofolio milik temanmu. Tantang dirimu untuk menganalisis dan memahami apa yang membuat sebuah portofolio terlihat lebih menarik dan efektif.

Langkah-langkah yang bisa dilakukan:

- Pilih dua portofolio dari temanmu (di GitHub atau LinkedIn).
- Lihat struktur, kejelasan, dan daya tarik visualnya.
- Catat kekuatan dan kekurangan masing-masing.
- Simpulkan mana yang menurutmu lebih efektif dan kenapa.

Aktivitas 4: Hack README Day: Upgrade Dokumentasi Proyek Lama

Deskripsi Aktivitas:
Ambil satu proyek lamamu dan ubah README-nya seolah kamu sedang menjelaskan pada orang asing yang baru pertama melihatnya. Ini latihan komunikasi teknis yang sangat penting!

Langkah-langkah yang bisa dilakukan:

- Pilih proyek yang kamu suka, tapi README-nya masih seadanya.
- Tambahkan struktur yang jelas: Judul, Deskripsi, Data, Tools, dan Hasil.
- Gunakan bahasa yang ringkas tapi informatif.

- Coba minta satu temanmu membaca README baru itu, apakah mereka paham?

Aktivitas 5: Buat Portofolio Sederhana di Google Sites

Deskripsi Aktivitas:
Gunakan Google Sites untuk membuat website portofolio pribadi. Tambahkan dua proyek AI yang menurutmu paling keren, lengkap dengan screenshot, link, dan narasi singkat.

Langkah-langkah yang bisa dilakukan:

- Buka Google Sites dan pilih template dasar.
- Buat satu halaman per proyek, sertakan screenshot dan tautan GitHub.
- Tambahkan deskripsi singkat untuk menjelaskan proses dan hasil proyek.
- Publikasikan dan bagikan link ke teman atau guru untuk mendapatkan masukan.

Aktivitas 6: Coding Mini-Challenge: Buat Generator README

Deskripsi Aktivitas:
Tantang dirimu dengan mini coding challenge: buat pseudocode (atau script Python sederhana) yang bisa menghasilkan template README secara otomatis!

Langkah-langkah yang bisa dilakukan:

- Rancang input untuk bagian penting README: judul, deskripsi, data, tools, dan hasil.
- Gunakan bahasa Python atau pseudocode untuk menampilkannya dalam format markdown.
- Jalankan kodenya dan lihat apakah hasilnya siap digunakan di GitHub.
- Tambahkan fitur bonus jika bisa, seperti simpan ke file `.md`!

Rangkuman

Bayangkan kalian bisa menunjukkan semua pencapaian dan keterampilan AI yang kalian miliki, bukan hanya lewat nilai rapor, tapi lewat sesuatu yang *keren* dan profesional—itulah **portofolio digital**! Lewat platform seperti **GitHub**, **LinkedIn**, atau bahkan situs pribadi, kalian bisa mengunggah proyek-proyek AI yang kalian buat. Ini bukan hanya bisa dilihat guru atau teman, tapi juga bisa dilirik oleh **kampus impian**, **recruiter kerja**, atau bahkan **mentor industri**!

Tapi portofolio digital bukan cuma kumpulan hasil kerja—ini adalah **cerita perjalanan kalian di dunia AI**. Dari awal belajar *machine learning*, bikin chatbot, sampai bikin visualisasi data, semua itu bisa kalian tampilkan di satu tempat. Di sinilah kalian bisa menunjukkan

kemampuan coding, berpikir kritis, kerja sama tim, dan juga cara kalian menyampaikan ide lewat **visualisasi yang menarik**.

Agar portofolio kalian benar-benar *stand out*, isi dengan proyek-proyek yang berkualitas, tambahkan **deskripsi yang jelas**, **dokumentasi visual**, dan jangan lupa sertakan **refleksi pribadi**. Ini bukan cuma alat untuk mencari peluang, tapi juga jadi **cermin perkembangan dan semangat belajar kalian**. Jadi, siap tantang diri kalian untuk bikin portofolio yang bikin orang lain bilang: *"Wow, ini siswa SMA yang serius banget di AI!"*

Latihan Soal

A. Benar atau Salah (5 Soal)

1. Portofolio digital hanya berguna jika kamu ingin bekerja di luar negeri.
   Jawaban: Salah

2. GitHub dapat digunakan untuk menyimpan dan membagikan kode proyek AI.
   Jawaban: Benar

3. Deskripsi proyek dalam portofolio tidak perlu menyebutkan data yang digunakan.
   Jawaban: Salah

4. Menampilkan tantangan dan solusi dalam proyek dapat meningkatkan kualitas portofolio.
   Jawaban: Benar

5. LinkedIn hanya digunakan oleh orang yang sudah bekerja, tidak cocok untuk siswa.
   Jawaban: Salah

B. Pilihan Ganda (10 Soal)

6. Tujuan utama dari portofolio digital adalah...
   a. Menyimpan nilai rapor
   b. Menunjukkan bukti keterampilan dan proyek
   c. Mencari hiburan
   d. Menyimpan file pribadi
   Jawaban: b

7. Di bawah ini yang **bukan** platform portofolio digital adalah...
   a. GitHub
   b. TikTok
   c. LinkedIn
   d. Google Sites
   Jawaban: b

8. File `README.md` pada GitHub berfungsi untuk...
   a. Menyimpan data
   b. Menyimpan foto profil
   c. Menjelaskan proyek
   d. Menghapus repositori
   Jawaban: c

9. Komponen yang **harus ada** dalam deskripsi proyek AI adalah, kecuali...
   a. Problem Statement
   b. Harga Proyek
   c. Metodologi
   d. Hasil & Analisis
   Jawaban: b

10. Apa manfaat menambahkan visualisasi seperti grafik atau diagram dalam portofolio?
    a. Untuk memperpanjang isi
    b. Agar lebih berwarna
    c. Untuk menjelaskan hasil proyek secara jelas
    d. Untuk menarik perhatian media sosial
    Jawaban: c

11. Berikut adalah **contoh metrik evaluasi** dalam proyek AI, kecuali...
    a. Akurasi
    b. Presisi
    c. Kecepatan TikTok
    d. Recall
    Jawaban: c

12. Manfaat portofolio digital dalam proses beasiswa adalah...
    a. Mengisi waktu luang
    b. Membantu reviewer melihat bukti kemampuan
    c. Menghindari tugas sekolah
    d. Untuk menyimpan meme
    Jawaban: b

13. Pernyataan yang benar tentang GitHub adalah...
    a. Hanya bisa digunakan oleh programmer profesional
    b. Tidak bisa menampilkan deskripsi proyek
    c. Bisa digunakan untuk kerja kolaboratif
    d. Tidak perlu akun untuk menggunakannya
    Jawaban: c

14. Portofolio yang baik sebaiknya...
    a. Mengandung semua proyek meskipun belum selesai
    b. Mengutamakan kuantitas
    c. Diupdate secara berkala dan mudah diakses
    d. Berisi banyak animasi

Jawaban: c

15. Alasan mencantumkan pembelajaran dari proyek adalah...
    a. Untuk mengisi ruang kosong
    b. Untuk menunjukkan kamu belajar dari proses
    c. Agar terlihat panjang
    d. Agar bisa di-copy oleh orang lain
    Jawaban: b

C. Isian Singkat (5 Soal)

16. _____ adalah file penting di GitHub yang menjelaskan proyek secara keseluruhan.
    Jawaban: README.md

17. Dua platform populer untuk membangun portofolio digital adalah _____ dan _____.
    Jawaban: GitHub, LinkedIn

18. Dalam bagian "Hasil dan Analisis", sebaiknya disertakan _____ seperti grafik dan metrik.
    Jawaban: Visualisasi

19. Portofolio digital sebaiknya disusun dengan narasi yang menceritakan _____ belajar kita.
    Jawaban: Perjalanan

20. Komponen dalam deskripsi proyek yang menjelaskan masalah yang ingin diselesaikan disebut _____.
    Jawaban: Problem Statement

D. Soal Eksplorasi

Soal:

Bayangkan kamu sedang membuat deskripsi proyek AI yang mendeteksi emosi dari teks. Buatlah deskripsi singkat proyek yang mencakup: judul menarik, tujuan, pendekatan yang digunakan, dan hasilnya.

Soal:

Kamu melihat dua portofolio: satu memiliki 10 proyek biasa, dan satu lagi hanya 3 proyek tetapi sangat rapi dan terstruktur. Mana yang lebih baik menurutmu dan mengapa? Jelaskan dengan contoh konkret.

Soal:

Seorang siswa membuat proyek sistem rekomendasi buku, tapi deskripsinya hanya “Sistem rekomendasi buku pakai Python.” Bantu perbaiki deskripsi tersebut agar sesuai dengan standar portofolio digital yang efektif.

Tugas Proyek

Proyek 1: Buat akun GitHub dan unggah proyek AI sederhana

Langkah:

1. Buat akun di GitHub.
2. Unggah proyek (misalnya: klasifikasi gambar, chatbot, prediksi harga).
3. Sertakan file `README.md` berisi:

Contoh Struktur README:

```
# Image Classification with CNN

## Deskripsi
Proyek ini adalah model klasifikasi gambar menggunakan Convolutional Neural Networks (CNN) untuk mengenali jenis pakaian dari dataset Fashion MNIST.

## Data
Dataset: Fashion MNIST, terdiri dari 60.000 gambar latih dan 10.000 gambar uji.

## Cara Menjalankan
1. Clone repositori ini
2. Install requirements: `pip install -r requirements.txt`
3. Jalankan: `python train.py`

## Hasil
Model mencapai akurasi 89% pada data uji.

## Pembelajaran
Saya belajar bagaimana membangun CNN sederhana, memahami overfitting, dan mengevaluasi model klasifikasi.
```

Proyek 2: Deskripsi pendek untuk LinkedIn (maks 5 kalimat)

Contoh:

> Membangun model klasifikasi gambar menggunakan CNN dengan dataset Fashion MNIST. Proyek ini menggunakan TensorFlow dan mencapai akurasi 89%. Saya mengimplementasikan preprocessing data, arsitektur CNN, dan evaluasi model. Proyek ini melatih keterampilan saya dalam deep learning dasar. Kode tersedia di GitHub.

Proyek 3: Bangun portofolio digital (2 proyek + narasi)

Platform yang bisa dipakai:

- GitHub Pages
- Google Sites
- LinkedIn bagian “Projects”

Contoh narasi proyek:

1. **Image Classification with CNN**

   Model deep learning sederhana yang saya buat untuk mengenali jenis pakaian dari gambar menggunakan dataset Fashion MNIST. Saya menggunakan TensorFlow dan CNN dengan tiga lapis konvolusi. Proyek ini membantu saya memahami arsitektur CNN dan proses pelatihan model. Akurasi akhir mencapai 89%.

2. **Chatbot dengan NLP**

   Saya membangun chatbot sederhana menggunakan NLTK dan TF-IDF. Bot mampu menjawab pertanyaan sederhana dari pengguna seputar topik tertentu. Proyek ini mengasah pemahaman saya tentang tokenisasi, preprocessing teks, dan sistem pencocokan berbasis similaritas.

Proyek 4: Analisa portofolio mahasiswa AI (300 kata)

Contoh ringkasan:
Saya menemukan portofolio GitHub milik seorang mahasiswa AI bernama John Doe. Portofolionya menonjol karena struktur repositorinya rapi, setiap proyek memiliki README yang informatif, dan ada visualisasi hasil model (grafik, confusion matrix). John juga menuliskan teknologi yang dipakai, tujuan proyek, tantangan yang dihadapi, serta solusi yang diterapkan. Proyek-proyeknya tidak hanya menampilkan kode, tapi juga konteks, seperti aplikasi nyata dan motivasi di baliknya.

Yang membuatnya kuat:

1. **Keterbacaan**: README-nya mudah dipahami, dengan poin-poin jelas.
2. **Visualisasi**: Setiap proyek menampilkan grafik atau output, membantu pembaca memahami hasil.
3. **Dokumentasi Proyek Nyata**: Ada proyek akhir seperti “Pendeteksi Emosi dari Suara” yang menunjukkan keterlibatan dalam data dunia nyata.
4. **Profesionalisme**: Profil LinkedIn-nya ditautkan, dan dia menyusun narasi proyek layaknya CV – singkat tapi bermakna.

Secara keseluruhan, portofolionya menunjukkan bahwa dia bukan hanya bisa coding, tapi juga mampu menjelaskan, mendokumentasikan, dan menampilkan pekerjaannya dengan cara yang profesional dan menarik.

Proyek 5: Revisi deskripsi lama (tambahkan tantangan, solusi, pembelajaran)

Contoh Revisi:
Proyek Lama: “Klasifikasi Sentimen Twitter”
Revisi:

> Saya membangun model klasifikasi sentimen dari data Twitter menggunakan Naive Bayes dan TF-IDF.
> **Tantangan:** Banyak data noisy seperti emoji, URL, dan singkatan.
> **Solusi:** Saya membuat preprocessing pipeline: pembersihan teks, stemming, dan filtering kata-kata tidak relevan.
> **Pembelajaran:** Saya memahami pentingnya preprocessing dalam NLP dan bagaimana model sederhana bisa efektif jika datanya bersih.
> **Visualisasi:** Wordcloud untuk kata positif/negatif dan confusion matrix dari evaluasi model.

BAB: Mengenal Domain .id



Apa Itu Domain?

Bayangkan internet seperti **kota besar** yang punya **jutaan rumah**. Setiap rumah di kota itu punya alamat agar bisa dikunjungi. Nah, nama **domain** itu seperti **alamat kita di internet**, nanti bisa digunakan sebagai alamat website maupun alamat email kita. Dan kita bisa mempunyai alamat email sendiri misalnya halo@meita.id. Misalnya alamat website

Contohnya:

- pandi.id
- home.s.id
- meita.my.id

Alamat itu disebut **nama domain**, dan tanpa domain, kamu harus mengingat angka-angka ribet yang disebut **IP address**. Pasti gak seru, kan?

Apa Itu .id?

Sekarang coba perhatikan domain ini: www.sekolahku.id

Nah, yang bagian **".id"** itu disebut **ekstensi domain**. ".id" menunjukkan bahwa website itu **berasal dari Indonesia**. Setiap negara punya ekstensinya masing-masing, contohnya:

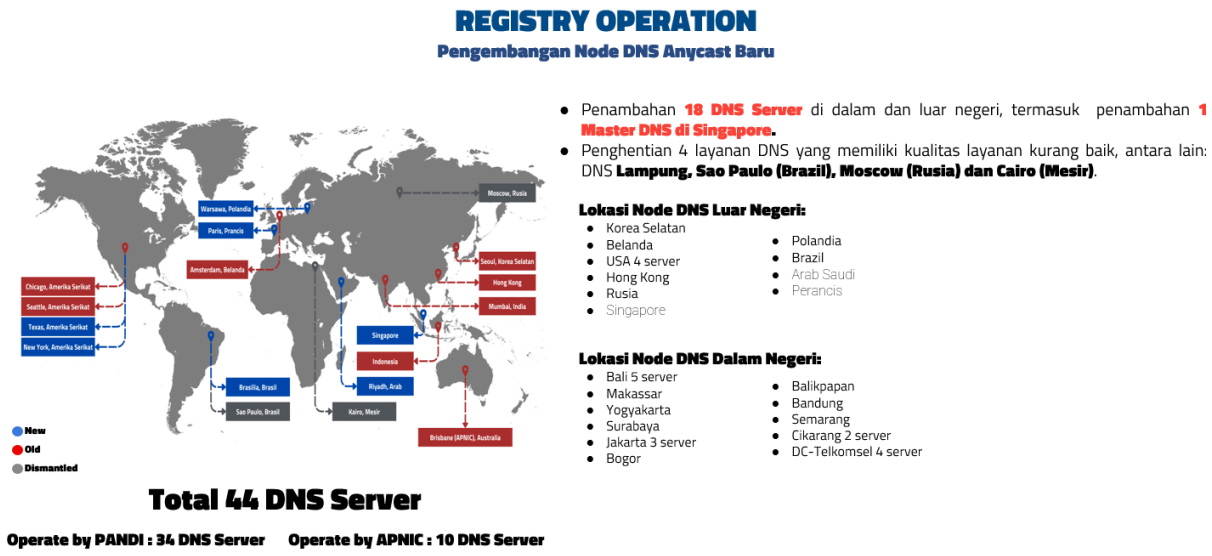
- .jp → Jepang
- .us → Amerika Serikat
- .id → Indonesia 🇮🇩

Menggunakan domain .id itu keren, karena:

- Menunjukkan kebanggaan sebagai orang Indonesia
- Membuat situsmu terlihat lebih terpercaya di mata orang Indonesia
- Cocok banget buat bisnis lokal, sekolah, komunitas, atau bahkan blog pribadi

Keunggulan domain .id

- Representasi Indonesia
  Menunjukan bahwa ini entitas ataupun personal dari Indonesia karena .id adalah kode negara Indonesia
- Sebagai identity
  Selain reprensentasi Indonesia .id juga bisa berarti sebagai identity atau identitas, sehingga sangat menarik untuk dijadikan sebuah alamat website, email maupun brand di internet



Siapa Itu PANDI?

PANDI adalah singkatan dari **Pengelola Nama Domain Internet Indonesia.**
Coba bayangkan **PANDI** seperti **penjaga gerbang dunia .id**. Mereka:

- Mengatur dan mengelola semua domain dengan **akhiran .id**
- Menjaga agar **domain .id aman, tertata, dan tidak disalahgunakan**
- Memberi kesempatan buat siapa saja — termasuk kamu — **punya domain sendiri!**

Misalnya kamu ingin punya website "tokohfavoritku.id", kamu bisa daftar lewat penyedia domain resmi, dan **PANDI** yang mengatur sistemnya agar website-mu bisa diakses semua orang.

Kenapa Kamu Harus Tahu Ini?

- Siapa tahu **nanti kamu bikin website** untuk tugas sekolah, bisnis kecil-kecilan, atau komunitas temanmu
- Supaya kamu melek teknologi, karena **dunia digital adalah masa depan**
- Karena kamu bisa bikin identitas online-mu sendiri dan itu sangat keren!

Kalau kamu bisa punya akun Instagram, kamu juga bisa punya domain sendiri. Bayangkan: www.namamu.id – tempat kamu membagikan karya, foto, tulisan, atau bahkan jualan!

Lebih dalam tentang PANDI

PANDI adalah Registry Nama Domain Indonesia yang berperan untuk mengoperasikan, memelihara dan mengelola Nama Domain Indonesia (.id)

Pengelola Nama Domain Internet Indonesia (PANDI) merupakan perkumpulan yang terdiri dari berbagai pemangku kepentingan, termasuk perwakilan dari pemerintah, operator industri internet, dan akademisi.

Didirikan pada tahun 2006, **PANDI** menerima Redelegasi dari **Internet Assigned Numbers Authority (IANA)** sebagai Registry .id pada tahun 2013.

Hingga 31 Desember 2024 jumlah domain .id yang terdaftar tumbuh menjadi yang terbesar di Asia Tenggara, dengan total 1.215.714 Nama Domain. Nama Domain .id sendiri bisa diartikan sebagai Indonesia, identitas, ide, dan lainnya. Sejalan dengan makna tersebut, **PANDI** juga memiliki kebijakan khusus dalam menyelesaikan perselisihan nama domain .id.

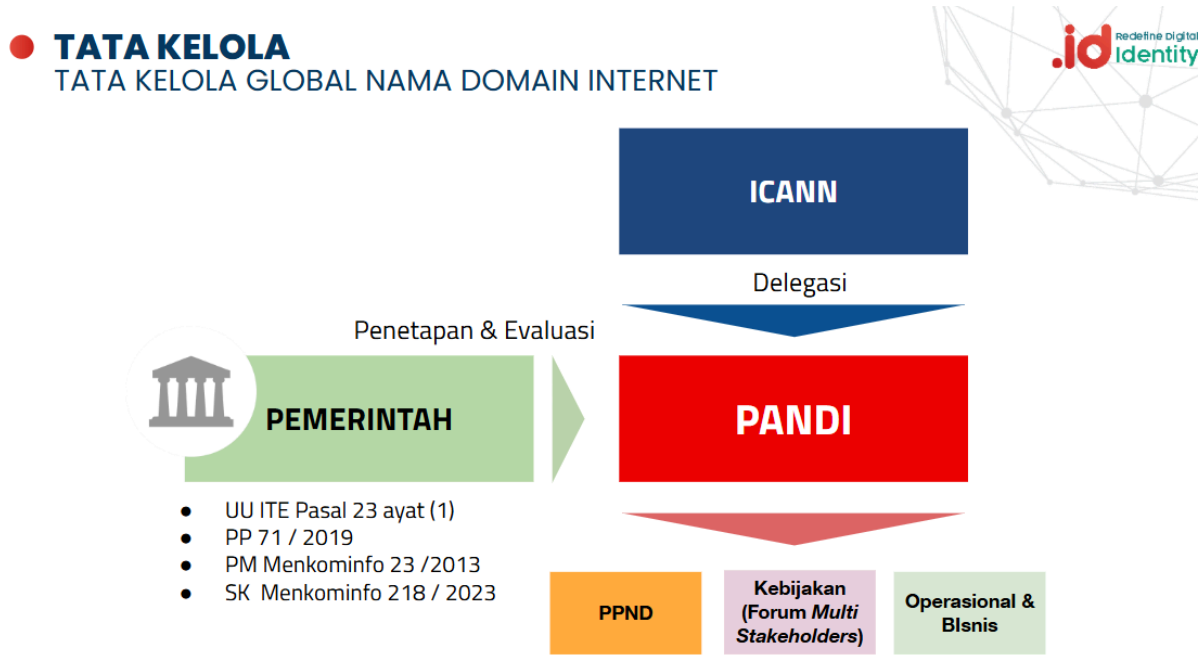
Tugas dan Kewajiban PANDI

Tugas

- Merumuskan kebijakan di bidang pengelolaan Nama Domain Tingkat Tinggi Indonesia
- Menyiapkan, mengoperasikan, dan memelihara infrastruktur yang dibutuhkan serta menyediakan sistem elektronik untuk pengelolaan Nama Domain Tingkat Tinggi Indonesia
- Menyelenggarakan pendaftaran Nama Domain Tingkat Tinggi Indonesia sesuai dengan ketentuan peraturan perundang-undangan, kepatutan yang berlaku dalam masyarakat, dan prinsip kehati-hatian
- Melaksanakan seleksi Registrar Nama Domain
- Memberikan peringatan kepada Registrar Nama Domain jika terindikasi melakukan pelanggaran
- Mencabut hak operasional Registrar Nama Domain jika terbukti melakukan pelanggaran, dan
- Melakukan pengawasan operasional dan teknis Registrar Nama Domain

Kewajiban

- Menjamin sistem elektronik Registri Nama Domain Indonesia beroperasi dengan baik, stabil, aman didukung dengan layanan yang dapat diandalkan
- Menempatkan pusat data dan pusat pemulihan bencana di wilayah Indonesia
- Melakukan pengawasan terhadap Registrar Nama Domain
- Memfasilitasi penyelesaian perselisihan Nama Domain Indonesia
- Melaporkan daftar Registrar Nama Domain kepada Menteri
- Mengikuti ketentuan pengelolaan Nama Domain internasional dan peraturan perundang-undangan
- Menyampaikan laporan berkala kepada Menteri sekurang-kurangnya 1 (satu) kali dalam setahun
- Membayar pungutan biaya Pengelolaan Nama Domain Indonesia



Tata Kelola Internet

PANDI berkomitmen penuh untuk terus berkontribusi aktif dalam tata kelola internet global dengan mendukung infrastruktur yang aman dan inovatif. Hal ini karena domain .id merupakan bagian penting dari ekosistem digital dunia yang inklusif dan berkelanjutan.

Di tahun 2024, PANDI menginisiasi berbagai wadah tata kelola internet nasional serta berpartisipasi aktif dalam forum tata kelola internet global, seperti:

- Forum Multi Stakeholder (FMS),
- PANDI Meeting 2024,
- ICANN 79 di Puerto Riko,
- ICANN 81 di Turki,
- APTLD 86 di Vietnam,
- APAC DNS Forum 2024 yang diselenggarakan oleh PANDI di Bali.

Selain APAC DNS Forum, PANDI juga menggelar program Indonesia Internet Governance Academy (IDIGA) yang bertujuan mencetak pemimpin muda di bidang tata kelola internet, baik di tingkat nasional maupun internasional.

Institut Teknologi Tangerang Selatan (ITTS) dan Penulis.

Para penulis adalah **dosen aktif di Institut Teknologi Tangerang Selatan (ITTS)**, sebuah **institusi pendidikan tinggi berbasis teknologi** yang berperan aktif dalam pengembangan **solusi digital** dan *kecerdasan buatan (AI)*. Tidak hanya mengajar, mereka juga merupakan **praktisi dan inovator** yang menghasilkan **karya-karya nyata di bidang teknologi mandiri**.

Berlokasi di **Komplek Komersial BSD, Serpong Utara, Tangerang Selatan**, **ITTS hadir sebagai pusat inovasi** yang mengintegrasikan **pembelajaran, penelitian, dan pengembangan teknologi secara menyeluruh**. Fokus utama ITTS mencakup bidang strategis seperti *AI, Machine Learning, Deep Learning*, jaringan, *cloud*, keamanan siber, pemrograman, sistem informasi, dan multimedia.

ITTS menawarkan lingkungan akademik yang dinamis dan kolaboratif. Setiap minggu, diselenggarakan berbagai *workshop*, *demo teknologi*, seminar, hingga *webinar*—mayoritas **gratis dan tersedia secara terbuka di YouTube**. Ini menciptakan **kultur belajar yang aktif dan relevan dengan perkembangan industri**.

Fasilitas ITTS menunjang kegiatan riset dan inovasi secara optimal. Lab komputer dilengkapi *GPU RTX 4060* untuk eksperimen *AI*, serta **laboratorium teknologi dengan puluhan server, perangkat *IoT*, sistem komunikasi radio, dan infrastruktur jaringan jarak jauh**. Dari ekosistem ini lahirlah **produk-produk mandiri** seperti **ChatGPT versi lokal** untuk institusi pendidikan dan **sistem perpustakaan digital yang dapat diakses tanpa koneksi internet**.

Daya saing ITTS juga tercermin dari karakter dosen dan mahasiswa yang tidak hanya akademik, tetapi juga **produktif secara praktis**. Mahasiswa didorong aktif menjadi **narasumber, penulis, dan peneliti** sejak dini. Banyak diantaranya telah menghasilkan **karya buku, artikel ilmiah**, serta **terlibat langsung dalam proyek-proyek teknologi**.

Kolaborasi strategis dengan berbagai mitra industri—seperti *IDCloudHost, Ubuntu, XecureIT, Dinas KOMINFO Tangerang Selatan*, berbagai *ISP*, dan komunitas IT nasional—menjadi **jembatan nyata antara kampus dan dunia profesional**. Hasilnya, **lebih dari 75% mahasiswa ITTS telah bekerja bahkan sebelum lulus**, dan **hampir seluruh lulusan langsung terserap industri**.

ITTS bukan hanya tempat belajar, tapi tempat mencipta. Dosen sebagai **penulis dan praktisi**, mahasiswa sebagai **peneliti dan kontributor**—semuanya berperan **membangun ekosistem teknologi yang mandiri dan berdaya saing tinggi**.

Institut Teknologi Tangerang Selatan (ITTS)
Komplek Komersial BSD
Jl. Raya Serpong No. Kav. 9, Lengkong Karya, Serpong Utara
Kota Tangerang Selatan, Banten 15331
📞 (+62) 877-7277-1775 📧 info@itts.ac.id 🌐 www.itts.ac.id