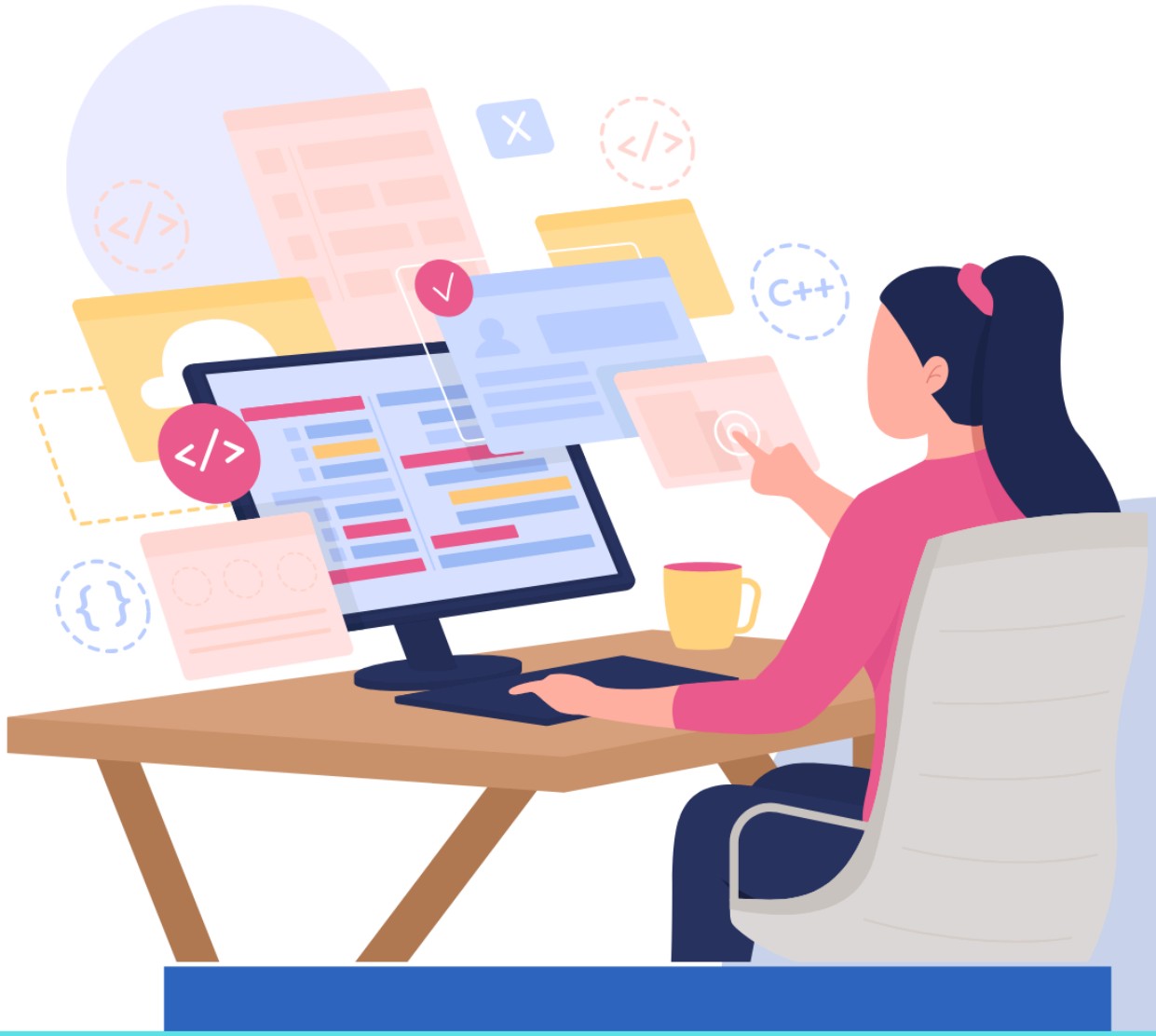
ARTIFICIAL INTELLIGENCE

UNTUK SEKOLAH MENENGAH PERTAMA (SMP)

Jilid 4



Onno W. Purbo, Chendri Irawan SN, Muhamad Yusuf, Luthfia Saskia, Devi Fatwanti, Anas Nasrulloh

Artificial Intelligence
Sekolah Menengah Pertama (SMP)

Karya ini dipersembahkan melalui kerja sama penerbitan Intech Press dan kontribusi keilmuan dari dosen Institut Teknologi Tangerang Selatan (ITTS), dengan apresiasi atas dukungan dari PANDI dan Astra.

Onno W. Purbo, Chendri Irawan SN, Muhamad Yusuf,
Luthfia Saskia, Devi Fatwanti, Anas Nasrulloh

Institut Teknologi Tangerang Selatan (ITTS)
2025

Artificial Intelligence
Sekolah Menengah Pertama (SMP)

Penulis : Onno W. Purbo, Chendri Irawan SN,
Muhamad Yusuf, Luthfia Saskia, Devi Fatwanti, Anas Nasrulloh
Desain Sampul : Irwan Siswanto
Tata Letak : Fathiya Nurul Ilmi, Meiliza Putri Saefulloh

Ukuran:
viii, 115 hlm., Uk.: 21x29,7 cm

ISBN :
No ISBN

Cetakan Pertama :
Agustus 2025

Hak Cipta 2025, Pada Penulis

Isi diluar tanggung jawab percetakan
Copyright © 2025 by InTech Press
All Right Reserved

Hak cipta dilindungi undang-undang
Dilarang keras menerjemahkan, memfotokopi, atau
memperbanyak sebagian atau seluruh isi buku ini
tanpa izin tertulis dari Penerbit.

InTech Press
Komp Komersial BSD, Jl. Raya Serpong No.Kav.9,
Lengkong Karya, Serpong Utara, Tangerang Selatan, Banten 15331

Website: https://www.intechpress.itts.ac.id
E-mail: intechpress@itts.ac.id

Hak Cipta © 2025
Seluruh hak cipta dilindungi undang-undang.

UU No 28 tahun 2014 tentang Hak Cipta

Fungsi dan sifat hak cipta Pasal 4
Hak Cipta sebagaimana dimaksud dalam Pasal 3 huruf a merupakan hak eksklusif yang terdiri atas hak moral dan hak ekonomi.

Pembatasan Perlindungan Pasal 26
Ketentuan sebagaimana dimaksud dalam Pasal 23, Pasal 24, dan Pasal 25 tidak berlaku terhadap:

i. Penggunaan kutipan singkat Ciptaan dan/atau produk Hak Terkait untuk pelaporan peristiwa aktual yang ditujukan hanya untuk keperluan penyediaan informasi aktual;

ii. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk kepentingan penelitian ilmu pengetahuan;

iii. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk keperluan pengajaran, kecuali pertunjukan dan Fonogram yang telah dilakukan Pengumuman sebagai bahan ajar; dan

iv. Penggunaan untuk kepentingan pendidikan dan pengembangan ilmu pengetahuan yang memungkinkan suatu Ciptaan dan/atau produk Hak Terkait dapat digunakan tanpa izin Pelaku Pertunjukan, Produser Fonogram, atau Lembaga Penyiaran.

Sanksi Pelanggaran Pasal 113

1. Setiap Orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak Rp100.000.000 (seratus juta rupiah).

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

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. Silakan 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

Daftar Isi

Kata Pengantar **5**
Daftar Isi **6**
Lisensi & Catatan Karya **9**
Disclaimer **9**
BAB 1: Pengenalan App Inventor 2 10
Tujuan Pembelajaran 10
Peta Konsep 10
Penjelasan Konsep (teori) 11
Ilustrasi App Inventor 2 12
Contoh Kasus 13
Contoh Soal dan Pembahasan 14
Tips Cepat Menghafal 16
Aktivitas Siswa 17
Rangkuman 19
Tugas Proyek 25
BAB 2: Membuat Aplikasi Sederhana: Kalkulator / Survey 28
Tujuan Pembelajaran 28
Peta Konsep 28
Apersepsi 29
Penjelasan Konsep (teori) 29
"Memberi Nyawa pada Aplikasi Kita!" 29
Blok Peristiwa (Event Handling) 29
Variabel (Variables) 30
Blok Matematika (Math Blocks) 31
Menampilkan Teks di Label (Label Blocks) 31
Mengambil Input dari TextBox (TextBox Blocks) 31
Ringkasan 32
Tantangan Seru! 32
Contoh Kasus atau Ilustrasi 33
Contoh Soal dan Pembahasan 35
Fakta Menarik 38
Tips Cepat Menghafal 38
Aktivitas Siswa 39
Rangkuman 41
Latihan Soal 42
Tugas Proyek 46
BAB 3: Pengenalan Google Teachable Machine (Teori) 48
Tujuan Pembelajaran 48
Peta Konsep 48
Apersepsi 48

Penjelasan Konsep (teori)..... 49
Apa itu Google Teachable Machine?..... 49
Mengapa Belajar Machine Learning (ML) Itu Penting?..... 49
Bagaimana Cara Kerja Machine Learning?..... 50
Langkah-Langkah Proses Machine Learning:..... 50
Perbedaan Penting..... 51
Jenis Proyek di Google Teachable Machine..... 51
Kesimpulan Seru!..... 52
Contoh Kasus atau Ilustrasi..... 52
Contoh Soal dan Pembahasan..... 54
Fakta Menarik..... 55
Tips Cepat Menghafal..... 55
Aktivitas Siswa..... 56
Rangkuman..... 58
Latihan Soal..... 59
Tugas Proyek..... 62
BAB 4: Praktik dengan Google Teachable Machine: Model Gambar Sederhana (3 Pertemuan)..... 65
Tujuan Pembelajaran..... 65
Peta Konsep..... 65
Apersepsi..... 65
Penjelasan Konsep (teori)..... 66
Langkah-langkah Membuat Model Gambar..... 66
Kesimpulan..... 69
Ilustrasi: Memecahkan Kasus dengan Teachable Machine..... 70
Contoh Kasus atau Ilustrasi..... 70
Contoh Soal dan Pembahasan..... 72
Fakta Menarik..... 74
Tips Cepat Menghafal..... 74
Aktivitas Siswa..... 75
Rangkuman..... 76
Latihan Soal..... 76
Tugas Proyek..... 80
BAB 5: Mengenal Machine Learning dengan Ekstensi Personal Image Classifier di App Inventor 2..... **83**
Tujuan Pembelajaran..... 83
Peta Konsep..... 83
Apersepsi..... 84
Penjelasan Konsep (Teori)..... 84
Apa Itu Machine Learning?..... 84
Apa Itu Personal Image Classifier (PIC)?..... 85
Membuat Model dan Training Data untuk PersonalImageClassifer..... 85
Aplikasi untuk Klasifikasi Gambar Sederhana..... 92

Ilustrasi..... 94
Contoh Kasus atau Ilustrasi..... 94
Fakta Menarik..... 97
Tips Cepat Menghafal.....97
Contoh Mind Mapping Sederhana:..... 97
Aktivitas Siswa.....98
Rangkuman..... 100
Latihan Soal.....101
Tugas Proyek..... 104
BAB 6: Refleksi: Apa Itu AI di Aplikasi?.....107
Tujuan Pembelajaran.....107
Peta Konsep.....107
Apersepsi..... 107
Penjelasan Konsep (teori)..... 108
Ilustrasi: Pemakaian Aplikasi dengan AI di Dunia Nyata..... 108
Contoh Kasus atau Ilustrasi.....109
Contoh Soal dan Pembahasan..... 111
Fakta Menarik..... 112
Tips Cepat Menghafal..... 112
Aktivitas Siswa.....113
Rangkuman..... 115
Latihan Soal.....116
Tugas Proyek..... 119
BAB 7: Mengenal Domain .id..... 122
Apa Itu Domain?..... 123
Apa Itu .id?..... 123
Keunggulan domain .id.....123
Siapa Itu PANDI?.....124
Kenapa Kamu Harus Tahu Ini?..... 124
Lebih dalam tentang PANDI..... 125
Institut Teknologi Tangerang Selatan (ITTS) dan Penulis.....127

BAB 1: Pengenalan App Inventor 2

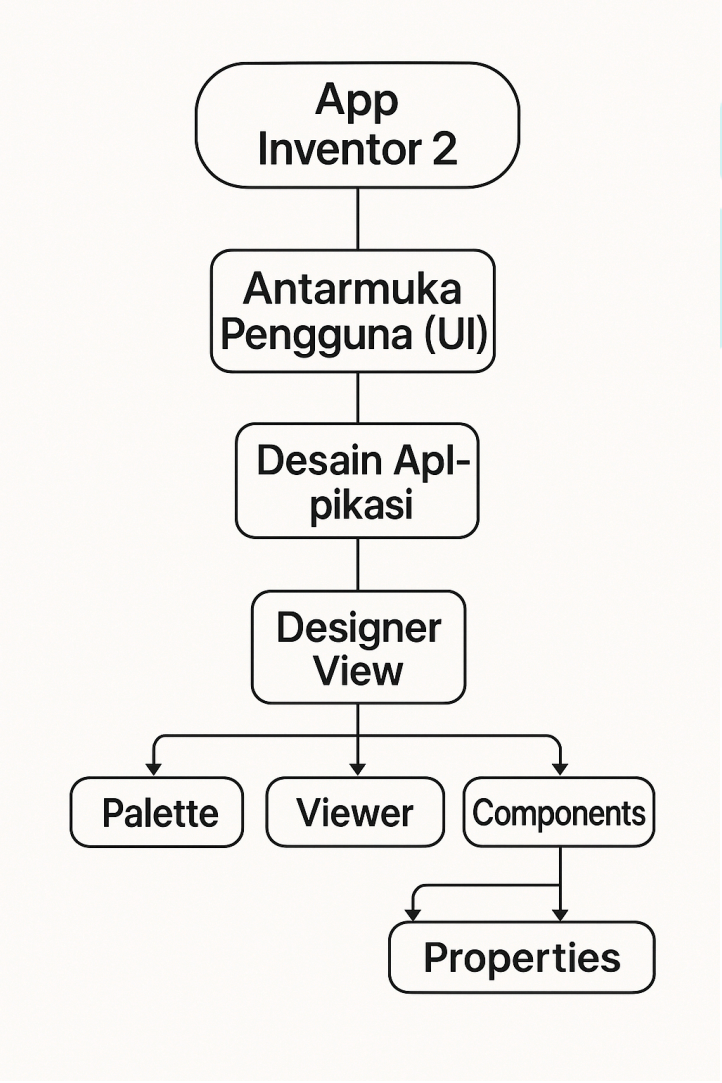
Tujuan Pembelajaran

Setelah mempelajari bab ini, kalian diharapkan mampu:

- Menggunakan **App Inventor 2** untuk membuat tampilan aplikasi (user interface).
- Menyusun tata letak komponen seperti **tombol, teks, gambar, dan kotak input** secara rapi dan menarik.
- Memahami fungsi dari setiap **panel utama** dalam lingkungan App Inventor 2, seperti **Palette, Viewer, Components, dan Properties**.

Jumlah Pertemuan: 2

Peta Konsep



Apersepsi

Pernahkah kalian memperhatikan tampilan aplikasi di ponsel?
Ada yang warna-warni, simpel, atau sangat estetik! Tapi... bagaimana cara membuatnya?

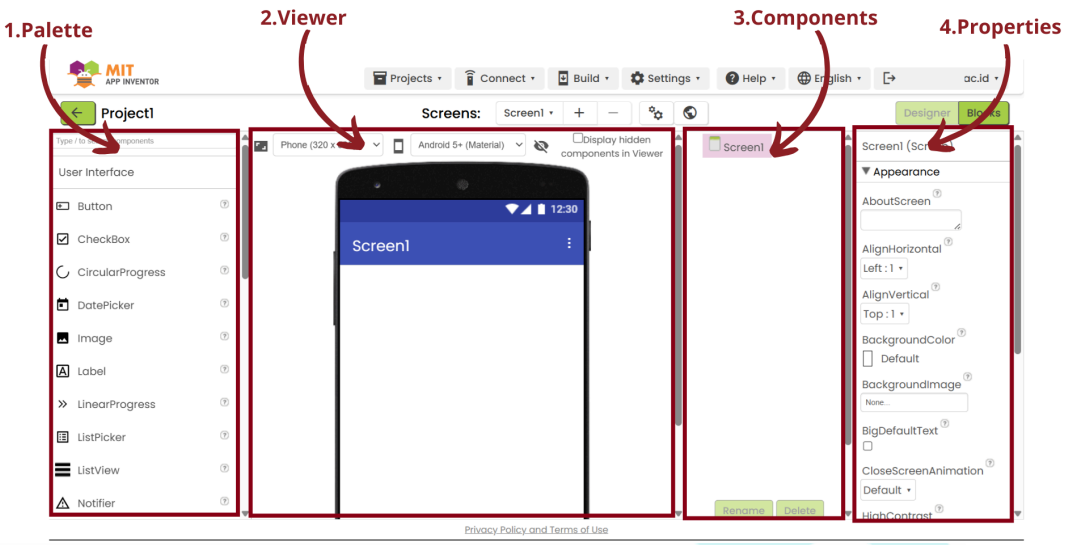
Nah, di bab ini, kita akan belajar **cara mendesain tampilan aplikasi seperti seorang desainer profesional**! Kalian akan bisa membuat tampilan aplikasi sendiri — lengkap dengan tombol, gambar, dan teks, hanya dengan **drag and drop**. Mudah, seru, dan menantang!

Siap jadi **desainer aplikasi pertama kalian**?

Penjelasan Konsep (teori)

App Inventor 2 adalah platform pengembangan aplikasi mobile berbasis blok grafis yang sangat mudah digunakan. Lingkungan kerjanya terbagi menjadi dua bagian utama:

1. **Designer View:** Ini adalah tempat kita mendesain tampilan aplikasi. Di Designer View, ada beberapa panel penting:
    - **Palette:** [Gambar: Screenshot App Inventor 2 Designer View dengan panel Palette disorot]. Terletak di sisi paling kiri. Panel ini berisi semua komponen yang bisa kita tambahkan ke aplikasi. Komponen-komponen ini dikelompokkan ke dalam kategori seperti User Interface (Button, Label, TextBox, Image), Layout (untuk menata komponen), Media (Sound, Camera), Sensors (BarcodeScanner, GPS), dan banyak lagi.
    - **Viewer:** [Gambar: Screenshot App Inventor 2 Designer View dengan panel Viewer disorot]. Ini adalah layar pratinjau aplikasi kita, terlihat seperti layar smartphone. Komponen yang kita pilih dari Palette akan kita “tarik dan lepas” ke sini. Di sinilah kita menempatkan dan mengatur posisi komponen-komponen.
    - **Components:** [Gambar: Screenshot App Inventor 2 Designer View dengan panel Components disorot]. Terletak di antara Viewer dan Properties. Panel ini menampilkan daftar semua komponen yang sudah kita tambahkan ke layar aplikasi. Kita bisa memilih komponen dari daftar ini untuk mengubah propertinya.
    - **Properties:** [Gambar: Screenshot App Inventor 2 Designer View dengan panel Properties disorot]. Terletak di sisi paling kanan. Setelah memilih sebuah komponen di panel Components, panel Properties akan menampilkan semua atribut atau properti yang bisa kita atur untuk komponen tersebut (misalnya, teks pada tombol, warna latar belakang, ukuran huruf, lebar, tinggi, dll.).



2. **Blocks Editor:** Ini adalah tempat kita menambahkan logika dan fungsi pada aplikasi menggunakan blok-blok kode visual. (Akan dibahas lebih lanjut di Bab 2).

   Kita akan mengenal komponen UI dasar seperti:

   - Button: Tombol yang bisa diklik. Properti penting: Text, BackgroundColor, Width, Height.
   - Label: Teks statis untuk menampilkan informasi. Properti penting: Text, FontSize, TextColor.
   - TextBox: Kotak input untuk pengguna memasukkan teks atau angka. Properti penting: Hint, Text.
   - Image: Untuk menampilkan gambar. Properti penting: Picture, Height, Width.
   - VerticalArrangement, HorizontalArrangement: Komponen layout untuk menata komponen lain secara vertikal atau horizontal agar rapi. Properti penting: Width, Height, AlignHorizontal, AlignVertical.

Ilustrasi App Inventor 2

Mari kita desain layar pembuka aplikasi sederhana. Layar ini akan memiliki judul aplikasi, gambar, dan sebuah tombol "Mulai".

1. Buka App Inventor 2.
2. Mulai proyek baru.
3. Ubah judul layar: Klik pada Screen1 di panel Components. Di panel Properties di sebelah kanan, gulir ke bawah dan cari properti Title. Ganti teks Screen1 menjadi “Halo!”.
4. Dari Palette, tarik komponen Label ke Viewer. Ubah properti Text Label menjadi "Selamat Datang di Aplikasi Saya". Atur ukuran tulisan (FontSize) dan warna tulisan (TextColor) sesuai keinginan.

5. Tarik komponen Image ke bawah Label. Upload gambar yang relevan dari komputer kalian ke Media, lalu pilih gambar tersebut di properti Picture komponen Image. Atur lebar (Width) dan tinggi (Height) gambar di panel Properties sesuai keinginan.
6. Tarik komponen Button ke bawah Image. Ubah properti Text Button menjadi "Mulai Sekarang".
7. Rapikan tata letak komponen: Klik pada komponen layar di panel Components. Di panel Properties, atur AlignHorizontal menjadi Center:3.

Selamat! Kamu telah berhasil membuat desain aplikasi pertamamu!



Contoh Kasus

Kasus 1: Aplikasi Salam Pembuka

Bayangkan kamu membuat aplikasi untuk menyambut pengguna. Aplikasinya hanya punya satu tulisan “Halo, Selamat Datang!”, gambar smiley, dan tombol “Mulai”. Komponennya:

- Label → Tulisan sambutan
- Image → Gambar smiley
- Button → Tombol "Mulai"

Kasus 2: Aplikasi Identitas Diri

Kamu bisa membuat aplikasi "Kartu Nama Digital" yang menampilkan:

- Foto kamu
- Nama lengkap
- Kontak (email/telepon)
- Tombol "Hubungi Saya"

Kasus 3: Aplikasi Catatan Harian

Buat aplikasi dengan satu kotak besar untuk menulis (TextBox) dan tombol “Simpan”. Sederhana, tapi berguna!

Kasus 4: Aplikasi Kalkulator Mini

Tampilan kalkulator dengan:

- 2 TextBox untuk angka
- 4 tombol operasi (Tambah, Kurang, Kali, Bagi)
- 1 Label untuk menampilkan hasil

Contoh Soal dan Pembahasan

Soal: Bagaimana cara menambahkan sebuah tombol ke layar aplikasi di App Inventor 2, dan bagaimana cara mengubah teks yang tertera pada tombol tersebut?

Pembahasan:

- **Menambahkan Tombol:** Di Designer View, cari Button di panel Palette (biasanya di kategori User Interface). Klik dan seret Button tersebut ke Viewer (area tengah yang menyerupai layar HP).
- **Mengubah Teks Tombol:** Setelah Button ditempatkan di Viewer, pastikan Button tersebut terpilih di panel Components. Kemudian, lihat panel Properties di sebelah kanan. Cari properti Text dan ubah nilainya dari "Text for Button1" menjadi teks yang kamu inginkan, misalnya "Klik Saya!".

Soal: Bagaimana cara menambahkan dan mengganti teks pada Button?

Jawaban:

- Seret komponen Button dari Palette ke Viewer.
- Pilih Button di panel Components.
- Di panel Properties, ubah tulisan di bagian **Text** menjadi "Klik Saya!".

Soal: Komponen apa yang digunakan untuk menampilkan gambar, dan properti apa yang harus diubah?

Jawaban:
Gunakan komponen Image. Untuk menampilkan gambar:

- Upload gambar ke Media.
- Di panel Properties, ubah properti **Picture** menjadi nama file gambar.

Soal: Apa perbedaan Label dan TextBox?

Jawaban:

- **Label:** Menampilkan teks statis, tidak bisa diubah pengguna.
- **TextBox:** Untuk input dari pengguna.
  Contoh: Gunakan Label untuk "Nama", **TextBox** untuk pengguna menulis nama mereka.

Soal: Dimana kita mengatur warna dan ukuran teks pada Label?

Jawaban:
Di panel Properties. Ubah:

- **FontSize** → untuk ukuran huruf
- **TextColor** → untuk warna teks

Fakta Menarik

Tahukah kalian?

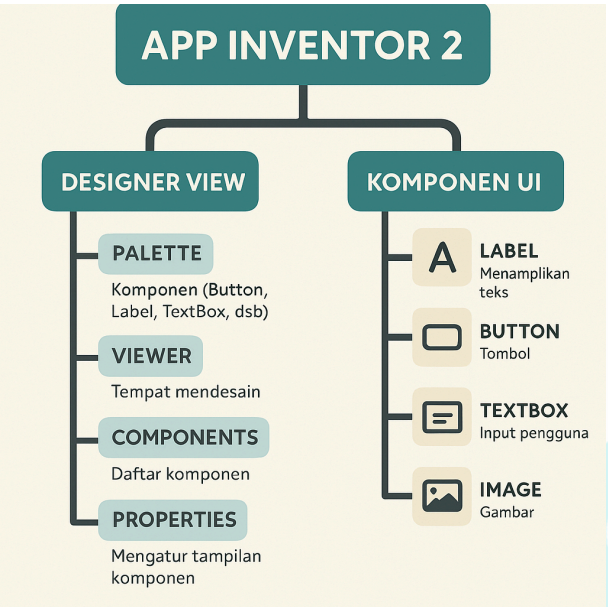
- **Desain aplikasi yang baik,** yaitu yang mudah digunakan dan enak dipandang, bisa membuat pengguna sangat senang dan betah berlama-lama memakai aplikasi tersebut!
- Sebaliknya, **desain yang buruk** bisa membuat pengguna frustasi dan segera menghapus aplikasi. Jadi, desain itu penting!

Tips Cepat Menghafal

Tips Mudah Ingat:

"Tarik – Lepas – Atur Properti"
(Drag from Palette, Drop to Viewer, Edit in Properties!)

Mind Mapping



Cerita Ringan

Bayangkan App Inventor itu seperti merakit mainan LEGO.

- Palette = Kotak LEGO
- Viewer = Meja kerja kamu
- Komponen = Potongan LEGO
- Properties = Warna, ukuran, dan bentuk potongan LEGO

Aktivitas Siswa

Aktivitas 1: Mendesain Aplikasi Kartu Nama Digital Sederhana

Buatlah sebuah aplikasi kartu nama digital yang menampilkan informasi pribadi kalian. Aplikasi ini harus memiliki:

1. Label untuk Nama Lengkap kalian.
2. Image untuk menampilkan foto diri atau avatar.
3. Label untuk Nomor Telepon/Email.
4. Button dengan teks "Hubungi Saya" (nantinya akan ditambahkan fungsi di babselanjutnya). Pastikan layoutnya rapi menggunakan HorizontalArrangement atau VerticalArrangement jika diperlukan.

Langkah-langkah:

- Mulai proyek baru di App Inventor.

- Tambahkan VerticalArrangement untuk menampung semua komponen.
- Di dalamnya, tambahkan Image dan atur properti Picture dengan foto Anda.
- Tambahkan Label untuk nama Anda dan atur properti Text.
- Tambahkan Label untuk kontak (misalnya "Email: nama@contoh.com") dan atur Text.
- Tambahkan Button dan atur properti Text menjadi "Hubungi Saya".
- Atur properti Width pada Image menjadi Fill parent agar foto menyesuaikan ukuran layar.
- Atur properti AlignHorizontal dan AlignVertical pada Screen1 atau VerticalArrangement agar komponen-komponen berada di tengah layar.

Aktivitas 2: Tebak Komponen!

Ini adalah kuis cepat di kelas. Guru menyebutkan fungsi dari sebuah komponen aplikasi, dan siswa harus menebak nama komponennya.

Langkah-langkah:

1. Guru menyebutkan satu fungsi, misalnya: “Menampilkan gambar”.
2. Siswa berpikir komponen apa yang cocok.
3. Siswa angkat tangan dan menjawab, misalnya: “Image!”.
4. Guru memberi poin jika benar.

Pseudocode:

```
WHILE permainan berlangsung:
Guru: tampilkan fungsi komponen
Siswa: jawab nama komponennya
IF jawaban benar:
Tambah poin siswa
ELSE:
Coba siswa lain
END WHILE
```

Aktivitas 3: Kompetisi Desain Kartu Nama

Siswa membuat kartu nama digital dengan desain semenarik mungkin menggunakan App Inventor. Harus ada gambar dan teks. Setelah selesai, semua siswa voting desain terbaik.

Langkah-langkah:

1. Buka App Inventor dan mulai proyek baru.
2. Tambahkan gambar (foto diri atau logo) dan teks (nama, kelas, hobi).
3. Hias tampilan dengan warna dan tata letak menarik.
4. Tampilkan hasil ke teman-teman.
5. Semua siswa voting desain favorit mereka.

Pseudocode:

Mulai proyek App Inventor
Tambah komponen: Image, Label, Canvas
Desain kartu nama digital
Simpan dan tampilkan ke teman
Setiap siswa: pilih desain terbaik
Hitung suara terbanyak → pemenang

Aktivitas 4: Aplikasi Favoritku

Siswa memilih satu aplikasi favorit (misalnya TikTok, WhatsApp), lalu mencoba mendesain ulang tampilan utamanya dengan versi sederhana di App Inventor.

Langkah-langkah:

1. Pilih aplikasi favorit.
2. Perhatikan tampilan aplikasi tersebut.
3. Di App Inventor, buat layout serupa (tombol, gambar, kolom chat, dll).
4. Gunakan komponen seperti Button, Image, TextBox, ListView.
5. Tampilkan ke teman dan jelaskan fungsinya.

Pseudocode:
Pilih aplikasi favorit
Buka App Inventor → buat proyek baru
Susun komponen sesuai tampilan aplikasi asli
Contoh:
Tambah Button untuk menu
Tambah Image untuk profil
Tambah ListView untuk daftar chat
Tampilkan hasil dan presentasi ke kelas

Aktivitas 5: Game Desain Cepat

Guru menyebutkan 3 komponen secara acak (misalnya Button, Image, TextBox). Siswa harus mendesain layout menggunakan ketiga komponen tersebut dalam waktu 10 menit!

Langkah-langkah:

1. Guru mengumumkan 3 komponen wajib.
2. Siswa buka App Inventor.
3. Dalam waktu 10 menit, buat tampilan menggunakan 3 komponen tersebut.
4. Kreativitas bebas, tapi semua komponen harus dipakai.

Pseudocode:
Guru: sebutkan 3 komponen acak
Timer mulai → 10 menit
Siswa:
Buat proyek baru
Tambah komponen sesuai instruksi
Atur layout se-kreatif mungkin
Timer habis → tampilkan hasil
Guru/teman beri masukan

Rangkuman

Di bab ini, kita telah belajar bahwa **App Inventor 2** adalah **alat yang sangat berguna** untuk membuat tampilan aplikasi di HP Android. Dengan menggunakan *Designer View*, kita bisa dengan mudah **membuat desain aplikasi** dengan cara **"tarik dan lepas"** komponen.

Komponen seperti **Button (tombol)**, **Label (teks)**, **TextBox (kolom isian)**, dan **Image (gambar)** bisa kita ambil dari bagian *Palette*, lalu **diletakkan di Viewer**, yaitu area kerja utama tempat kita menyusun tampilan aplikasi. Setelah itu, tampilannya bisa kita **atur lebih lanjut di panel *Properties***.

Agar tampilan aplikasi lebih **rapi dan teratur**, kita bisa menggunakan pengaturan *Layout* dengan komponen seperti **VerticalArrangement**. Ini membantu menyusun komponen secara **terstruktur dan enak dilihat**.

Secara keseluruhan, **App Inventor 2 sangat cocok untuk pemula** karena tidak perlu mengetik kode yang rumit. Cukup ikuti tiga langkah utama:

1. **Tarik komponen** dari *Palette*,
2. **Lepas di Viewer*,
3. **Atur tampilannya** di *Properties*.

Dengan memahami cara kerja komponen dan *Layout*, kamu sudah bisa mulai membuat **aplikasi Android yang menarik dan fungsional**!

Latihan Soal

A. Benar atau Salah (5 Soal)

1. App Inventor 2 digunakan untuk membuat aplikasi mobile berbasis teks.
   Jawaban: Salah

2. Komponen Label dapat digunakan untuk menampilkan teks statis.
   Jawaban: Benar

3. Viewer adalah tempat untuk mengatur logika program.
   Jawaban: Salah

4. Properties digunakan untuk mengubah atribut suatu komponen.
   Jawaban: Benar

5. Button, Label, dan TextBox adalah contoh komponen antarmuka pengguna.
   Jawaban: Benar

B. Pilihan Ganda (10 Soal)

6. Komponen mana yang digunakan untuk menampilkan gambar di aplikasi?
   a. Label
   b. Button
   c. Image
   d. TextBox
   Jawaban: c. Image

7. Untuk menyusun komponen secara vertikal, kita menggunakan:
   a. Viewer
   b. HorizontalArrangement
   c. VerticalArrangement
   d. Palette
   Jawaban: c. VerticalArrangement

8. Panel yang menampilkan semua komponen yang sudah ditambahkan ke layar adalah:
   a. Properties
   b. Components
   c. Viewer
   d. Palette
   Jawaban: b. Components

9. Fungsi dari komponen TextBox adalah:
   a. Menampilkan teks
   b. Menyusun layout
   c. Menyimpan gambar
   d. Menerima input teks dari pengguna
   Jawaban: d. Menerima input teks dari pengguna

10. Di panel mana kita mengatur warna, ukuran, dan teks suatu komponen?
    a. Palette
    b. Viewer
    c. Properties

d. Components
Jawaban: c. Properties

11. Untuk mengubah tulisan pada tombol, properti yang digunakan adalah:
    a. FontSize
    b. Text
    c. Width
    d. Picture
    Jawaban: b. Text

12. Komponen Button memiliki fungsi utama untuk:
    a. Menampilkan gambar
    b. Menyusun layout
    c. Memicu aksi ketika diklik
    d. Menyimpan data
    Jawaban: c. Memicu aksi ketika diklik

13. Jika ingin menempatkan dua tombol secara horizontal, maka digunakan komponen:
    a. VerticalArrangement
    b. Viewer
    c. HorizontalArrangement
    d. Image
    Jawaban: c. HorizontalArrangement

14. Di mana kita dapat menemukan komponen-komponen seperti Label dan Button?
    a. Components
    b. Viewer
    c. Properties
    d. Palette
    Jawaban: d. Palette

15. Properti Hint biasanya digunakan pada komponen:
    a. Label
    b. TextBox
    c. Button
    d. Image
    Jawaban: b. TextBox

C. Isian Singkat (5 Soal)

16. Komponen untuk menampilkan teks statis adalah ___.
    Jawaban: Label

17. Komponen yang digunakan untuk menerima masukan teks dari pengguna adalah ___.
    Jawaban: TextBox

18. Untuk menyusun komponen secara mendatar, digunakan komponen ___.
    Jawaban: HorizontalArrangement

19. Panel tempat melihat pratinjau desain aplikasi disebut ___.
    Jawaban: Viewer

20. Properti yang digunakan untuk mengubah warna latar belakang pada Button adalah ___.
    Jawaban: BackgroundColor

D. Soal Eksplorasi

Soal 1:

Bayangkan kamu membuat aplikasi pembuka yang berisi gambar dan tombol "Mulai Sekarang". Jelaskan langkah-langkah dari awal mulai proyek sampai komponen tampil rapi di layar. Sebutkan komponen yang digunakan dan bagaimana kamu mengatur layout agar tampilan menarik dan mudah digunakan.

Jawaban: Langkah pertama adalah membuat proyek baru di App Inventor 2. Lalu tambahkan VerticalArrangement untuk menyusun komponen dari atas ke bawah. Masukkan komponen Label dengan teks "Selamat Datang", Image untuk gambar, dan Button dengan teks "Mulai Sekarang". Atur ukuran komponen, gunakan AlignHorizontal dan AlignVertical agar semua komponen berada di tengah layar. Tambahkan gambar ke Media dan atur properti Picture pada Image. Periksa di Viewer apakah tata letaknya sudah rapi dan seimbang.

Soal 2:

Kamu diminta membuat aplikasi catatan sederhana dengan satu kotak teks besar dan satu tombol "Simpan". Buatlah desain layout-nya menggunakan App Inventor dan tulis urutan tindakan dari pemilihan komponen hingga pengaturan ukuran agar nyaman digunakan.

Jawaban: Buat proyek baru di App Inventor. Tambahkan VerticalArrangement sebagai kontainer utama. Masukkan satu TextBox, aktifkan properti MultiLine, dan atur Width menjadi Fill parent agar selebar layar. Tambahkan Button di bawahnya dengan teks "Simpan". Atur Height dari TextBox agar cukup panjang untuk menulis banyak teks. Gunakan pengaturan layout agar komponen berada di tengah dan tidak terlalu rapat. Periksa tampilan akhir di Viewer agar terlihat bersih dan mudah digunakan.

Soal 3:

Jelaskan perbedaan antara komponen Label dan TextBox dari segi fungsi dan properti. Berikan satu contoh kapan kamu harus menggunakan Label, dan kapan kamu harus memilih TextBox saat membuat aplikasi di App Inventor.

Jawaban: Label digunakan untuk menampilkan teks statis yang tidak bisa diubah oleh pengguna, seperti judul atau informasi. Sedangkan TextBox digunakan untuk menerima input dari pengguna. Label memiliki properti penting seperti Text, FontSize, dan TextColor, sedangkan TextBox memiliki Hint, Text, dan MultiLine. Misalnya, gunakan Label untuk menampilkan tulisan "Nama Lengkap" dan TextBox untuk tempat pengguna menulis nama mereka.

Soal 4:

Bayangkan kamu ingin membuat aplikasi "Kartu Nama Digital". Jelaskan langkah-langkah dasar yang harus kamu lakukan di App Inventor 2, mulai dari menambahkan komponen sampai mengatur tata letaknya agar rapi dan mudah dibaca.

Jawaban: Langkah pertama adalah membuka App Inventor dan membuat proyek baru. Tambahkan komponen VerticalArrangement sebagai wadah. Di dalamnya, masukkan komponen Image untuk foto, Label untuk nama, Label untuk email atau nomor telepon, dan Button bertuliskan "Hubungi Saya". Gunakan panel Properties untuk mengatur tampilan teks, gambar, dan posisi agar semua elemen berada di tengah dan tampak rapi.

Soal 5:

Jelaskan bagaimana kamu akan mendesain tampilan aplikasi kalkulator sederhana di App Inventor 2. Sebutkan komponen-komponen yang akan kamu gunakan dan bagaimana kamu menyusunnya agar pengguna mudah menggunakannya.

Jawaban: Saya akan menggunakan dua TextBox untuk input angka, lalu empat Button untuk operasi tambah, kurang, kali, dan bagi. Di bawahnya, saya tambahkan Label untuk menampilkan hasil. Semua komponen ditata menggunakan VerticalArrangement, sedangkan tombol operasi disusun secara horizontal dengan HorizontalArrangement. Saya pastikan ukuran tombol sama dan mudah ditekan.

Soal 6:

Misalkan kamu sedang mendesain tampilan aplikasi catatan harian di App Inventor 2. Bagaimana kamu akan membuat tampilan yang sederhana tetapi fungsional? Jelaskan komponen apa saja yang digunakan dan

bagaimana kamu mengatur ukurannya agar nyaman saat digunakan menulis catatan panjang.

Jawaban: Saya akan menambahkan satu TextBox besar dengan properti MultiLine diaktifkan agar bisa menulis banyak baris. Di bawahnya, saya tambahkan Button dengan teks "Simpan". Untuk layout, saya gunakan VerticalArrangement agar tampilan tertata. Saya atur Width dari TextBox menjadi Fill parent agar melebar memenuhi layar, dan saya posisikan komponen di tengah layar dengan pengaturan AlignHorizontal dan AlignVertical.

Soal 7:

Bayangkan kamu ingin membuat aplikasi "Kartu Nama Digital". Jelaskan langkah-langkah dasar yang harus kamu lakukan di App Inventor 2, mulai dari menambahkan komponen sampai mengatur tata letaknya agar rapi dan mudah dibaca.

Soal 8:

Jelaskan bagaimana kamu akan mendesain tampilan aplikasi kalkulator sederhana di App Inventor 2. Sebutkan komponen-komponen yang akan kamu gunakan dan bagaimana kamu menyusunnya agar pengguna mudah menggunakannya.

Soal 9:

Misalkan kamu sedang mendesain tampilan aplikasi catatan harian di App Inventor 2. Bagaimana kamu akan membuat tampilan yang sederhana tetapi fungsional? Jelaskan komponen apa saja yang digunakan dan bagaimana kamu mengatur ukurannya agar nyaman saat digunakan menulis catatan panjang.

Tugas Proyek

Proyek 1: Buku Harian Digital

Deskripsi: Buatlah aplikasi sederhana untuk mencatat kegiatan atau perasaan harian.

Komponen Utama:

- **TextBox** (multi-baris) untuk menulis catatan
- **Button** untuk menyimpan catatan

Langkah Saran:

1. Gunakan **TextBox** dengan ukuran besar agar nyaman saat mengetik.
2. Tambahkan **Button** bertuliskan “Simpan” atau “Tambahkan Catatan”.

3. Gunakan warna dan font yang menenangkan seperti biru atau hijau pastel.
4. Coba tambahkan **Notifikasi** saat catatan berhasil disimpan.

Tantangan: Tambahkan fitur tanggal otomatis menggunakan komponen **Clock**!

Proyek 2: Kalkulator Cerdas Mini

Deskripsi: Rancang kalkulator sederhana untuk melakukan operasi matematika dasar.

Komponen Utama:

- **2 TextBox** untuk memasukkan angka
- **4 Button** untuk operasi tambah, kurang, kali, dan bagi
- **1 Label** untuk menampilkan hasil

Langkah Saran:

1. Atur tampilan agar rapi dan mudah digunakan (gunakan **HorizontalArrangement** dan **VerticalArrangement**).
2. Tambahkan kode blok logika untuk masing-masing tombol operasi.
3. Pastikan hasil langsung muncul di **Label** setelah tombol ditekan.
4. Buat desain responsif yang tetap rapi di ukuran layar berbeda.

Tantangan: Tambahkan tombol "Hapus" untuk mengosongkan isian!

Proyek 3: Kartu Identitas Digital

Deskripsi: Ciptakan aplikasi yang menampilkan identitas dirimu dalam bentuk digital.

Komponen Utama:

- **Image** (foto profil)
- **Label** untuk nama lengkap, kelas, dan kontak
- **Button** untuk menampilkan informasi tambahan

Langkah Saran:

1. Gunakan **VerticalArrangement** agar komponen tersusun dari atas ke bawah.
2. Pastikan ukuran gambar proporsional dan teks mudah dibaca.
3. Buat tombol "Lihat Info" yang akan menampilkan detail tambahan saat ditekan.
4. Gunakan kombinasi warna yang serasi dan menarik perhatian.

Tantangan: Tambahkan tombol untuk mengganti warna latar aplikasi!

Proyek 4: Aplikasi Sambutan Interaktif

Deskripsi: Buat halaman pembuka aplikasi yang menyapa pengguna dengan tampilan menarik.

Komponen Utama:

- **Label** dengan teks "Selamat Datang"
- **Image** (logo sekolah atau ikon menarik)
- **Button** bertuliskan "Mulai"

Langkah Saran:

1. Gunakan **Screen1** sebagai halaman utama.
2. Posisikan semua komponen di tengah layar dengan **AlignHorizontal dan AlignVertical** ke "Center".
3. Tambahkan aksi saat tombol ditekan, misalnya berpindah ke screen lain.
4. Buat teks sambutan yang hangat, seperti "Hai, siap belajar hari ini?"

Tantangan: Tambahkan animasi sederhana saat membuka aplikasi!

Proyek 5: Mini App Favorit-ku

Deskripsi: Coba tiru aplikasi favoritmu dengan versi yang lebih sederhana dan kreatif!

Komponen Minimal:

- **3 Komponen visual** (misalnya Label, Image, Button)
- **1 Layout pengatur tampilan**
- **1 Button** sebagai aksi utama

Langkah Saran:

1. Pilih aplikasi yang kamu sukai (contoh: media sosial, game, pemutar musik).
2. Rancang tampilan versi sederhananya.
3. Gunakan **Button** untuk berpindah layar atau menampilkan konten.
4. Pastikan desainnya mirip, tetapi tetap original!

Tantangan: Tambahkan efek suara atau animasi saat tombol ditekan!

BAB 2: Membuat Aplikasi Sederhana: Kalkulator / Survey

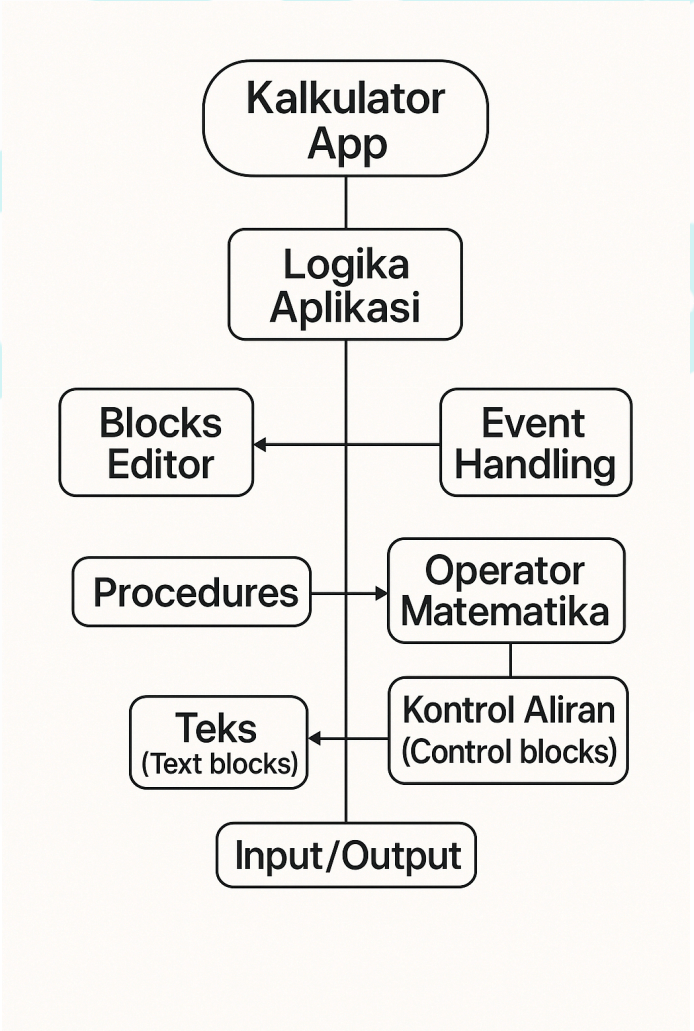
Tujuan Pembelajaran

Setelah mengikuti bab ini, kamu diharapkan mampu:

- Membuat aplikasi sederhana seperti **kalkulator** atau **survey** menggunakan **App Inventor 2**.
- Memahami dan menggunakan **blok kode** untuk menangani **peristiwa (event)**, **variabel**, dan **perhitungan**.
- Menyusun logika aplikasi agar bisa menerima input, memprosesnya, dan menampilkan hasilnya secara interaktif.

Jumlah Pertemuan: 3

Peta Konsep



Apersepsi

Pernahkah kamu bertanya-tanya **bagaimana kalkulator di HP tahu cara menghitung angka yang kamu tekan?**
Atau **bagaimana aplikasi survey bisa menyimpan jawaban kamu satu per satu?**
Rahasia di balik semua itu adalah **logika dan instruksi tersembunyi** yang membuat aplikasi bisa "berpikir" dan "bertindak".

Nah, di bab ini, kamu akan **menjadi programmer visual** yang bisa membuat aplikasi sederhana yang benar-benar bisa digunakan!

Penjelasan Konsep (teori)

"Memberi Nyawa pada Aplikasi Kita!"

Setelah kamu selesai mendesain tampilan aplikasi di **Designer View** — tempat kamu memilih dan mengatur tombol, label, dan elemen lainnya — sekarang saatnya membuat aplikasi tersebut benar-benar *hidup*! Tapi bagaimana caranya?

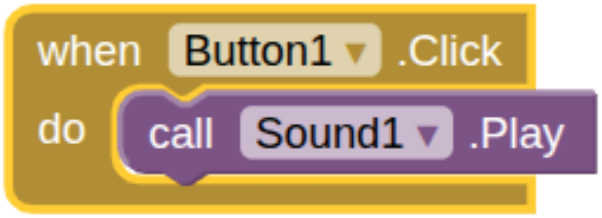
Nah, disinilah kita mengenal **Blocks Editor** dari App Inventor 2. Bayangkan kamu sedang bermain dengan balok-balok LEGO. Tapi kali ini, baloknya bukan untuk membangun rumah atau robot, melainkan untuk membangun **logika dan perintah aplikasi**!

Ayo kita kenali beberapa **komponen penting dalam Blocks Editor** yang akan kamu gunakan untuk membuat kalkulator atau aplikasi survei sederhana.

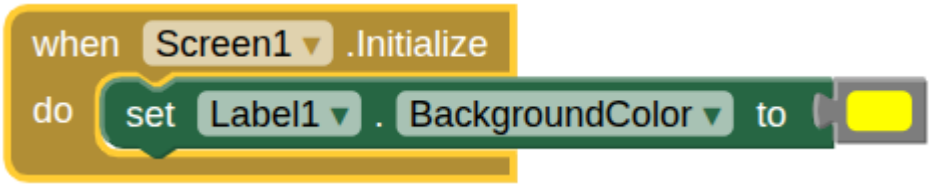
Blok Peristiwa (Event Handling)

Ini seperti "telinga" aplikasi yang mendengarkan peristiwa tertentu. Misalnya: saat tombol diklik, atau saat aplikasi baru dibuka.

- when Button1.Click do
  ➤ Artinya: Jalankan kode di dalam blok ini **setiap kali tombol Button1 ditekan**.



- when Screen1.Initialize do
  ➤ Artinya: Jalankan kode ini **saat aplikasi pertama kali dijalankan**.



Intinya: Blok ini adalah *pemicu aksi*. Tanpa blok event, aplikasi tidak tahu kapan harus melakukan sesuatu.

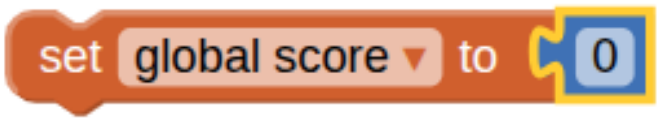
Variabel (Variables)

Variabel adalah **tempat penyimpanan sementara** dalam aplikasi. Seperti kotak kecil yang bisa menyimpan angka, teks, atau jawaban pengguna.

- initialize global score to
  ➤ Digunakan untuk **membuat variabel baru** dan memberi nilai awal. Misalnya: score = 0.



- set global score to
  ➤ Untuk **mengubah nilai** dari variabel yang sudah dibuat.



- get global score
  ➤ Untuk **mengambil nilai** dari variabel tersebut saat dibutuhkan.



Bayangkan variabel sebagai ransel: kamu bisa memasukkan sesuatu ke dalamnya, mengeluarkan, atau mengganti isinya kapan saja!

Blok Matematika (Math Blocks)

Ini adalah blok untuk melakukan **perhitungan matematika** — cocok banget untuk aplikasi kalkulator!

- + Tambah
- − Kurang
- × Kali
- ÷ Bagi



Misalnya: 5 + 3, atau nilai1 × nilai2 — semua bisa dilakukan dengan menggabungkan blok-blok ini.

Menampilkan Teks di Label (Label Blocks)

Setelah perhitungan selesai, tentu kita ingin **menampilkan hasilnya kepada pengguna**, bukan?

- set Label1.Text to
  ➤ Digunakan untuk **mengubah teks** yang tampil di layar pada komponen Label.



Misalnya, kalau hasil perhitungan = 10, maka Label bisa menampilkan: "Hasilnya adalah 10!"

Mengambil Input dari TextBox (TextBox Blocks)

Aplikasi perlu tahu **apa yang diketik pengguna**, misalnya angka yang ingin dihitung atau jawaban survei.

- TextBox1.Text
  ➤ Digunakan untuk **mengambil nilai atau teks** yang dimasukkan pengguna di kotak isian.



Jadi saat kamu menulis “5” di TextBox, aplikasi bisa mengambil angka itu untuk digunakan dalam perhitungan.

Ringkasan

Agar aplikasi kita bisa bekerja dengan baik, kita perlu:

1. **Menangkap aksi pengguna** lewat *event blocks*
2. **Menyimpan data sementara** lewat *variabel*
3. **Menghitung dan mengolah data** dengan *blok matematika*
4. **Menampilkan hasilnya** melalui *Label*
5. **Membaca input pengguna** dari *TextBox*

Tantangan Seru!

Setelah memahami konsep-konsep di atas, kamu akan bisa:

- Membuat aplikasi kalkulator sendiri!
- Mendesain aplikasi survei sederhana yang bisa menyimpan jawaban pengguna!
- Menjadi “programmer visual” dengan balok kode yang menyenangkan!

Ingat! Pemrograman itu seperti memberitahu komputer apa yang harus dilakukan — dan kamu adalah bosnya!
Mari kita mulai menyusun blok dan ciptakan aplikasi pertamamu!

Contoh Kasus atau Ilustrasi

Bayangkan kamu adalah siswa kelas 8 yang sering ditunjuk menjadi ketua kelas. Suatu hari, guru menugaskan untuk mengumpulkan pendapat teman-teman tentang makanan favorit mereka di kantin. Daripada menulis di kertas satu per satu, kamu berpikir: *“Bagaimana kalau aku buat aplikasi kecil untuk survei?”*

Dengan **MIT App Inventor 2**, kamu bisa membuat aplikasi seperti kalkulator atau survey dengan cara menyusun balok kode layaknya menyusun puzzle. Menyenangkan, bukan?

Kasus 1: Kalkulator Penjumlahan Ceria

Deskripsi:
Siswa membuat aplikasi kalkulator sederhana yang bisa menjumlahkan dua angka. Sangat cocok untuk mengenalkan konsep input, proses, dan output.

Blok Utama:

```
When Button1.Click do
  set Label1.Text to
    (convert to number TextBox1.Text) + (convert to number
TextBox2.Text)
```



Tantangan Menarik:
Ubah kalkulator agar bisa juga mengurangkan, mengalikan, dan membagi! Bisa kah kamu membuat pilihan operasi dengan komponen "Spinner"?

Kasus 2: Survey Pilihan Siswa

Deskripsi:
Aplikasi ini menanyakan pendapat pengguna, misalnya: "Apakah kamu suka pelajaran TIK?" dengan dua tombol: **Ya** dan **Tidak**.

Blok Utama:

```
When ButtonYa.Click do
  set LabelHasil.Text to "Kamu suka TIK!"
When ButtonTidak.Click do
  set LabelHasil.Text to "Wah, coba jelaskan kenapa ya!"
```

Tantangan Menarik:
Bisa nggak kamu menambahkan **jumlah total suara** dari tiap pilihan?

Kasus 3: Salam Nama Ajaib

Deskripsi:
Aplikasi akan menyapa pengguna berdasarkan nama yang diketik, contoh: "Halo, Budi! Selamat datang!".

Blok Utama:

```
When ButtonTampilkan.Click do
  set LabelSalam.Text to join("Halo, ", TextBoxNama.Text)
```

Tantangan Menarik:
Buat versi lanjutan: jika nama yang diketik adalah "Guru", tampilkan pesan spesial seperti “Selamat datang, Guru Hebat!”

Kasus 4: Kuis Mini Matematika

Deskripsi:
Tampilkan pertanyaan matematika dan periksa jawaban siswa. Misalnya: "Berapa 7 + 5?"

Blok Utama:

```
When ButtonPeriksa.Click do
  if (convert to number TextBoxJawaban.Text = 12)
    then set LabelHasil.Text to "Benar!"
    else set LabelHasil.Text to "Coba lagi!"
```

Tantangan Menarik:
Bisa nggak kamu buat soal berubah-ubah setiap kali aplikasi dibuka?

Kasus 5: Hitung Umur dari Tahun Lahir

Deskripsi:
Aplikasi ini meminta pengguna memasukkan tahun lahir, lalu menampilkan umur berdasarkan tahun saat ini.

Blok Utama:

```
When ButtonHitung.Click do
  set LabelUmur.Text to
    "Umurmu: " + (2025 - convert to number TextBoxTahun.Text)
```

Tantangan Menarik:
Kembangkan agar bisa menampilkan kategori umur: Anak-anak, Remaja, atau Dewasa.

Kasus 6: Stopwatch Sederhana

Deskripsi:
Menggunakan komponen Clock, aplikasi ini bisa menghitung waktu saat tombol Start ditekan.

Blok Utama:

```
When ButtonStart.Click do
  call Clock1.TimerEnabled to true
When Clock1.Timer do
  set LabelWaktu.Text to (convert to number LabelWaktu.Text) + 1
```

Tantangan Menarik:
Buat juga tombol **Stop** dan **Reset**. Berani coba?

Semua kasus di atas tidak hanya **mengasah logika dan kreativitas**, tapi juga mendorong siswa memahami cara kerja aplikasi modern dengan pendekatan **visual dan menyenangkan**.

Contoh Soal dan Pembahasan

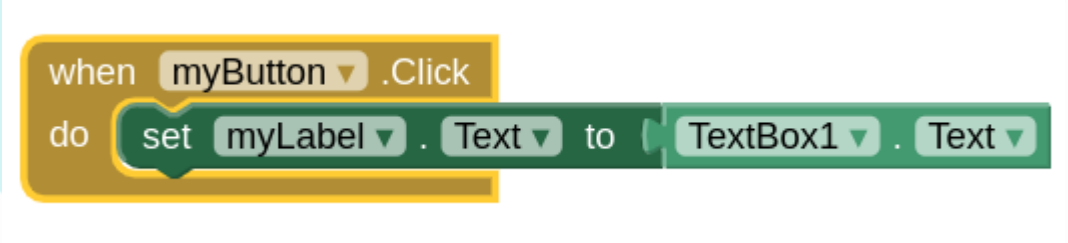
Soal:

Buat kode di App Inventor agar ketika sebuah Button bernama myButton diklik, sebuah Label bernama myLabel akan menampilkan pesan "Hello World!".

Pembahasan:

1. Di Blocks Editor, cari blok When myButton.Click do dari kategori myButton (komponen Button yang telah Anda beri nama myButton). Tarik blok ini ke area kerja.
2. Dari kategori myLabel, cari blok set myLabel.Text to. Tarik dan letakkan di dalam blok When myButton.Click do.
3. Dari kategori Text (blok berwarna ungu), cari blok teks kosong (string kosong ""). Tarik dan letakkan di tempat nilai set myLabel.Text to.

Ketik Hello World! di dalam blok teks kosong tersebut.



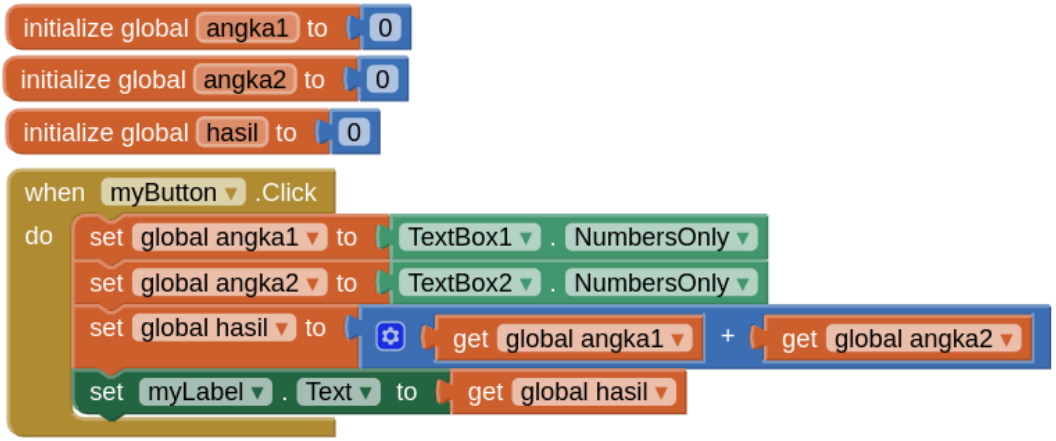
Soal: Aplikasi Penjumlahan Sederhana

Desain:

- **TextBox1** (angka pertama)
- **TextBox2** (angka kedua)
- **myButton** (Hitung)
- **myLabel** (Hasil)

Pseudocode:

```
When myButton.Click do:
    angka1 ← TextBox1.Text
    angka2 ← TextBox2.Text
    hasil ← angka1 + angka2
    set Label.Text to hasil
```



Pembahasan:

- Menggunakan blok **Math:** +
- Ambil input dari **TextBox1** dan **TextBox2**
- Tampilkan hasil ke **Label**

Soal: Menyimpan dan Menampilkan Nama

Desain:

- **TextBox1** (nama)
- **myButton** (Simpan)
- **myLabel** (Halo)

Pseudocode:

```
When myButton.Click do:
    nama ← TextBox1.Text
    set myLabel.Text to "Halo, " + nama
```



Pembahasan:

- Variabel nama diambil dari **TextBox**
- Ditambahkan dengan teks "Halo, "
- Tampil di **Label**

Soal: Aplikasi Kalkulator Pengurangan

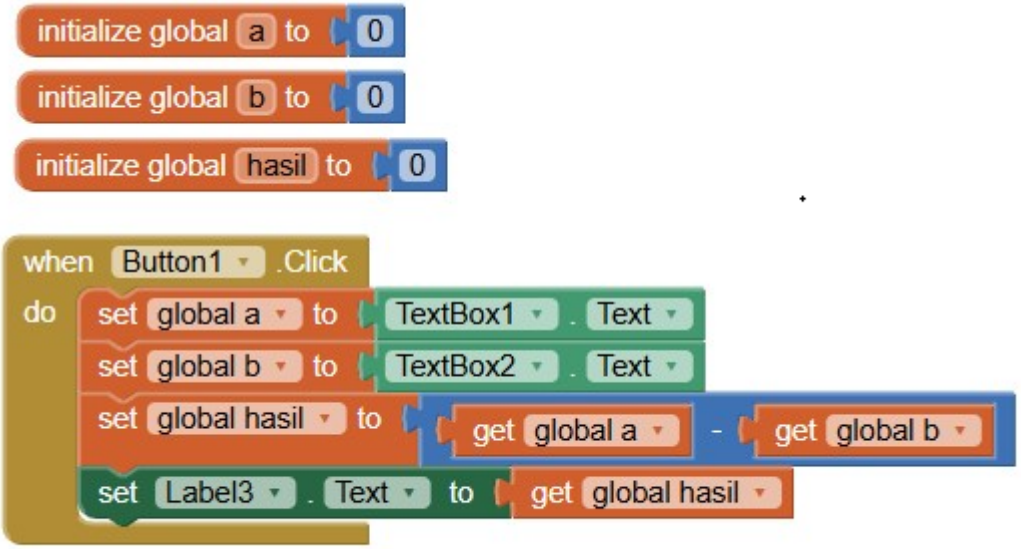
Desain:

- **TextBox1** (angka pertama)
- **TextBox2** (angka kedua)
- **Button** (Kurangi)

- **Label** (Hasil)

  Pseudocode:

  ```
  When Button.Click do:
      a ← TextBox1.Text
      b ← TextBox2.Text
      hasil ← a - b
      set Label.Text to hasil
  ```



Pembahasan:

- Menggunakan blok **Math: -**
- Hasil pengurangan ditampilkan di **Label**

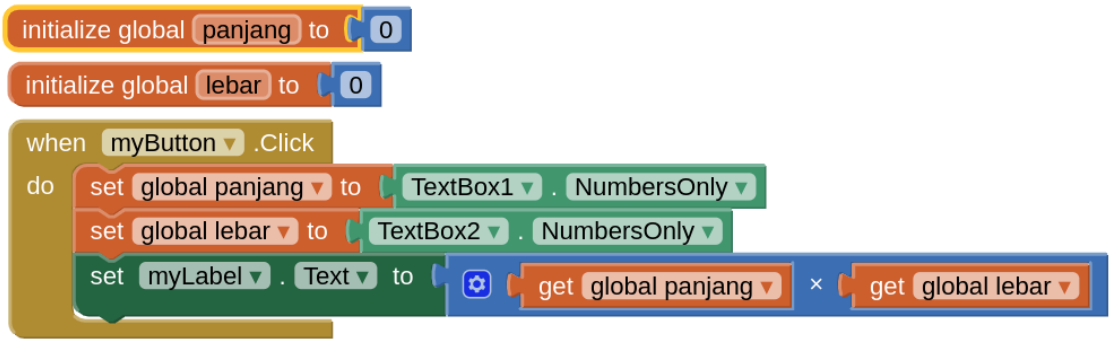
Soal: Aplikasi Luas Persegi Panjang

Desain:

- **TextBox1** (Panjang)
- **TextBox2** (Lebar)
- **Button** (Hitung Luas)
- **Label** (Luas)

  Pseudocode:

  ```
  When Button.Click do:
      panjang ← TextBox1.Text
      lebar ← TextBox2.Text
      luas ← panjang × lebar
      set Label.Text to "Luas = " + luas
  ```



Pembahasan:

- Perkalian menggunakan blok **Math: ×**
- Teks "Luas = ..." ditambahkan di hasil

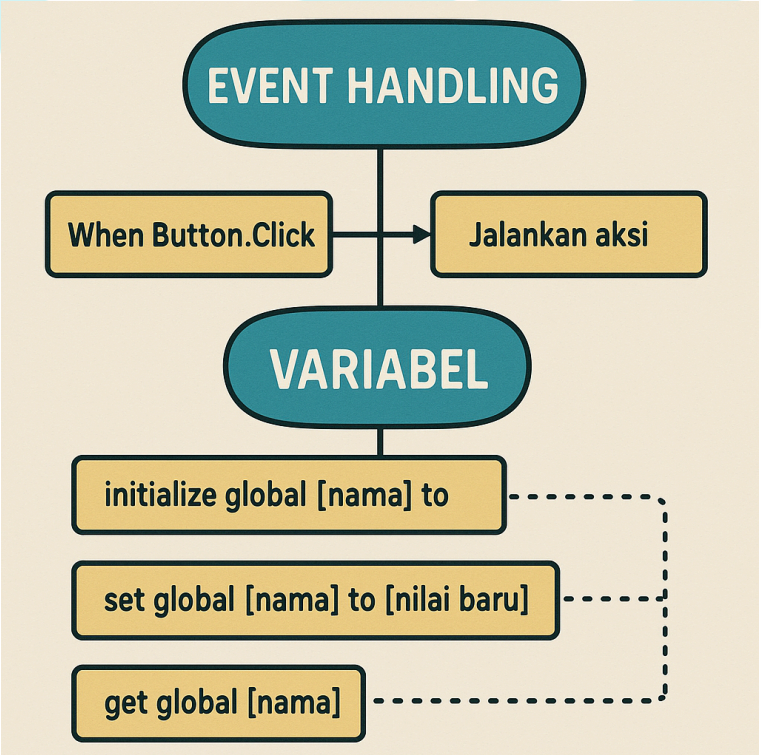
Fakta Menarik

Percaya atau tidak, aplikasi kalkulator adalah salah satu aplikasi paling dasar dan pertama yang dibuat untuk komputer dan smartphone! Dulu, membuat kalkulator digital adalah pencapaian teknologi yang besar!

Tips Cepat Menghafal

Ingat konsep Event Handling dengan kalimat sederhana: **"Apa yang terjadi saat sebuah komponen diklik atau diubah, maka lakukan ini!"** (When [component].[event] do [action]).

Mind Mapping: Event Handling & Variabel



Cerita Pengingat:

Bayangkan tombol adalah seperti bel rumah. Ketika kamu tekan (event), orang dalam rumah akan membuka pintu (aksi). Begitu juga tombol di App Inventor: kita menyuruh aplikasi melakukan sesuatu saat tombol ditekan!

Aktivitas Siswa

Tombol Ganti Warna Latar

Tujuan: Siswa belajar membuat tombol yang bisa mengubah warna latar layar aplikasi.
Langkah-langkah:

1. Buka App Inventor 2.
2. Buat proyek baru dengan nama "GantiWarna".
3. Tambahkan komponen **Button** dan **Screen.**
4. Ubah tulisan tombol jadi "Ganti Warna".
5. Masuk ke tab **Blocks** untuk membuat kode.

Pseudocode:

```
Ketika Button ditekan:
Jika warna latar saat ini adalah putih:
Ubah warna latar jadi biru
Kalau tidak:
Ubah warna latar jadi putih
```

Kalkulator Penjumlahan Sederhana

Tujuan: Siswa belajar menerima input dan menampilkan hasil perhitungan.
Langkah-langkah:

1. Tambahkan 2 komponen **TextBox** untuk input angka.
2. Tambahkan **Button** dengan teks "Hitung".
3. Tambahkan **Label** untuk menampilkan hasil.
4. Gunakan tab **Blocks** untuk membuat logika perhitungan.

Pseudocode:

```
Ketika Button ditekan:
Ambil angka1 dari TextBox1
Ambil angka2 dari TextBox2
jumlah = angka1 + angka2
Tampilkan jumlah di Label
```

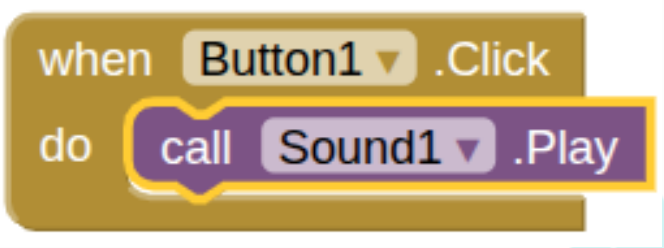
Aplikasi Suara (Play Sound)

Tujuan: Siswa belajar menambahkan suara dan menjalankannya lewat tombol.
Langkah-langkah:

1. Tambahkan komponen Button dan Sound (dari Media).

2. Upload file suara (contoh: “suara.mp3”).
3. Setel **Sound1** ke file suara tersebut.
4. Buat blok agar tombol memainkan suara.

Pseudocode:
Ketika Button ditekan:
Mainkan suara menggunakan komponen Sound



Quiz Sederhana (Pertanyaan Ya/Tidak)

Tujuan: Siswa membuat aplikasi tanya-jawab sederhana dan mendapat umpan balik.
Langkah-langkah:

1. Tambahkan komponen **Label** untuk pertanyaan.
2. Tambahkan dua **Button:** satu bertuliskan “Ya”, satu “Tidak”.
3. Tambahkan **Label** untuk jawaban/umpan balik.

Pseudocode:
Ketika tombol "Ya" ditekan:
Tampilkan "Jawaban benar!" di Label
Ketika tombol "Tidak" ditekan:
Tampilkan "Jawaban salah!" di Label

Rangkuman

Di **bab ini**, kita sudah berhasil **membuat aplikasi kita "hidup"** dengan menambahkan *logika* menggunakan **Blocks Editor**. Kita mulai dari **mendesain tampilan** aplikasi, lalu belajar bagaimana menambahkan "perintah" atau *kode visual* agar aplikasi bisa **merespons tindakan pengguna**, seperti saat tombol ditekan.

Kita mengenal konsep penting seperti ***Event Handling*** (*When [komponen].[peristiwa] do*) yang digunakan untuk memberi tahu aplikasi *apa yang harus dilakukan saat sesuatu terjadi*—misalnya saat tombol diklik. Selain itu, kita juga belajar tentang **variabel**, yaitu "wadah data sementara", dan **blok matematika** yang digunakan untuk **menghitung angka** serta **blok teks** untuk **menampilkan hasil** di layar.

Dengan menguasai blok-blok seperti ***When Button.Click do***, ***set Label.Text to***, dan ***convert to number***, kita sekarang bisa membuat **aplikasi interaktif sederhana sendiri**! Ini adalah **langkah awal yang seru dan penting** sebelum kita belajar membuat aplikasi yang lebih kompleks dan canggih. Yuk, terus **eksplorasi dan berkreasi** dengan aplikasi buatanmu sendiri!

Latihan Soal

A. Benar atau Salah (5 Soal)

1. Blok *When Button1.Click do* digunakan untuk memulai suatu aksi ketika tombol diklik.
   Jawaban: Benar

2. Blok *set Label1.Text to* digunakan untuk mengganti warna latar belakang aplikasi.
   Jawaban: Salah

3. Untuk menampilkan hasil perhitungan, kita harus menggunakan komponen Label.
   Jawaban: Benar

4. Variabel di App Inventor tidak bisa menyimpan data teks.
   Jawaban: Salah

5. Blok TextBox1.Text digunakan untuk mengambil nilai dari input pengguna.
   Jawaban: Benar

B. Pilihan Ganda (10 Soal)

6. Blok yang digunakan untuk memulai aksi saat tombol diklik adalah...
   A. *set Label1.Text to*
   B. *TextBox1.Text*
   C. *When Button1.Click do*
   D. *get global name*
   Jawaban: C

7. Untuk menambahkan dua angka di App Inventor, kita menggunakan blok...
   A. *Logic =*
   B. *Text join*
   C. *Math +*
   D. *Control if*
   Jawaban: C

8. Fungsi dari blok *convert to number* adalah...
   A. Menggabungkan dua angka
   B. Mengubah teks menjadi angka
   C. Menyimpan data
   D. Menampilkan teks
   Jawaban: B

9. Blok *initialize global name to* digunakan untuk...
   A. Menghapus variabel
   B. Menambahkan tombol
   C. Membuat variabel baru dan memberikan nilai awal
   D. Mengambil input pengguna
   Jawaban: C

10. Untuk menampilkan hasil pada layar, kita menggunakan blok...
    A. *TextBox1.Text*
    B. *set Label1.Text to*
    C. *join*
    D. *get global name*
    Jawaban: B

11. Komponen yang digunakan untuk menampilkan teks hasil perhitungan adalah...
    A. TextBox
    B. Label
    C. Button
    D. ListView
    Jawaban: B

12. Blok *join* dari kategori Text digunakan untuk...
    A. Menyimpan data
    B. Menjumlahkan angka
    C. Menggabungkan beberapa teks
    D. Menghapus teks
    Jawaban: C

13. Blok *get global name* digunakan untuk...
    A. Membuat variabel
    B. Mengambil nilai dari variabel
    C. Menambahkan variabel
    D. Menyimpan teks
    Jawaban: B

14. Untuk membuat aplikasi survey, kita membutuhkan...
    A. Satu Label saja
    B. Dua TextBox
    C. Label dan beberapa Button
    D. Kompas digital
    Jawaban: C

15. Blok *TextBox1.Text* sebaiknya dibungkus dengan blok *convert to number* jika...
    A. Mengambil data teks
    B. Menyimpan hasil
    C. Melakukan operasi matematika
    D. Menampilkan warna
    Jawaban: C

C. Isian Singkat (5 Soal)

16. Blok yang digunakan untuk menjalankan kode saat aplikasi pertama kali dibuka adalah _____.
    Jawaban: When Screen1.Initialize do

17. Blok untuk mengubah nilai dari sebuah variabel global adalah _____.
    Jawaban: set global name to

18. Untuk menampilkan hasil penjumlahan di Label, kita menggunakan blok _____.
    Jawaban: set Label1.Text to

19. Jika kita ingin menyimpan jawaban “Ya” dalam survey, kita bisa menggunakan blok _____.
    Jawaban: set global jawabanPertanyaan1 to "Ya"

20. Komponen yang digunakan pengguna untuk mengetik angka disebut _____.
    Jawaban: TextBox

D. Soal Explorasi

Soal:

Jelaskan secara langkah demi langkah bagaimana kamu membuat aplikasi penjumlahan dua angka di App Inventor mulai dari menambahkan komponen di Designer View hingga menyusun logikanya di Blocks Editor. Tuliskan langkah-langkah dengan urutan yang jelas.

Jawaban: Langkah pertama dalam membuat aplikasi penjumlahan dua angka di App Inventor adalah membuka proyek baru, lalu di **Designer View**, tambahkan dua komponen **TextBox** (TextBox1 dan TextBox2) untuk input angka, satu **Button** (ButtonAdd) dengan teks "Jumlahkan", dan satu **Label** (LabelResult) untuk menampilkan hasil. Setelah itu, pindah ke **Blocks Editor** dan buat blok *When ButtonAdd.Click do*. Di dalamnya, gunakan blok *set LabelResult.Text to*, lalu hubungkan dengan blok *Math* +. Ambil nilai dari *TextBox1.Text* dan *TextBox2.Text*, lalu bungkus masing-masing dengan blok *convert to number* agar bisa dijumlahkan. Gabungkan hasil perhitungan dengan teks "Hasil: " menggunakan blok *join*, lalu tampilkan hasil di LabelResult.

Soal:

Kamu ingin membuat aplikasi survey sederhana yang menanyakan 3 pertanyaan "Ya/Tidak", dan menampilkan ringkasan jawaban ketika tombol "Lihat Hasil" ditekan. Buatlah rancangan logika blok yang perlu disusun di Blocks Editor agar aplikasi ini bisa bekerja. Jelaskan bagaimana kamu menyimpan dan menampilkan data pengguna.

Jawaban: Untuk membuat aplikasi survey dengan 3 pertanyaan "Ya/Tidak", pertama di Designer View tambahkan 3 **Label** untuk menampilkan pertanyaan dan masing-masing diberi 2 **Button** untuk jawaban "Ya" dan "Tidak", lalu tambahkan **ButtonLihatHasil** dan **LabelSummary**. Di Blocks Editor, buat 3 variabel global untuk menyimpan jawaban (misal: *jawabanPertanyaan1*, *jawabanPertanyaan2*, *jawabanPertanyaan3*). Saat tombol "Ya" atau "Tidak" diklik pada setiap pertanyaan, gunakan blok *set global jawabanX to* untuk menyimpan jawaban sesuai tombol yang dipilih. Saat *ButtonLihatHasil* diklik, buat blok *When ButtonLihatHasil.Click do*, lalu gunakan blok *set LabelSummary.Text to* dan gabungkan semua jawaban menggunakan blok *join* agar muncul ringkasan seperti: "Pertanyaan 1: Ya, Pertanyaan 2: Tidak, Pertanyaan 3: Ya".

Soal 3:

Modifikasi aplikasi kalkulatormu agar bisa menjumlahkan, mengurangkan, dan mengalikan dua angka. Jelaskan bagaimana kamu menambahkan tombol dan logika baru untuk operasi pengurangan dan perkalian. Buat urutan blok yang kamu perlukan agar aplikasi bekerja sesuai harapan.

Jawaban: Untuk memodifikasi aplikasi kalkulator agar bisa menjumlahkan, mengurangkan, dan mengalikan dua angka, tambahkan dua **Button** baru di Designer View: satu bernama *ButtonSubtract* dengan teks "Kurangi" dan satu lagi *ButtonMultiply* dengan teks "Kali". Di Blocks Editor, buat blok *When ButtonSubtract.Click do* dan *When ButtonMultiply.Click do*. Di masing-masing blok, tambahkan blok *set LabelResult.Text to*, lalu gunakan blok *Math* - untuk pengurangan dan

Math × untuk perkalian. Ambil nilai dari *TextBox1.Text* dan *TextBox2.Text*, bungkus dengan blok *convert to number*, dan lakukan operasi sesuai tombol. Gunakan blok *join* untuk menambahkan teks seperti “Hasil: ” sebelum hasil perhitungan ditampilkan di LabelResult.

Tugas Proyek

Proyek 1: Aplikasi Mini Survei "Ya atau Tidak"

Deskripsi:
Buatlah sebuah aplikasi sederhana yang menyajikan tiga pertanyaan pilihan "Ya" atau "Tidak" kepada pengguna. Aplikasi ini akan menyimpan setiap jawaban dan menampilkan ringkasan respon di akhir.

Langkah-langkah:

- Rancang tampilan dengan tiga pertanyaan dan dua tombol ("Ya" dan "Tidak") untuk masing-masing pertanyaan.
- Gunakan **variabel** untuk menyimpan jawaban pengguna.
- Tambahkan tombol "Lihat Hasil" untuk menampilkan ringkasan jawaban dalam sebuah **Label**.

Proyek 2: Kalkulator Mini – Tiga Operasi Matematika

Deskripsi:
Buat kalkulator sederhana yang mampu melakukan tiga operasi: penjumlahan, pengurangan, dan perkalian antara dua angka.

Langkah-langkah:

- Tambahkan dua **TextBox** untuk input angka.
- Sediakan tiga tombol: “Tambah (+)”, “Kurang (–)”, dan “Kali (×)”.
- Gunakan blok **Math +, -, dan *** untuk menghitung hasil.
- Tampilkan hasil perhitungan di **Label**.

Proyek 3: Aplikasi Ucapan Nama

Deskripsi:
Aplikasi ini akan meminta pengguna mengetikkan nama mereka, lalu menampilkan sapaan ramah, seperti: "Halo, Budi!"

Langkah-langkah:

- Tambahkan **TextBox** untuk input nama.
- Buat tombol "Tampilkan Salam".
- Saat tombol ditekan, gunakan blok **join** untuk menggabungkan teks "Halo, " dengan nama pengguna.
- Tampilkan sapaan di **Label**.

Proyek 4: Kuis Pilihan Ganda Seru

Deskripsi:
Rancang kuis interaktif dengan 3 pertanyaan pilihan ganda (A, B, C). Aplikasi akan menghitung skor berdasarkan jawaban yang benar.

Langkah-langkah:

- Buat tiga pertanyaan, masing-masing dengan tiga tombol jawaban.
- Gunakan **variabel** untuk menyimpan skor.
- Tambahkan logika untuk memeriksa apakah jawaban benar, lalu tambahkan nilai.
- Setelah semua dijawab, tampilkan skor akhir di **Label**.

Proyek 5: Konversi Panjang – Meter ke Sentimeter

Deskripsi:
Buat aplikasi yang bisa mengubah satuan dari meter ke sentimeter (1 meter = 100 cm). Cocok untuk melatih konsep konversi satuan.

Langkah-langkah:

- Tambahkan **TextBox** untuk input angka dalam meter.
- Tambahkan tombol "Konversi".
- Saat ditekan, kalikan angka dengan 100 menggunakan blok **Math ×**.
- Tampilkan hasil dalam **Label**.

Proyek 6: Stopwatch Sederhana

Deskripsi:
Kembangkan stopwatch digital yang bisa memulai, menghentikan, dan mereset waktu. Proyek ini menantang dan menyenangkan!

Langkah-langkah:

- Gunakan **Clock** component dari App Inventor.
- Tambahkan tiga tombol: “Mulai”, “Berhenti”, dan “Reset”.
- Gunakan variabel untuk menghitung waktu setiap detik.
- Tampilkan waktu secara real-time di **Label**.

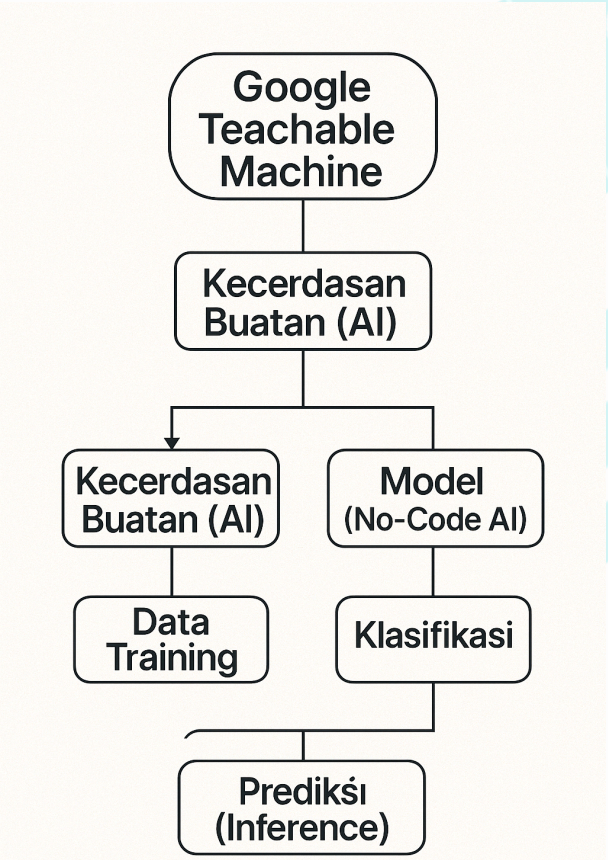
BAB 3: Pengenalan Google Teachable Machine (Teori)

Tujuan Pembelajaran

Peserta didik memahami konsep dasar pengenalan gambar, suara, dan pose menggunakan Google Teachable Machine sebagai pengantar Machine Learning sederhana, serta perbedaan antara data training dan prediksi.

Jumlah Pertemuan: 2

Peta Konsep

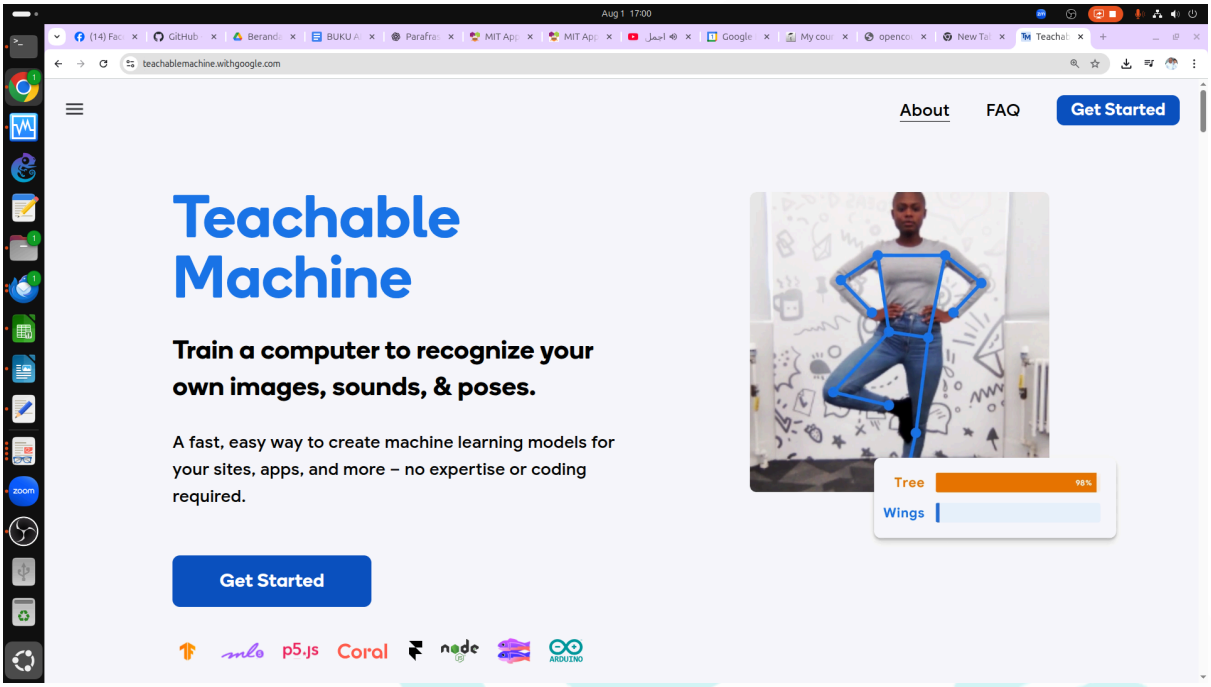


Apersepsi

Pernahkah kalian perhatikan bagaimana HP kalian bisa mengenali wajah kalian saat membuka kunci, atau bagaimana Instagram bisa secara otomatis menambahkan filter ke wajah kalian? Itu semua adalah contoh bagaimana komputer "belajar" dan "melihat" seperti kita. Di bab ini, kita akan belajar tentang alat ajaib bernama Google Teachable Machine yang memungkinkan kita "mengajari" komputer melakukan hal-hal pintar seperti itu!

Penjelasan Konsep (teori)

Apa itu Google Teachable Machine?



Bayangkan jika kalian bisa "mengajari" komputer untuk mengenali wajah teman, membedakan suara tepuk tangan dari suara kucing, atau bahkan mengenali pose tubuh saat kalian sedang berdiri atau duduk — semuanya **tanpa perlu menulis kode pemrograman**!

Nah, inilah kehebatan dari **Google Teachable Machine**.

> **Google Teachable Machine** adalah alat berbasis web dari Google yang memungkinkan siapa saja — termasuk kalian! — untuk membuat model pembelajaran mesin (Machine Learning) secara **mudah dan menyenangkan**.

Kalian hanya perlu memberikan contoh-contoh data (seperti gambar, suara, atau gerakan), dan komputer akan belajar mengenali pola dari data tersebut. Tidak perlu jadi ahli komputer atau programmer untuk memulainya!

Mengapa Belajar Machine Learning (ML) Itu Penting?

Coba perhatikan hal-hal di sekitar kalian:

- HP yang bisa membuka kunci hanya dengan wajah.
- Aplikasi kamera yang secara otomatis mengenali wajah dan memberi filter.
- Asisten virtual seperti Google Assistant atau Siri yang bisa menjawab pertanyaan kalian.

Semua itu bekerja menggunakan teknologi yang disebut **Machine Learning (ML)**. Tapi bagaimana caranya komputer bisa "belajar"?

Bagaimana Cara Kerja Machine Learning?

Kita akan mulai dengan contoh sederhana:
Mengajarkan komputer membedakan antara apel dan jeruk.

Kalau tanpa Machine Learning:

Kita harus membuat aturan manual, contohnya:

- "Kalau buahnya bulat dan merah, itu apel."
- "Kalau kulitnya oranye dan kasar, itu jeruk."

Tapi… bagaimana jika ada **apel hijau** atau **jeruk yang halus**? Aturan kita bisa gagal.

Nah, dengan Machine Learning:
Komputer **belajar sendiri** dari banyak contoh, tanpa perlu kita kasih aturan satu per satu.

Langkah-Langkah Proses Machine Learning:

1. **Mengumpulkan Data Training**
    - Kita berikan banyak contoh ke komputer.
    - Contohnya: 100 gambar apel (merah, hijau, besar, kecil) dan 100 gambar jeruk.
    - Ini disebut *data training*.

2. **Training Model**
    - Komputer mempelajari pola dari data tersebut.
    - Misalnya: warna, bentuk, tekstur, dan lain-lain.
    - Komputer membangun "model" berdasarkan pola-pola itu.

3. **Menguji Model dengan Data Baru (Prediksi)**
    - Kita beri gambar buah yang belum pernah dilihat komputer.
    - Komputer menebak: ini apel atau jeruk?
    - Proses ini disebut *prediksi* atau *inference*.

Perbedaan Penting

Istilah	Penjelasan
Data Training	Data awal yang digunakan untuk melatih komputer. Semakin beragam dan jelas, hasil belajarnya makin bagus.
Prediksi (Inference)	Saat model yang sudah terlatih mencoba mengenali data baru dan memberikan hasil tebakan.

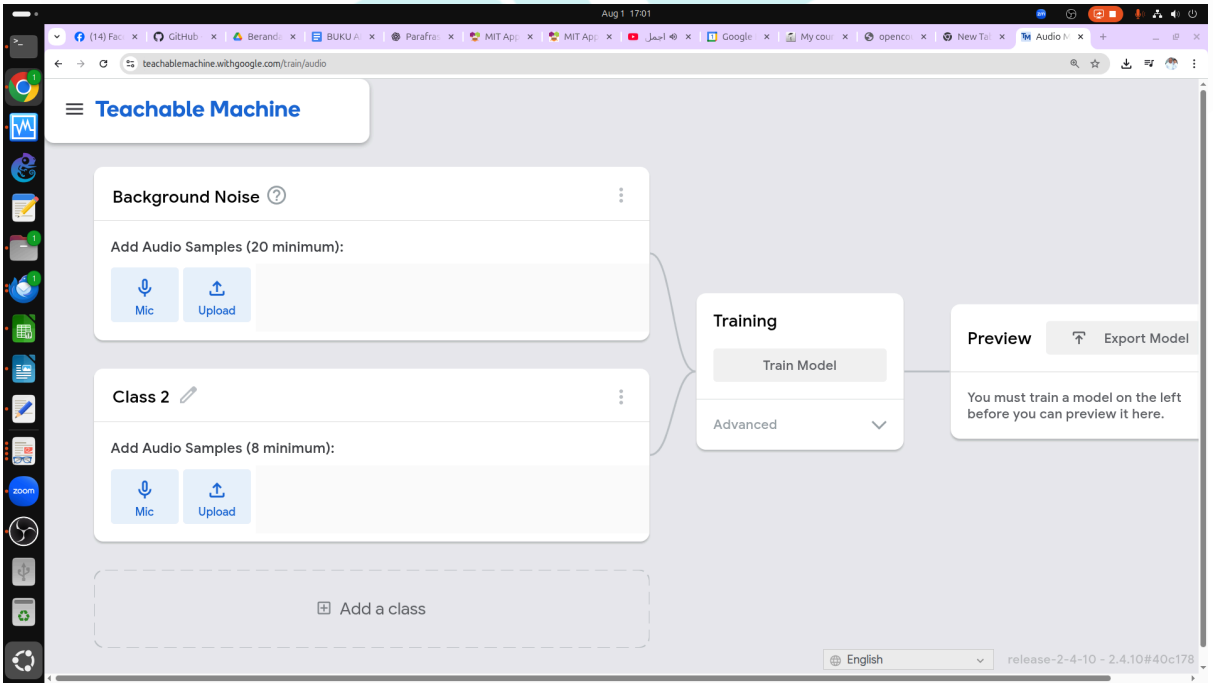
Jenis Proyek di Google Teachable Machine

Setelah membuka https://teachablemachine.withgoogle.com, kalian akan menemukan tiga jenis proyek seru. Masing-masing punya kegunaan yang berbeda:

1. **Image Project**
   - Untuk mengenali **gambar**.
   - Contoh: komputer mengenali ekspresi wajah atau benda di sekitar.

2. **Audio Project**
   - Untuk mengenali **suara**.
   - Contoh: suara tepuk tangan, tawa, atau suara binatang.

3. **Pose Project**
   - Untuk mengenali **gerakan atau posisi tubuh**.
   - Contoh: saat berdiri, jongkok, mengangkat tangan.



Kesimpulan Seru!

Dengan Google Teachable Machine, kalian bisa:

- Mengerti dasar-dasar bagaimana **komputer bisa belajar** seperti manusia.
- Bereksperimen langsung membuat model AI.
- Merasakan langsung serunya dunia teknologi cerdas.

Siap jadi pelatih untuk komputer?
Mari kita mulai petualangan seru di dunia Machine Learning!

Contoh Kasus atau Ilustrasi

Kasus 1: Mengenali Hewan Peliharaan: Si Kucing Miko dan Si Anjing Bimo

Proyek: Image Project
Tujuan: Melatih komputer mengenali hewan yang berbeda berdasarkan gambar.

Ilustrasi:
Bayangkan kamu punya dua sahabat berbulu: Miko si kucing lucu dan Bimo si anjing setia. Saat kamu pergi sekolah, kamu penasaran siapa yang paling sering muncul di depan kamera. Nah, kamu bisa melatih komputer untuk mengenali Miko dan Bimo hanya dengan foto mereka!
Kamu kumpulkan banyak foto Miko dan Bimo dari berbagai sudut dan pencahayaan, lalu masukkan ke Teachable Machine. Setelah model dilatih, kamu tinggal tunjukkan gambar baru, dan komputer akan menebak: “Itu Miko!” atau “Itu Bimo!”

Kasus 2: Mengenali Ekspresi Wajah: Senyum vs Sedih

Proyek: Image Project
Tujuan: Komputer membedakan wajah senyum dan wajah sedih.

Ilustrasi:
Kamu dan temanmu sedang bermain game ekspresi! Sekarang, giliran komputer yang jadi penebak. Kumpulkan foto wajah teman-teman saat mereka tersenyum dan saat mereka menunjukkan ekspresi sedih.
Masukkan semua gambar itu ke Teachable Machine dalam dua kelas: "Senyum" dan "Sedih". Setelah dilatih, komputer bisa jadi juri lomba ekspresi: cukup tunjukkan wajahmu, dan dia bisa bilang: "Wah, kamu lagi sedih, ya?" atau "Senyum cerah banget hari ini!"

Kasus 3: Membedakan Suara: Meong vs Gonggong

Proyek: Audio Project
Tujuan: Komputer membedakan suara kucing dan anjing.

Ilustrasi:
Bayangkan kamu punya alat pendeteksi hewan otomatis di rumah. Kalau terdengar suara "meong", artinya kucing sedang kelaparan. Kalau terdengar "gonggong", mungkin ada tamu datang!
Dengan Google Teachable Machine, kamu bisa merekam suara meong dari beberapa kucing dan suara gonggongan dari anjing. Setelah data dilatih, komputer bisa menebak suara baru: "Itu suara kucing!" atau "Anjing lagi menggonggong!"

Kasus 4: Mengenali Gerakan: Duduk vs Berdiri

Proyek: Pose Project
Tujuan: Komputer membedakan apakah seseorang duduk atau berdiri.

Ilustrasi:
Bayangkan kamu sedang membuat sistem otomatis di ruang kelas: kalau siswa berdiri, lampu menyala, kalau duduk, lampu mati.
Dengan menggunakan kamera, kamu bisa melatih komputer mengenali dua pose: duduk dan berdiri. Kamu minta teman-teman bergaya sambil direkam. Setelah dilatih, komputer bisa mendeteksi secara otomatis siapa yang sedang duduk dan siapa yang sedang berdiri!

Kasus 5: Mengenali Isyarat Tangan: "Stop!" dan "Lanjut!"

Proyek: Image Project
Tujuan: Komputer mengenali sinyal tangan tertentu.

Ilustrasi:
Bayangkan kamu jadi pengatur lalu lintas mini! Tangan ke atas artinya "STOP!", dan tangan ke depan artinya "LANJUT!".
Kamu bisa mengajari komputer mengenali isyarat tangan ini dengan mengambil banyak gambar pose tangan saat "stop" dan saat "lanjut". Setelah dilatih, kamu bisa gerakkan tanganmu ke depan kamera dan komputer akan langsung merespon: "Lanjutkan!" atau "Berhenti dulu!"

Kasus 6: Mengenali Suasana Ruangan: Terang vs Gelap

Proyek: Image Project
Tujuan: Komputer membedakan kondisi pencahayaan ruangan.

Ilustrasi:
Kamu ingin membuat alarm otomatis yang menyala saat ruangan tiba-tiba jadi gelap. Caranya? Foto-foto ruangan saat terang dan saat gelap (bisa dimatikan lampu atau tutup jendela).
Masukkan foto-foto ini ke dalam dua kelas di Teachable Machine: “Terang” dan “Gelap”. Setelah model dilatih, komputer bisa menebak kondisi cahaya hanya dari satu gambar ruangan!

Contoh Soal dan Pembahasan

Soal:

Apa fungsi utama dari tombol "Train Model" di Google Teachable Machine?
Pembahasan: Tombol "Train Model" berfungsi untuk memulai proses pembelajaran pada model AI. Saat Anda mengklik tombol ini, Google Teachable Machine akan mengambil semua Data Training yang telah Anda kumpulkan untuk setiap kelas, menganalisisnya, dan menemukan pola-pola atau fitur-fitur unik yang membedakan satu kelas dari kelas lainnya. Hasil dari proses training ini adalah sebuah "model" yang siap digunakan untuk melakukan Prediksi terhadap data baru. Ibarat seorang guru, tombol "Train Model" adalah saat guru itu sedang mengajari siswanya dari semua contoh yang diberikan.

Soal:

Apa fungsi tombol "Train Model" di Google Teachable Machine?
Jawaban:
Untuk memulai proses belajar AI dari data yang telah dikumpulkan. Tombol ini membuat komputer memproses semua data training dan mengenali pola.

Soal:

Apa perbedaan antara Data Training dan Data Prediksi?
Jawaban:

- **Data Training:** Digunakan untuk melatih model AI agar mengenal pola dari data.
- **Data Prediksi:** Data baru yang belum dikenali, digunakan untuk menguji apakah AI bisa menebak dengan benar.

Soal:

Jika kamu ingin mengenali suara bel sepeda dan suara lonceng sekolah, proyek apa yang digunakan?

Jawaban:

Audio Project. Karena proyek ini khusus untuk mengenali jenis suara.

Soal:

Mengapa data training yang beragam penting?

Jawaban:

Agar model AI bisa mengenali banyak variasi data di dunia nyata dan tidak hanya menghafal satu jenis gambar/suara saja.

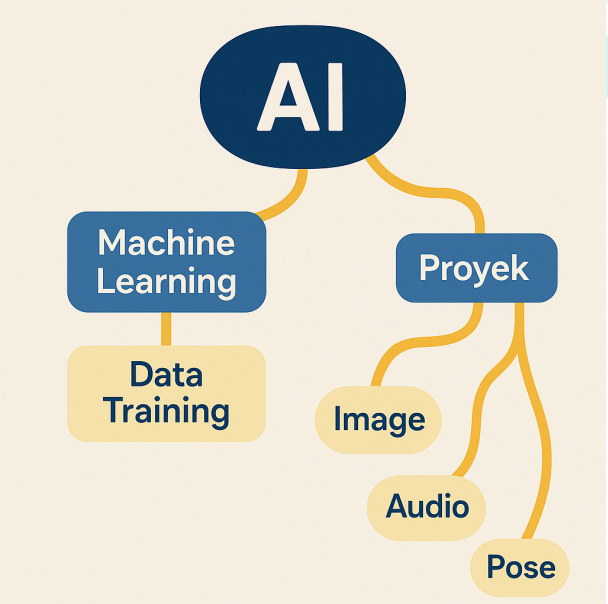
Fakta Menarik

Dulu, hanya para ilmuwan dan peneliti yang bisa membuat AI. Tapi sekarang, dengan alat seperti Teachable Machine, siapa pun bisa "mengajari" AI, bahkan mengenali kucing dan anjing hanya dengan beberapa klik! Ini menunjukkan betapa mudahnya AI semakin bisa diakses oleh semua orang.

Tips Cepat Menghafal

Ingat analogi: **"Kita Ajar Komputer dengan Contoh, Komputer Belajar, Lalu Dia Bisa Menebak!"** Proses ini adalah inti dari Machine Learning.

Mind Mapping



Cerita Mudah Diingat:

Bayangkan kalian sedang di kelas komputer, dan saya kasih banyak gambar apel dan jeruk ke komputer—ada yang merah, kuning, oranye, besar, kecil,

semuanya! Komputer melihat satu per satu dan mulai belajar membedakan mana apel, mana jeruk. Setelah belajar cukup banyak, saya tunjukkan gambar baru dan tanya, "Ini buah apa, ya?" Komputer langsung jawab, "Itu apel!" Nah, itulah yang disebut proses prediksi—komputer menebak berdasarkan apa yang sudah dipelajari sebelumnya. Seru, kan?

Tips Cepat Menghafal:

- Gunakan singkatan: TIP = Training-Inference-Prediction
- Gabungkan belajar dengan gambar atau video.
- Latihan langsung membuat proyek kecil.

Aktivitas Siswa

Aktivitas 1: Menonton Video Tutorial tentang Google Teachable Machine

1. Akses situs Google Teachable Machine: teachablemachine.withgoogle.com
2. Tonton video perkenalan singkat yang ada di halaman utama atau cari video tutorial dasar "Getting Started with Teachable Machine" di YouTube.
3. Catatlah hal-hal penting yang kalian pelajari dari video tersebut, seperti:
    - Jenis-jenis proyek yang bisa dibuat.
    - Bagaimana cara menambahkan kelas.
    - Bagaimana cara mengumpulkan data.
    - Bagaimana proses training model terjadi.
    - Bagaimana cara menguji model yang sudah dilatih.

Aktivitas 2: Ekspresi Detektor

Deskripsi:
Kamu akan membuat proyek AI yang bisa mengenali ekspresi wajah temanmu, seperti senyum, sedih, atau kaget.
Langkah-langkah:

1. Foto wajah teman dengan berbagai ekspresi (senyum, sedih, kaget).
2. Kumpulkan minimal 5 foto untuk tiap ekspresi.
3. Latih model AI untuk mengenali ekspresi.
4. Uji coba model ke wajah teman yang lain, dan lihat apakah prediksi benar.

Pseudocode:

```
Jika gambar_input = wajah_temanku
    jika model_mengenali = "senyum"
        tampilkan "Kamu sedang senyum!"
    jika model_mengenali = "sedih"
        tampilkan "Kamu terlihat sedih."
    jika model_mengenali = "kaget"
        tampilkan "Kamu kaget ya?"
```

Aktivitas 3: Game Tepuk vs Ketukan

Deskripsi:

Kamu dan temanmu akan bermain game AI. Satu merekam suara, satu lagi melatih AI untuk mengenali suara tepuk dan ketukan.

Langkah-langkah:

1. Rekam 5 suara tepuk dan 5 suara ketukan.
2. Simpan suara dengan nama yang benar (misal: tepuk1.wav).
3. Gunakan AI untuk belajar membedakan suara tepuk dan ketukan.
4. Uji AI dengan suara baru untuk melihat prediksi.

Pseudocode:

```
Jika suara_input = suara_baru
    jika model_mengenali = "tepuk"
        tampilkan "Ini suara tepuk."
    jika model_mengenali = "ketukan"
        tampilkan "Ini suara ketukan."
```

Aktivitas 4: Buah Ajaib

Deskripsi:

Kamu akan membuat model AI untuk mengenali jenis buah seperti apel, pisang, dan jeruk.

Langkah-langkah:

1. Ambil 10 gambar buah (apel, jeruk, pisang).
2. Latih AI menggunakan gambar yang kamu ambil.
3. Cari gambar buah dari internet dan uji ke AI.
4. Lihat apakah AI bisa mengenali buahnya dengan benar.

Pseudocode:

```
Jika gambar_input = gambar_buah
    jika model_mengenali = "apel"
        tampilkan "Ini apel"
    jika model_mengenali = "pisang"
        tampilkan "Ini pisang"
    jika model_mengenali = "jeruk"
        tampilkan "Ini jeruk"
```

Aktivitas 5: Deteksi Suasana Musik

Deskripsi:

Kamu akan melatih AI untuk mengenali suasana dari musik, seperti sedih, bahagia, atau menegangkan.

Langkah-langkah:

1. Cari 3 lagu dengan suasana yang berbeda: sedih, bahagia, dan menegangkan.
2. Potong bagian pendek dari tiap lagu (10–15 detik).
3. Latih AI dengan potongan lagu tersebut.
4. Uji AI dengan lagu lain dan lihat prediksinya.

Pseudocode:

```
Jika musik_input = lagu_baru
   jika model_mengenali = "sedih"
      tampilkan "Lagu ini terdengar sedih."
   jika model_mengenali = "bahagia"
      tampilkan "Lagu ini membuat senang!"
   jika model_mengenali = "menegangkan"
      tampilkan "Lagu ini terasa menegangkan!"
```

Rangkuman

Di bab ini, kita mengenal **Google Teachable Machine**, alat *ajaib* yang membantu kita memahami **AI (Artificial Intelligence)** tanpa harus menulis kode. Dengan alat ini, kita bisa membuat model AI sendiri hanya dengan memberikan **contoh data**. Inilah yang disebut **Machine Learning**—bukan memprogram komputer secara rinci, tapi **"mengajari" komputer lewat contoh**, atau yang kita sebut **Data Training**. Setelah belajar dari data, model AI bisa melakukan **prediksi** saat melihat data baru. Kita juga belajar membuat proyek seru dengan tiga jenis input: **gambar (*Image*)**, **suara (*Audio*)**, dan **gerakan tubuh (*Pose*)**. Proses pentingnya adalah:

Data Training → Train Model → Prediksi

Selain itu, kita jadi paham pentingnya **variasi data** agar AI makin pintar, dan kita langsung mencoba sendiri lewat proyek nyata yang **menyenangkan**. Ini adalah langkah awal yang **seru dan mudah dimengerti** untuk mulai menjelajahi dunia AI sejak SMP!

Latihan Soal

A. Benar atau Salah (5 Soal)

1. Google Teachable Machine bisa digunakan tanpa harus menulis kode.
   Jawaban: Benar

2. Data Training digunakan hanya setelah proses prediksi selesai.
   Jawaban: Salah

3. Salah satu jenis proyek di Teachable Machine adalah mengenali suara.
   Jawaban: Benar

4. Setelah model AI dilatih, ia tidak bisa digunakan lagi.
   Jawaban: Salah

5. Prediksi dilakukan menggunakan data yang sudah pernah dilatih.
   Jawaban: Salah

B. Pilihan Ganda (10 Soal)

6. Apa fungsi utama dari tombol "Train Model" di Teachable Machine?
   A. Menghapus data
   B. Menyimpan proyek
   C. Melatih model AI
   D. Mencetak hasil
   Jawaban: C

7. Apa yang dimaksud dengan "Data Training"?
   A. Data yang digunakan untuk membuat aplikasi
   B. Data yang digunakan untuk melatih model AI
   C. Data yang dihapus setelah uji coba
   D. Data yang digunakan untuk mempercantik tampilan
   Jawaban: B

8. Apa nama proyek di Teachable Machine untuk mengenali gambar?
   A. Audio Project
   B. Text Project
   C. Image Project
   D. Video Project
   Jawaban: C

9. Model AI membuat "Prediksi" saat:
   A. Menyimpan gambar
   B. Menggunakan data baru yang belum dikenali

C. Mengedit proyek
D. Mengulang data training
Jawaban: B

10. Berikut ini yang termasuk contoh Data Training adalah:
A. File PDF
B. Gambar hewan yang dikenali sebelumnya
C. Video musik
D. Link internet
Jawaban: B

11. Untuk mengenali suara tepuk tangan, proyek yang dipilih adalah:
A. Pose Project
B. Image Project
C. Audio Project
D. Video Project
Jawaban: C

12. Teachable Machine bisa mengenali pose seperti:
A. Berlari dan membaca
B. Duduk dan berdiri
C. Tidur dan bermain
D. Menulis dan menyanyi
Jawaban: B

13. Proses mengenali pola dari data disebut:
A. Printing
B. Scanning
C. Training
D. Zooming
Jawaban: C

14. Data baru yang dimasukkan untuk diuji ke model disebut:
A. Data Training
B. Data Lama
C. Data Sementara
D. Data Prediksi
Jawaban: D

15. Kelebihan Teachable Machine adalah:
A. Harus diinstal dulu
B. Hanya bisa digunakan dengan coding
C. Bisa digunakan tanpa coding
D. Hanya bisa dipakai di HP
Jawaban: C

C. Isian Singkat (5 Soal)

16. Google Teachable Machine adalah alat berbasis _____ dari Google.
    Jawaban: web

17. Data yang digunakan untuk menguji kemampuan model disebut _____.
    Jawaban: prediksi

18. Proyek yang digunakan untuk mengenali posisi tubuh disebut _____ Project.
    Jawaban: Pose

19. Semakin banyak dan beragam data training, maka model AI akan semakin _____.
    Jawaban: akurat

20. Untuk mengenali wajah tersenyum, kita harus membuat kelas "Senyum" dan mengumpulkan banyak _____ wajah tersenyum.
    Jawaban: gambar

D. Soal Explorasi

Soal:

Jelaskan bagaimana kamu akan membuat proyek di Google Teachable Machine yang bisa membedakan antara suara ketukan jari dan suara tepuk tangan. Tuliskan langkah-langkahmu dari memilih jenis proyek, mengumpulkan data, melatih model, sampai mengujinya.

Jawaban: Saya akan memilih proyek "Audio Project" karena ingin mengenali suara. Lalu saya membuat dua kelas: satu untuk suara ketukan jari, dan satu lagi untuk suara tepuk tangan. Saya merekam beberapa contoh suara untuk masing-masing kelas dari berbagai arah dan kekuatan suara. Setelah data cukup, saya klik "Train Model" agar komputer belajar membedakannya. Terakhir, saya coba tepuk tangan di depan mikrofon untuk melihat apakah AI bisa menebak dengan benar.

Soal:

Bayangkan kamu ingin membuat model yang bisa membedakan buah apel, pisang, dan jeruk. Jelaskan apa yang kamu lakukan untuk mengumpulkan Data Training yang baik. Tuliskan strategi kamu agar modelnya bisa akurat.

Jawaban: Saya akan membuat tiga kelas: Apel, Pisang, dan Jeruk. Untuk tiap buah, saya akan mengambil foto dari berbagai sudut, pencahayaan, dan ukuran buah yang berbeda. Misalnya apel hijau dan merah, pisang

mentah dan matang, jeruk kecil dan besar. Saya juga akan memastikan gambarnya jelas dan tidak buram. Dengan data yang beragam, model akan lebih pintar membedakan ketiganya.

Soal:

Menurutmu, mengapa Google Teachable Machine disebut sebagai "No-Code AI"? Jelaskan kelebihan alat ini bagi pelajar seperti kamu yang belum bisa coding.

Jawaban: Google Teachable Machine disebut "No-Code AI" karena kita tidak perlu menulis kode untuk membuat AI. Semua prosesnya bisa dilakukan dengan klik dan rekam data. Ini sangat bagus untuk pelajar SMP karena kita bisa langsung mencoba membuat AI tanpa harus belajar bahasa pemrograman dulu. Jadi, lebih cepat paham dan menyenangkan untuk belajar AI sejak dini.

Tugas Proyek

Proyek 1: "Detektor Ekspresi Wajah"

Tujuan: Mengenali ekspresi wajah seperti **senyum**, **sedih**, dan **kaget**.
Jenis Proyek: *Image Project*

Langkah-Langkah:

1. Buat tiga kelas: *Senyum*, *Sedih*, dan *Kaget*.
2. Kumpulkan minimal 10 gambar wajah dari teman-teman untuk setiap ekspresi.
3. Latih model dengan gambar-gambar tersebut.
4. Uji apakah model dapat mengenali ekspresi orang baru secara akurat.

Kenapa Menarik?
Ekspresi wajah sering digunakan dalam aplikasi seperti kamera pintar atau robot sosial. Cocok untuk mengenalkan siswa pada konsep **komputer vision** dan **pengolahan citra**.

Proyek 2: "Suara Unik Sehari-hari"

Tujuan: Membedakan suara **bel rumah**, **tepuk tangan**, dan **tertawa**.
Jenis Proyek: *Audio Project*

Langkah-Langkah:

1. Rekam suara bel rumah, tepuk tangan, dan tertawa masing-masing minimal 10 kali.
2. Buat tiga label sesuai suara.
3. Latih dan uji model dengan suara baru.

4. Gunakan fitur live-mic untuk menguji secara real-time.

Kenapa Menantang?
Siswa belajar bahwa **suara** adalah data penting bagi AI dan bagaimana AI bisa membedakan pola-pola audio.

Proyek 3: "Pose Pemain Bola"

Tujuan: Membedakan pose tubuh: **berdiri siap**, **menendang**, dan **melompat**.
Jenis Proyek: *Pose Project*

Langkah-Langkah:
1. Ambil video singkat atau pose dari teman-teman saat berolahraga.
2. Buat tiga label berdasarkan pose.
3. Latih model dengan berbagai contoh agar variasi pose tertangkap.
4. Uji model dengan gerakan tubuh secara langsung.

Kenapa Seru?
Siswa bisa aktif bergerak sambil belajar AI. Ini juga memperkenalkan **pengolahan gerakan dan postur tubuh** dalam AI.

Proyek 4: "Tebak Isyarat Tangan"

Tujuan: Mengenali isyarat seperti **jempol ke atas**, **tangan terbuka**, dan **peace sign**.
Jenis Proyek: *Image Project*

Langkah-Langkah:
1. Ambil foto tangan dengan 3 pose berbeda.
2. Latih model untuk mengenali setiap pose.
3. Uji dengan tangan orang lain untuk melihat akurasinya.

Kenapa Layak Dicoba?
Isyarat tangan bisa digunakan untuk **kendali tanpa sentuh** – cocok untuk pengenalan AI dalam kehidupan nyata (misalnya remote pintar, atau sistem tanpa tombol).

Proyek 5: "AI Penolong Harian – Cerita AI-ku"

Tujuan: Membuat cerita pendek tentang pengalaman membangun proyek AI.
Jenis Proyek: *Penulisan Naratif Kreatif*

Langkah-Langkah:

1. Ceritakan dalam 5–10 kalimat tentang proses membuat proyek AI (misalnya mengenali ekspresi wajah atau suara).
2. Tambahkan imajinasi tentang bagaimana AI tersebut bisa membantu orang di rumah, sekolah, atau masyarakat.
3. Boleh dihias dengan gambar atau animasi jika diinginkan.

Kenapa Penting?
Melatih siswa untuk **berpikir reflektif dan kreatif** tentang peran AI dalam kehidupan manusia.

Proyek 6: "Kuis AI: Prediksi vs Latihan"

Tujuan: Memahami konsep **Data Training**, **Prediksi**, dan **Akurasi Model AI**.
Jenis Proyek: *Diskusi dan Simulasi*

Langkah-Langkah:

1. Buat dua set data: satu untuk melatih model, satu lagi untuk menguji.
2. Coba dengan jumlah data yang sedikit dan yang banyak – bandingkan hasilnya.
3. Diskusikan: Mengapa AI kadang salah menebak? Apa yang bisa diperbaiki?
4. Bahas konsep **No-Code AI** dan mengapa Google Teachable Machine masuk kategori itu.

Pengetahuan Tambahan yang Wajib Dipahami Siswa:

- **Data Training** adalah data yang digunakan untuk *melatih* AI agar mengenali pola.
- **Prediksi** adalah hasil *dugaan AI* berdasarkan data baru.
- **AI bisa salah** karena data training kurang, tidak bervariasi, atau terlalu mirip antar kelas.
- **No-Code AI** artinya kita bisa membuat AI tanpa perlu menulis kode — Google Teachable Machine adalah contohnya.
- **Kualitas dan jumlah data sangat penting!** Jika data terlalu sedikit atau mirip-mirip, AI bisa bingung dan salah menebak.

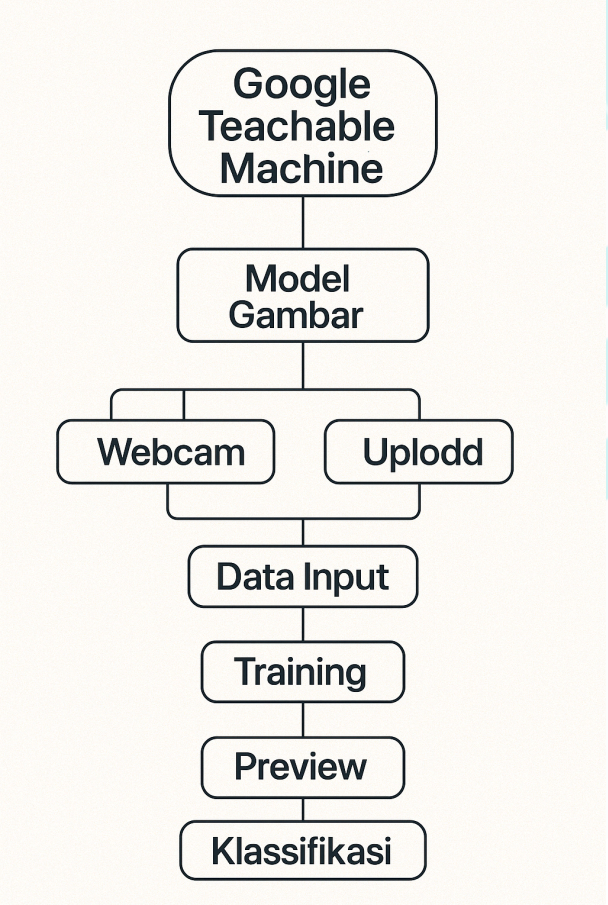
BAB 4: Praktik dengan Google Teachable Machine: Model Gambar Sederhana (3 Pertemuan)

Tujuan Pembelajaran

Peserta didik mampu secara mandiri membuat model AI sederhana untuk mengenali gambar menggunakan Google Teachable Machine, termasuk mengumpulkan data, melatih model, dan menguji akurasinya.

Jumlah Pertemuan: 3

Peta Konsep



Apersepsi

Di bab sebelumnya, kita sudah belajar teori tentang bagaimana Teachable Machine bekerja. Sekarang, mari kita coba langsung! Bayangkan kita akan membuat AI yang bisa mengenali apakah tangan kita terbuka atau tertutup. Ini seperti kita mengajari AI cara "melihat" dan "memahami" gambar!

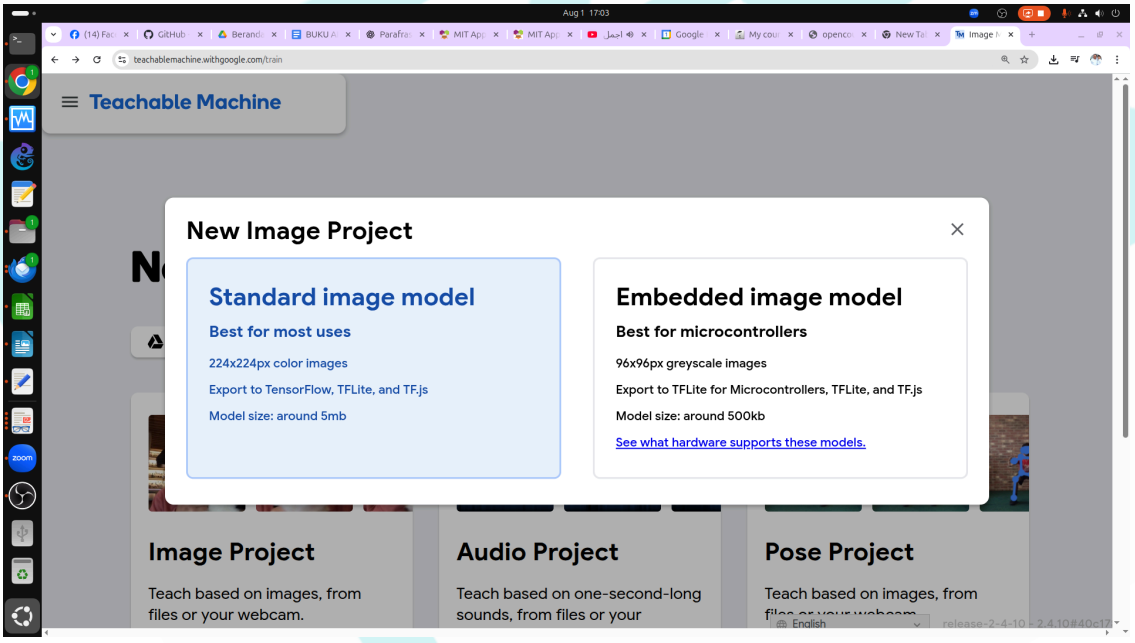
Penjelasan Konsep (teori)

Pernahkah kamu berpikir bagaimana komputer bisa mengenali wajahmu atau membedakan antara tangan terbuka dan tertutup? Nah, itulah salah satu kemampuan dari **kecerdasan buatan (Artificial Intelligence atau AI)**, dan kamu bisa membuatnya sendiri menggunakan alat online yang mudah dan menyenangkan, yaitu **Google Teachable Machine**!

Teachable Machine adalah platform dari Google yang memungkinkan siapa saja — termasuk kamu — untuk **melatih komputer agar bisa mengenali gambar, suara, atau pose tubuh**, hanya dengan beberapa langkah sederhana. Yuk, kita pelajari bagaimana cara membuat **model pengenalan gambar (image model)** menggunakan alat keren ini!

Langkah-langkah Membuat Model Gambar

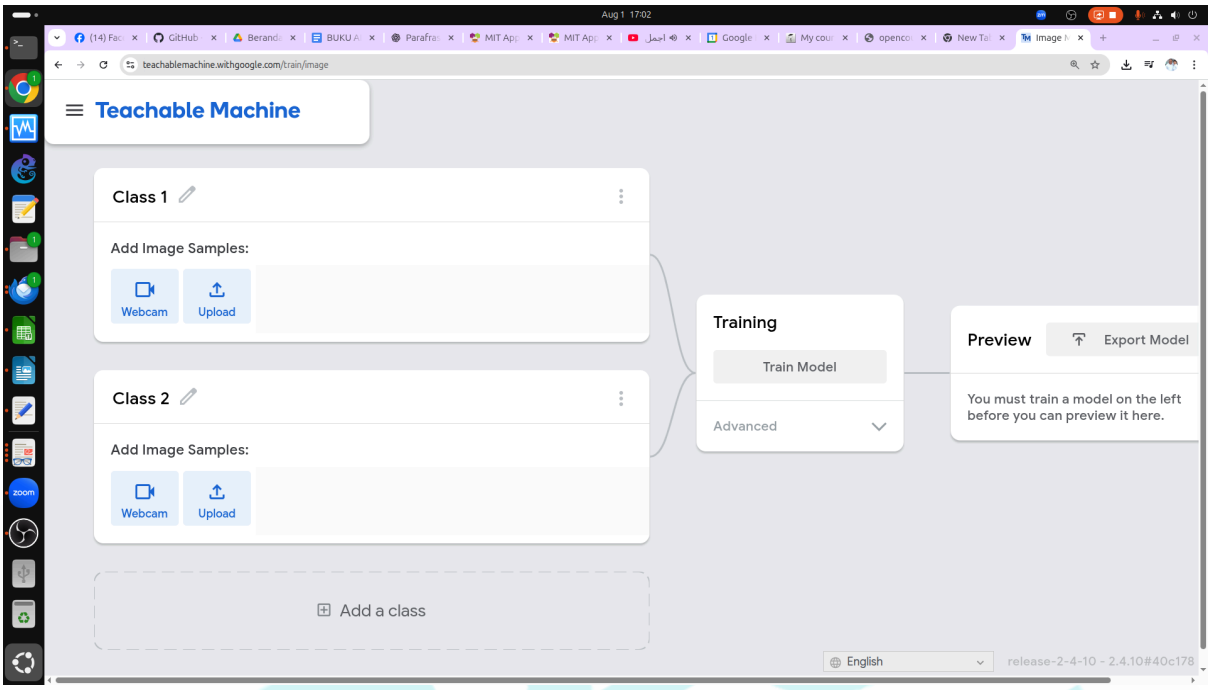
Memulai Proyek Baru



- Buka browser (seperti Chrome atau Firefox), lalu ketik alamat berikut: teachablemachine.withgoogle.com.
- Klik tombol **"Get Started"** untuk memulai.
- Pilih jenis proyek: **"Image Project"**.
- Setelah itu, akan muncul dua pilihan:
    - **Standard Image Model**: model biasa untuk penggunaan umum.
    - **Embedded Image Model**: untuk perangkat kecil seperti Arduino.
    - ➤ *Pilih yang "Standard Image Model" ya!*

Tujuan: Di sini kamu memulai lembar kerja untuk melatih AI mengenali gambar yang kamu pilih.

Menambahkan dan Mengatur Kelas (Class)



Setiap model harus bisa mengenali **beberapa kategori**. Nah, kategori-kategori ini disebut sebagai **kelas (class)**.

- Di awal, kamu akan melihat dua kelas default: **Class 1** dan **Class 2**.
- Klik ikon **pensil** di sebelah nama kelas, dan ubah namanya menjadi sesuatu yang bermakna. Misalnya:
    - Ganti **Class 1** menjadi "Tangan Terbuka"
    - Ganti **Class 2** menjadi "Tangan Tertutup"
- Jika perlu, kamu bisa **menambahkan kelas lain** (misalnya: "Tangan Membentuk Huruf L")

Tujuan: Setiap kelas mewakili objek atau kondisi yang ingin kamu ajarkan kepada AI.

Mengumpulkan Data Latih (Training Data)

Nah, sekarang saatnya "mengajarkan" AI dengan memberi contoh sebanyak-banyaknya!
Ada dua cara untuk mengisi data gambar ke setiap kelas:

Menggunakan Webcam

- Klik tombol **"Webcam"** di bawah kelas yang kamu pilih.
- Izinkan browser mengakses kamera.
- Arahkan objek (misalnya tanganmu) ke kamera sesuai kelas.
- Tekan dan tahan tombol **"Hold to Record"** untuk merekam gambar dari berbagai sudut dan kondisi pencahayaan.

Mengunggah Gambar

- Kamu juga bisa klik tombol **"Upload"** dan memilih gambar dari komputer.

Catatan Penting

- Berikan **jumlah gambar yang seimbang** untuk setiap kelas!
- Gunakan variasi posisi, latar belakang, dan pencahayaan agar model menjadi pintar dan akurat.

Tujuan: AI butuh belajar dari banyak contoh, sama seperti kita!

Melatih Model (Training)

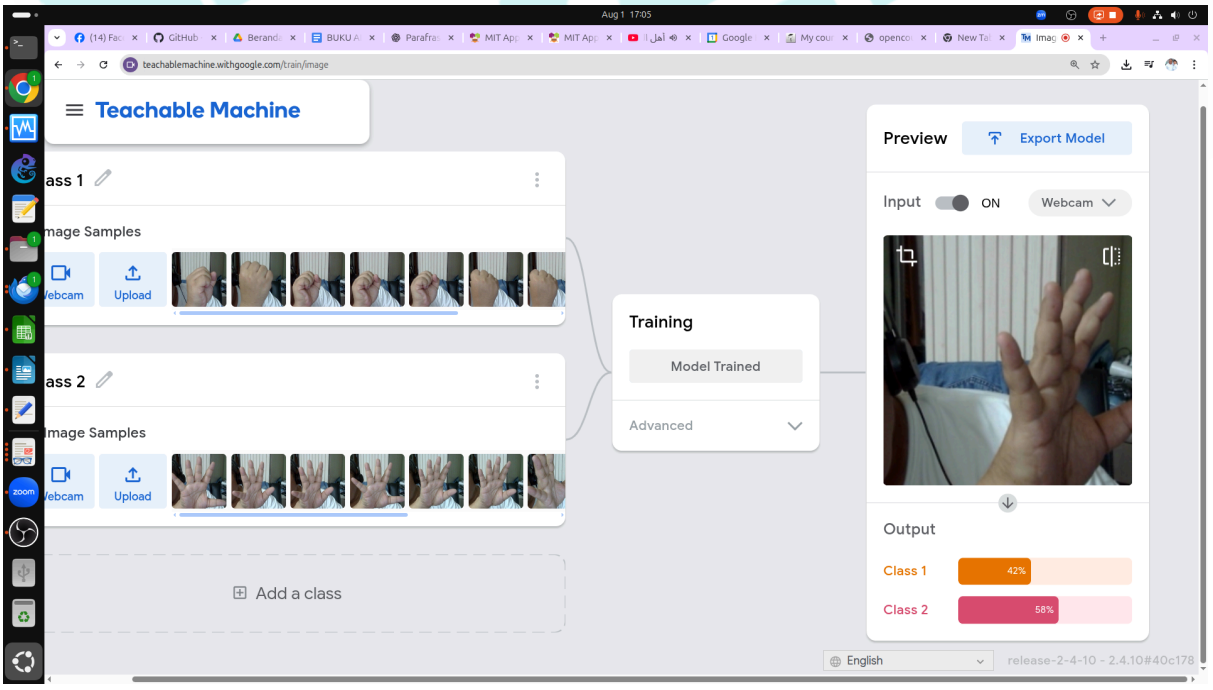
Setelah kamu selesai mengisi gambar untuk semua kelas...

- Klik tombol besar **"Train Model"** (berwarna biru).
- Tunggu sebentar. Komputer akan memproses data, biasanya hanya dalam hitungan detik atau menit.
- Akan muncul status seperti:
  - "Preparing training data..."
  - "Training model..."
  - "Model trained!" (berarti AI kamu siap digunakan)

Jangan tutup tab atau pindah halaman saat proses training berlangsung!

Tujuan: Di sini, sistem mulai "belajar" dari data yang kamu berikan.

Menguji Model (Testing / Preview)



Sekarang saatnya melihat apakah AI kamu bekerja!

- Lihat panel sebelah kanan dengan judul **"Preview"**.
- Pilih input: **Webcam** atau **Upload File**.
- Arahkan objek ke kamera, misalnya tangan terbuka.
- Hasil akan muncul dalam bentuk **persentase** untuk setiap kelas.

Misalnya:
"Tangan Terbuka: 95%"
"Tangan Tertutup: 5%"

Semakin tinggi persentasenya, semakin yakin AI bahwa itulah kelas yang sesuai.

Jika hasilnya belum akurat:

- Tambahkan lebih banyak contoh gambar.
- Ulangi proses training.

Tujuan: Ini adalah tahap untuk menguji seberapa baik AI kamu dalam mengenali gambar.

Kesimpulan

Dengan Google Teachable Machine, kamu tidak hanya belajar tentang teknologi AI, tetapi juga **menjadi pelatihnya langsung!** Seru, kan?

Kegiatan ini melatih:

- **Logika berpikir**
- **Kreativitas dalam pengambilan gambar**
- **Kesabaran dan ketelitian**

Dan yang paling penting, kamu bisa **bereksperimen sambil bermain**!

Jika kamu sudah menguasai cara ini, kamu bisa lanjut belajar:

- Pengenalan suara (Sound Project)
- Pengenalan pose tubuh (Pose Project)

Ayo, ciptakan model AI buatanmu sendiri. Siapa tahu suatu hari nanti kamu bisa menciptakan teknologi yang mengubah dunia!

Ilustrasi: Memecahkan Kasus dengan Teachable Machine

Bayangkan kamu menjadi seorang ilmuwan muda yang sedang mencoba membuat robot asisten yang bisa membantumu membedakan antara *apel* dan *jeruk*. Robot ini belum tahu apa-apa, jadi tugasmu adalah *mengajarinya* dengan cara menunjukkan banyak gambar apel dan jeruk. Setiap gambar seperti "pelajaran" bagi si robot!

Nah, lewat **Google Teachable Machine**, kamu bisa *melatih AI* (Artificial Intelligence) untuk mengenali gambar hanya dengan menggunakan **kamera laptop** dan koneksi internet. Cukup rekam beberapa gambar objek, latih modelnya, dan lihat hasilnya di *Preview*! Mudah dan seru, kan?

Contoh Kasus atau Ilustrasi

Kasus 1: Si Juara Buah — Apel vs. Jeruk

Ilustrasi Kasus:
Bayangkan kamu menjadi juri dalam lomba buah paling fotogenik. Tugasmu adalah membuat model AI yang bisa membedakan **apel** dan **jeruk**, hanya dengan melihat gambar.

Langkah Petualangan:

- Buat dua kelas: **"Apel"** dan **"Jeruk"**.
- Kumpulkan banyak gambar apel dan jeruk (bisa dari internet atau langsung difoto dengan webcam).
- Latih model dan uji keakuratannya.

Coba tantang temanmu dengan menunjukkan buah sungguhan—apakah modelmu bisa menebak dengan benar?

Kasus 2: Detektif Ekspresi — Senyum, Netral, atau Sedih?

Ilustrasi Kasus:
Bayangkan kamu punya robot teman yang bisa membaca perasaanmu hanya dari wajahmu. Di sini, kamu akan melatih AI untuk mengenali **ekspresi wajah**:
Senyum 😊, **Netral** 😐, dan **Sedih** 😢.

Langkah Petualangan:

- Buat tiga kelas berdasarkan ekspresi.
- Rekam wajahmu sendiri dengan ekspresi berbeda-beda.
- Uji: Tunjukkan ekspresi secara acak, dan lihat apakah AI bisa "membaca hatimu".

Seru untuk pelajaran tentang emosi dan komunikasi non-verbal!

Kasus 3: Tangan Ajaib — Terbuka atau Tertutup?

Ilustrasi Kasus:
Bayangkan kamu membuat game sederhana yang dikontrol dengan **gerakan tangan**. AI-nya harus bisa tahu apakah tanganmu **terbuka** atau **tertutup**.

Langkah Petualangan:

- Buat dua kelas: "Tangan Terbuka" dan "Tangan Tertutup".
- Ambil gambar tanganmu dari berbagai arah dan pencahayaan.
- Uji model dengan berbagai posisi tangan—bisa jadi dasar membuat game atau robot mini!

Tantangan: Bisa tidak kamu buat AI-nya mengenali "pose salam lima"?

Kasus 4: Detektor Alat Tulis — Pensil, Penghapus, Penggaris

Ilustrasi Kasus:
Bayangkan AI yang bisa membantu temanmu mencari alat tulis di meja—tanpa harus mencarinya satu per satu! Di sini, AI akan dilatih untuk mengenali **pensil**, **penghapus**, dan **penggaris**.

Langkah Petualangan:

- Buat tiga kelas: satu untuk setiap benda.
- Foto setiap benda dari berbagai sudut, pencahayaan, bahkan warna.
- Coba uji model: apakah bisa membedakan pensil dan pulpen? Bisa dijadikan alat bantu visual.

Berguna untuk teman yang memiliki keterbatasan penglihatan atau dalam situasi gelap!

Kasus 5: Sensor Camilan — Cokelat, Keripik, atau Biskuit?

Ilustrasi Kasus:
Bayangkan kamu membuat vending machine pintar yang tahu jenis camilan hanya dari tampilannya. AI akan mengenali apakah itu **keripik**, **cokelat**, atau **biskuit**.

Langkah Petualangan:

- Buat tiga kelas: "Keripik", "Biskuit", "Cokelat".
- Foto camilan dari berbagai merek, ukuran, dan posisi.
- Latih model dan uji: bisakah dia mengenali camilan favoritmu?

Belajar sambil ngemil? Siapa takut!

Kasus 6: Pemindai Suara — Suara "Ya" vs. "Tidak"

Ilustrasi Kasus:
Kamu bisa membuat AI yang bisa membedakan dua suara: **"Ya"** dan **"Tidak"**. Bisa untuk kontrol suara mini!

Langkah Petualangan:

- Rekam suara "ya" dan "tidak" dalam berbagai nada dan kecepatan.
- Uji dengan suara kamu dan temanmu. Apakah AI bisa mengenalinya?

Cocok untuk membuat tombol virtual pakai suara!

Setiap kasus ini seperti **misi sains seru**, dan dengan Google Teachable Machine, kamu bisa langsung mencobanya di kelas atau di rumah. Tantang dirimu dan temanmu untuk membuat AI paling pintar dan paling seru! Mau mencoba salah satu sekarang?

Contoh Soal dan Pembahasan

Soal:

Bagaimana cara melatih model di Google Teachable Machine agar hasil prediksinya lebih akurat? Sebutkan minimal 3 hal yang bisa dilakukan.

Pembahasan: Untuk melatih model agar lebih akurat, ada beberapa hal yang bisa dilakukan:

1. **Tambah Jumlah Data Training:** Semakin banyak contoh yang Anda berikan untuk setiap kelas, semakin banyak pola yang dapat dipelajari oleh model. Targetkan minimal 50-100 contoh per kelas, atau bahkan lebih jika memungkinkan.
2. **Variasi Data Training:** Jangan hanya memberikan contoh yang sama. Misalnya, untuk wajah senyum, rekam wajah senyum dengan pencahayaan berbeda, sudut berbeda, ekspresi senyum yang sedikit berbeda (lebar, tipis), dan latar belakang yang bervariasi. Untuk objek, foto objek dari berbagai sudut pandang, jarak, dan kondisi cahaya. Variasi membantu model untuk menggeneralisasi dengan lebih baik.
3. **Kualitas Data Training:** Pastikan gambar atau suara yang Anda gunakan jelas dan tidak buram. Hindari gambar yang terlalu gelap, terlalu terang, atau terpotong.
4. **Seimbangkan Data Training:** Pastikan jumlah contoh untuk setiap kelas relatif sama. Jika satu kelas memiliki 100 contoh dan kelas lain hanya 10, model mungkin akan bias dan lebih sering memprediksi kelas dengan data lebih banyak.

5. **Hilangkan "Noise" atau Gangguan:** Pastikan data training hanya berisi apa yang ingin Anda kenali. Misalnya, jika melatih tangan, pastikan tidak ada objek lain yang ikut terekam secara signifikan di latar belakang yang bisa membingungkan model.

Soal:

Apa yang harus dilakukan agar model AI lebih akurat?
Jawaban:
Tambahkan banyak gambar
Berikan gambar dari berbagai sudut dan kondisi
Pastikan kualitas gambar baik dan jelas

Soal:

Mengapa harus membuat kelas berbeda saat membuat model gambar?
Jawaban:
Karena setiap kelas mewakili satu kategori yang ingin dikenali AI. Misalnya: "Buku", "Botol", dan "Pensil" harus dipisah agar AI tidak bingung.

Soal:

Apa yang terjadi jika data training hanya sedikit?
Jawaban:
Model AI tidak bisa belajar dengan baik, sehingga prediksinya sering salah atau tidak akurat.

Soal:

Bagaimana cara melihat apakah model AI sudah akurat?
Jawaban:
Lihat hasil di panel Preview. Jika persentase prediksi tinggi untuk kelas yang benar, artinya AI sudah cukup baik.

Fakta Menarik

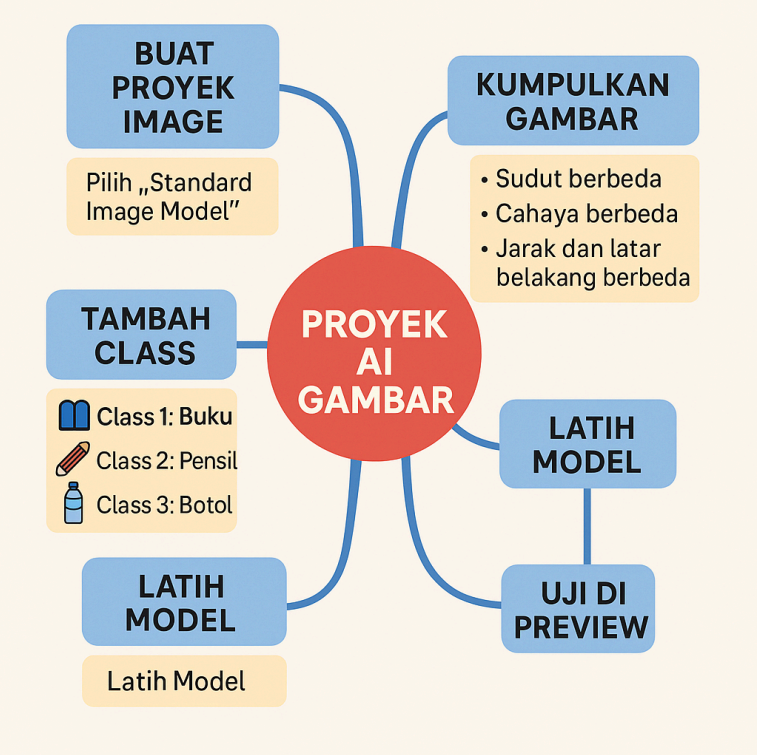
Model AI yang kita buat ini seperti bayi yang belajar dari contoh-contoh di sekitarnya. Semakin banyak contoh yang benar dan bervariasi yang kita berikan, semakin pintar dan "paham" dia tentang dunia di sekelilingnya!

Tips Cepat Menghafal

Ingat pepatah sederhana ini: **"Banyak Contoh yang Bervariasi, Hasil Akurat Pasti!"** Ini adalah kunci sukses dalam melatih model AI.

Mind Mapping

Buat peta pikiran seperti ini saat memulai proyek Teachable Machine:



Cerita: "AI Si Bayi Belajar"

Bayangkan AI seperti bayi. Jika kamu tunjukkan hanya **satu gambar jeruk**, dia mungkin pikir semua buah itu jeruk! Tapi kalau kamu tunjukkan banyak gambar jeruk dari berbagai jenis, barulah dia mulai paham: "Ooh, yang bulat dan oranye itu jeruk!" Sama seperti kita belajar sejak kecil.

Aktivitas Siswa

Membuat Model Teachable Machine untuk Mengenali Ekspresi Wajah (Senyum, Datar, Sedih)

Ikuti langkah-langkah di atas untuk membuat model AI yang bisa mengenali setidaknya 3 ekspresi wajah kalian:

1. **Kelas 1:** Senyum
2. **Kelas 2:** Wajah Datar/Netral
3. **Kelas 3:** Sedih

Pastikan untuk mengumpulkan minimal 50-70 contoh per kelas menggunakan webcam. Berikan variasi pada setiap ekspresi (misalnya, senyum dengan gigi,

senyum tanpa gigi; sedih dengan mata sedikit tertutup, sedih dengan mata terbuka lebar). Setelah training, uji model kalian dan lihat seberapa baik ia bisa memprediksi ekspresi yang berbeda.

Game Tangan AI

Buat model untuk mengenali 3 posisi tangan: *tangan terbuka*, *tangan tertutup*, dan *dua jari membentuk "V"*. Bisa jadi game tebak gerakan!

AI Detektor Barang Sekolah

Latih model AI untuk mengenali *penghapus*, *pensil*, dan *spidol*. Gunakan berbagai latar dan pencahayaan untuk tantangan tambahan.

Kuis AI dengan Teman

Buat proyek AI bersama teman, saling tukar model, lalu coba tebak benda masing-masing dengan bantuan AI.

Contoh Pseudocode:

```
IF gambar di depan kamera dikenali sebagai “Pensil”
        THEN tampilkan: “Ini adalah Pensil!”
        ELSE
        tampilkan: “Coba lagi!”
```

Rangkuman

Bab ini mengajak kamu untuk terjun langsung membuat model AI gambar menggunakan Google Teachable Machine. Dengan langkah-langkah yang mudah, kamu belajar membuat *Class*, mengumpulkan data dari webcam, melakukan *training*, dan menguji hasilnya.

Proyek ini melatih kemampuan berpikir kritis dan eksperimen. Kamu juga diajak memahami bahwa kualitas dan variasi data sangat penting agar AI bisa belajar dengan benar. Ini bukan hanya soal teknologi, tapi juga cara berpikir seperti ilmuwan: teliti, sabar, dan kreatif!

Latihan Soal

A. Benar atau Salah (5 Soal)

1. Kita bisa membuat model gambar di Teachable Machine tanpa menggunakan kamera.
   Jawaban: Salah

2. Jumlah data training yang tidak seimbang bisa membuat model kurang akurat.

Jawaban: Benar

3. Tombol "Train Model" digunakan untuk menghapus semua kelas yang sudah dibuat.
   Jawaban: Salah

4. Preview hasil prediksi akan muncul setelah proses training selesai.
   Jawaban: Benar

5. Menambahkan banyak variasi gambar bisa membuat model lebih akurat.
   Jawaban: Benar

B. Pilihan Ganda (10 Soal)

6. Langkah pertama untuk membuat model gambar di Teachable Machine adalah:
   A. Klik Preview
   B. Klik Train Model
   C. Pilih Image Project
   D. Upload suara
   Jawaban: C

7. Fitur “Hold to Record” digunakan untuk:
   A. Melatih model
   B. Menyimpan proyek
   C. Menghapus kelas
   D. Mengambil data gambar dari webcam
   Jawaban: D

8. Mengapa penting menggunakan gambar dari berbagai sudut dan pencahayaan saat melatih model?
   A. Agar file lebih besar
   B. Agar model tidak membosankan
   C. Agar model bisa mengenali variasi nyata
   D. Supaya cepat selesai
   Jawaban: C

9. Jika Anda ingin melatih model mengenali “Tangan Terbuka” dan “Tangan Tertutup”, berapa kelas yang harus dibuat?
   A. 1
   B. 2
   C. 3
   D. 5
   Jawaban: B

10. Dimana kita bisa melihat hasil prediksi dari model yang sudah dilatih?
    A. Di halaman utama
    B. Di menu Upload
    C. Di panel Preview
    D. Di menu Webcam
    Jawaban: C

11. Salah satu penyebab model memberikan prediksi yang salah adalah:
    A. Jumlah kelas terlalu banyak
    B. Browser tidak di-refresh
    C. Data training terlalu sedikit atau mirip-mirip
    D. Menggunakan mode gelap
    Jawaban: C

12. Saat membuat model pengenalan gambar, kita sebaiknya menggunakan:
    A. Webcam atau upload gambar
    B. Mikrofon dan video
    C. Musik dan teks
    D. Spreadsheet dan tabel
    Jawaban: A

13. Proses mengajari AI dengan gambar disebut:
    A. Preview
    B. Coding
    C. Training
    D. Loading
    Jawaban: C

14. Jika kita ingin mengenali 5 jenis buah yang berbeda, berapa banyak Class yang harus kita buat?
    A. 2
    B. 3
    C. 5
    D. 10
    Jawaban: C

15. Apa yang terjadi jika kita menutup tab browser saat model sedang dilatih?
    A. Model akan selesai lebih cepat
    B. Model tetap berjalan di latar belakang
    C. Proses training gagal
    D. Kelas akan bertambah otomatis
    Jawaban: C

C. Isian Singkat (5 Soal)

16. Tombol untuk memulai proses pembelajaran model di Teachable Machine disebut _____.
Jawaban: Train Model

17. Kita bisa mengumpulkan gambar menggunakan webcam atau dengan cara _____ gambar.
Jawaban: mengunggah

18. Menu untuk melihat seberapa baik model mengenali gambar disebut _____.
Jawaban: Preview

19. Kumpulan contoh yang digunakan untuk mengajari AI disebut _____ _____.
Jawaban: Data Training

20. Jika jumlah gambar pada satu kelas terlalu sedikit, model akan menjadi tidak _____.
Jawaban: akurat

D. Soal Explorasi

Soal:

Kamu diminta membuat model yang bisa membedakan antara "Buku", "Botol", dan "Pensil". Jelaskan bagaimana kamu akan mengumpulkan data training untuk proyek ini agar model bisa mengenali ketiganya dengan baik.

Jawaban: Saya akan membuat tiga kelas: Buku, Botol, dan Pensil. Untuk setiap kelas, saya akan mengambil minimal 70 gambar menggunakan webcam dari berbagai sudut. Buku akan saya foto saat tertutup, terbuka, berdiri, dan tidur. Botol akan saya foto dalam keadaan penuh dan kosong, dari samping dan atas. Pensil akan saya foto dalam posisi horizontal, vertikal, dan diagonal. Saya juga akan mengganti pencahayaan dan latar belakang agar model tidak hanya mengenali objek di satu kondisi saja.

Soal:

Misalnya kamu sudah melatih model, tapi saat di-Preview hasil prediksinya sering salah. Tuliskan tiga langkah apa saja yang akan kamu lakukan untuk memperbaikinya.

Jawaban: Pertama, saya akan menambah lebih banyak data training, terutama untuk kelas yang sering salah ditebak. Kedua, saya akan memastikan bahwa gambar-gambar tersebut bervariasi dari segi sudut, cahaya, dan latar belakang. Ketiga, saya akan mengecek apakah jumlah

gambar di setiap kelas sudah seimbang agar model tidak condong ke satu kelas saja. Setelah itu, saya akan retrain model dan uji ulang.

Soal:

Jelaskan mengapa kualitas gambar sangat penting saat melatih model AI. Berikan satu contoh gambar yang buruk dan dampaknya terhadap hasil prediksi.

Jawaban: Kualitas gambar sangat penting karena gambar yang buram, terlalu gelap, atau tidak jelas bisa membuat model sulit mengenali pola. Misalnya, jika saya memasukkan gambar kucing yang gelap dan hanya terlihat bayangannya, model bisa salah menebak itu sebagai objek lain. Dampaknya, hasil prediksi bisa salah walaupun inputnya benar, karena model tidak bisa "melihat" fitur penting dari gambar tersebut.

Tugas Proyek

Proyek 1: Si Detektif Benda Sekitar

Tujuan: Melatih AI untuk mengenali 3 benda berbeda di lingkungan sekitarmu (contoh: buku, pensil, dan botol minum).

Langkah-langkah:

1. Buka Google Teachable Machine dan pilih *Image Project*.
2. Buat **3 kelas (class)**, masing-masing diberi nama sesuai benda pilihan.
3. Gunakan webcam untuk mengambil minimal **70 gambar per benda**. Ambil dari berbagai sudut, jarak, pencahayaan, dan latar belakang.
4. Latih modelnya dan klik *Train Model*.
5. Uji coba: Arahkan webcam ke benda aslinya dan lihat apakah AI bisa menebaknya dengan benar.
6. Evaluasi: Jika hasil belum akurat, tambahkan lebih banyak variasi gambar.

Tantangan tambahan: Gunakan benda dengan bentuk mirip (misal: pulpen dan spidol) dan lihat apakah AI bisa membedakannya!

Proyek 2: Detektor Ekspresi Wajah

Tujuan: Mengajarkan AI untuk mengenali tiga ekspresi wajahmu: senang, sedih, dan marah.

Langkah-langkah:

1. Pilih *Image Project* dan buat 3 kelas ekspresi: Senang 😄, Sedih 😢, Marah 😠.

2. Ambil minimal **70 foto** untuk tiap ekspresi, dengan berbagai sudut wajah dan pencahayaan.
3. Latih modelnya, lalu coba ekspresikan wajahmu dan lihat apakah AI bisa mengenalinya.
4. Uji dengan temanmu juga, apakah model bisa mengenali ekspresi orang lain?

Tantangan tambahan: Tambahkan ekspresi "bingung" atau "kaget" agar AI lebih pintar!

Proyek 3: Tebak Gerakan Tangan!

Tujuan: Membangun AI yang mengenali 3 gestur tangan: tangan terbuka, tangan mengepal, dan jempol ke atas 👍

Langkah-langkah:

1. Pilih *Image Project* dan buat 3 kelas sesuai gestur.
2. Ambil foto tanganmu dari berbagai sudut dan pencahayaan, minimal 70 gambar per gestur.
3. Latih modelnya.
4. Uji model: apakah bisa menebak gerakanmu secara real time?
5. Gunakan hasilnya untuk membuat kuis gestur—temanmu harus menebak gestur berdasarkan prediksi AI!

Tantangan tambahan: Tambahkan gestur "peace ✌️" dan lihat seberapa baik AI membedakannya.

Proyek 4: AI Penebak Buah

Tujuan: Melatih AI mengenali 3 buah asli (bukan gambar): apel, pisang, dan jeruk.

Langkah-langkah:

1. Siapkan buah asli dan pilih *Image Project*.
2. Buat 3 kelas sesuai nama buah.
3. Ambil foto dari berbagai sudut dan jarak (minimal 70 per buah).
4. Latih dan uji model dengan buah aslinya.
5. Lakukan tes silang: apakah AI bisa mengenali buah jika kamu mengganti meja/latar belakang?

Tantangan tambahan: Tambahkan buah lain yang mirip (misalnya jeruk dan lemon) untuk menguji ketelitian AI.

Proyek 5: Uji Kualitas Data – Siapa Lebih Hebat?

Tujuan: Mengamati pengaruh kualitas gambar terhadap akurasi model AI.

Langkah-langkah:

1. Buat **dua model berbeda**:
    - **Model A**: Gunakan gambar buram, gelap, atau terlalu terang.
    - **Model B**: Gunakan gambar jelas, terang, dan rapi.
2. Gunakan jenis benda yang sama (misalnya: penghapus, penggaris, dan pulpen).
3. Latih masing-masing model secara terpisah.
4. Bandingkan akurasi hasil prediksi.
5. Buat laporan singkat: model mana lebih akurat? Mengapa?

Tantangan tambahan: Tambahkan kategori baru dan ulangi eksperimen untuk melihat konsistensi hasil.

Proyek 6: AI Teman Belajar

Tujuan: Ciptakan AI yang bisa membantu mengenali alat belajar: penghapus, penggaris, dan pensil warna.

Langkah-langkah:

1. Pilih *Image Project* dan buat kelas sesuai alat belajar.
2. Kumpulkan data gambar dengan variasi tinggi.
3. Latih model dan gunakan dalam skenario "tebak alat belajar" di ruang kelas atau rumah.
4. Gunakan model untuk membantu adik atau teman dalam bermain sambil belajar.

Tantangan tambahan: Gabungkan proyek ini dengan proyek deteksi ekspresi untuk membuat asisten belajar yang bisa membaca ekspresimu!

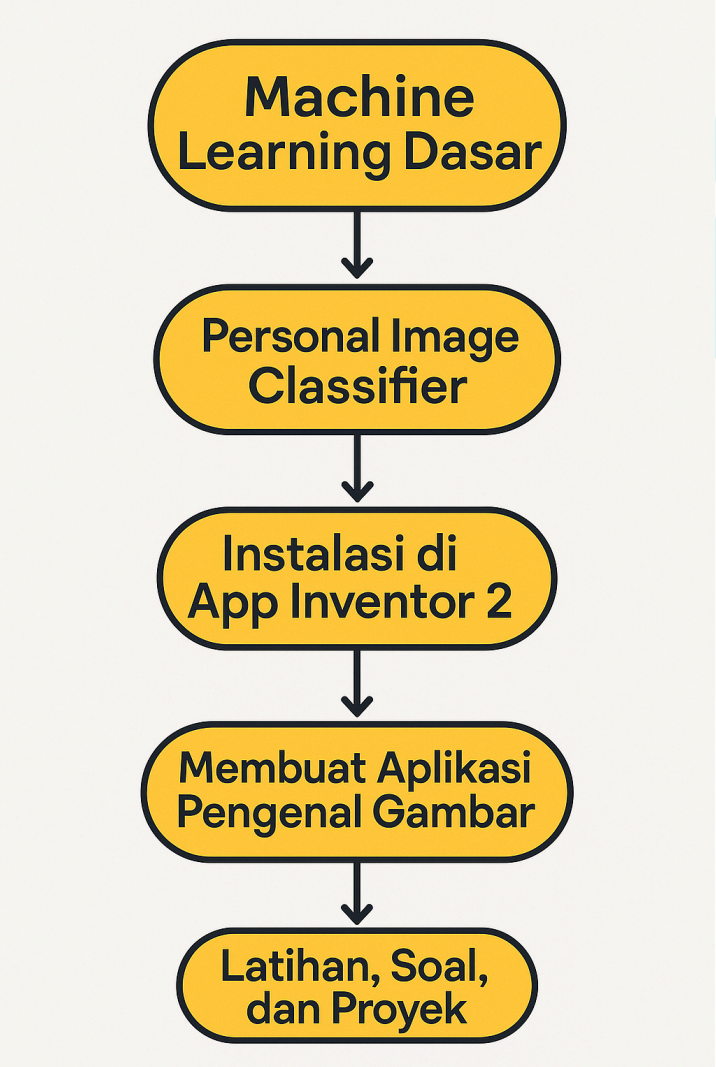
BAB 5: Mengenal Machine Learning dengan Ekstensi Personal Image Classifier di App Inventor 2

Tujuan Pembelajaran

Setelah mempelajari bab ini, siswa diharapkan mampu:

- Memahami apa itu **machine learning** dan cara kerjanya secara dasar.
- Mengetahui fungsi **Personal Image Classifier (PIC)**.
- Menginstal dan menggunakan ekstensi PIC di App Inventor 2.
- Membangun aplikasi sederhana yang bisa mengenali gambar menggunakan machine learning.
- Menumbuhkan rasa ingin tahu dan semangat belajar teknologi masa depan.

Peta Konsep



Apersepsi

Coba bayangkan kamu punya **robot teman** yang bisa mengenali gambar hewan, buah, atau mainanmu. Menarik, bukan? Itulah yang bisa kamu buat sendiri di App Inventor 2 dengan bantuan **machine learning**!

Sebelum kita mulai, yuk pikirkan:

- Apa kamu pernah melihat kamera HP bisa mengenali wajah?
- Pernah pakai filter Instagram atau TikTok yang berubah sesuai wajahmu?

Semua itu menggunakan teknologi yang disebut **machine learning** – dan kamu juga bisa mempelajarinya!

Penjelasan Konsep (Teori)

Apa Itu Machine Learning?

Pernahkah kamu membayangkan sebuah komputer yang bisa "belajar" seperti manusia? Nah, itulah yang disebut dengan **Machine Learning** atau **Pembelajaran Mesin**. Machine Learning adalah salah satu cabang dari **Kecerdasan Buatan (Artificial Intelligence/AI)** yang memungkinkan komputer untuk belajar dari **data** dan kemudian membuat **prediksi** atau keputusan **tanpa harus diprogram secara langsung** untuk setiap tugas.

Misalnya, bayangkan kamu menunjukkan **ratusan gambar kucing** kepada komputer. Dengan bantuan Machine Learning, komputer akan menganalisis gambar-gambar itu dan mulai **belajar** seperti apa bentuk kucing—misalnya punya telinga runcing, mata besar, atau kumis panjang. Setelah "belajar", komputer bisa mengenali gambar kucing lainnya **meskipun belum pernah melihatnya sebelumnya**. Keren, kan?

Machine Learning membuat komputer bisa:

- Mengenali wajah di foto
- Menebak genre lagu yang kamu suka
- Memprediksi cuaca
- Bahkan membantu mobil berjalan sendiri!

Dengan kata lain, Machine Learning adalah cara agar komputer bisa menjadi **lebih pintar** setiap harinya dengan belajar dari data, sama seperti kamu yang semakin pintar setiap kali belajar hal baru di sekolah!

Apa Itu Personal Image Classifier (PIC)?

Personal Image Classifier, atau disingkat **PIC**, adalah sebuah alat pintar yang memungkinkan kita membuat **model kecerdasan buatan (AI)** sendiri untuk mengenali gambar-gambar tertentu. Dengan PIC, kita bisa **melatih komputer** agar mampu membedakan berbagai jenis gambar sesuai keinginan kita.

Misalnya, kamu ingin komputer bisa membedakan antara gambar **kucing** dan **anjing**. Kamu cukup mengunggah beberapa contoh gambar kucing dan anjing ke PIC, lalu alat ini akan belajar dari gambar-gambar itu. Setelah belajar, komputer akan bisa menebak gambar baru—apakah itu kucing atau anjing—dengan cukup akurat!

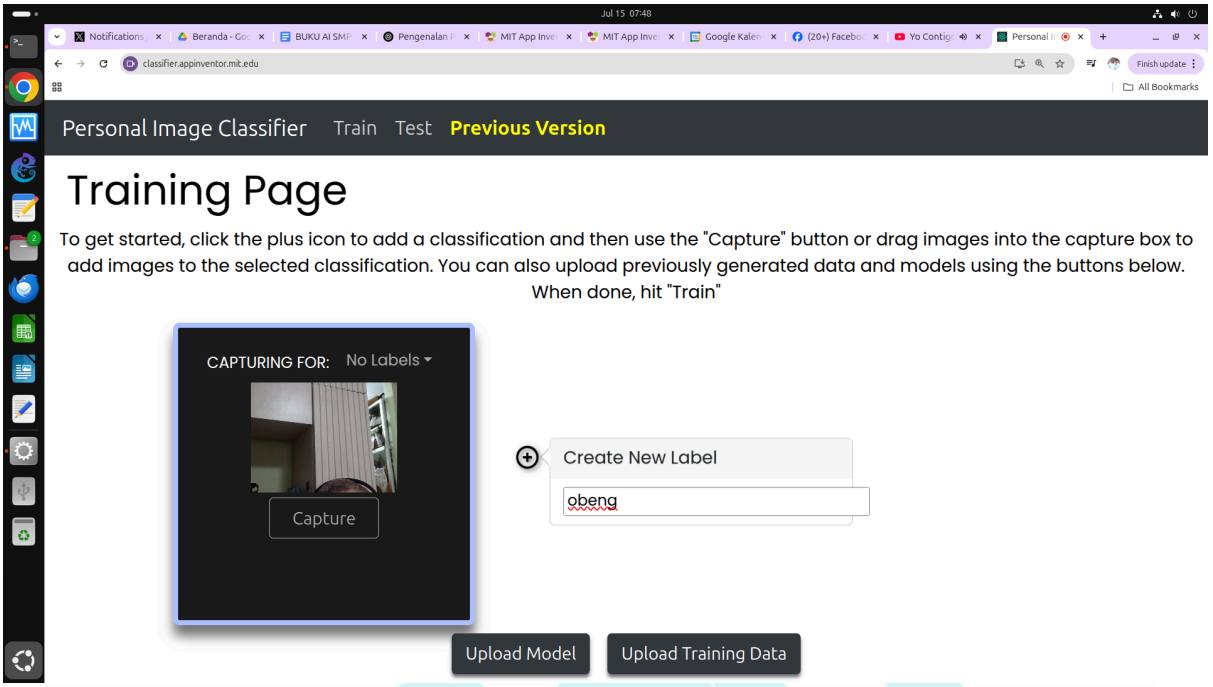
Yang menarik, setelah kamu membuat model pengenalan gambar ini, kamu bisa langsung **menggunakannya di App Inventor 2**, sebuah platform untuk membuat aplikasi Android secara mudah. Jadi, kamu bisa membuat aplikasi buatanmu sendiri yang bisa mengenali gambar—keren, kan?

Dengan Personal Image Classifier, belajar teknologi dan AI jadi lebih menyenangkan karena kamu bisa langsung praktik dan melihat hasilnya secara nyata!

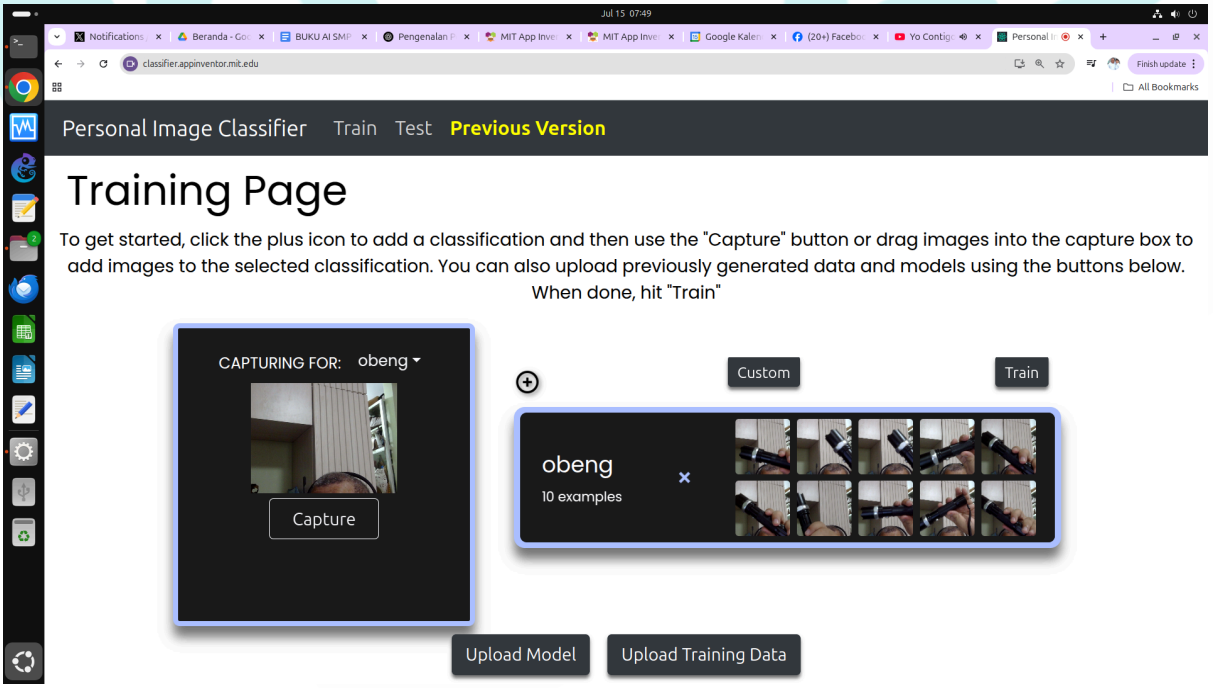
Membuat Model dan Training Data untuk PersonalImageClassifer

Langkah pertama, buka situs berikut:
🔗 https://classifier.appinventor.mit.edu

Setelah halaman terbuka, klik tombol **"+"** untuk mulai menambahkan kategori gambar yang ingin kamu kenali. Misalnya, kamu bisa membuat kategori dengan nama **"Obeng"** jika kamu ingin melatih komputer agar bisa mengenali gambar obeng.

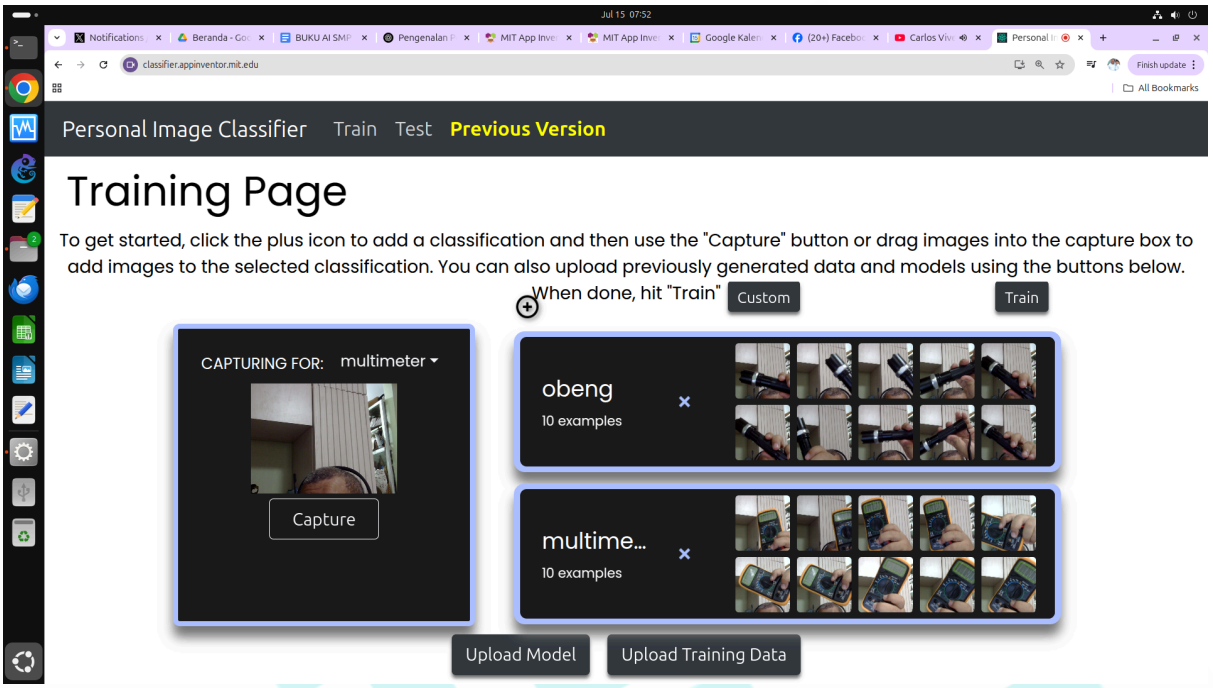


Selanjutnya, kita akan melatih sistem untuk mengenali benda yang ingin kita analisis. Caranya, kita perlu menggunakan beberapa gambar benda tersebut dari berbagai sudut pandang, seperti yang ditunjukkan pada gambar. Dengan begitu, sistem bisa 'belajar' lebih baik dan mengenali bentuk benda dari sudut mana pun!

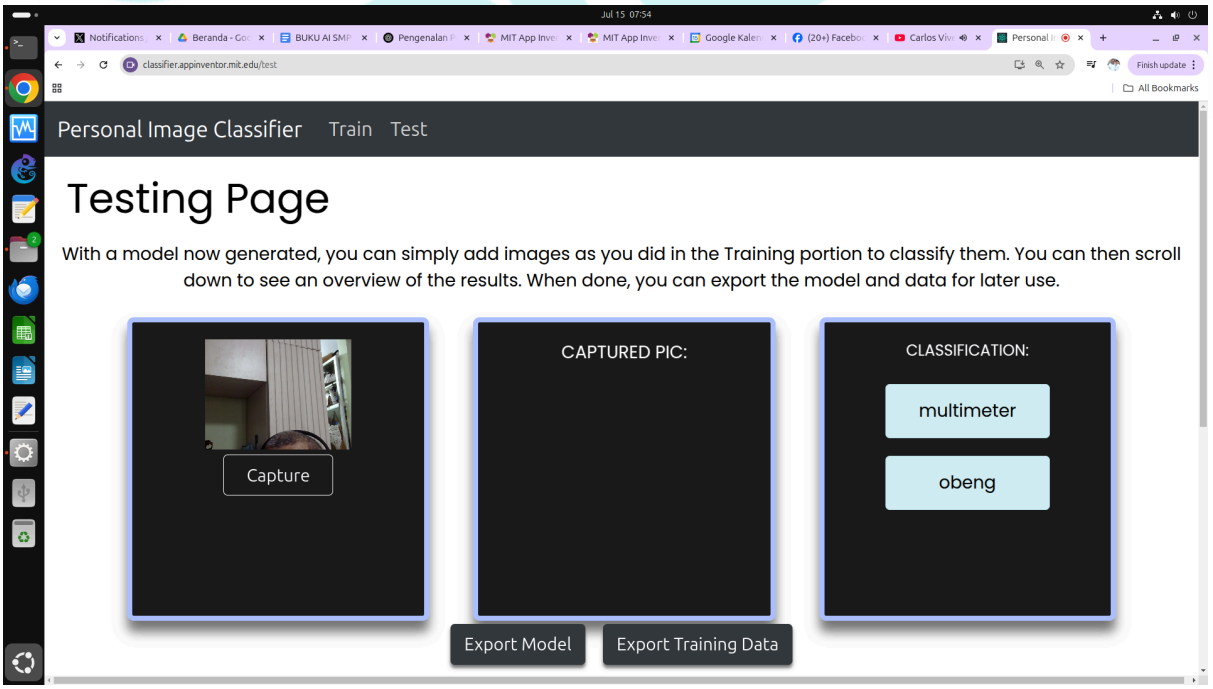


Misalnya, kita ingin mengenali dua jenis benda. Salah satu contohnya adalah multimeter. Untuk menambah benda baru, kita cukup klik tombol **"+"**, lalu ketikkan nama kategorinya, misalnya **"multimeter"**. Setelah itu, kita bisa mulai melatih sistem dengan menampilkan multimeter di depan webcam. Hasilnya,

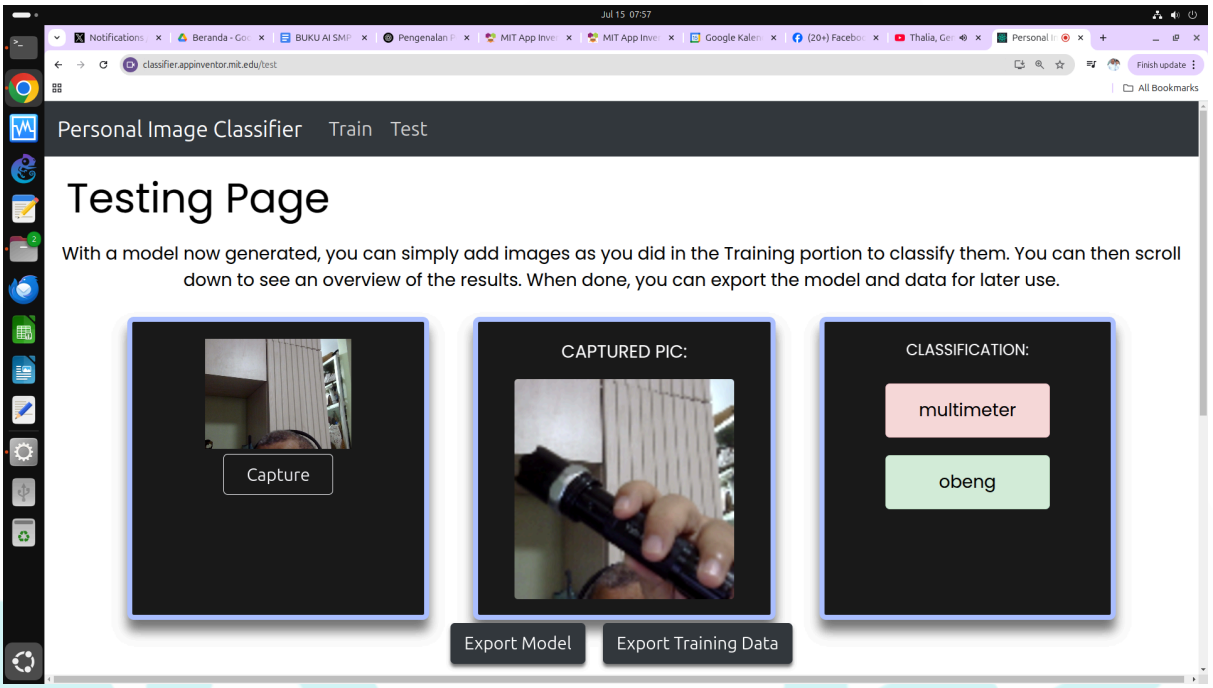
kita akan memiliki dua kumpulan data (dataset) yang siap digunakan untuk mendeteksi kedua benda tersebut.



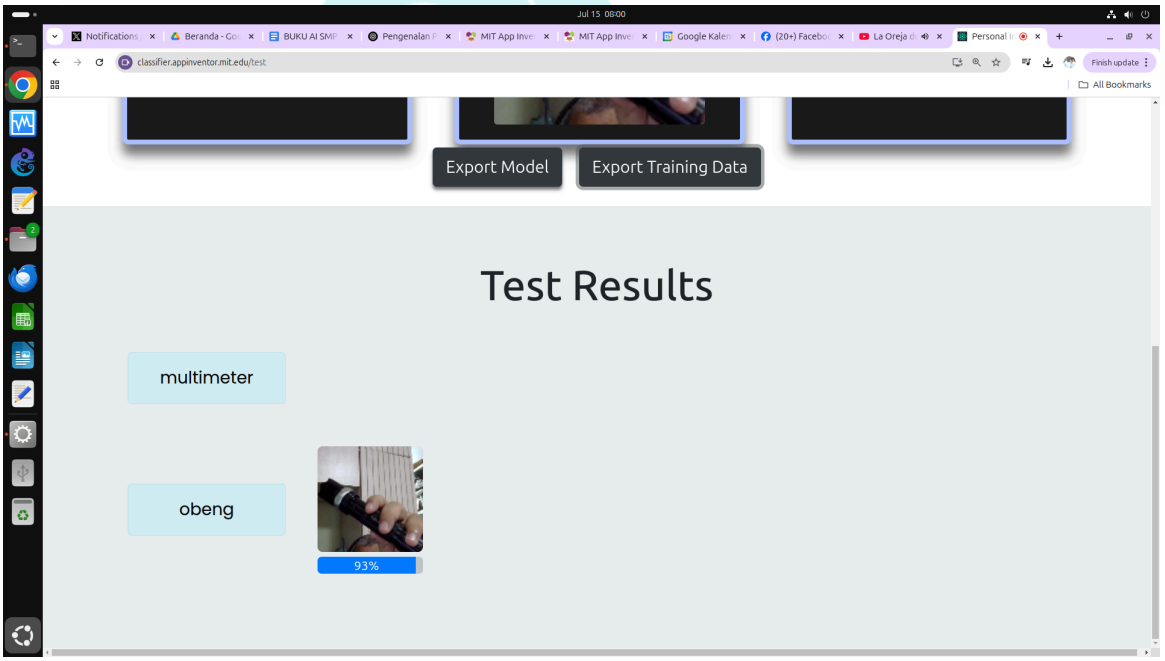
Setelah data latihan yang kita butuhkan berhasil dikumpulkan, langkah selanjutnya adalah menekan tombol **"Train"** untuk mulai melatih data tersebut. Nah, proses ini akan membuat komputer belajar dari data yang kita berikan supaya bisa mengenali alat yang kita inginkan. Setelah beberapa saat, kita bisa langsung mencoba dan melihat apakah model yang sudah kita buat benar-benar bisa mengenali objeknya dengan tepat. Seru, kan? Seperti melatih robot agar jadi pintar!



Pada gambar, terlihat bahwa aplikasi web berhasil mengenali obeng yang kita arahkan ke kamera dengan sangat baik. Klasifikasi menunjukkan obeng berwarna hijau—artinya terdeteksi dengan benar! Sementara itu, multimeter ditandai dengan warna merah, menandakan bahwa benda tersebut tidak sesuai dengan kategori yang dicari. Seru ya, melihat teknologi bisa membedakan alat-alat hanya dari gambar!



Nah, setelah kita selesai membuat model dan melatihnya dengan data, kita bisa menyimpan hasil kerja kita untuk digunakan lagi nanti. Caranya gampang banget! Tinggal klik tombol 'Export Model' untuk menyimpan modelnya, dan 'Export Training Data' untuk menyimpan data latihnya. Praktis, kan?

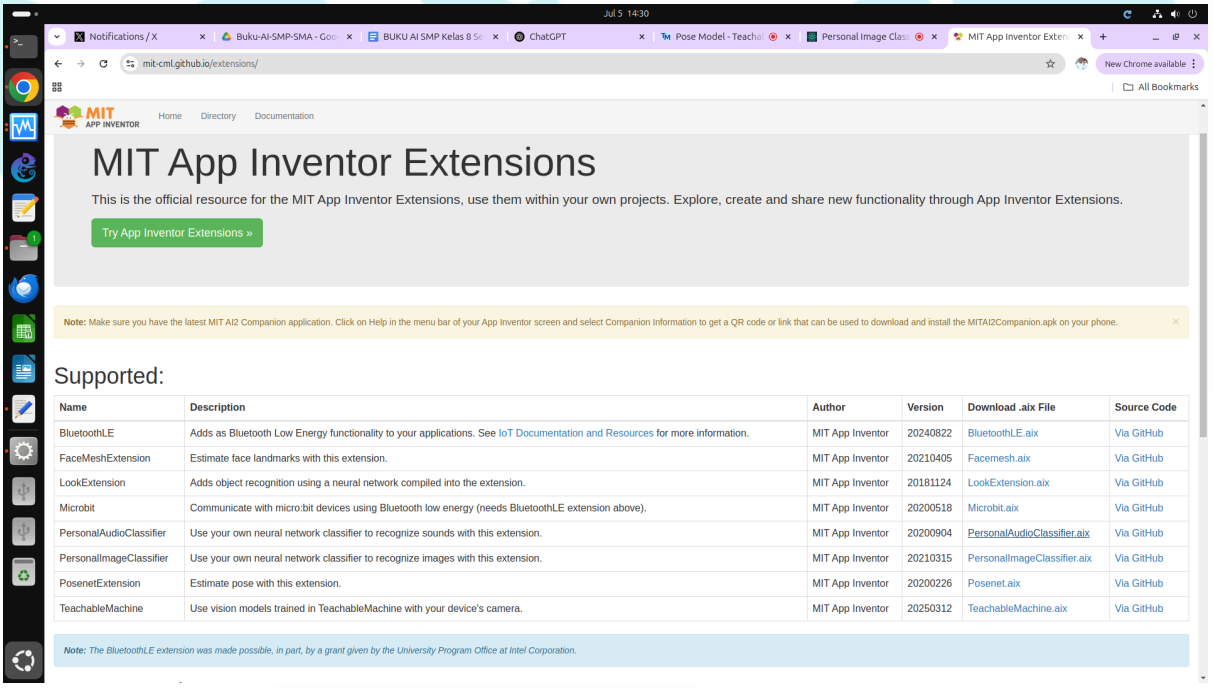


Instalasi Ekstensi Personal Image Classifier di App Inventor 2

Agar proyek App Inventor kalian makin canggih, yuk kita tambahkan ekstensi *Personal Image Classifier* atau disingkat PIC! Langkah-langkahnya mudah, lho. Ikuti petunjuk di bawah ini:

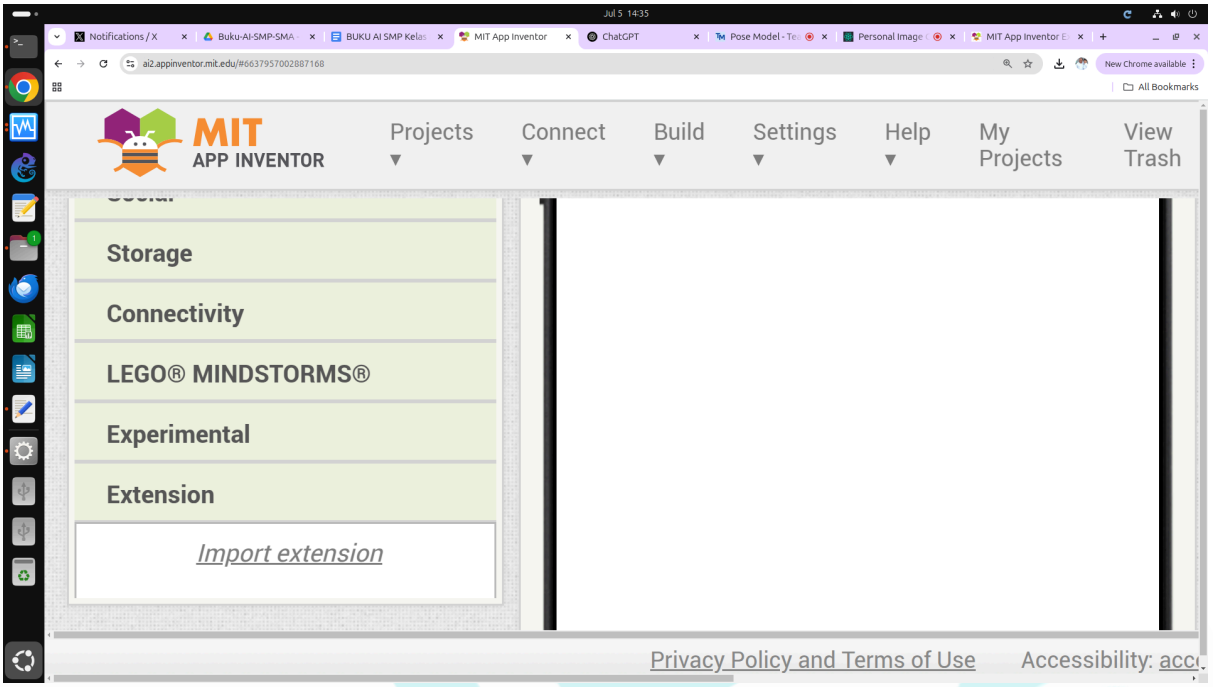
- **Unduh Ekstensi PIC**
  Pertama-tama, kalian perlu mengunduh file ekstensi PIC dengan format .aix. Ekstensi ini bisa kalian dapatkan dari situs resmi MIT App Inventor atau dari sumber lain yang terpercaya.
  Caranya gampang! Cukup cari di internet dengan kata kunci:
  Personal Image Classifier extension App Inventor
  Atau, kalian bisa langsung mengunjungi tautan berikut untuk mengunduhnya:
  - https://mit-cml.github.io/extensions/
  - Langsung ke file PIC (.aix)
    https://mit-cml.github.io/extensions/data/extensions/PersonalImageClassifier.aix

Dengan ekstensi ini, kalian bisa membuat aplikasi yang bisa mengenali gambar sesuai pelatihan yang kalian lakukan sendiri.

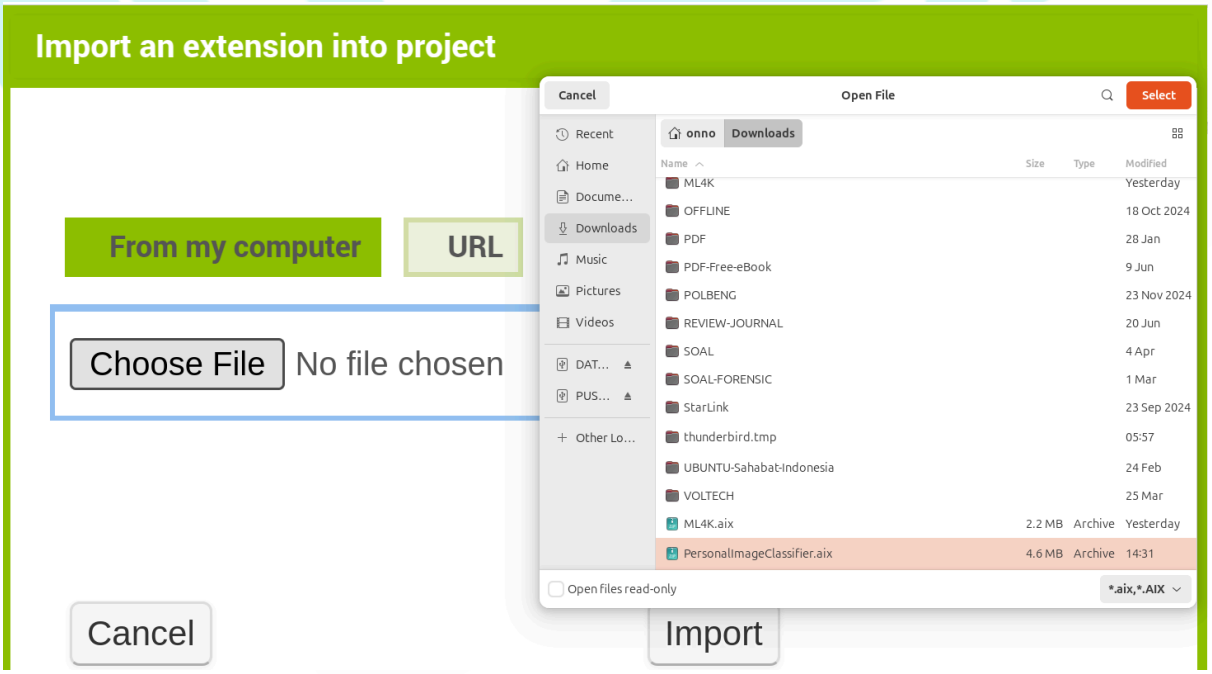


- Buka situs **App Inventor 2** melalui tautan: ai2.appinventor.mit.edu. Pastikan koneksi internet kamu lancar, ya!
- Setelah halaman terbuka, klik menu **"Ekstensi"** di bagian atas.
- Pilih opsi **"Impor Ekstensi"**.
  Di panel sebelah kiri, gulir ke bawah hingga kamu menemukan bagian **"Extension"**, lalu klik tombol **"Import extension"**.

- Cari dan pilih file ekstensi dengan format **.aix** yang sudah kamu unduh sebelumnya. File ini berisi fitur tambahan keren yang akan membuat aplikasi buatanmu jadi lebih menarik!

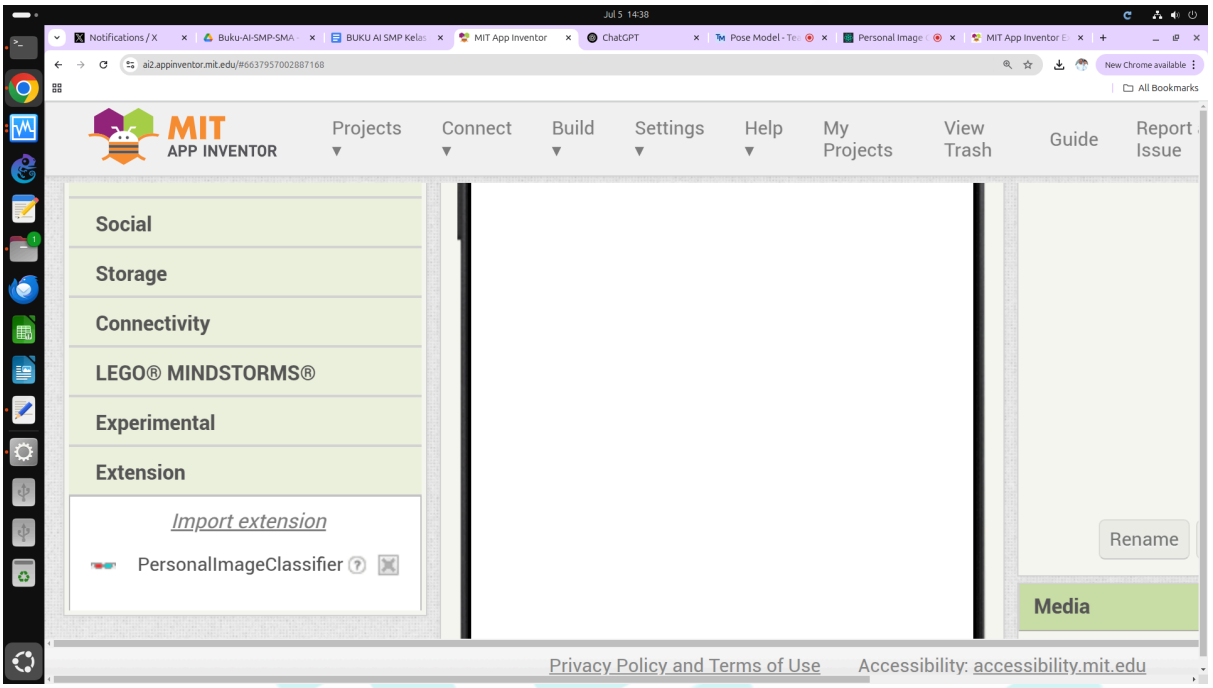


Yuk, pilih file **.aix** yang sudah kamu unduh untuk diunggah ke proyekmu. Atau, kamu juga bisa langsung ambil dari URL berikut ini: https://mit-cml.github.io/extensions/data/extensions/PersonalImageClassifier.aix

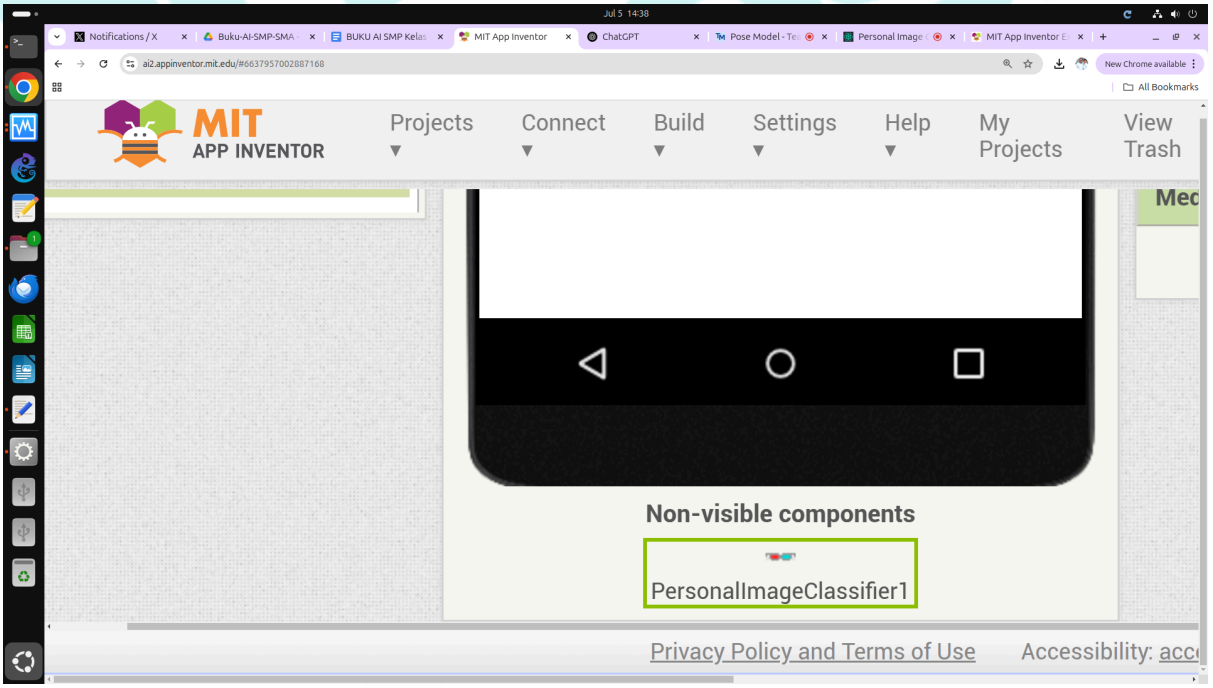


Yuk, kita mulai! Seret komponen **PersonalImageClassifier** dari bagian **"Extension"** ke area kerja di layar kamu. Pada gambar berikut tampilan layar di

App Inventor 2 setelah kamu berhasil menginstal ekstensi **Personal Image Classifier**. Keren, kan? Siap lanjut ke langkah berikutnya!



"Untuk mulai menggunakan, kalian bisa klik dan seret *PersonalImageClassifier* ke sebelah kanan. Komponen ini akan masuk sebagai bagian yang tidak terlihat di layar, tapi tetap bekerja di balik layar untuk membantu aplikasi kalian!"



Aplikasi untuk Klasifikasi Gambar Sederhana

Yuk, kita belajar membuat aplikasi seru yang bisa mengenali gambar menggunakan kecerdasan buatan (AI)! Dalam proyek ini, kamu akan membuat aplikasi Android yang bisa mengambil gambar dan menebak isinya, misalnya menebak emosi atau jenis objek dalam foto.

Langkah 1: Melatih Model AI

Pertama-tama, kita perlu melatih model AI terlebih dahulu. Ikuti langkah berikut:

1. Buka situs 🔗 classifier.appinventor.mit.edu.
2. Latih model dengan mengunggah contoh-contoh gambar yang kamu inginkan.
3. Setelah selesai, download file model-nya, misalnya bernama model.mdl.

Langkah 2: Mengimpor Ekstensi Personal Image Classifier

1. Kunjungi situs 🔗 mit-cml.github.io/extensions dan download ekstensi **Personal Image Classifier (.aix)**.
2. Di MIT App Inventor, buka bagian **Extensions** (seperti terlihat pada gambar pertama), lalu klik **Import Extension** dan unggah file .aix tersebut.

Langkah 3: Upload Model ke Proyek

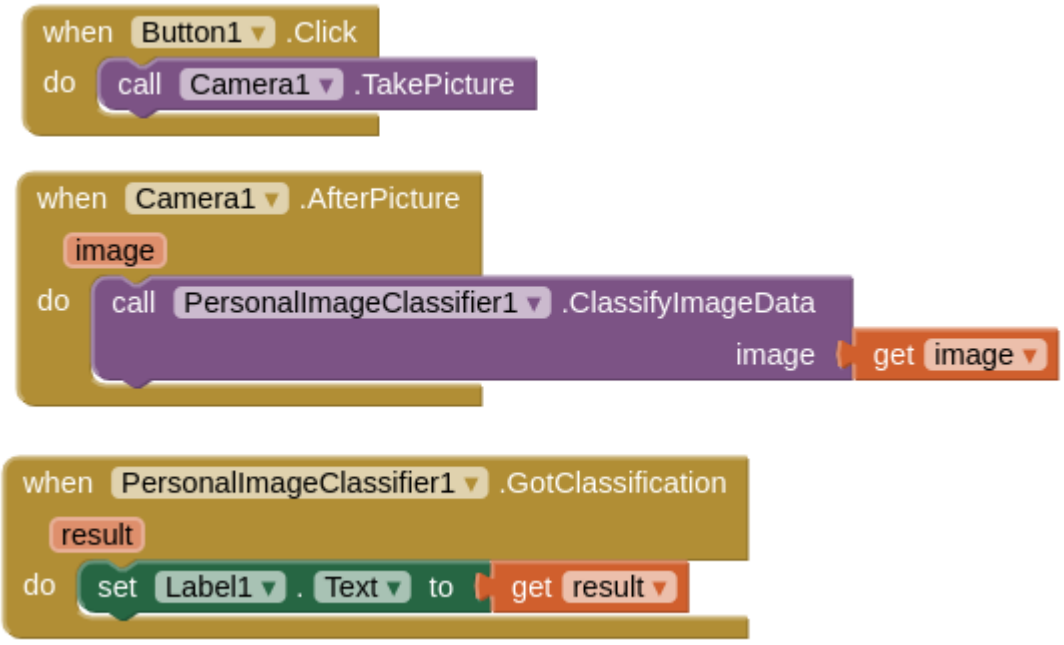
1. Tambahkan komponen **PersonalImageClassifier** ke layar aplikasi kamu.
2. Pada bagian **Properties**, setel properti:
    - **Model** ke file model.mdl yang sudah kamu upload.
    - **WebViewer** ke komponen **WebViewer1**.

Langkah 4: Pemrograman Aplikasi

Sekarang waktunya membuat logika aplikasi!
Buka tampilan **Blocks** di MIT App Inventor, dan ikuti struktur blok seperti yang terlihat di gambar kedua:

Penjelasan Blok:

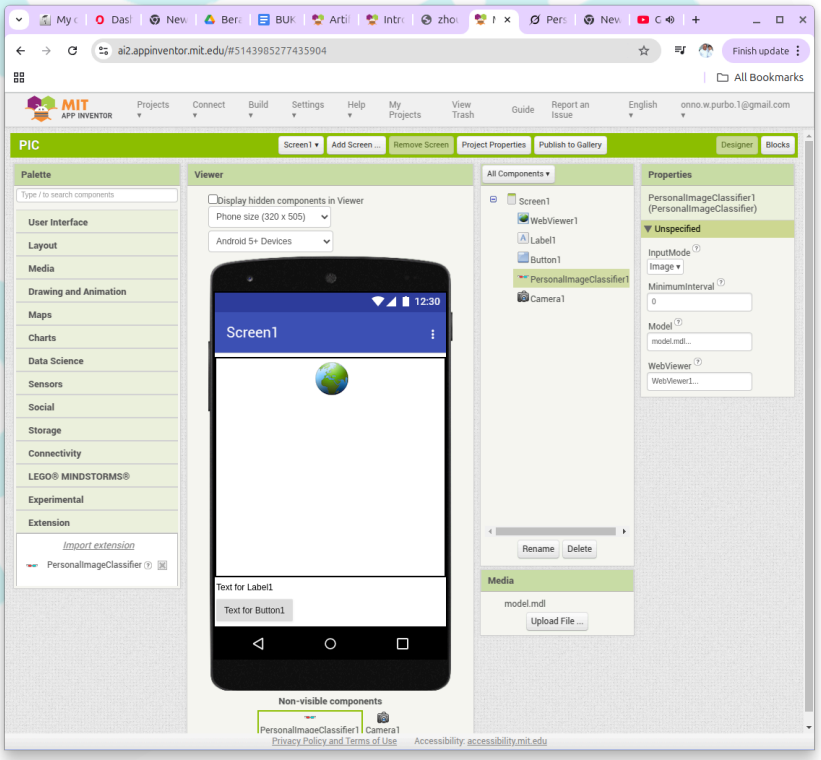
- Ketika tombol **Button1** diklik, aplikasi akan **mengambil gambar** menggunakan **Camera1**.
- Setelah gambar diambil (blok Camera1.AfterPicture), aplikasi akan **mengirim gambar tersebut ke model AI** untuk diklasifikasi menggunakan PersonalImageClassifier1.ClassifyImageData.
- Setelah hasil klasifikasi didapat (blok GotClassification), hasilnya akan **ditampilkan ke pengguna** melalui **Label1**.



Tampilan Aplikasi

Seperti terlihat pada gambar pertama:

- Di tengah layar terdapat gambar bumi (ikon WebViewer).
- Di bawahnya ada teks untuk hasil klasifikasi dan tombol untuk mengambil gambar.
- Komponen tidak terlihat (seperti kamera dan classifier) ada di bagian bawah.



Dengan aplikasi ini, kamu bisa membuat AI yang bisa mengenali gambar hanya dengan beberapa blok kode. Seru dan sangat cocok untuk belajar teknologi masa depan!

Ilustrasi

Bayangkan kamu punya robot teman di rumah. Tapi… robot itu cuma bisa diam. Ia tak tahu mana apel, mana pisang. Dia bingung membedakan penghapus dan pulpen. Wah, kurang berguna, ya?

Nah, kamu mulai berpikir, "Bagaimana kalau aku ajari robotku mengenali benda-benda ini?" Kamu mulai mengambil foto pulpen dan penghapus dari berbagai sudut, melatih robot dengan sabar, bahkan saat gagal berkali-kali. Tapi kamu tak menyerah!

Akhirnya… dengan bantuan **Personal Image Classifier (PIC)** dan **App Inventor**, robotmu bisa bilang, “Itu penghapus!” sambil menunjuk dengan yakin. Wah, usaha dan rasa ingin tahumu tidak sia-sia!

Contoh Kasus atau Ilustrasi

Kasus 1: Misteri Buah di Dapur – Pisang atau Apel?

Bayangkan ini: Kamu sedang ikut ibu di dapur, dan di atas meja ada dua keranjang buah. Tapi, tiba-tiba lampu padam, dan hanya senter kamera HP yang bisa kamu gunakan untuk mengenali buahnya. Nah, kamu buat aplikasi mini yang bisa **membedakan antara pisang dan apel hanya dengan kamera!**

Tantangan serunya:
Kamu harus melatih aplikasi agar bisa "belajar" dari gambar. Upload 10 gambar pisang dan 10 gambar apel. Setelah itu, uji dengan buah asli. Bisakah aplikasi mengenali buah dengan benar saat kamu mengarahkannya ke keranjang?

Kasus 2: Ekspresi Sahabat – Senang atau Sedih?

Ceritanya begini: Temanmu tampak murung hari ini. Tapi dia bilang, “Aku baik-baik saja.” Nah, kamu punya ide keren: **buat aplikasi yang bisa membaca ekspresi wajah**!

Tantangan serunya:
Latih aplikasi dengan foto ekspresi senang dan sedih. Nanti, kamu dan temanmu bisa selfie dan lihat apakah aplikasi bisa membaca perasaan kalian. Cocok untuk yang suka psikologi, drama, atau yang ingin tahu teknologi di balik filter media sosial!

Kasus 3: Tim Detektif Mainan – Mana Boneka, Mana Mobilan?

Kisahnya: Di ruang bermain adikmu, mainan berantakan! Kamu ditantang untuk **membuat aplikasi yang bisa membedakan antara boneka beruang dan mobil-mobilan** – langsung dari kamera!

Tantangan serunya:
Ambil gambar dari dua jenis mainan ini, latih model, dan uji di dunia nyata. Arahkan ke mainan di rumah. Apakah aplikasi bisa jadi "detektif mainan" yang akurat?

Kasus 4: Pahlawan Lingkungan – Deteksi Sampah vs Daur Ulang

Bayangkan kamu ikut klub lingkungan sekolah. Tugasmu: mengajarkan teknologi bagaimana cara membedakan **sampah organik (daun, sisa makanan)** dengan **sampah anorganik (plastik, botol, kaleng).**

Tantangan serunya:
Gunakan kamera dan model gambar untuk mengenali jenis sampah. Uji aplikasi ini di taman sekolah atau saat kerja bakti. Kalau berhasil, bisa jadi alat bantu memilah sampah yang keren!

Kasus 5: Belanja Cerdas di Pasar – Tomat atau Wortel?

Ceritanya: Kamu ikut ibu ke pasar. Dia minta kamu ambil tomat, tapi kamu malah ambil wortel! Ups! Daripada bingung, kamu buat aplikasi yang bisa **membedakan sayur dan buah, seperti tomat dan wortel.**

Tantangan serunya:
Latih aplikasi dengan gambar sayuran tersebut. Bawa HP-mu ke pasar, uji di lapak pedagang, dan lihat apakah aplikasi bisa jadi "asisten belanja" digital milikmu.

Kasus 6 (Opsional Tambahan): Tebak Suasana di Kelas – Ramai atau Sepi?

Kondisi nyata: Guru keluar sebentar, dan kamu ingin tahu apakah suasana kelas tenang atau ribut. Nah, kamu buat aplikasi yang bisa mengenali gambar suasana kelas: **apakah sedang hening atau ramai?**

Tantangan serunya:
Kumpulkan foto kelas dalam dua kondisi, latih model, lalu uji secara langsung. Cocok untuk yang suka eksperimen sosial atau ingin membuat alat bantu pengawasan otomatis.

Contoh Soal dan Pembahasan

Soal:

Apa yang dimaksud dengan *machine learning*, dan mengapa teknologi ini penting dalam kehidupan modern?

Pembahasan:

Machine learning adalah teknologi cerdas yang membuat komputer dapat "belajar" dari data tanpa harus diprogram secara khusus untuk setiap tugas. Misalnya, kamu bisa mengajarkan komputer mengenali gambar kucing dan anjing hanya dengan memberi contoh-contoh gambar tersebut. Teknologi ini digunakan di berbagai bidang, seperti aplikasi kamera yang bisa mengenali wajah, hingga mobil tanpa sopir. Keren, kan?

Soal:

Mengapa kita perlu memberikan banyak contoh gambar saat melatih sebuah model pengenal gambar? Apa yang terjadi jika hanya sedikit gambar yang digunakan?

Pembahasan:

Bayangkan kamu sedang belajar mengenali jenis-jenis bunga. Kalau kamu hanya melihat 2 gambar bunga mawar, kamu mungkin akan kesulitan mengenali mawar lain dengan bentuk yang sedikit berbeda. Nah, komputer juga sama! Semakin banyak gambar yang diberikan, semakin pintar modelnya mengenali pola dan variasi bentuk. Jika hanya diberi 2 gambar, model bisa bingung dan salah mengenali objek lain.

Soal:

Mengapa penting memberi gambar dari berbagai sudut saat melatih model pengenal gambar di App Inventor?

Pembahasan:

Misalnya kamu ingin komputer mengenali botol minum. Kalau kamu hanya memberikan gambar dari depan, saat melihat gambar dari samping, komputer bisa salah paham. Oleh karena itu, kita perlu memberi gambar dari berbagai sudut, supaya model bisa “mengerti” bentuk botol dari sudut manapun. Ini membuat hasil prediksi lebih akurat!

Soal:

Apa fungsi dari ekstensi Personal Image Classifier (PIC) di App Inventor, dan mengapa itu berguna untuk pemula?

Pembahasan:

PIC adalah alat tambahan di App Inventor yang membantu kamu membuat aplikasi yang bisa mengenali gambar. Dengan PIC, kamu bisa melatih model sendiri — misalnya mengenali jenis buah atau benda

sekolah — tanpa perlu jadi ahli komputer. Ini cocok banget buat pemula yang ingin membuat aplikasi cerdas tanpa coding yang rumit!

Soal:

Apa resiko atau kekurangan jika model machine learning dilatih hanya dengan sedikit data dan tidak bervariasi?

Pembahasan:

Model seperti itu akan "kurang pintar" karena hanya belajar dari sedikit contoh. Akibatnya, ia bisa salah mengenali objek baru yang bentuk atau warnanya agak berbeda. Ini seperti kamu hanya belajar satu soal matematika lalu disuruh mengerjakan ujian — kemungkinan besar hasilnya keliru!

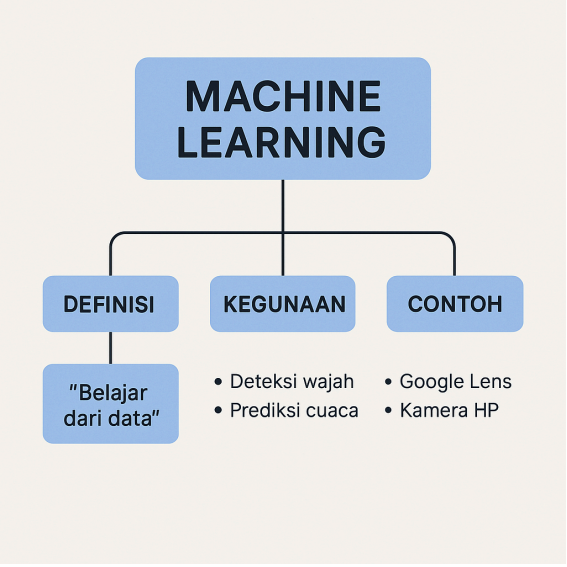
Fakta Menarik

- **Google Lens** bisa mengenali hampir semua benda di dunia hanya dengan kamera—teknologi ini berbasis machine learning!
- Aplikasi deteksi wajah atau deteksi senyum di kamera HP juga menggunakan teknologi ini!
- Aplikasi kamera kamu yang bisa "blur" latar belakang juga pakai AI.
- Anak-anak seusiamu di Jepang sudah membuat aplikasi AI untuk mengenali jenis makanan untuk bayi adik mereka!

Tips Cepat Menghafal

- Ingat rumus ML sederhana: **Data → Dilatih → Prediksi**
- Gunakan singkatan: **PIC = Personal Image Classifier**
- Belajar sambil praktek → Lebih cepat paham daripada hanya membaca!

Contoh Mind Mapping Sederhana:



Tips Cepat Ingat:

- **Singkatan cepat:**
  PIC = Personal Image Classifier
  ML = Machine Learning = Komputer Belajar

- **Cerita pendek mudah diingat:**
  "Komputer seperti murid. Kasih data = pelajaran, dia latihan, lalu bisa menjawab soal = prediksi."

- **Latihan sambil praktek:**
  Jangan hanya baca! Coba buat aplikasinya, dijamin lebih paham!

Aktivitas Siswa

Aktivitas 1: Eksplorasi Benda Sekitar: Pencarian Objek Unik

Deskripsi Aktivitas:
Ajak siswa menjelajah lingkungan kelas atau rumah untuk mencatat berbagai benda yang mereka temukan (seperti botol, tas, spidol, dll), lalu **membayangkan aplikasi AI** yang bisa mengenali benda-benda tersebut secara otomatis.

Langkah Pelaksanaan:

- Minta siswa membuat daftar minimal 5 benda di sekitar mereka.
- Dorong mereka untuk berpikir kreatif: "Kalau ada aplikasi yang bisa mengenali benda ini, akan berguna untuk apa?"
- Bahas ide-ide bersama teman kelompok. Bisa dikaitkan dengan kehidupan sehari-hari seperti aplikasi untuk siswa tunanetra, pengelompokan barang, atau keamanan.

Aktivitas 2: Petualangan Visual: Latih AI Kenali Benda!

Deskripsi Aktivitas:
Gunakan kamera untuk mengambil gambar 3 benda berbeda (misalnya: buku, penghapus, penggaris), lalu latih model sederhana menggunakan platform seperti **PIC (Personal Image Classifier)**.

Langkah Pelaksanaan:

- Foto setiap benda dari beberapa sudut.
- Memasukkan gambar ke model PIC dan beri label yang sesuai.
- Uji model dengan kamera: apakah bisa mengenali benda secara akurat?
- Diskusikan: Apa yang membuat model berhasil atau gagal mengenali benda?

Aktivitas 3: Ekspresi Detektif: Kenali Senyum dan Sedih!

Deskripsi Aktivitas:
Buat model sederhana untuk **membedakan ekspresi wajah senang dan sedih**. Aktivitas ini cocok sebagai lomba mini antarkelompok!

Langkah Pelaksanaan:

- Minta teman berpose dengan ekspresi senyum dan sedih. Ambil beberapa gambar.
- Latih model wajah dengan PIC atau alat serupa.
- Uji model dengan selfie baru. Cocokkah hasilnya?
- Lakukan pertandingan antar kelompok: siapa yang paling akurat deteksinya?

Aktivitas 4: Tantangan AI Quiz Show!

Deskripsi Aktivitas:
Lakukan kuis cepat dan seru berisi pertanyaan seputar **AI, Machine Learning, App Inventor, dan PIC**. Aktivitas ini mengasah pemahaman konsep sambil membangun semangat kompetisi.

Langkah Pelaksanaan:

- Siapkan 5 pertanyaan pilihan ganda atau benar/salah.
- Gunakan waktu terbatas, misalnya 30 detik per soal.
- Bisa menggunakan kertas, lisan, atau platform digital seperti Kahoot.
- Kelompok dengan jawaban tercepat dan paling tepat menjadi pemenang.

Aktivitas 5: Simulasi Coding: Mainkan Pseudocode!

Deskripsi Aktivitas:
Ajak siswa menyusun dan membaca alur logika **dengan pseudocode** sederhana. Ini melatih logika pemrograman tanpa harus menulis kode sebenarnya.

Langkah Pelaksanaan:

- Tampilkan skenario: "Jika tombol foto ditekan, maka ambil gambar dan kirim ke model."
- Minta siswa menyusun langkah-langkahnya dalam bentuk pseudocode.
- Bisa dilakukan secara berpasangan, lalu diskusikan hasilnya bersama.
- Tantangan tambahan: ubah skenario menjadi hal lain, misalnya deteksi suara atau warna.

Aktivitas 6: Desain Poster: Ide Aplikasi AI Kamu!

Deskripsi Aktivitas:
Dalam kelompok, siswa membuat **poster kreatif** berisi ide aplikasi AI yang mereka inginkan, berdasarkan diskusi sebelumnya.

Langkah Pelaksanaan:

- Pilih satu ide dari eksplorasi benda atau deteksi wajah.
- Desain poster yang mencakup: nama aplikasi, fungsinya, siapa penggunanya, dan manfaatnya.
- Presentasikan di depan kelas atau pamerkan di dinding kelas sebagai galeri AI!

Semua aktivitas di atas bersifat eksploratif, kolaboratif, dan menantang untuk siswa SMP. Guru berperan sebagai fasilitator, bukan pemberi tugas. Jika kamu ingin, aku bisa bantu buatkan lembar panduan per aktivitas.

Rangkuman

Bayangkan kalau komputer bisa *belajar* seperti manusia—nah, itulah yang dilakukan oleh **Machine Learning (ML)**! Teknologi ini memungkinkan komputer memahami pola dari data, misalnya gambar, dan membuat prediksi atau keputusan berdasarkan itu. Serunya, kita bisa mencobanya sendiri dengan alat bernama **Personal Image Classifier (PIC)**. Alat ini memungkinkan kita melatih komputer supaya mengenali berbagai benda dari kumpulan gambar yang kita pilih sendiri.

Dengan **PIC**, kamu bisa membuat proyek ML yang menyenangkan! Setelah model dilatih, kamu bisa menggabungkannya dengan **MIT App Inventor**, sebuah platform yang memudahkan kita membuat aplikasi Android tanpa harus jago *coding*. Jadi, kamu bisa membuat aplikasi cerdas yang bisa mengenali *buah*, *hewan peliharaan*, bahkan *alat tulis* hanya dari gambar. Proyek ini cocok banget untuk eksplorasi kreatif sambil belajar teknologi masa depan.

Langkah-langkahnya cukup sederhana tapi menantang: **latih model** dengan gambar yang kamu kumpulkan → **impor ke App Inventor** → **uji coba langsung** di aplikasimu. Ini bukan cuma soal membuat aplikasi, tapi juga belajar berpikir seperti ilmuwan data. Yuk, coba bikin aplikasi ML-mu sendiri—belajarnya asik, hasilnya bikin bangga!

Latihan Soal

A. Benar atau Salah (5 Soal)

1. Personal Image Classifier digunakan untuk mengenali suara dalam aplikasi.
   Jawaban: Salah

2. Machine learning membuat komputer bisa belajar dari data.
   Jawaban: Benar

3. App Inventor adalah platform yang hanya digunakan oleh programmer profesional.
   Jawaban: Salah

4. Model yang dilatih dengan lebih banyak gambar akan menghasilkan prediksi yang lebih akurat.
   Jawaban: Benar

5. Ekstensi .aix harus diimpor ke App Inventor sebelum digunakan.
   Jawaban: Benar

B. Pilihan Ganda (10 Soal)

6. **Apa fungsi dari Personal Image Classifier?**
   A. Mendeteksi suara dalam aplikasi
   B. Membangun aplikasi pesan singkat
   C. Melatih model untuk mengenali gambar
   D. Membuat animasi bergerak
   Jawaban: C

7. **Komponen apa yang digunakan untuk mengambil gambar dalam aplikasi?**
   A. WebViewer
   B. Label
   C. Camera
   D. Notifier
   Jawaban: C

8. **Untuk mengimpor ekstensi ke App Inventor, kita harus menggunakan file berformat:**
   A. .zip
   B. .apk
   C. .aix
   D. .exe

Jawaban: C

9. **Langkah terakhir dalam proses pelatihan model adalah:**
   A. Menghapus data
   B. Menyimpan aplikasi
   C. Mengimpor gambar
   D. Menekan tombol "Train"
   Jawaban: D

10. **Mengapa gambar perlu diambil dari berbagai sudut saat melatih model?**
   A. Agar ukuran gambar lebih kecil
   B. Supaya model tidak terlalu akurat
   C. Agar model mengenali objek lebih baik
   D. Untuk membuat gambar lebih menarik
   Jawaban: C

11. **Setelah model dilatih, file model biasanya berekstensi:**
   A. .jpg
   B. .mdl
   C. .aix
   D. .txt
   Jawaban: B

12. **Salah satu contoh penggunaan machine learning adalah:**
   A. Mengetik di Microsoft Word
   B. Menggambar manual
   C. Mengenali wajah di kamera
   D. Menonton film
   Jawaban: C

13. **Proses ketika komputer belajar dari data disebut:**
   A. Programming
   B. Training
   C. Debugging
   D. Hacking
   Jawaban: B

14. **Apa nama situs untuk melatih model PIC?**
   A. ai2.appinventor.mit.edu
   B. classifier.appinventor.mit.edu
   C. picmodel.com
   D. ai-lab.org
   Jawaban: B

15. **Di App Inventor, hasil klasifikasi biasanya ditampilkan dengan komponen:**
    A. Button
    B. Label
    C. Notifier
    D. ListView
    Jawaban: B

C. Isian Singkat (5 Soal)

16. Machine learning adalah proses di mana komputer belajar dari _____.
    Jawaban: data

17. File ekstensi Personal Image Classifier memiliki format _____.
    Jawaban: .aix

18. Untuk mengklasifikasikan gambar, kita harus terlebih dahulu _____ modelnya.
    Jawaban: melatih

19. Aplikasi yang dibuat dengan App Inventor bisa berjalan di sistem operasi _____.
    Jawaban: Android

20. Komponen yang digunakan untuk menampilkan hasil klasifikasi adalah _____.
    Jawaban: Label

D. Soal Eksplorasi (3 Soal & Jawaban)

Soal:

Bayangkan kamu sedang mengembangkan aplikasi untuk mengenali berbagai jenis buah menggunakan PIC. Kamu hanya punya 3 gambar untuk tiap buah. Ceritakan apa risiko yang mungkin terjadi pada modelmu dan bagaimana cara mengatasinya.

Jawaban:

Risikonya adalah model akan memiliki akurasi rendah karena jumlah gambar terlalu sedikit. Model bisa salah mengenali objek baru karena tidak memiliki cukup variasi data. Solusinya adalah menambahkan lebih banyak gambar dari berbagai sudut dan pencahayaan agar model belajar dengan lebih baik.

Soal:

Kamu ingin membuat aplikasi yang bisa mengenali ekspresi senyum dan ekspresi datar dari wajah temanmu. Ceritakan langkah-langkah dari mulai mengumpulkan data, melatih model, hingga menguji hasilnya di App Inventor 2.

Jawaban:

Langkah pertama adalah mengumpulkan gambar wajah teman dengan ekspresi senyum dan datar. Lalu, buka situs classifier.appinventor.mit.edu dan unggah gambar tersebut ke dua kategori. Setelah itu, klik "Train" untuk melatih model. Unduh model yang telah dilatih, lalu impor ke App Inventor bersama ekstensi PIC. Tambahkan logika pada blok App Inventor agar hasil klasifikasi muncul setelah kamera mengambil gambar. Uji dengan foto teman untuk melihat hasilnya.

Soal:

Seorang temanmu membuat aplikasi pengenal alat tulis, tetapi aplikasinya sering salah mengenali penghapus sebagai penggaris. Jelaskan dua kemungkinan penyebabnya dan berikan solusi untuk masing-masing.

Jawaban:

Pertama, kemungkinan jumlah gambar penghapus dan penggaris tidak seimbang. Solusinya: tambahkan lebih banyak gambar untuk masing-masing kategori. Kedua, gambar diambil dari sudut yang terlalu mirip atau latar belakang terlalu seragam. Solusinya: ambil gambar dari berbagai sudut menggunakan latar belakang yang bervariasi.

Tugas Proyek

Proyek 1: Aplikasi Pengenal Alat Tulis

Deskripsi:

Kembangkan aplikasi sederhana yang dapat mengenali alat tulis seperti pulpen, pensil, dan penghapus menggunakan teknologi pembelajaran mesin.

Langkah-langkah:

1. Ambillah minimal 10 foto dari masing-masing alat tulis.
2. Latih model pengenalan gambar di **Platform Indonesia Cerdas (PIC)**.
3. Bangun aplikasi menggunakan **MIT App Inventor** dan tampilkan hasil klasifikasinya di layar.
4. Uji aplikasimu dengan benda nyata dan catat hasilnya.
5. Presentasikan aplikasi ke teman-teman di kelas!

Proyek 2: Detektor Makanan Ringan Otomatis

Deskripsi:
Buat aplikasi yang bisa mengenali jenis makanan ringan seperti keripik, biskuit, dan permen hanya dari gambar.

Langkah-langkah:

1. Kumpulkan setidaknya 10 gambar untuk tiap jenis makanan.
2. Kelompokkan gambar dan latih model klasifikasi di PIC.
3. Gunakan **App Inventor** untuk membangun aplikasi deteksi makanan.
4. Ambil gambar dengan kamera ponsel dan lihat apakah aplikasi dapat menebak dengan benar.
5. Tulis laporan singkat tentang proses dan akurasi aplikasi.

Proyek 3: Kamera Cuaca Pintar

Deskripsi:
Ciptakan aplikasi yang bisa menebak kondisi cuaca (cerah atau mendung) berdasarkan foto langit.

Langkah-langkah:

1. Ambil foto langit cerah dan langit mendung (masing-masing 10 gambar).
2. Latih model cuaca di PIC menggunakan gambar tersebut.
3. Buat aplikasi dengan App Inventor yang bisa memberikan label seperti "Cerah" atau "Mendung".
4. Uji aplikasimu di luar ruangan dan dokumentasikan hasilnya.
5. Tambahkan fitur suara atau ikon lucu untuk membuat aplikasi lebih menarik.

Proyek 4: Senyum atau Datar? Aplikasi Detektor Ekspresi

Deskripsi:
Bangun aplikasi yang bisa mendeteksi ekspresi wajah: senyum atau datar, dari foto selfie.

Langkah-langkah:

1. Kumpulkan 10 gambar wajah tersenyum dan 10 gambar wajah datar (bisa dari teman).
2. Gunakan PIC untuk melatih model ekspresi wajah.
3. Bangun aplikasi dengan fitur kamera di App Inventor.
4. Uji aplikasimu dengan selfie dari beberapa teman.
5. Tambahkan fitur respon seperti emoticon atau suara lucu berdasarkan hasil deteksi.

Proyek 5: Poster Digital "Aku dan Machine Learning"

Deskripsi:
Refleksikan pengalamanmu dalam membuat aplikasi berbasis AI melalui poster digital interaktif.

Langkah-langkah:

1. Gunakan **Canva**, **PowerPoint**, atau aplikasi desain lainnya.
2. Ceritakan langkah-langkah proyek: mulai dari mengumpulkan data, melatih model, membuat aplikasi, hingga pengujian.
3. Sertakan tangkapan layar dari aplikasimu.
4. Tambahkan kesan dan pesanmu tentang belajar Machine Learning.
5. Presentasikan hasil postermu dalam bentuk slideshow atau video pendek.

Proyek 6: Aplikasi Pengenal Benda Sekolah di Sekitarmu

Deskripsi:
Tantangan tambahan: kembangkan aplikasi pengenal benda lain yang sering digunakan di sekolah, misalnya penggaris atau penghapus papan.

Langkah-langkah:

1. Pilih 2–3 benda di sekitarmu yang ingin dikenali.
2. Kumpulkan masing-masing minimal 5–10 gambar.
3. Latih model di PIC dan bangun aplikasimu.
4. Uji dengan benda asli di kelas atau rumah.
5. Buat laporan singkat atau video demonstrasi.

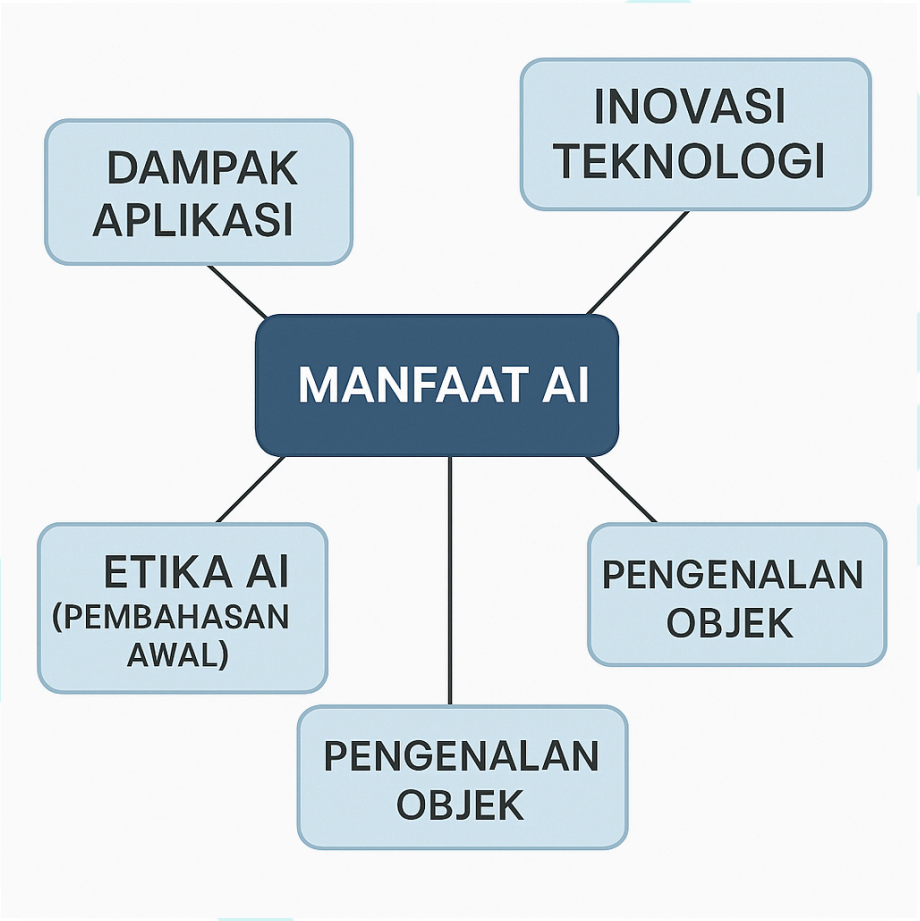
Semua proyek ini bertujuan melatih kreativitas, kemampuan teknologi, dan pemahaman dasar tentang kecerdasan buatan. Ajak teman-temanmu untuk berkolaborasi, dan jadikan proses belajarmu lebih seru dan bermakna!

BAB 6: Refleksi: Apa Itu AI di Aplikasi?

Tujuan Pembelajaran

Peserta didik mampu merefleksikan peran, manfaat, dan potensi dampak Artificial Intelligence (AI) dalam berbagai aplikasi mobile yang mereka gunakan sehari-hari, serta memahami bagaimana AI membuat aplikasi lebih cerdas dan inovatif.

Peta Konsep



Apersepsi

Sekarang kalian sudah bisa membuat aplikasi dan bahkan "mengajari" komputer. Coba sebutkan aplikasi di HP kalian yang menurutmu sudah menggunakan AI! Mungkin kalian sering menggunakannya tanpa menyadari bahwa di baliknya ada kecerdasan buatan yang bekerja. Mari kita bahas bagaimana AI mengubah cara kita berinteraksi dengan teknologi!

Penjelasan Konsep (teori)

Kecerdasan Buatan (AI) bukan lagi sesuatu yang hanya ada di film fiksi ilmiah. AI sudah ada di sekitar kita, terintegrasi dalam berbagai aplikasi mobile yang kita gunakan setiap hari, bahkan seringkali tanpa kita sadari. AI meningkatkan fungsionalitas aplikasi dengan membuatnya lebih personal, efisien, dan mampu melakukan tugas-tugas yang kompleks.

Beberapa contoh bagaimana AI meningkatkan fungsionalitas aplikasi:

- **Filter di Instagram atau TikTok:** Ketika Anda menggunakan filter wajah atau efek augmented reality, AI sedang bekerja untuk mendeteksi fitur wajah Anda secara real-time dan menempatkan filter dengan tepat. Ini menggunakan teknik pengenalan gambar dan pelacakan objek.
- **Rekomendasi di TikTok, YouTube, atau Spotify:** Algoritma AI menganalisis kebiasaan menonton atau mendengarkan Anda (video/lagu apa yang Anda suka, berapa lama Anda menonton, apa yang Anda cari). Berdasarkan pola ini, AI merekomendasikan konten baru yang kemungkinan besar Anda sukai, membuat pengalaman Anda lebih personal dan menarik.
- **Terjemahan Instan di Google Translate:** AI, khususnya di bidang Natural Language Processing (NLP), memungkinkan aplikasi untuk menerjemahkan teks atau suara dari satu bahasa ke bahasa lain secara instan, bahkan dengan kemampuan mengenali tulisan tangan atau teks di gambar (OCR - Optical Character Recognition).
- **Asisten Virtual (Siri, Google Assistant):** Mereka menggunakan AI untuk memahami perintah suara Anda (Speech Recognition), memproses bahasa alami (Natural Language Understanding), dan memberikan respons yang relevan.
- **Pengenalan Wajah untuk Membuka Kunci HP:** Ini adalah bentuk pengenalan gambar yang sangat canggih, di mana AI dilatih untuk mengenali pola unik pada wajah seseorang.
- **Navigasi dan Peta (Google Maps):** AI digunakan untuk mengoptimalkan rute, memprediksi kemacetan berdasarkan data lalu lintas real-time, dan memberikan estimasi waktu perjalanan yang akurat.

Ilustrasi: Pemakaian Aplikasi dengan AI di Dunia Nyata

Bayangkan kamu sedang duduk di taman. Kamu melihat bunga unik dan penasaran namanya. Kamu buka Google Lens, arahkan kamera ke bunga itu, dan... *Tadaa!* Langsung muncul nama bunga, asal-usulnya, dan cara menanamnya. Nah, itu contoh **AI (Artificial Intelligence)** sedang bekerja membantumu secara real-time!

Atau bayangkan kamu lagi bosan, buka TikTok, dan langsung muncul video-video yang *kamu banget*. Kamu gak nyari, tapi kok pas semua? Itu karena AI membaca kebiasaanmu, dan memberikan *konten yang cocok* untuk kamu.

Kesimpulannya: AI membuat aplikasi lebih pintar, lebih cepat paham kamu, dan jauh lebih berguna!

Contoh Kasus atau Ilustrasi

Kasus 1: Kamera Ajaib Pengenal Benda

Bayangkan kamu lagi jalan-jalan ke taman kota. Kamu lihat bunga unik yang belum pernah kamu lihat sebelumnya. Kamu buka aplikasi **Google Lens**, arahkan kamera ke bunga itu... dan *taraa!* muncul nama latinnya, asalnya dari mana, bahkan cara merawatnya!

Ilustrasi: Seolah-olah HP-mu punya "mata ajaib" yang bisa mengenali dunia sekitar. Google Lens ini menggunakan AI untuk membaca gambar dan memberi informasi. Bukan cuma bunga — coba arahkan ke gedung, makanan, bahkan tulisan dari buku cetak!

Kasus 2: Bahasa Alien dari Teman Baru

Kamu main game online dan dapat pesan dari pemain luar negeri. Isinya pakai bahasa asing, kamu bingung banget! Tapi tenang, kamu salin pesannya ke Google Translate, dan langsung muncul terjemahannya dalam Bahasa Indonesia.

Ilustrasi: AI seperti penterjemah super cepat yang bisa bantu kamu ngobrol dengan siapa saja di seluruh dunia. Ia mengenali ribuan kata, memahami konteks, dan menyesuaikan artinya agar tetap nyambung.

Kasus 3: Guru Virtual yang Paham Kamu Banget

Kamu belajar bahasa Inggris lewat aplikasi seperti Duolingo. Ternyata, kamu sering salah kalau soal *verb* atau kata kerja. Aplikasinya tiba-tiba ngasih latihan khusus bagian itu, lho!

Ilustrasi: Seolah-olah kamu punya guru pribadi yang terus mengamati pola belajarmu dan tahu bagian mana yang kamu butuh bantuan. Ini kerjaan AI — dia membaca kebiasaanmu dan memberikan materi yang pas buatmu.

Kasus 4: Penjaga Video Pintar

Saat kamu buka YouTube Kids, videonya selalu tentang eksperimen sains, fakta hewan, dan dongeng seru. Kok bisa ya? AI bekerja seperti penjaga gerbang yang memastikan kamu hanya menonton video yang sesuai usia, edukatif, dan aman.

Ilustrasi: AI seperti kakak pengasuh virtual yang selalu memilihkan tontonan terbaik buatmu, berdasarkan apa yang kamu suka dan juga aturan keamanan untuk anak seusiamu.

Kasus 5: Musik dari Suara Ajaib

Kamu sedang makan siang dan tangannya kotor. Tapi kamu ingin dengar musik. Tinggal bilang, *"OK Google, putar lagu favoritku!"* dan musik langsung jalan.

Ilustrasi: Seolah-olah kamu punya asisten pribadi yang selalu siap mendengarkan dan melayani perintah hanya lewat suara. Teknologi AI ini mendengarkan ucapanmu, mengenal suaramu, lalu menjalankan perintah dengan cerdas.

Kasus 6: Pengenalan Tulisan Otomatis

Kamu lihat buku teman berisi catatan rapi, dan kamu ingin menyalin cepat. Kamu foto catatannya pakai Google Lens, dan... teksnya bisa langsung disalin ke HP kamu, bahkan diterjemahkan!

Ilustrasi: AI bertindak seperti mesin fotokopi digital yang bisa mengubah tulisan menjadi teks, lalu bisa diedit, disalin, atau diterjemahkan. Canggih dan super membantu belajar!

Jika kamu tertarik, kamu bisa bereksperimen sendiri di rumah dengan mencoba aplikasi-aplikasi tersebut. Dunia AI itu luas dan seru — siapa tahu kamu bisa menciptakan teknologi hebat juga suatu hari nanti!

Contoh Soal dan Pembahasan

Soal:

Menurutmu, bagaimana AI bisa membantu dalam aplikasi pendidikan? Berikan minimal dua contoh konkret.

Jawaban:

AI memiliki potensi besar untuk merevolusi pendidikan. Contohnya:

1. **Personalisasi Pembelajaran:** Aplikasi pendidikan berbasis AI dapat menganalisis gaya belajar, kekuatan, dan kelemahan setiap siswa. AI kemudian dapat menyesuaikan materi pelajaran, kecepatan, dan jenis latihan yang paling efektif untuk setiap siswa, mirip dengan tutor pribadi. Misalnya, jika AI mendeteksi siswa kesulitan pada topik Matematika tertentu, ia akan memberikan latihan tambahan khusus topik itu.
2. **Sistem Penilaian Otomatis dan Umpan Balik Instan:** AI dapat digunakan untuk menilai tugas-tugas atau esai secara otomatis, memberikan umpan balik instan kepada siswa tentang kesalahan

mereka. Ini membebaskan waktu guru dan memungkinkan siswa untuk segera memperbaiki kesalahannya tanpa menunggu lama.

3. **Aplikasi Belajar Bahasa:** AI dapat membantu siswa berlatih berbicara dan memahami bahasa asing dengan menyediakan fitur pengenalan suara untuk memeriksa pelafalan atau fitur chatbot yang bisa diajak berbicara dalam bahasa target.

Soal:

Apa manfaat AI dalam aplikasi seperti Google Maps?

Jawaban: AI menganalisis data lalu lintas secara real-time dan memberi rute tercepat untuk sampai ke tujuan. Ini sangat membantu agar kita tidak terjebak macet.

Soal:

Bagaimana AI membantu dalam aplikasi belanja?

Jawaban: AI merekomendasikan produk berdasarkan apa yang sering kita cari, lihat, atau beli. AI juga membantu menampilkan diskon atau penawaran yang cocok buat kita.

Soal:

Beri contoh aplikasi AI yang digunakan untuk belajar bahasa.

Jawaban: Duolingo. AI disana menyesuaikan tingkat kesulitan soal berdasarkan jawaban kita sebelumnya, dan memberikan tips belajar yang sesuai dengan kelemahan kita.

Soal:

Mengapa filter wajah bisa pas mengikuti wajah kamu saat kamu bergerak?

Jawaban: Karena AI mengenali titik-titik penting di wajah dan mengikutinya secara real-time menggunakan teknologi *computer vision*.

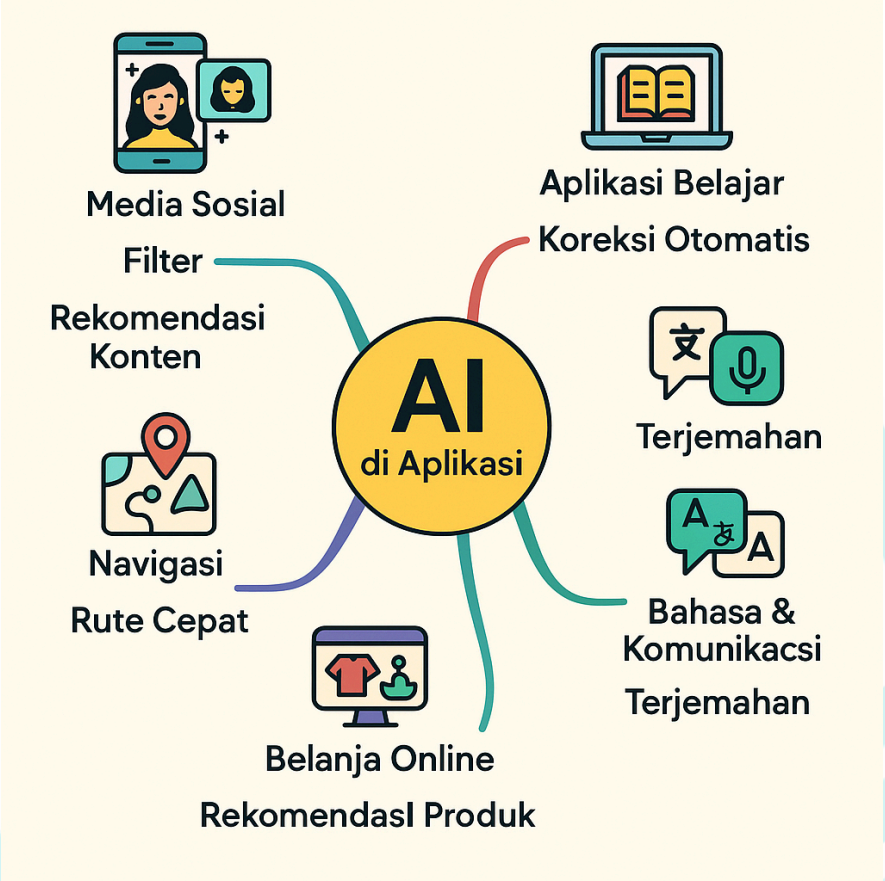
Fakta Menarik

Hampir semua aplikasi populer yang kalian gunakan setiap hari, mulai dari media sosial, e-commerce, hingga game, sekarang menggunakan AI dalam berbagai bentuknya! Dari rekomendasi produk, pengaturan privasi, hingga mendeteksi *spam*, AI ada di mana-mana!

Tips Cepat Menghafal

Ingat intinya: **"AI membuat Aplikasi Lebih Pintar, Lebih Personal, dan Lebih Bermanfaat!"**

Mind Mapping AI:



Cerita Lucu & Gampang Diingat:

"Bayu ingin traveling, tapi gak tahu arah, gak bisa bahasa asing, dan bingung mau makan apa. Tapi HP-nya supercerdas! Ada Google Maps buat arah, Google Translate buat ngobrol, dan Google Lens buat tahu menu makanan. Semua pakai AI. Jadi Bayu tetap bisa keliling dunia dengan tenang!"

Aktivitas Siswa

Aktivitas 1: Eksplorasi Aplikasi Cerdas: Diskusi Seru Tentang AI di Genggamanmu

Langkah-langkah:

- Bagi siswa ke dalam kelompok kecil (3–4 orang).
- Minta mereka menyebutkan 3–5 aplikasi favorit yang sering mereka gunakan.

- Ajak siswa mengamati dan mendiskusikan fitur-fitur aplikasi tersebut yang menurut mereka "cerdas" atau terasa seperti menggunakan teknologi AI.
- Tantang kelompok untuk mencari tahu **bagaimana AI membuat aplikasi tersebut lebih menyenangkan atau bermanfaat**.
- Akhiri dengan presentasi singkat dari tiap kelompok untuk saling bertukar wawasan.

Tujuan: Siswa belajar mengenali peran AI dalam kehidupan sehari-hari melalui diskusi aktif dan kolaboratif.

Aktivitas 2: AI Hunter: Misi Menemukan Kecerdasan Buatan di Ponselmu

Langkah-langkah:

- Siswa diajak menjadi "detektif teknologi" dengan menjelajahi 5 aplikasi di HP mereka.
- Tantangan: Temukan fitur-fitur yang kemungkinan menggunakan AI! (Misalnya: filter wajah, rekomendasi musik, prediksi teks).
- Ajak siswa menuliskan **nama aplikasi, fitur AI-nya, dan manfaatnya bagi pengguna**.
- Guru bisa memfasilitasi sesi berbagi temuan paling unik di kelas.

Tujuan: Melatih kepekaan siswa dalam mengenali teknologi cerdas di sekitar mereka.

Aktivitas 3: Tantangan Tebak Aplikasi: Siapa Paling Paham AI?

Langkah-langkah:

- Guru menyebutkan fungsi dari sebuah fitur AI (misalnya: "Memprediksi kata selanjutnya saat mengetik").
- Siswa secara individu atau kelompok kecil menebak aplikasi yang memiliki fitur tersebut.
- Bisa dikembangkan menjadi kuis seru dengan skor dan hadiah kecil.

Tujuan: Melatih logika dan pengetahuan siswa tentang penerapan AI dalam aplikasi populer.

Aktivitas 4: Simulasi Logika AI: Pseudocode Seru untuk Otak Cerdas

Langkah-langkah:

- Guru memberikan contoh sederhana logika AI, seperti sistem adaptif dalam latihan soal.
- Ajak siswa menulis pseudocode (tanpa harus bisa coding) untuk situasi tertentu. Contoh: Bagaimana AI tahu kapan harus memberi soal lebih sulit atau lebih mudah?
- Diskusikan alur logika yang mereka buat. Dorong kreativitas dan penalaran!

Tujuan: Mengenalkan logika berpikir komputasional yang digunakan AI dengan cara yang menyenangkan.

Aktivitas 5: Desain AI Superhero: Kreativitas dalam Dunia Teknologi

Langkah-langkah:

- Ajak siswa menciptakan tokoh superhero berbasis AI yang bisa membantu manusia di dunia nyata.
- Siswa menggambar dan menuliskan fitur-fitur AI tokoh tersebut: Apa kemampuannya? Siapa yang dia bantu? Bagaimana cara kerjanya?
- Pamerkan hasil karya di kelas sebagai galeri mini.

Tujuan: Menggabungkan kreativitas seni dan pemahaman konsep AI dalam konteks sosial.

Aktivitas 6: AI di Masa Depan: Cerita Pendek dari Masa Depan

Langkah-langkah:

- Ajak siswa menulis cerita pendek tentang kehidupan seseorang di masa depan yang sangat bergantung pada AI.
- Minta mereka menyertakan setidaknya 2 teknologi AI yang realistis namun inovatif.
- Diskusikan bersama: Apakah teknologi tersebut membantu atau justru menantang manusia?

Tujuan: Mendorong siswa berpikir kritis dan imajinatif tentang dampak jangka panjang AI.

Rangkuman

Melalui bab refleksi ini, kita telah melihat bahwa AI bukan hanya teknologi masa depan, tetapi sudah menjadi bagian integral dari kehidupan sehari-hari kita melalui berbagai Aplikasi Mobile. AI meningkatkan fungsionalitas aplikasi dengan menyediakan fitur-fitur cerdas seperti Rekomendasi, Filter, Terjemahan Instan, dan Pengenalan Objek. Pemahaman tentang Manfaat AI dan Dampak Aplikasi yang menggunakannya akan mendorong kita untuk berinovasi dan memanfaatkan potensi AI lebih jauh.

AI sudah menjadi bagian penting dalam banyak aplikasi yang kita pakai sehari-hari, dari media sosial, pendidikan, hiburan, sampai belanja online. Teknologi AI membantu membuat aplikasi jadi lebih *pintar*, lebih *personal*, dan lebih *berguna*. AI bisa mengenali wajah, memahami suara, menerjemahkan bahasa, dan memberi rekomendasi berdasarkan kebiasaan kita.

Dengan memahami bagaimana AI bekerja di balik aplikasi-aplikasi tersebut, kita bukan hanya menjadi pengguna yang cerdas, tetapi juga bisa menjadi calon inovator yang menciptakan aplikasi cerdas untuk masa depan!

Latihan Soal

A. Benar atau Salah (5 Soal)

1. AI hanya bisa digunakan di film dan robot, tidak di aplikasi mobile.
   Jawaban: Salah

2. Filter wajah di Instagram bekerja dengan bantuan AI.
   Jawaban: Benar

3. AI bisa memahami perintah suara kita melalui teknologi pengenalan suara.
   Jawaban: Benar

4. Aplikasi navigasi seperti Google Maps tidak menggunakan AI.
   Jawaban: Salah

5. AI dapat membantu menerjemahkan tulisan tangan ke dalam teks digital.
   Jawaban: Benar

B. Pilihan Ganda (10 Soal)

6. Apa fungsi AI dalam aplikasi Spotify?
   A. Mengedit foto
   B. Memprediksi cuaca
   C. Memberi rekomendasi lagu
   D. Menambah sinyal internet
   Jawaban: C

7. Teknologi yang memungkinkan AI menerjemahkan suara ke teks disebut:
   A. Face Recognition
   B. Speech Recognition
   C. Image Tracking
   D. Text Mining
   Jawaban: B

8. Salah satu manfaat AI dalam aplikasi seperti TikTok adalah:
   A. Menambah followers otomatis
   B. Menyaring komentar
   C. Memberikan rekomendasi video sesuai minat
   D. Menghapus akun palsu
   Jawaban: C

9. AI dalam Google Maps digunakan untuk:
   A. Membuat peta baru
   B. Memprediksi lalu lintas dan memberikan rute tercepat
   C. Mengambil foto jalan
   D. Menghapus jalan yang rusak
   Jawaban: B

10. Teknologi AI yang mengenali objek melalui kamera disebut:
    A. Natural Language Processing
    B. Virtual Reality
    C. Computer Vision
    D. Voice Assistant
    Jawaban: C

11. Aplikasi yang menggunakan AI untuk mengenali tulisan tangan adalah:
    A. WhatsApp

B. Google Translate
C. TikTok
D. YouTube
Jawaban: B

12. Rekomendasi playlist "Daily Mix" di Spotify dibuat oleh:
A. DJ
B. AI berdasarkan kebiasaan pengguna
C. Pilihan acak
D. Pengguna lain
Jawaban: B

13. AI membuat pengalaman pengguna lebih:
A. Lambat dan rumit
B. Pribadi dan efisien
C. Sulit dan teknis
D. Seragam untuk semua orang
Jawaban: B

14. Google Lens menggunakan AI untuk:
A. Menyimpan foto ke cloud
B. Menghapus latar belakang foto
C. Mengidentifikasi objek melalui kamera
D. Memblokir sinyal
Jawaban: C

15. Teknologi AI yang memahami bahasa manusia disebut:
A. Natural Language Processing
B. Augmented Reality
C. Data Mining
D. Neural Network
Jawaban: A

C. Isian Singkat (5 Soal)

16. AI pada fitur pengenalan wajah digunakan untuk membuka kunci _____.
Jawaban: HP

17. Algoritma yang menganalisis pola penggunaan pengguna disebut _____.
Jawaban: AI

18. _____ adalah aplikasi yang bisa menerjemahkan teks dan suara secara langsung menggunakan AI.

Jawaban: Google Translate

19. Contoh aplikasi yang menggunakan filter wajah berbasis AI adalah _____.
Jawaban: Instagram

20. Aplikasi peta yang menggunakan AI untuk mencari rute tercepat adalah _____.
Jawaban: Google Maps

D. Soal Eksplorasi

Soal:

Bayangkan kamu sedang membuat aplikasi belajar bahasa untuk anak-anak. Jelaskan satu fitur AI yang ingin kamu tambahkan ke aplikasi tersebut, dan bagaimana fitur itu akan membantu pengguna belajar dengan lebih menyenangkan.

Jawaban: Saya ingin menambahkan fitur chatbot berbasis AI yang bisa berbicara dan merespons anak-anak dalam bahasa asing yang sedang dipelajari. Fitur ini akan membuat anak bisa berlatih percakapan dengan cara yang menyenangkan, seperti bermain dengan teman virtual.

Soal:

Ceritakan pengalamanmu menggunakan salah satu aplikasi yang menurutmu memakai AI. Fitur apa yang kamu gunakan, dan bagaimana menurutmu AI membantu meningkatkan pengalaman kamu saat menggunakan aplikasi itu?

Jawaban: Saya sering menggunakan YouTube dan merasa AI-nya sangat membantu karena rekomendasi video yang muncul selalu sesuai dengan minat saya. Setelah menonton beberapa video tentang hewan, AI langsung menyarankan video serupa, jadi saya tidak perlu mencari sendiri.

Soal:

Pilih satu aplikasi yang belum disebutkan dalam materi, lalu jelaskan bagaimana aplikasi itu mungkin menggunakan AI. Apa manfaatnya bagi pengguna?

Jawaban: Shopee adalah aplikasi belanja yang menurut saya menggunakan AI untuk merekomendasikan produk berdasarkan riwayat pencarian dan pembelian saya. Dengan AI, saya bisa menemukan barang yang saya butuhkan lebih cepat tanpa harus mencarinya satu per satu.

Tugas Proyek

Berikut beberapa soal proyek yang bisa kalian kerjakan secara individu atau kelompok:

Proyek 1: Jelajahi 5 Aplikasi AI di Sekitarmu

Tujuan: Mengenali penerapan AI di aplikasi yang digunakan sehari-hari.

Langkah-langkah:

1. Pilih 5 aplikasi mobile yang sering kamu gunakan (contoh: Spotify, TikTok, WhatsApp).
2. Cari tahu fitur AI apa yang digunakan dalam aplikasi tersebut.
3. Jelaskan bagaimana AI bekerja di dalamnya dan apa manfaatnya bagi pengguna.
4. Buatlah dalam bentuk tabel atau infografik agar menarik dan mudah dipahami.

Saran Fun: Tambahkan emoji atau ikon dari masing-masing aplikasi agar presentasi lebih hidup!

Proyek 2: Ciptakan Aplikasi AI Impianmu

Tujuan: Melatih kreativitas dalam merancang solusi AI untuk dunia pendidikan.

Langkah-langkah:

1. Bayangkan aplikasi impian yang bisa membantu kamu dan teman-teman belajar lebih mudah.
2. Tuliskan nama aplikasi, siapa penggunanya, dan apa yang dilakukan aplikasi tersebut.
3. Jelaskan fitur AI-nya, misalnya: chatbot pengingat tugas, membuat rangkuman otomatis, atau kuis pintar.
4. Tambahkan manfaatnya dan bagaimana AI membuat aplikasi tersebut lebih pintar.

Saran Menantang: Coba buat sketsa tampilan aplikasi kamu di kertas atau aplikasi desain sederhana seperti Canva!

Proyek 3: AI: Teman atau Tantangan?

Tujuan: Mengembangkan kemampuan berpikir kritis tentang dampak AI.

Langkah-langkah:

1. Tuliskan 2 manfaat AI yang kamu rasakan dari aplikasi (misalnya: efisiensi belajar, hiburan yang sesuai selera).

2. Tuliskan 1 kemungkinan dampak negatif dari penggunaan AI (misalnya: kecanduan, privasi terganggu).
3. Buat perbandingan dalam bentuk tabel "Positif vs Negatif".
4. Tambahkan pendapat pribadi: apakah kamu merasa AI lebih banyak manfaatnya atau bahayanya?

Saran Diskusi: Coba tanyakan pendapat teman atau guru, lalu bandingkan pandangan mereka dengan milikmu.

Proyek 4: AI di Balik Media Sosial

Tujuan: Memahami cara kerja AI dalam aplikasi yang paling populer di kalangan remaja.

Langkah-langkah:

1. Pilih salah satu aplikasi media sosial: TikTok, Instagram, atau YouTube.
2. Telusuri bagaimana AI digunakan untuk mengatur konten (misalnya: "FYP", rekomendasi video, deteksi komentar kasar).
3. Buat poster atau slide berisi penjelasan singkat + ilustrasi.
4. Sertakan diagram sederhana atau komik mini untuk menjelaskan proses AI.

Saran Fun: Gunakan warna cerah dan gaya desain ala media sosial agar hasilnya makin keren!

Proyek 5: AI Membantu Belanja Online

Tujuan: Menyadari peran AI dalam kehidupan sehari-hari, khususnya saat belanja online.

Langkah-langkah:

1. Telusuri bagaimana aplikasi belanja seperti Shopee, Tokopedia, atau Lazada menggunakan AI.
2. Jelaskan 2 fitur berbasis AI, seperti:
    - **Rekomendasi produk**: AI menyarankan barang yang mungkin kamu suka.
    - **Chatbot pintar**: AI membantu menjawab pertanyaan pembeli secara otomatis.
3. Buatlah tulisan singkat atau video berdurasi 1 menit yang menjelaskan fitur-fitur ini.

Saran Kreatif: Coba pura-pura jadi penjual dan buat simulasi tanya-jawab dengan chatbot.

Proyek 6: Proyek Bonus: AI Detektif Data

Tujuan: Mengenal bagaimana AI "belajar" dari data.

Langkah-langkah:

1. Kumpulkan data kecil (misalnya: selera warna teman-teman, makanan favorit, atau lagu kesukaan).
2. Kelompokkan data tersebut dan coba "prediksi" pilihan berdasarkan pola.
3. Jelaskan bahwa AI juga bekerja dengan cara serupa, tapi dengan data yang jauh lebih besar!

Saran Eksperimen: Buat kuis prediksi gaya teman berdasarkan jawaban mereka – kamu jadi "AI" mini!

BAB 7: 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:

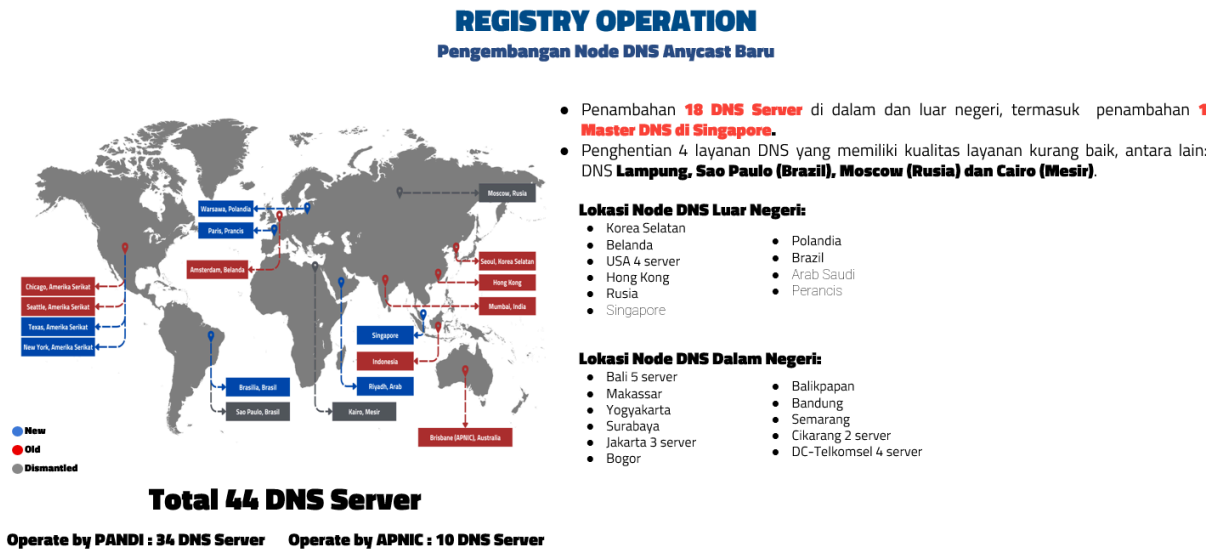
- .jp → Jepang
- .us → Amerika Serikat
- .id → Indonesia 🇮🇩

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

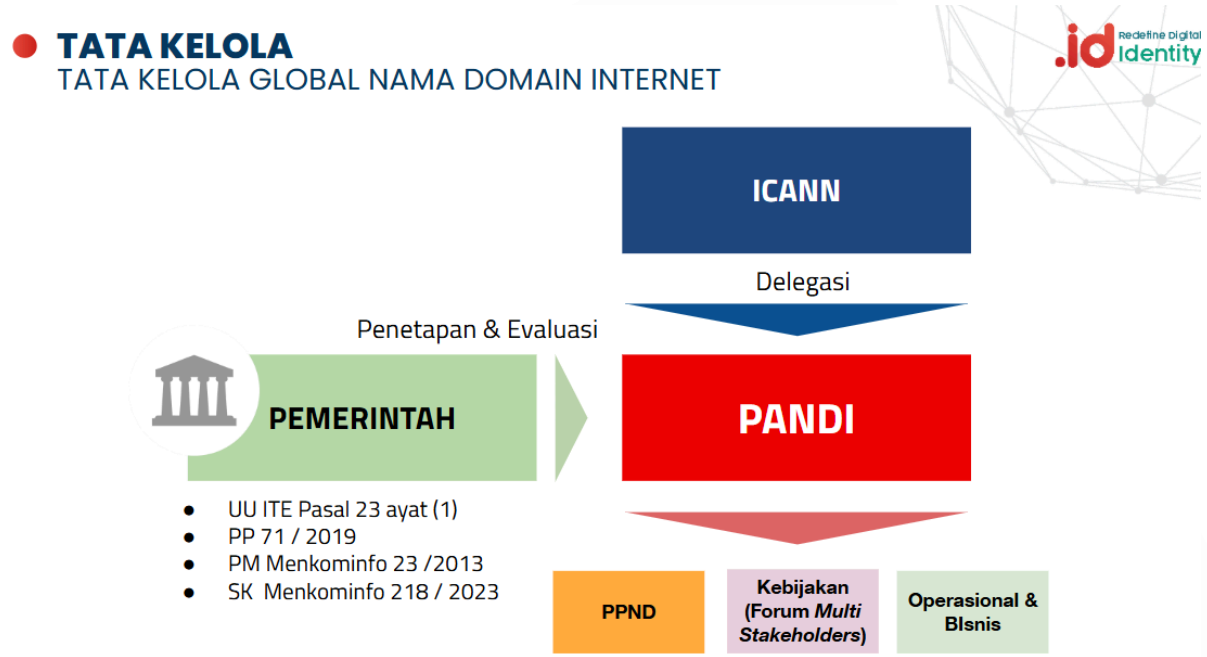
Tugas

- 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

Artificial Intelligence Sekolah Menengah Pertama (SMP) dirancang khusus untuk membantu siswa SMP memahami dunia kecerdasan buatan (AI) dan pemrograman aplikasi secara menyenangkan dan aplikatif. Dimulai dari pengenalan App Inventor 2 hingga membuat aplikasi sederhana seperti kalkulator dan survey, siswa diajak memahami konsep dasar, blok pemrograman, serta praktik langsung yang interaktif.

Lebih jauh, buku ini mengenalkan Google Teachable Machine dan Personal Image Classifier (PIC) di App Inventor 2, sehingga siswa dapat belajar membuat model machine learning sendiri, melakukan klasifikasi gambar, dan memahami bagaimana AI digunakan di aplikasi sehari-hari. Selain teori, buku ini diperkaya dengan contoh kasus, ilustrasi, fakta menarik, tips cepat menghafal, serta tugas proyek yang menantang kreativitas dan keterampilan siswa.

Di bagian akhir, siswa juga diperkenalkan pada domain .id dan peran PANDI, sehingga tidak hanya belajar coding dan AI, tetapi juga memahami aspek digital yang relevan dengan dunia teknologi informasi. Buku ini menjadi panduan lengkap bagi siswa SMP untuk mengembangkan kemampuan teknis, berpikir kritis, dan kreativitas digital sejak dini.

InTech press