

# MBG (Makan Bergizi Ga Bocor)

## 1. Pilar AI: Penentuan Prioritas & Alokasi Sasaran

(Alur ini tetap sama. AI bekerja di awal untuk menentukan "Sekolah X" dan "Dana Y").

1. **Input:** AI mengumpulkan data (Kemiskinan, Dapodik, Stunting).
2. **Proses:** AI melakukan *scoring* dan *ranking* untuk menghasilkan "Daftar Prioritas Sekolah".
3. **Output:** AI menetapkan "Sekolah X mendapat alokasi program senilai Y Rupiah, yang akan dieksekusi oleh Katering Z (mitra pemenang tender)."

## 2. Pilar Blockchain: Alur Dana "Escrow" (Pusat ke Katering)

Ini adalah alur uang (dana) yang baru, di mana sekolah tidak memegang *wallet*.

**Aktor dalam Alur Ini:**

- **Payer:** Pemerintah (Pusat/Dinas Daerah).
- **Payee:** Mitra Katering/Pemasok.
- **Verifier:** Sekolah (Kepala Sekolah/Guru yang ditunjuk).

**Alur Kerja Teknis:**

1. **Inisiasi Anggaran (Pusat ke Daerah):**
  - Pemerintah Pusat mentransfer total anggaran program ke *digital wallet* Dinas Pendidikan Daerah yang terdaftar di Blockchain. (Ini untuk akuntabilitas anggaran daerah).
2. **Dana Dikunci (Escrow):**
  - Berdasarkan *output* AI (Poin 1.3), Dinas Daerah **mengunci (lock)** dana ke dalam **Smart Contract Escrow**.
  - **Contoh:** Smart contract di-setting: "Dana Y (misal: Rp 15.000.000) untuk Katering Z. Dana ini akan cair jika *Verifier* (Sekolah X) memberikan sinyal konfirmasi."
  - Pada tahap ini, Katering Z bisa melihat di *dashboard* bahwa dana untuk mereka sudah tersedia dan *dijamin* oleh sistem, tapi mereka belum bisa mengambilnya.
3. **Pengiriman dan Verifikasi (Aksi di Lapangan):**
  - Katering Z mengirimkan makanan (sesuai kontrak) ke Sekolah X.
  - Sekolah X (Kepala Sekolah/Guru) menerima makanan. Melalui aplikasi NutriChain (yang terhubung ke *backend* blockchain), mereka melakukan verifikasi.
  - **Aksi Verifikasi:** Cukup menekan tombol "Konfirmasi Penerimaan: 100 porsi diterima, kualitas sesuai" atau memindai QR *delivery order* dari katering.
4. **Eksekusi Pembayaran Otomatis (Pelepasan Dana):**
  - Smart Contract secara *real-time* mendeteksi sinyal "Konfirmasi" dari *verifier* (Sekolah X).

- Saat itu juga, Smart Contract *otomatis* melepaskan **(release)** dana Y (Rp 15.000.000) yang tadi terkunci, dan mentransfernya langsung ke *digital wallet* Katering Z.
- 

### Ringkasan Sederhana (Sesuai Revisi Anda)

1. **AI:** Menentukan "Sekolah X butuh 100 porsi makanan (senilai Y Rupiah) dari Katering Z."
2. **Pemerintah (Dinas):** Menaruh/mengunci dana Y ke dalam "Brankas Digital" (Smart Contract Escrow).
3. **Katering Z:** Mengirim 100 porsi makanan ke Sekolah X.
4. **Sekolah X:** Menerima makanan, lalu (lewat HP/aplikasi) menekan tombol "Barang Sudah Diterima."
5. **Blockchain:** "Brankas Digital" otomatis terbuka dan langsung mengirimkan dana Y ke Katering Z.

**Intinya:** Anda benar. Sekolah X tidak pernah menyentuh uangnya (Dana Y). Sekolah hanya bertindak sebagai "pemegang kunci" untuk membuka brankas agar Katering Z bisa dibayar. Ini memotong birokrasi, mempercepat pembayaran ke katering, dan mencegah sekolah terbebani urusan administrasi keuangan.

# Prompt untuk Mendesain Website "NutriChain"

**Judul Proyek:** Website Platform "NutriChain"

**Tujuan Utama Website:** Menciptakan sebuah *platform* (website) yang memiliki dua fungsi utama:

1. **Dashboard Publik:** Sebagai etalase transparansi *real-time* bagi publik (LSM, orang tua, auditor) untuk melacak alokasi dana, status distribusi, dan data prioritas.
2. **Portal Internal:** Sebagai alat kerja fungsional bagi para pemangku kepentingan (Dinas/Pemerintah, Mitra Katering, dan Sekolah) untuk menjalankan proses bisnis inti (eksekusi dana, verifikasi, dan pelaporan).

## ## 1. Target Pengguna (User Personas) & Hak Akses

Tolong definisikan fitur dan tampilan berdasarkan 4 persona pengguna utama ini:

- **A. Publik (Auditor, LSM, Orang Tua):**
  - **Kebutuhan:** Ingin transparansi penuh. Mau tahu "Uang saya dipakai ke mana?", "Sekolah mana yang dapat?", dan "Apakah programnya jalan?".
  - **Hak Akses:** Tidak perlu *login*. Hanya bisa melihat (read-only) data yang sudah disterilisasi (data publik).
- **B. Sekolah (Verifier):**
  - **Kebutuhan:** Butuh alat yang *sangat sederhana* untuk konfirmasi. Tidak mau direpotkan dengan administrasi.
  - **Hak Akses:** Perlu *login* (misal: menggunakan NIP/NPSN). Hanya punya satu fungsi: "Verifikasi Penerimaan".
- **C. Mitra Katering (Payee/Penerima Dana):**
  - **Kebutuhan:** Ingin kepastian pembayaran dan jadwal. Mau tahu "Kapan saya harus kirim?", "Status pembayaran saya bagaimana?", "Dokumen apa yang kurang?".
  - **Hak Akses:** Perlu *login*. Bisa melihat jadwal pengiriman, status pembayaran (Menunggu, Terbukti, Cair), dan riwayat transaksi.
- **D. Pemerintah/Dinas (Payer/Admin):**
  - **Kebutuhan:** Ingin mengelola program, mengeksekusi dana, dan melihat laporan kepatuhan.
  - **Hak Akses:** Perlu *login* (level admin). Bisa mengunci dana ke *smart contract*, melihat *output* AI (peta prioritas), dan memonitor performa semua katering dan sekolah.

## ## 2. Fitur Utama (Features)

Tolong jabarkan fitur-fitur berikut berdasarkan penggunaannya:

- **Untuk Publik (Dashboard Utama):**
  - **Peta Interaktif Prioritas (AI):** Peta Indonesia yang menunjukkan skor prioritas per wilayah (hasil *scoring* AI). Warna merah untuk prioritas tinggi, hijau untuk rendah.

- **Statistik Utama:** *Widget* besar menunjukkan: "Total Dana Teralokasi", "Jumlah Porsi Makanan Terdistribusi", "Jumlah Sekolah Terlayani".
- **Pelacak Transparansi (Blockchain):** *Feed* transaksi *real-time* yang disederhanakan. (Misal: "Pembayaran Rp 15.000.000 ke Katering Z untuk Sekolah X telah terverifikasi").
- **Fitur "Lihat Alokasi Sekolah Anda":** *Search bar* untuk publik mencari sekolah dan melihat apakah sekolah itu termasuk penerima atau tidak.
- **Untuk Sekolah (Portal Verifikasi):**
  - **Halaman Login Sederhana.**
  - **Dashboard "Tugas Hari Ini":** Tampilan seperti kartu yang berisi (Misal: "Pengiriman dari Katering Z - 100 Porsi - 10 Nov 2025").
  - **Tombol Verifikasi:** Dua tombol besar: [☒ Konfirmasi Sesuai] dan [☐ Laporkan Masalah].
- **Untuk Katering (Portal Mitra):**
  - **Dashboard Keuangan:** Menampilkan "Dana Terkunci (Escrow)", "Total Pendapatan", "Status Pembayaran Terakhir".
  - **Jadwal Pengiriman:** Kalender yang menunjukkan jadwal dan detail pengiriman (ke sekolah mana, berapa porsi).
  - **Riwayat Transaksi:** Bukti pencairan dana yang terhubung ke *link* Etherscan/Polygonscan (jika perlu).
- **Untuk Pemerintah/Dinas (Portal Admin):**
  - **Manajemen Anggaran:** Fitur untuk mengunci (lock) dana ke *smart contract escrow* untuk setiap katering/sekolah.
  - **Dashboard Monitoring:** Peta yang menunjukkan status *real-time* (mana sekolah yang sudah/belum verifikasi hari ini).
  - **Akses Laporan AI:** Melihat data mentah hasil *scoring* AI untuk perencanaan.
  - **Manajemen Mitra:** Mengelola akun Katering dan Sekolah.

### ## 3. Fungsionalitas (Functionality)

Tolong jelaskan *bagaimana* fitur-fitur kunci ini harus bekerja secara teknis di *backend*:

- **Fungsi Login:** Harus menggunakan sistem *role-based access control* (RBAC). *User* Sekolah tidak boleh bisa melihat *dashboard* Admin.
- **Fungsi Dashboard AI:** Website harus bisa mengambil *output* (misal: file JSON atau CSV) dari model AI yang sudah dilatih, lalu me-render-nya menjadi peta visual (menggunakan library seperti Mapbox atau Leaflet.js).
- **Fungsi Verifikasi (Koneksi ke Blockchain):**
  1. Tombol [☒ Konfirmasi Sesuai] yang ditekan Sekolah *tidak* langsung berinteraksi dengan blockchain (karena sekolah tidak punya *wallet*).
  2. Tombol itu mengirim sinyal (API call) ke *backend server* (milik NutriChain).
  3. *Backend server* inilah yang memiliki *wallet* (sebagai *service account*) dan *private key* untuk memanggil fungsi `releaseEscrow()` di *smart contract*.
  4. Ini memastikan prosesnya aman dan sekolah tidak perlu tahu apa itu *gas fee*.
- **Fungsi Dashboard Publik (Blockchain):** Website harus memiliki *listener* yang memantau *event* di *smart contract* (misal: *event* `PaymentReleased`). Ketika *event* terdeteksi, *database* website (misal: PostgreSQL) di-update, dan data baru itu yang

ditampilkan di *dashboard* publik (ini lebih cepat daripada memanggil *node* blockchain setiap kali *user* me-refresh halaman).

## ## 4. Alur Pengguna (User Flow)

Tolong buat alur *step-by-step* untuk skenario paling krusial:

- **Alur 1: Proses Pembayaran (Happy Path)**
  1. **Admin (Dinas):** Login -> Buka menu "Manajemen Anggaran" -> Pilih "Katering Z" dan "Sekolah X" -> Masukkan nilai "Rp 15.000.000" -> Klik "Kunci Dana ke Escrow".
  2. **(Sistem):** *Backend* memanggil *smart contract* `lockFund()`.
  3. **Katering Z:** Login -> Lihat *dashboard* -> Status berubah jadi "Dana Rp 15.000.000 Terkunci". -> Katering Z mengirim makanan.
  4. **Sekolah X:** Login -> Lihat *dashboard* -> Ada notifikasi "Menunggu Verifikasi dari Katering Z". -> Klik "✅ Konfirmasi Sesuai".
  5. **(Sistem):** *Backend* menerima sinyal -> Memanggil *smart contract* `releaseEscrow()`.
  6. **Katering Z:** Menerima notifikasi -> *Dashboard* Keuangan berubah jadi "Dana Rp 15.000.000 Telah Cair".
  7. **Publik:** Melihat *feed* transparansi -> "Pembayaran Rp 15.000.000 ke Katering Z untuk Sekolah X telah selesai."

## ## 5. Tech Stack yang Direkomendasikan

Tolong sarankan *tech stack* modern yang *scalable* dan cocok untuk proyek ini:

- **Frontend (Dashboard):** Next.js atau React (untuk interaktivitas tinggi dan SEO di *dashboard* publik).
- **Backend (API & Logic):** Node.js (Express/NestJS) atau Python (Django/FastAPI). Python lebih disukai jika *backend* ini juga yang akan men-servis model AI.
- **Database:** PostgreSQL (untuk data relasional seperti akun *user*, jadwal, riwayat) DAN/ATAU MongoDB (jika datanya lebih tidak terstruktur).
- **Visualisasi Data:** D3.js, Chart.js (untuk grafik), Mapbox/Leaflet.js (untuk peta).
- **Blockchain:** Solidity (untuk *Smart Contract*) di-deploy di *chain* L2 seperti **Polygon** atau **Arbitrum** (untuk menekan *gas fee* agar operasional pemerintah murah).
- **Interaksi Blockchain:** Ethers.js / Web3.js (di sisi *backend*).
- **Hosting:** AWS (menggunakan EC2, RDS, S3) atau Vercel (untuk Frontend) dan Heroku/Render (untuk Backend).

---

**Tugas Anda:** Berdasarkan semua informasi di atas, tolong hasilkan draf spesifikasi yang lebih rinci, *sitemap* (peta situs), dan *wireframe* konseptual (bisa dalam bentuk teks/deskripsi) untuk setiap halaman utama.

Berdasarkan analisis dan penelusuran saya, jawabannya adalah:

**Belum, solusi spesifik yang mengintegrasikan AI *dan* Blockchain untuk gizi sekolah seperti "NutriChain" belum diimplementasikan secara penuh dan terpadu di Indonesia.**

Namun, ide Anda sangat relevan dan tepat waktu. Yang menarik adalah, **komponen-komponen dari solusi Anda saat ini sedang dikerjakan atau didiskusikan secara terpisah** oleh berbagai pihak di Indonesia, terutama terkait program "Makan Bergizi Gratis" (MBG).

Berikut adalah temuan yang paling relevan dengan konsep Anda:

### **1. Digitalisasi "Makan Bergizi Gratis" (MBG)**

Pemerintah secara aktif mendorong digitalisasi program MBG. Inisiatif ini memiliki tujuan yang sama dengan "NutriChain": **transparansi dan efisiensi**.

- **Sistem Pemantauan Digital:** Ada penelitian dan pengembangan (R&D) yang sedang dilakukan untuk membangun *platform* digital MBG. Fitur-fiturnya sangat mirip dengan proposal Anda (minus blockchain), seperti:
  - Integrasi data (termasuk Dapodik).
  - Pelacakan logistik (distribusi).
  - *Dashboard* monitoring untuk pemerintah.
- **Pengawasan Berbasis Bukti Visual:** Badan Gizi Nasional (BGN) dilaporkan akan menggunakan "**bukti visual**" (foto) dalam sistem pengawasan digitalnya. Ini adalah langkah awal yang sempurna untuk diintegrasikan dengan pilar **AI Computer Vision** Anda (untuk memverifikasi menu dan kualitas).

## 2. Diskusi Blockchain untuk Program MBG

Ide menggunakan Blockchain untuk transparansi dana MBG sudah muncul di ruang publik.

- **Opini Publik:** Terdapat artikel dan opini di media nasional yang secara spesifik mengusulkan penggunaan teknologi **blockchain untuk transparansi** dalam program Makan Bergizi Gratis. Ini menunjukkan bahwa ide Anda sejalan dengan apa yang dipikirkan oleh para ahli.

## 3. AI & Blockchain di Rantai Pasok (Secara Umum)

Di luar program sekolah, teknologi ini sudah mulai diadopsi di Indonesia:

- **AI untuk Rantai Pasok:** Perusahaan besar di Indonesia (seperti Unilever, Astra) sudah menggunakan AI untuk optimasi logistik, manajemen inventaris, dan prediksi permintaan. Pemerintah juga mendorong penggunaan AI untuk efisiensi nasional.
- **Blockchain untuk Rantai Pasok Pangan:** Beberapa perusahaan swasta telah mengimplementasikan blockchain untuk keterlacakan (*traceability*) produk. Contohnya, **Sunpride** menggunakan blockchain untuk melacak produk pisang mereka dari kebun hingga ke konsumen.

---

## Kesimpulan

Solusi "NutriChain" Anda **belum ada sebagai satu produk jadi** di Indonesia.

Kondisi saat ini adalah:

- **Pemerintah** sedang membangun sistem digital untuk **monitoring dan efisiensi** MBG (Pilar AI Anda, khususnya bagian *analytics* dan *computer vision*).
- **Sektor Swasta** sudah membuktikan bahwa **Blockchain** bisa dipakai untuk **rantai pasok pangan** di Indonesia.
- **Publik** sudah mendiskusikan perlunya **Blockchain** untuk **transparansi dana** MBG (Pilar Blockchain Anda).

Proposal Anda **sangat inovatif** karena mengusulkan untuk **menggabungkan semua bagian ini** (AI *scoring*, AI *vision*, dan Blockchain *escrow*) ke dalam satu *platform* terpadu untuk menyelesaikan masalah spesifik gizi sekolah.

Dataset:

<https://referensi.data.kemendikdasmen.go.id/> [ALTERNATIF] -> scraping manual!

**BPS = KKN GO**

map -> leaflet

<https://gis.bmkg.go.id/portal/dataapi>

<https://docs.xendit.co/docs/api-keys>

## PENJELASAN INTERFACE DAN PAGE:

# INTERFACE PUBLIK (Public Transparency Dashboard)

## Tujuan Utama

Memberi **pandangan helikopter** ke masyarakat:

“Uang negara untuk Makan Bergizi Gratis dipakai di mana, sekolah mana yang dapat, dan apakah benar jalan?”

Publik **tidak mengubah apa-apa**, tapi melihat hasil interaksi antara Pemerintah, Sekolah, dan Katering.

---

## Page-Level & Relasi ke Role Lain

## 1.1. Public Home Dashboard

### Fungsi utamanya:

- Menampilkan:
  - Total dana dialokasikan (dari keputusan Pemerintah).
  - Total dana cair (hasil aksi Sekolah + smart contract bayar Katering).
  - Total porsi makanan didistribusikan (akumulasi verifikasi distribusi dari Sekolah).
- Feed transaksi real-time:
  - Dibentuk dari **event di blockchain** ketika smart contract membayar Katering.

### Relasi ke role lain:

- **Pemerintah:** saat admin lock dana → angka “Total dana dialokasikan” naik.
- **Sekolah:** saat klik “Konfirmasi diterima” → memicu pencairan dana.
- **Katering:** saat dana cair → muncul sebagai transaksi di feed publik.

Publik bisa melihat:

“Sekolah X dapat bantuan, dikerjakan oleh Katering Z, dan sudah dibayar tanggal 10 Nov.”

---

## 1.2. Peta Prioritas Nasional (AI Scoring Map)

### Fungsinya:

- Menampilkan heatmap, misal:
  - Kabupaten A (merah) = prioritas tinggi → kemiskinan/stunting tinggi.
  - Kabupaten B (hijau) = prioritas rendah.

### Relasi ke role lain:

- **Pemerintah:**
  - AI scoring ini dipakai di admin dashboard untuk menentukan alokasi.

- **Sekolah:**
  - Sekolah di daerah prioritas tinggi akan **lebih dulu dimasukkan ke skema MBG.**
- **Katering:**
  - Vendor akan dapat kontrak lebih banyak di daerah-daerah tertentu, sesuai hasil AI.

Publik bisa cross-check:

“Oh pantes daerah X dapat duluan, scoring datanya memang tinggi.”

---

### 1.3. Sekolah Explorer & Detail Sekolah

**Fungsinya:**

- Cari sekolah → lihat:
  - Dapat MBG atau tidak,
  - Katering yang melayani,
  - Porsi/hari, status verifikasi, status masalah.

**Relasi ke role lain:**

- **Sekolah:**
  - Data di sini langsung berasal dari tindakan mereka (verifikasi / laporan masalah).
- **Katering:**
  - Nama vendor yang terikat kontrak dengan sekolah itu ditampilkan.
- **Pemerintah:**
  - Menentukan bahwa sekolah ini termasuk penerima & menetapkan katering-nya.

Publik bisa tanya:

“Sekolah adik saya di desa X, dapat MBG nggak? Realisasinya gimana?”

---

#### 1.4. Transparansi Dana (Blockchain Feed)

Fungsinya:

- Menampilkan semua pembayaran yang **benar-benar terjadi**:
  - Tanggal, sekolah, vendor, nominal, hash transaksi.

Relasi ke role lain:

- **Pemerintah**: memicu `lockFund()` di awal.
- **Sekolah**: memicu `releaseEscrow()` (melalui backend) saat konfirmasi.
- **Katering**: menjadi penerima dana.

Publik melihat **hasil final** sinergi tiga pihak ini.

---

## 2. INTERFACE SEKOLAH (Verifier Dashboard)

### Tujuan Utama

Membuat peran sekolah **sederhana & aman**:

“Sekolah hanya perlu bilang: makanan ini sudah benar-benar sampai & dibagikan ke murid.”  
tanpa harus pegang uang.

---

### Page-Level & Relasi ke Role Lain

#### 2.1. Dashboard “Tugas Hari Ini”

Apa yang sekolah lihat?

- Daftar pengiriman untuk hari itu:
  - “Katering Z → 100 porsi”

- Status: “Menunggu konfirmasi”

#### Relasi ke role lain:


- Data ini muncul karena:
  - **Pemerintah** membuat alokasi & menjadwalkan katering.
  - **Katering** menerima jadwal & mengirim.

Sekolah adalah “check point” terakhir.

---

## 2.2. Halaman “Konfirmasi Penerimaan”

#### Aksi sekolah:

- Klik “ Konfirmasi diterima” jika:
  - Porsi sesuai,
  - Makanannya layak.

#### Di backend terjadi:

1. Status **deliveries** untuk pengiriman itu → **CONFIRMED**.
2. Backend memanggil fungsi **releaseEscrow()** di smart contract.
3. Smart contract membayar Katering.

#### Relasi ke role lain:

- **Katering:**
  - Dashboard vendor menunjukkan: “Pembayaran cair.”
- **Pemerintah:**
  - Admin melihat: “Sekolah X sudah menerima hari ini.”
- **Publik:**

- Dashboard publik update feed: “Rp 15.000.000 dibayar ke Katering Z untuk Sekolah X.”

Sekolah di sini seperti:

“Penjaga kunci brankas digital”  
— hanya jika mereka bilang OK → brankas terbuka.

---

### 2.3. Halaman “Laporkan Masalah”

**Jika ada masalah:**

- Sekolah pilih:
  - Porsi kurang
  - Telat
  - Makanan tidak layak
  - dsb.
- Mengirim laporan ke sistem.

**Relasi ke role lain:**

- **Pemerintah:**
  - Laporan masuk ke halaman “Investigasi Masalah”.
  - Admin bisa:
    - Menahan dana,
    - Meminta pengiriman ulang,
    - Menandai vendor bermasalah.
- **Katering:**
  - Melihat status: “Payment on hold” dan laporan detail.
- **Publik:**

- Bisa melihat (secara agregat) bahwa di sekolah X pernah ada issue, tapi detail sensitif bisa disembunyikan.

Sekolah punya power untuk:

Menghentikan pencairan dana jika ada masalah di lapangan.

---

## 2.4. Riwayat Verifikasi

Sekolah bisa melihat:

- Semua pengiriman sebelumnya,
- Kapan mereka konfirmasi,
- Kapan ada masalah.

Relasi ke role lain:

- **Pemerintah:**
    - Data ini dipakai sebagai KPI: apakah sekolah rajin verifikasi atau sering telat.
  - **Katering:**
    - Bisa refleksi performa di sekolah tersebut.
- 

## 3. INTERFACE KATERING (Vendor Dashboard)

### Tujuan Utama

Memberi vendor:

**Kepastian jadwal + kepastian pembayaran,**  
tanpa harus kejar tanda tangan & birokrasi manual.

---

### Page-Level & Relasi ke Role Lain

#### 3.1. Beranda Vendor (Ringkasan)

Vendor melihat:

- “Dana terkunci untuk Anda: Rp xxx”
- “Distribusi hari ini: x sekolah, y porsi”

**Relasi ke role lain:**

- **Pemerintah:**
    - Saat admin lock dana untuk vendor – sekolah, angka “Dana terkunci” naik.
  - **Sekolah:**
    - Bertindak sebagai gatekeeper pembayaran – jadi vendor tau pentingnya deliver tepat.
- 

### 3.2. Jadwal Pengiriman

Tampilkan:

- Sekolah mana harus dikirim hari ini,
- Jam pengiriman,
- Jumlah porsi.

**Relasi ke role lain:**

- Ini adalah hasil:
  - Keputusan Pemerintah (alokasi & kontrak),
  - Hasil AI (sekolah mana diprioritaskan).

Vendor bisa menyiapkan operasionalnya sesuai jadwal resmi.

---

### 3.3. Status Pembayaran

Vendor melihat:

- Untuk setiap kontrak:
  - Dana: Terkunci / Menunggu Verifikasi / Cair.

**Relasi ke role lain:**

- Status berubah karena:
  - **Pemerintah** → Lock (escrow).
  - **Sekolah** → Konfirmasi (trigger release).
  - **Blockchain** → Mencatat pembayaran.

Vendor jadi punya **cashflow visibility**:

“Kalau semua sekolah verifikasi tepat waktu, tanggal segini saya cair sekian.”

---

### 3.4. Laporan Masalah & Reputasi

Vendor bisa melihat:

- Apakah ada pengiriman yang dilaporkan bermasalah,
- Statistik:
  - Berapa kali terlambat,
  - Berapa kali masalah kualitas.

**Relasi ke role lain:**

- Laporan ini dibuat oleh **Sekolah**, dibaca oleh **Pemerintah**, dan mempengaruhi reputasi vendor.
  - Bisa jadi faktor menentukan apakah vendor masih dipakai tahun depan.
- 

## 4. INTERFACE PEMERINTAH / DINAS (Admin Dashboard)

## Tujuan Utama

Memberi Pemerintah:

**kendali penuh tapi berbasis data & otomatis,**  
dengan jejak kuat untuk audit & akuntabilitas publik.

---

## Page-Level & Relasi ke Role Lain

### 4.1. Dashboard Nasional/Daerah

Admin melihat:

- Realisasi harian (berapa sekolah sudah menerima).
- Peta status:
  - Hijau = sekolah sudah terlayani.
  - Kuning = menunggu verifikasi.
  - Merah = belum ada distribusi / bermasalah.

Relasi ke role lain:

- **Sekolah:**
    - Kalau sekolah belum verifikasi, akan terlihat “belum selesai”.
  - **Katering:**
    - Jika sering telat → banyak sekolah merah.
  - **Publik:**
    - Angka agregat di sini diteruskan ke dashboard publik.
- 

### 4.2. AI Priority Allocation View

Admin melihat hasil AI:

- Ranking sekolah prioritas,

- Rekomendasi besaran alokasi.

**Relasi ke role lain:**

- Output ini menetapkan:
    - Sekolah mana yang masuk program duluan.
    - Vendor mana yang dikontrak di wilayah tertentu.
  - Dampaknya:
    - **Sekolah terpencil** → dapat akses lebih cepat.
    - **Sekolah elit** → bisa opt-out / bukan prioritas.
- 

### **4.3. Manajemen Anggaran (Lock Dana Escrow)**

Inilah “titik awal aliran dana”.

**Aksi:**

- Pilih: vendor, sekolah, nominal.
- Klik “**LOCK ESCROW**”.

**Di backend:**

- Mencatat alokasi di DB.
- Memanggil smart contract `lockFund(...)`.

**Relasi ke role lain:**

- **Katering:**
  - Melihat dana terkunci untuk kontrak tersebut.
- **Sekolah:**
  - Akan punya tugas verifikasi untuk periode itu.

- **Publik:**

- Bisa lihat bahwa “sekian dana sudah dialokasikan ke sekolah X via vendor Z.”
- 

#### **4.4. Monitoring Sekolah & Vendor**

Dua halaman terpisah, tapi konsepnya sama:

- **Monitoring Sekolah:**
  - Menilai apakah sekolah disiplin dalam verifikasi & pelaporan.
- **Monitoring Vendor:**
  - Menilai apakah vendor tepat waktu dan minim masalah.

**Relasi ke role lain:**

- **Data diambil dari:**
    - Aktivitas Sekolah (verifikasi & laporan).
    - Aktivitas Katering (pengiriman sesuai jadwal).
  - **Bisa berujung ke:**
    - Sanksi / penggantian vendor.
    - Pendampingan untuk sekolah tertentu.
- 

#### **4.5. Investigasi Masalah**

Menampilkan:

- Semua laporan masalah dari sekolah (detail: kapan, vendor mana, jenis masalah).

**Relasi ke role lain:**

- **Sekolah:** sumber laporan.
- **Katering:** bisa diminta kirim ulang / diberi teguran.

- **Publik:**

- Bisa ditampilkan versi diringkas, misal: “Ada 5 masalah di Kabupaten X bulan ini.”

Pemerintah jadi punya **alat intervensi cepat**, bukan hanya menunggu laporan manual.

---

#### 4.6. Laporan Program

Dipakai untuk:

- Ke Menteri,
- Ke BPK,
- Ke publik (versi ringkas).

**Relasi ke role lain:**

- Menggunakan semua data:
    - Alokasi (admin),
    - Realisasi (sekolah),
    - Pembayaran (katering),
    - AI scoring (prioritas).
- 

## Gambaran Besar Relasi Antar Role (bahasa manusia)

- **Pemerintah:**

- Menentukan **siapa** yang dapat, **berapa** dana, dan **vendor** mana → lock dana di smart contract.

- **Katering:**

- Melihat, “Oh, sudah ada dana terkunci untuk saya, tinggal saya kirim makanan ke sekolah.”

- **Sekolah:**

- Setelah benar-benar menerima makanan → klik **OK** → sistem melepaskan dana ke katering.

- **Publik:**

- Hanya duduk manis, buka website, dan bisa lihat:
  - “Apa yang dijanjikan pemerintah benar-benar jalan atau tidak, secara data.”

---

Kalau kamu mau, next step aku bisa:

- Bikin **tabel ringkas** “**Role** → **Pages** → **API** → **Dampak ke Role Lain**” untuk langsung kamu taruh di laporan teknis, atau
- Bikin **sequence diagram** 1 skenario lengkap (misal: 1 minggu distribusi di Sekolah X dari awal alokasi sampai pembayaran vendor).