

Perancangan dan Implementasi Smart Contract untuk Traceability Pangan pada Program Makanan Bergizi Gratis (MBG) di Indonesia

Abdillah AG^{1*}, Alhadi Saputra² Irfan Afriantoro³

^{1*,2,3} Program Studi Teknik Informatika, Fakultas Teknik, Universitas Pelita Bangsa, Kabupaten Bekasi, Jawa Barat, Indonesia.

Corresponding Email: abdillahag@pelitabangsa.ac.id ^{1*} alhadi.saputra@pelitabangsa.ac.id ² irfanafriantoro@pelitabangsa.ac.id ³

Histori Artikel:

Dikirim 26 Agustus 2026; *Diterima dalam bentuk revisi* 07 September 2026; *Diterima* 14 September 2026; *Diterbitkan* 28 September 2026. Semua hak dilindungi oleh Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) STMIK Indonesia Banda Aceh.

Abstrak

Ketertelusuran pangan merupakan kebutuhan penting dalam rantai pasok Program Makanan Bergizi Gratis (MBG) karena melibatkan banyak aktor, bahan pangan, proses produksi, dan distribusi. Kompleksitas tersebut membutuhkan mekanisme pencatatan yang mampu merepresentasikan riwayat bahan pangan serta hubungan antarkegiatan sehingga proses distribusi dapat ditelusuri dari sumber hingga penerima maupun sebaliknya. Penelitian ini bertujuan merancang dan mengimplementasikan MBGTraceability, yaitu smart contract berbasis blockchain untuk mendukung *food traceability* pada Program Makanan Bergizi Gratis di Indonesia. Penelitian menggunakan pendekatan *Design Science Research* yang mencakup identifikasi kebutuhan, perancangan, implementasi, pengujian, dan validasi fungsional. Model yang dikembangkan mencakup pengelolaan batch, perpindahan kepemilikan, penggunaan sebagian batch dalam produksi, hubungan produksi, distribusi, pelaporan insiden, dan *recall*. Pengujian dan validasi fungsional dilakukan melalui *automated test suite* dengan skenario positif dan negatif. Hasil pengujian menunjukkan bahwa seluruh 17 *test case* berhasil dijalankan (17 *passing*, 0 *failing*). Pengujian juga menunjukkan bahwa sistem mampu mempertahankan konsistensi kuantitas batch, menghubungkan bahan input dengan hasil produksi, serta mengaitkan distribusi dan insiden dengan unit *traceability*. Kontribusi penelitian terletak pada penerjemahan kebutuhan *traceability* spesifik MBG ke dalam model dan logika transaksi *smart contract* yang dapat diimplementasikan serta divalidasi secara fungsional. Hasil ini menunjukkan kelayakan awal pendekatan tersebut sebagai *proof of concept*, dengan keterbatasan bahwa skalabilitas, efisiensi gas dan biaya transaksi, performa jaringan, serta keamanan formal *smart contract* belum dievaluasi.

Kata Kunci: *Ood Traceability; Blockchain; Smart Contract; Program Makanan Bergizi Gratis; MBGTraceability.*

Abstract

Food traceability is an essential requirement in the supply chain of the Free Nutritious Meals Program (MBG) because it involves multiple actors, food ingredients, production processes, and distribution activities. This complexity requires a recording mechanism capable of representing the history of food ingredients and the relationships among activities so that the distribution process can be traced from the source to the recipient and vice versa. This study aims to design and implement MBGTraceability, a blockchain-based smart contract to support food traceability in the Free Nutritious Meals Program in Indonesia. The study applies the Design Science Research approach, which includes needs identification, design, implementation, testing, and functional validation. The developed model covers batch management, ownership transfer, partial batch usage in production, production relationships, distribution, incident reporting, and recall. Functional testing and validation were conducted through an automated test suite with positive and negative scenarios. The test results show that all 17 test cases were successfully executed (17 *passing*; 0 *failing*). The testing also shows that the system can maintain batch quantity consistency, link input materials with production outputs, and associate distribution and incidents with traceability units. The contribution of this study lies in translating MBG-specific traceability requirements into a smart contract transaction model and logic that can be implemented and functionally validated. The results indicate the initial feasibility of the approach as a proof of concept, with limitations in that scalability, gas efficiency and transaction cost, network performance, and formal smart contract security have not yet been evaluated.

Keyword: Food Traceability; Blockchain; Smart Contract; Free Nutritious Meals Program; MBGTraceability.

1. Pendahuluan

Traceability merupakan elemen penting dalam pengelolaan rantai pasok pangan karena memungkinkan pergerakan dan riwayat produk ditelusuri mulai dari sumber bahan, produksi, penyimpanan, hingga distribusi. Dalam konteks keamanan pangan, traceability mencakup penelusuran ke belakang (*backward traceability*) dan ke depan (*forward traceability*) serta membutuhkan koordinasi antarpelaku dan standarisasi proses (Behnke & Janssen, 2020). Blockchain telah digunakan dalam berbagai sistem *agri-food traceability* untuk mendukung pencatatan asal produk, proses produksi, transportasi, penyimpanan, dan riwayat transaksi (Demestichas *et al.*, 2020; Wang *et al.*, 2022).

Kompleksitas traceability meningkat ketika rantai pasok melibatkan banyak organisasi dengan sistem pencatatan yang berbeda. Kondisi tersebut dapat menyulitkan rekonstruksi riwayat produk dan membutuhkan standarisasi, pengendalian akses, keamanan pertukaran data, serta interoperabilitas antarpelaku (Behnke & Janssen, 2020; Lei *et al.*, 2022). Pemanfaatan teknologi digital pada sektor pangan juga menunjukkan pentingnya penguatan aliran informasi antara pelaku hulu dan hilir (Darodjat & Utarie, 2022; Perdana *et al.*, 2022). Selain itu, penerapan blockchain pada *agri-food provenance* perlu mempertimbangkan *scalability*, *latency*, *throughput*, waktu *query*, dan biaya transaksi (Subashini & Hemavathi, 2023). Kajian sistematis terbaru menunjukkan bahwa blockchain dalam *food supply chain* masih menghadapi tantangan kualitas data, interoperabilitas, skalabilitas, privasi, dan adopsi teknologi (Vasileiou *et al.*, 2026).

Dalam implementasi blockchain, *smart contract* dapat menerjemahkan aturan dan transaksi *supply chain* menjadi logika yang dapat dieksekusi secara otomatis. Penelitian terdahulu telah menerapkan blockchain dan *smart contract* untuk mendukung *food traceability* melalui pencatatan *provenance*, kepemilikan, distribusi, serta otomatisasi aturan keamanan pangan (Casino *et al.*, 2019; Markovic *et al.*, 2020; Wang *et al.*, 2022; Patel *et al.*, 2023; Kechagias *et al.*, 2023). Fernández-Iglesias *et al.* (2024) menunjukkan bahwa desain traceability berbasis *smart contract* juga perlu mempertimbangkan keseimbangan penyimpanan data *on-chain* dan *off-chain* untuk mendukung skalabilitas dan privasi. Temuan tersebut menunjukkan bahwa blockchain dan *smart contract* dapat digunakan untuk merepresentasikan hubungan antara pelaku, produk, aktivitas, kepemilikan, dan riwayat distribusi dalam rantai pasok pangan.

Kebutuhan traceability menjadi semakin penting ketika terjadi masalah keamanan pangan yang memerlukan tindakan *recall*. Identifikasi asal produk, histori pergerakan, dan pihak yang terkait dapat mendukung penentuan produk terdampak dan respons terhadap *recall*. Keramati *et al.* (2025) menunjukkan adanya hubungan antara penggunaan blockchain dan *responsiveness* terhadap *recall* pada industri makanan dan minuman. Namun, temuan tersebut belum secara langsung menjelaskan bagaimana mekanisme *recall* direpresentasikan sebagai logika transaksi *smart contract* dalam sistem *food traceability*.

Kebutuhan tersebut relevan dengan Program Makanan Bergizi Gratis (MBG) di Indonesia yang melibatkan berbagai pelaku, mulai dari pemasok bahan pangan, Satuan Pelayanan Pemenuhan Gizi (SPPG), hingga penerima manfaat. Pada awal Januari 2026, Badan Gizi Nasional (BGN) mengoperasikan 19.188 SPPG, menargetkan perluasan program hingga 82,9 juta penerima manfaat, serta melibatkan lebih dari 61 ribu pemasok (Badan Gizi Nasional, 2026a, 2026b, 2026c). Petunjuk teknis BGN juga menempatkan *monitoring*, evaluasi, dan pengawasan sebagai bagian penting dalam tata kelola program (Badan Gizi Nasional, 2025). Penelitian mengenai implementasi MBG menunjukkan adanya tantangan distribusi, fasilitas, pembagian peran, koordinasi, dan operasional, sementara kajian lain menyoroti pentingnya keamanan pangan, SOP, *monitoring*, dan koordinasi antarpemangku kepentingan (Babo *et al.*, 2026; Habibah *et al.*, 2025).

Dalam konteks tersebut, traceability MBG tidak cukup hanya mencatat pengiriman atau penerimaan bahan pangan, tetapi perlu mempertahankan hubungan *provenance* antara *batch* bahan, proses produksi, hasil produksi, distribusi, penerima, dan kejadian keamanan pangan. Hubungan ini diperlukan untuk mendukung *forward* dan *backward traceability*, termasuk ketika *batch* harus ditarik kembali. Oleh karena itu, kebutuhan traceability MBG perlu diterjemahkan ke dalam model

transaksi yang mencakup *batch lifecycle*, *ownership management*, *production*, *delivery*, *incident*, *recall*, dan *traceability query*.

Meskipun penelitian terdahulu telah menunjukkan kemampuan blockchain dan *smart contract* dalam mendukung *food traceability*, penelitian tersebut umumnya berfokus pada aspek tertentu, seperti pencatatan produk, *provenance*, perpindahan kepemilikan, distribusi, atau arsitektur penyimpanan data (Casino *et al.*, 2019; Markovic *et al.*, 2020; Wang *et al.*, 2022; Patel *et al.*, 2023; Kechagias *et al.*, 2023; Fernández-Iglesias *et al.*, 2024). Penelitian terkait *provenance* juga lebih banyak mengevaluasi performa blockchain, seperti *throughput*, *latency*, *query time*, dan biaya transaksi (Subashini & Hemavathi, 2023). Di sisi lain, penelitian mengenai MBG lebih banyak membahas kebijakan, keamanan pangan, distribusi, koordinasi, fasilitas, dan tata kelola (Babo *et al.*, 2026; Habibah *et al.*, 2025). Dengan demikian, masih terdapat kesenjangan dalam penerjemahan kebutuhan *traceability* program pangan publik multi-pelaku ke dalam model *smart contract* yang secara terintegrasi merepresentasikan *partial batch consumption*, hubungan input-output produksi, konfirmasi penerimaan, pelaporan insiden, dan mekanisme *recall*.

Berdasarkan kesenjangan tersebut, penelitian ini bertujuan untuk: (1) merancang model *smart contract* yang merepresentasikan kebutuhan *food traceability* pada Program Makanan Bergizi Gratis; (2) mengimplementasikan fungsi *batch lifecycle*, *ownership transfer*, *quantity-based traceability*, *production*, *delivery*, *incident*, dan *recall* dalam *smart contract* MBGTraceability; serta (3) memvalidasi fungsi tersebut melalui pengujian otomatis menggunakan skenario positif dan negatif. Penelitian dibatasi pada penerjemahan kebutuhan *traceability* MBG ke dalam model data dan logika transaksi *smart contract* yang dapat diimplementasikan dan diuji pada blockchain.

2. Metode Penelitian

2.1 Pendekatan Penelitian

Penelitian ini menggunakan pendekatan *Design Science Research* (DSR) untuk merancang, mengimplementasikan, dan memvalidasi *smart contract* berbasis blockchain sebagai artefak teknologi untuk mendukung *food traceability* pada Program Makanan Bergizi Gratis (MBG) di Indonesia. DSR dipilih karena penelitian berorientasi pada pengembangan dan evaluasi artefak untuk menyelesaikan permasalahan pada domain tertentu (Hevner *et al.*, 2004). Proses penelitian mengadaptasi *Design Science Research Methodology* (DSRM) dari Peffers *et al.* (2007) menjadi lima tahap, yaitu: (1) identifikasi masalah dan analisis kebutuhan, (2) perancangan *smart contract*, (3) implementasi, (4) pengujian, dan (5) validasi fungsional. Adaptasi dilakukan dengan mengintegrasikan analisis kebutuhan pada tahap identifikasi masalah serta merealisasikan demonstrasi dan evaluasi melalui implementasi dan *automated functional testing*. Dengan demikian, alur penelitian menghubungkan kebutuhan *traceability*, rancangan artefak, implementasi fungsi *smart contract*, dan evaluasi fungsional.



Gambar 1. Tahapan penelitian berbasis *Design Science Research*

2.2 Sumber dan Teknik Pengumpulan Data

Data penelitian terdiri atas data kebutuhan sistem dan data hasil pengujian artefak. Data kebutuhan diperoleh melalui studi literatur dan analisis dokumen terkait *food traceability*, blockchain, *smart contract*, *supply chain* pangan, serta tata kelola dan proses operasional MBG. Informasi yang relevan diekstraksi dan dikelompokkan menjadi kebutuhan fungsional, meliputi *role management*,

registrasi dan perpindahan *batch*, penggunaan sebagian *batch* dalam produksi, hubungan input-output produksi, distribusi, konfirmasi penerimaan, pelaporan insiden, *recall*, dan *traceability query*. Data empiris diperoleh dari eksekusi *automated test suite* terhadap *smart contract* MBGTraceability. Data pengujian meliputi keberhasilan atau penolakan transaksi, perubahan *state*, *event* transaksi, perubahan kuantitas dan kepemilikan *batch*, hubungan input-output produksi, status distribusi, pencatatan insiden, mekanisme *recall*, dan hasil fungsi *query*. Dengan demikian, data evaluasi diperoleh langsung dari eksekusi artefak, bukan dari penilaian subjektif pengguna.

2.3 Identifikasi Masalah dan Analisis Kebutuhan

Identifikasi masalah dilakukan dengan memetakan aktivitas rantai pasok MBG yang membutuhkan keterlacakan, meliputi pengadaan bahan pangan, perpindahan kepemilikan, penggunaan bahan dalam produksi, distribusi, penerimaan, pelaporan insiden, dan *recall*. Aktivitas tersebut diterjemahkan menjadi entitas, hubungan data, hak akses, dan transaksi yang perlu direpresentasikan dalam *smart contract*. Empat peran operasional yang direpresentasikan adalah Badan Gizi Nasional (BGN), *Supplier*, Satuan Pelayanan Pemenuhan Gizi (SPPG), dan *Recipient*. Selain aktor operasional tersebut, terdapat *admin/deployer* yang berfungsi sebagai administrator kontrak untuk pengelolaan *role* dan tidak diposisikan sebagai aktor operasional MBG. Pembagian peran disajikan pada Tabel 1.

Tabel 1. Model Peran pada *Smart Contract*

Stakeholder	Role	Tanggung Jawab pada <i>Smart Contract</i>
Badan Gizi Nasional	BGN	Pengawasan, pembaruan status <i>batch</i> , dan <i>food recall</i>
<i>Supplier</i>	<i>Supplier</i>	Registrasi <i>batch</i> bahan pangan dan pengiriman <i>batch</i>
SPPG	SPPG	Pengelolaan bahan, produksi, dan distribusi
<i>Recipient</i>	<i>Recipient</i>	Menerima dan mengonfirmasi pengiriman serta pelaporan insiden

Catatan: *Admin/deployer* bertindak sebagai administrator kontrak untuk pengelolaan *role* dan tidak termasuk aktor operasional MBG.

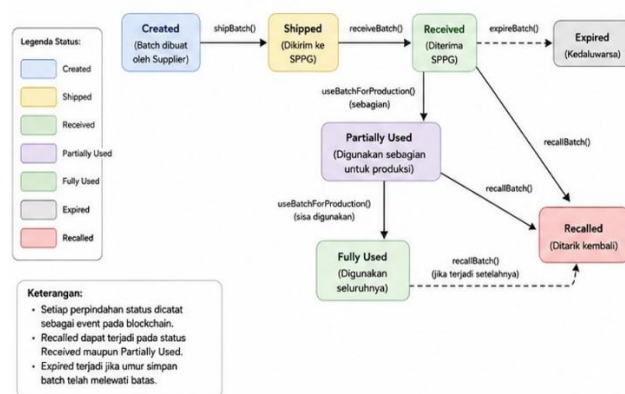
Kebutuhan fungsional kemudian dikelompokkan menjadi lima area, yaitu: (1) pengelolaan peran dan kontrol akses, (2) pengelolaan *batch* dan kepemilikan, (3) produksi dan *quantity-based traceability*, (4) distribusi dan konfirmasi penerimaan, serta (5) insiden, *recall*, dan *traceability query*. Hasil analisis kebutuhan tersebut menjadi acuan langsung dalam perancangan struktur data, fungsi transaksi, *access control*, dan skenario pengujian.

2.4 Perancangan *Smart Contract*

Kebutuhan fungsional diterjemahkan ke dalam model data, aturan akses, dan transaksi *smart contract* yang mencakup tujuh komponen utama, yaitu *role model*, *batch model*, *ownership model*, *production model*, *delivery model*, *incident model*, dan *recall mechanism*.

- 1) *Role model* mengatur hak akses berdasarkan peran.
- 2) *Batch model* merepresentasikan unit *traceability* beserta kuantitas, satuan, kepemilikan, dan status.
- 3) *Ownership model* mencatat perpindahan kepemilikan.
- 4) *Production model* menghubungkan *batch input* dengan hasil produksi.
- 5) *Delivery model* mencatat distribusi dan konfirmasi penerimaan.
- 6) *Incident model* menghubungkan insiden dengan *batch* dan distribusi terkait.
- 7) *Recall mechanism* memungkinkan aktor BGN yang berwenang menandai *batch* sebagai *recalled*.

Perancangan juga mencakup mekanisme *quantity-based traceability* untuk mempertahankan konsistensi kuantitas ketika sebagian *batch* digunakan dalam proses produksi. *Event* pada transaksi penting dirancang untuk mencatat perubahan *state* dan menyediakan *audit trail*. Rancangan tersebut menjadi dasar implementasi fungsi *smart contract* dan penyusunan *test case*.



Gambar 2. Lifecycle dan Perubahan Status Smart Contract MBGTraceability

2.5 Implementasi Smart Contract

Smart contract MBGTraceability diimplementasikan menggunakan Solidity dan dikembangkan serta diuji dengan Hardhat pada Hardhat Network menggunakan Node.js dan Viem. Skenario implementasi menggunakan aktor BGN, *Supplier*, beberapa profil SPPG, dan *Recipient* untuk merepresentasikan struktur multiaktor MBG. Fungsi yang diimplementasikan meliputi pengelolaan *role*, pembuatan dan transfer *batch*, pencatatan produksi, pembuatan dan konfirmasi *delivery*, pelaporan *incident*, *recall*, pembaruan status, serta *traceability query*. Struktur data dan fungsi transaksi diimplementasikan berdasarkan kebutuhan yang diperoleh pada tahap analisis kebutuhan. Setiap fungsi yang menjadi kebutuhan fungsional dipetakan ke satu atau lebih skenario pengujian untuk menyediakan mekanisme verifikasi pada tahap evaluasi.

2.6 Perancangan dan Pelaksanaan Pengujian

Pengujian dilakukan untuk memverifikasi kesesuaian fungsi smart contract dengan kebutuhan yang telah dirumuskan. Pengujian menggunakan *automated test suite* pada Hardhat Network dengan Node.js dan Viem. Skenario pengujian mencakup aktor BGN, *Supplier*, beberapa profil SPPG, dan *Recipient* untuk merepresentasikan struktur multiaktor MBG. Fungsi yang diuji meliputi pengelolaan *role*, pembuatan dan transfer *batch*, pencatatan produksi, pembuatan dan konfirmasi *delivery*, pelaporan *incident*, *recall*, pembaruan status, serta *traceability query*. Struktur data dan fungsi transaksi diuji berdasarkan kebutuhan yang diperoleh pada tahap analisis kebutuhan. Setiap fungsi yang menjadi kebutuhan fungsional dipetakan ke satu atau lebih skenario pengujian untuk menyediakan mekanisme verifikasi pada tahap evaluasi.

Tabel 2. Ringkasan Functional Test Cases

Kategori Pengujian	Jumlah Test Case	Contoh Skenario Uji	Hasil yang Diharapkan	Hasil Uji
Pengelolaan Peran dan Kontrol Akses	1	Admin menetapkan <i>role</i> BGN, <i>Supplier</i> , SPPG, dan <i>Recipient</i> ; <i>role</i> diverifikasi	<i>Role</i> tersimpan penetapan	sesuai 1 <i>passing</i>
Batch dan Kepemilikan	5	<i>Supplier</i> membuat <i>batch</i> ; non- <i>Supplier</i> mencoba membuat; ID duplikat ditolak; transfer oleh <i>Supplier</i> ; transfer oleh non- <i>owner</i>	Pembuatan oleh <i>Supplier</i> ; ID unik; transfer hanya oleh pemilik sah	hanya 5 <i>passing</i>
Produksi dan <i>Quantity-based</i>	2	SPPG menggunakan 400 kg dari <i>batch</i> 1.000 kg; non- <i>owner</i> mencoba produksi	Kuantitas sisa tetap konsisten; produksi hanya oleh pihak berwenang	2 <i>passing</i>

<i>Traceability</i>				
Distribusi dan Konfirmasi Penerimaan	2	SPPG membuat <i>delivery</i> ; <i>Recipient</i> mengonfirmasi; pengujian pada profil Sekolah, Posyandu, Puskesmas, dan Komunitas Ibu Hamil	<i>Delivery</i> tercatat; penerimaan dapat dikonfirmasi; kepemilikan berpindah dan status menjadi <i>Received</i>	2 <i>passing</i>
Insiden, <i>Recall</i> , dan <i>Traceability Query</i>	7	BGN melakukan <i>recall</i> ; non-BGN mencoba; SPPG/ <i>Auditor</i> /BGN memperbarui status; <i>query</i> produksi, <i>delivery</i> , kepemilikan, riwayat <i>trace</i> ; <i>Recipient</i> melaporkan insiden	<i>Recall</i> hanya oleh BGN; status oleh SPPG/BGN; <i>query</i> mengembalikandata benar; insiden tercatat	7 <i>passing</i>
Total	17	Skenario positif dan negatif	Seluruh skenario berjalan sesuai hasil yang diharapkan	17 <i>passing</i> , 0 <i>failing</i>

Pengujian menghasilkan 17 *test case* yang mencakup seluruh lima kelompok kebutuhan fungsional. Hasil aktual dibandingkan dengan *expected result* berdasarkan keberhasilan transaksi dan perubahan *state* yang relevan. Pada skenario *quantity-based traceability*, *batch* awal sebesar 1.000 kg diuji dengan penggunaan 400 kg sehingga sistem diharapkan mempertahankan sisa 600 kg. Pada skenario akses, transaksi oleh aktor yang tidak memiliki *role* atau kepemilikan yang sesuai harus ditolak.

2.7 Teknik Analisis Data Pengujian

Data pengujian dianalisis secara deskriptif berdasarkan kesesuaian hasil aktual dengan *expected result*. *Test case* dikategorikan *passing* apabila seluruh kondisi skenario terpenuhi dan *failing* apabila terdapat kondisi yang tidak terpenuhi. Analisis juga memeriksa perubahan *state*, kuantitas, kepemilikan, hubungan input-output produksi, status *delivery*, data *incident*, mekanisme *recall*, dan hasil *traceability query*. Karena data empiris berupa hasil eksekusi *test case*, penelitian tidak menggunakan statistik inferensial. Berdasarkan hasil pengujian, seluruh 17 *test case* berhasil sehingga *pass rate* sebesar 100% dan *fail rate* sebesar 0%. Hasil tersebut dianalisis berdasarkan kategori pengujian untuk menilai kesesuaian kebutuhan fungsional pada setiap kelompok.

2.8 Validitas, Reliabilitas, dan Batasan Evaluasi

Validitas fungsional dijaga melalui pemetaan kebutuhan sistem terhadap fungsi *smart contract* dan *test case* yang memiliki kondisi awal, tindakan, dan hasil yang diharapkan. Skenario positif memverifikasi operasi yang seharusnya diizinkan, sedangkan skenario negatif memverifikasi penolakan operasi yang tidak memenuhi aturan akses atau kepemilikan. Reliabilitas pengujian didukung oleh *automated test suite* dan Hardhat Network sehingga prosedur dapat dijalankan kembali dengan kondisi serta kriteria evaluasi yang sama. Dengan demikian, validasi didasarkan pada prosedur pengujian yang dapat direproduksi, bukan observasi subjektif peneliti. Evaluasi dibatasi pada kesesuaian fungsional *smart contract* terhadap kebutuhan dan skenario pengujian yang ditentukan. Keberhasilan seluruh *test case* tidak digunakan untuk menyimpulkan keamanan formal, skalabilitas, efisiensi gas, biaya transaksi, *throughput*, atau performa *smart contract* pada lingkungan produksi. Dengan demikian, hasil uji menunjukkan bahwa fungsi berjalan sesuai kriteria pada lingkungan pengujian, tetapi belum menunjukkan kesiapan implementasi pada skala nasional.

3. Hasil dan Pembahasan

3.1 Hasil

3.1.1 Hasil Implementasi *Smart Contract*

Implementasi menghasilkan artefak *smart contract* bernama MBGTraceability yang merepresentasikan fungsi utama *food traceability* pada rantai pasok MBG. Kontrak mengelola lima kelompok data utama, yaitu *role*, *batch*, *production*, *delivery*, dan *incident*, sedangkan *recall* direpresentasikan melalui perubahan status pada *batch*. Data yang diperlukan untuk validasi transaksi, seperti *owner*, *quantity*, *status*, dan *timestamp*, disimpan pada *state blockchain*, sedangkan *metadata* berukuran lebih besar direpresentasikan melalui referensi *dataHash*, yang pada skenario pengujian menggunakan format URI seperti *ipfs://...* dan dikaitkan ke *smart contract*. Struktur tersebut menempatkan *smart contract* sebagai lapisan transaksi dan *provenance*, sementara komponen antarmuka, autentikasi pengguna, dan penyimpanan *metadata* berada di luar kontrak. Hasil implementasi juga menunjukkan bahwa kebutuhan *traceability* dapat diterjemahkan menjadi fungsi transaksi yang terpisah berdasarkan *lifecycle* aset. *Batch* dapat dibuat oleh *Supplier*, ditransfer kepada aktor berikutnya, digunakan sebagai *input* produksi oleh SPPG, dikaitkan dengan *delivery*, dilaporkan apabila terjadi *incident*, dan ditandai sebagai *recalled* oleh BGN. Dengan demikian, implementasi tidak hanya menyimpan informasi produk, tetapi membentuk hubungan antarkegiatan yang memungkinkan rekonstruksi *provenance* berdasarkan perubahan *state* dan transaksi *blockchain*.

3.1.2 Hasil Validasi Fungsional

Validasi dilakukan melalui *automated test suite* yang mencakup skenario berhasil dan skenario negatif. Pengujian dikelompokkan ke dalam lima kategori, yaitu: (1) pengelolaan peran dan kontrol akses; (2) *batch* dan kepemilikan; (3) produksi dan *quantity-based traceability*; (4) distribusi dan konfirmasi penerimaan; serta (5) insiden, *recall*, dan *traceability query*. Secara keseluruhan terdapat 17 *test case*, dan seluruh pengujian berhasil (17 *passing*; 0 *failing*).

Tabel 3. Hasil Pengujian Fungsional MBGTraceability

No.	Kategori Pengujian	Test Case	Hasil
1	Pengelolaan Peran & Kontrol Akses	1	1 <i>passing</i>
2	<i>Batch</i> & Kepemilikan	5	5 <i>passing</i>
3	Produksi & <i>Quantity-based Traceability</i>	2	2 <i>passing</i>
4	Distribusi & Konfirmasi Penerimaan	2	2 <i>passing</i>
5	Insiden, <i>Recall</i> , & <i>Traceability Query</i>	7	7 <i>passing</i>
	Total	17	17 <i>passing</i> , 0 <i>failing</i>

Hasil pengujian menunjukkan bahwa fungsi yang diuji memberikan perilaku sesuai dengan skenario yang ditetapkan, termasuk validasi *role* dan *ownership*, perubahan kuantitas *batch*, hubungan *input-output* produksi, distribusi dan penerimaan, pelaporan insiden, *recall*, serta fungsi *query*. Keberhasilan 17/17 menunjukkan kesesuaian fungsional pada lingkungan pengujian, tetapi tidak digunakan untuk menyimpulkan keamanan formal, skalabilitas, efisiensi gas, atau kesiapan pada lingkungan produksi.

Tabel 4. Pemetaan Fungsi *Smart Contract* dan *Automated Test Suite*

Fungsi <i>Smart Contract</i>	Fungsi Utama	Diuji dalam <i>Test Suite</i>
<i>assignRole()</i>	Menetapkan <i>role</i> aktor	✓
<i>createBatch()</i>	Membuat <i>batch</i> bahan pangan	✓
<i>transferBatch()</i>	Transfer kepemilikan <i>batch</i>	✓
<i>createProduction()</i>	Mencatat proses produksi	✓
<i>updateBatchStatus()</i>	Memperbarui status <i>batch</i>	✓

createDelivery()	Mencatat pengiriman	✓*
confirmDelivery()	Mengonfirmasi penerimaan	✓
recallBatch()	Melakukan <i>food recall</i>	✓
reportIncident()	Mencatat insiden	✓
getBatch()	Mengambil data <i>batch</i>	✓
getProduction()	Mengambil data produksi	✓
getProductionInputs()	Mengambil <i>batch input</i> produksi	✓
getDelivery()	Mengambil data <i>delivery</i>	✓
getDeliveryTargetBatch()	Mengambil <i>batch</i> tujuan <i>delivery</i>	✓
isBatchDelivery()	Memeriksa tipe <i>delivery</i>	✓
isBatchOwner()	Memeriksa kepemilikan <i>batch</i>	✓
traceBatchHistory()	Menelusuri riwayat <i>batch</i>	✓
getIncident()	Mengambil data insiden	✓

createDelivery() diuji pada skenario *Batch reference*; RefType.Production belum diuji secara eksplisit.

Pemetaan pada Tabel 4 menunjukkan fungsi *smart contract* yang telah digunakan dalam *automated test suite*. Tanda (✓) menunjukkan bahwa fungsi telah diuji setidaknya pada satu skenario, tetapi tidak menunjukkan *exhaustive coverage* terhadap seluruh *execution path* atau kondisi kesalahan.

3.1.3 Hasil *Quantity-Based Traceability*

Salah satu temuan penting dari implementasi adalah kemampuan *smart contract* mempertahankan kuantitas awal dan sisa *batch* ketika sebagian bahan digunakan dalam produksi. Mekanisme tersebut menggunakan *batchInitialQuantity* untuk menyimpan kuantitas awal dan *batchRemaining* untuk merepresentasikan kuantitas yang masih tersedia. Pada saat produksi, nilai *batchRemaining* dikurangi berdasarkan parameter *usedQuantities*. Pada skenario pengujian, *batch* sebesar 1.000 kg digunakan dan 400 kg dikonsumsi untuk proses produksi. Hasil yang diharapkan sekaligus diverifikasi adalah sisa kuantitas sebesar 600 kg. Implementasi menetapkan bahwa ketika seluruh sisa kuantitas digunakan, status *batch* dapat berubah menjadi *FullyUsed*, sedangkan penggunaan sebagian mempertahankan status *PartiallyUsed*. Pengujian yang dilakukan secara eksplisit memverifikasi skenario penggunaan 400 kg dari *batch* 1.000 kg sehingga tersisa 600 kg. Temuan tersebut penting karena *traceability* berbasis identitas *batch* saja tidak cukup untuk menggambarkan penggunaan bahan yang bersifat parsial. Dalam rantai pasok MBG, satu *batch* bahan dapat digunakan untuk beberapa aktivitas produksi. Dengan mempertahankan kuantitas awal dan sisa secara terpisah, sistem dapat menjaga konsistensi *state* bahan sekaligus mempertahankan hubungan antara bahan yang tersedia dan bahan yang telah digunakan. Dengan demikian, kontribusi *quantity-based traceability* terletak pada kemampuan model untuk merepresentasikan status konsumsi bahan, bukan hanya keberadaan *batch*.

3.1.4 Hasil *Provenance dan Lifecycle Traceability*

Implementasi membentuk hubungan *provenance* melalui rangkaian *Batch* → *Production* → *Output Batch* → *Delivery* → *Recipient*. Pada proses produksi, satu atau lebih *batch input* dapat dikaitkan dengan satu *production* dan *output batch*. Pengurangan kuantitas bahan dilakukan pada *batch input*, sedangkan hasil produksi direpresentasikan sebagai *batch* baru yang tetap memiliki hubungan dengan proses yang menghasilkannya. Hasil ini memungkinkan penelusuran dilakukan dari bahan baku menuju produk hasil produksi. Sebaliknya, informasi mengenai *inputBatches* pada *Production* menyediakan dasar untuk melakukan penelusuran balik dari produk menuju bahan yang digunakan. Dengan demikian, model mendukung dua arah *traceability*, yaitu *forward traceability* dari bahan menuju distribusi dan *backward traceability* dari hasil produksi menuju bahan *input*. Pada tahap distribusi, *Delivery* dapat mengacu pada *Batch* maupun *Production*. *createDelivery()* mencatat pengirim dan

penerima, sedangkan `confirmDelivery()` hanya dapat dilakukan oleh alamat yang tercatat sebagai penerima. Setelah konfirmasi, status *delivery* berubah menjadi *Delivered* dan waktu konfirmasi dicatat. Model tersebut membuat *delivery* berfungsi bukan hanya sebagai catatan pengiriman, tetapi sebagai bagian dari *ownership provenance*. Temuan ini memperlihatkan bahwa desain *smart contract* berhasil mengubah alur logistik menjadi rangkaian transaksi yang saling berhubungan. Setiap aktivitas menghasilkan perubahan *state* yang dapat digunakan untuk merekonstruksi riwayat aset. Hal tersebut memperkuat fungsi blockchain sebagai lapisan *provenance* dalam sistem *traceability*.

3.1.5 Access Control, Incident, dan Recall

Hasil implementasi menunjukkan bahwa kontrol akses diterapkan pada tingkat fungsi menggunakan `onlyAdmin`, `onlyRole`, dan `onlySPPGorBGN`. Selain *role*, fungsi yang berkaitan dengan aset juga memeriksa kepemilikan *batch*. Dengan kombinasi tersebut, kewenangan tidak hanya ditentukan berdasarkan identitas aktor, tetapi juga berdasarkan hubungan aktor dengan aset yang ditransaksikan. Mekanisme tersebut diuji melalui skenario positif dan negatif. Dalam skenario akses yang diuji, aktor yang tidak memiliki *role* atau kepemilikan yang sesuai ditolak oleh *smart contract*. Sebagai contoh, pembuatan *batch* dibatasi kepada *Supplier*, transfer *batch* memerlukan kepemilikan oleh pemanggil, sedangkan *recall* hanya dapat dijalankan oleh BGN. Untuk *incident*, struktur data menghubungkan *incidentId* dengan *batchId* dan *deliveryId*. Dengan demikian, kejadian dapat dikaitkan langsung dengan aset dan aktivitas distribusi yang relevan. *Smart contract* mendefinisikan empat tipe insiden, yaitu *Contamination*, *ExpiredFood*, *DamagedFood*, dan *DistributionIssue*, sedangkan pengujian yang tersedia menggunakan skenario *ExpiredFood*. Ketika BGN menjalankan `recallBatch()`, status *batch* berubah menjadi *Recalled*. Pada implementasi, status *Recalled* digunakan sebagai kondisi pembatas pada operasi produksi dan *delivery* tertentu. Model ini menunjukkan bahwa *recall* tidak direpresentasikan sebagai catatan terpisah, tetapi sebagai perubahan *state* pada unit *traceability* yang menjadi pusat hubungan *provenance*. Secara konseptual, hubungan tersebut menghasilkan alur: *Batch* → *Delivery* → *Incident* → *Recall* yang memungkinkan kejadian keamanan pangan dikaitkan dengan aset dan distribusi yang terdampak. Namun, penelitian ini hanya memvalidasi fungsi *incident* dan *recall* berdasarkan skenario pengujian yang tersedia dan belum mengevaluasi efektivitas operasional *recall* pada kondisi nyata.

3.1.6 Auditability dan Traceability Query

Selain transaksi yang mengubah *state*, MBGTraceability menyediakan fungsi *read-only* seperti `getBatch()`, `getProduction()`, `getProductionInputs()`, `getDelivery()`, `getIncident()`, dan `traceBatchHistory()`. Fungsi tersebut memungkinkan data *provenance* yang tersimpan pada blockchain dibaca kembali tanpa mengubah *state* kontrak. Implementasi juga mendefinisikan sembilan *event* utama, yaitu *RoleAssigned*, *BatchCreated*, *BatchTransferred*, *ProductionCreated*, *DeliveryCreated*, *DeliveryConfirmed*, *BatchStatusUpdated*, *IncidentReported*, dan *BatchRecalled*. *Event* tersebut dirancang sebagai *audit trail* untuk merekam aktivitas penting sepanjang *lifecycle traceability*. Temuan ini menunjukkan adanya pemisahan fungsi antara *state-based provenance* dan *event-based audit trail*. *State* digunakan untuk mengetahui kondisi terkini suatu entitas, sedangkan *event* menyediakan jejak aktivitas yang dapat dibaca oleh sistem eksternal. Beberapa *event* utama juga diverifikasi secara eksplisit dalam *automated test*, termasuk *BatchCreated*, *BatchTransferred*, *ProductionCreated*, *DeliveryCreated*, *DeliveryConfirmed*, *IncidentReported*, dan *BatchRecalled*. Namun, belum seluruh *event* yang didefinisikan pada *smart contract* diuji secara individual. Oleh karena itu, hasil penelitian mendukung klaim bahwa model *event* telah dirancang dan diimplementasikan serta sebagian *event* telah divalidasi, tetapi belum cukup untuk menyimpulkan *exhaustive event coverage*.

3.2 Pembahasan

Hasil penelitian menunjukkan bahwa kebutuhan *food traceability* pada program MBG dapat diterjemahkan menjadi logika transaksi *smart contract* yang mengintegrasikan *batch lifecycle*, *ownership*, *quantity-based traceability*, *production*, *delivery*, *incident*, dan *recall*. Temuan utama penelitian bukan hanya keberhasilan pencatatan data pada blockchain, tetapi juga kemampuan model untuk

mempertahankan hubungan antarkegiatan sehingga riwayat suatu *batch* dapat direkonstruksi melalui transaksi yang saling terhubung. Dengan demikian, *smart contract* berfungsi sebagai lapisan logika transaksi yang mengikat identitas aset, aktor, kepemilikan, aktivitas produksi, distribusi, dan kejadian keamanan pangan.

Dari sisi *food traceability*, hasil ini sejalan dengan penelitian Kechagias et al. (2023) yang menggunakan *smart contract* dalam aplikasi terdesentralisasi berbasis Ethereum untuk mendukung *traceability* dan pencatatan kepemilikan pada rantai pasok pangan. Penelitian Markovic et al. (2020) juga menunjukkan pentingnya integrasi *provenance*, blockchain, dan aktivitas *last-mile delivery*. Namun, MBGTraceability memperluas representasi tersebut dengan menghubungkan proses produksi dengan *batch input* dan *batch output* serta mempertahankan kuantitas bahan yang tersisa setelah sebagian bahan digunakan. Dengan demikian, fokus penelitian ini tidak hanya pada perpindahan aset, tetapi juga pada hubungan antara konsumsi bahan, transformasi produksi, dan distribusi. Literatur menunjukkan bahwa *smart contract* dapat digunakan sebagai *business logic layer* untuk *food traceability*, sedangkan penelitian ini menerapkannya secara khusus pada kebutuhan program pangan publik MBG.

Temuan mengenai *quantity-based traceability* menjadi salah satu pembeda penting. Pada skenario pengujian, penggunaan 400 kg dari *batch* sebesar 1.000 kg menghasilkan sisa 600 kg. Mekanisme tersebut memungkinkan satu *batch* tetap dipertahankan sebagai unit *traceability* meskipun hanya sebagian kuantitasnya telah digunakan. Pendekatan ini melengkapi model *provenance* yang umumnya berfokus pada identitas dan hubungan antarkomponen. Penelitian Subashini dan Hemavathi (2023), misalnya, lebih menekankan aspek *scalability*, *latency*, *throughput*, *query time*, dan biaya transaksi dalam *agri-food provenance*. Penelitian tersebut menunjukkan pentingnya evaluasi performa, sedangkan penelitian ini berfokus pada representasi logika konsumsi parsial dan konsistensi *state*. Dengan demikian, kedua pendekatan berada pada dimensi evaluasi yang berbeda dan tidak dapat dibandingkan berdasarkan performa karena penelitian ini memang belum melakukan *benchmarking* tersebut.

Dari sisi tata kelola transaksi, penerapan *role-based access control* dan validasi kepemilikan menunjukkan bahwa *traceability* tidak hanya berkaitan dengan ketersediaan riwayat data, tetapi juga dengan siapa yang berwenang mengubah *state*. Temuan ini sejalan dengan Lei et al. (2022) yang menekankan bahwa sistem *blockchain-based food traceability* perlu memperhatikan *access control*, privasi, dan keamanan pertukaran data di antara banyak pemangku kepentingan. Dalam MBGTraceability, prinsip tersebut diterjemahkan ke dalam pembatasan fungsi berdasarkan *role* serta pemeriksaan kepemilikan *batch*. Namun, implementasi ini baru memvalidasi mekanisme otorisasi pada tingkat fungsi dan belum mengevaluasi perlindungan privasi atau mekanisme *privacy-preserving* secara empiris.

Pada aspek dukungan terhadap keamanan pangan, integrasi *incident* dan *recall* memberikan hubungan antara kejadian, *batch*, dan aktivitas distribusi. Ketika *recallBatch()* dijalankan oleh BGN, status *batch* berubah menjadi *Recalled*, sehingga status tersebut menjadi bagian dari *state* yang dapat digunakan oleh fungsi transaksi lain. Temuan ini relevan dengan Keramati et al. (2025), yang menunjukkan hubungan antara penggunaan blockchain dan *responsiveness* terhadap *food recall*. Namun, terdapat perbedaan fokus. Keramati et al. menganalisis *responsiveness* terhadap *recall*, sedangkan penelitian ini merancang dan menguji mekanisme transaksi *recall* pada tingkat *smart contract*. Oleh karena itu, hasil 17 *test case* yang berhasil tidak dapat ditafsirkan sebagai bukti bahwa sistem dapat mempercepat *recall* dalam kondisi operasional nyata. Pengujian efek pembatasan transaksi setelah status *Recalled* dan efektivitas operasional *recall* masih menjadi bagian dari evaluasi lanjutan.

Hasil penelitian juga menunjukkan adanya pemisahan antara *state-based provenance* dan *event-based audit trail*. *State* digunakan untuk merepresentasikan kondisi terkini suatu *batch*, sedangkan *event* merekam aktivitas penting yang dapat digunakan untuk membangun jejak audit. Pendekatan ini sejalan dengan kebutuhan *auditability* dalam sistem blockchain *traceability*. Fernández-Iglesias et al. (2024) menunjukkan bahwa desain sistem *traceability* berbasis *smart contract* perlu mempertimbangkan keseimbangan antara data yang disimpan *on-chain* dan *off-chain*. Dalam MBGTraceability, *metadata* yang lebih besar direpresentasikan melalui *dataHash*, sedangkan informasi yang diperlukan untuk validasi transaksi disimpan pada *state blockchain*. Namun, berbeda

dengan penelitian tersebut, penelitian ini belum melakukan evaluasi kuantitatif terhadap penghematan penyimpanan, skalabilitas, privasi, atau biaya transaksi. 17 dari 17 *test case* berhasil menunjukkan bahwa fungsi yang diuji memenuhi kebutuhan fungsional yang direpresentasikan dalam skenario pengujian. Hasil tersebut memberikan bukti awal bahwa kebutuhan traceability spesifik MBG dapat diterjemahkan menjadi fungsi transaksi yang dapat dieksekusi dan diuji pada blockchain. Namun, keberhasilan *functional testing* tidak dapat disamakan dengan keberhasilan implementasi pada skala nasional. Kajian *blockchain food supply chain* menunjukkan bahwa skalabilitas, interoperabilitas, privasi, kualitas data, dan adopsi masih menjadi tantangan penting. Penelitian Subashini dan Hemavathi (2023) secara khusus menunjukkan pentingnya pengujian *throughput*, *latency*, dan biaya transaksi, sedangkan Fernández-Iglesias et al. (2024) menekankan persoalan penyimpanan *on-chain* dan *off-chain*. Aspek-aspek tersebut belum diuji dalam penelitian ini.

Dalam penelitian ini, dampak terhadap traceability belum diukur menggunakan indikator operasional pada lingkungan MBG nyata. Bukti kuantitatif yang tersedia terbatas pada hasil pengujian fungsional, yaitu keberhasilan 17 dari 17 *test case* dan konsistensi perubahan *state* pada skenario konsumsi parsial. Oleh karena itu, pengukuran dampak traceability dalam praktik nyata masih memerlukan validasi lapangan dengan data operasional dan indikator yang relevan. Dengan demikian, implikasi utama penelitian terletak pada penerjemahan kebutuhan traceability program pangan publik yang multiaktor ke dalam model transaksi *smart contract* yang terintegrasi. Model tersebut menyediakan fondasi untuk penelusuran *forward* dan *backward*, pengelolaan konsumsi sebagian *batch*, hubungan *input-output* produksi, pencatatan distribusi dan penerimaan, serta keterkaitan insiden dan *recall*. Secara praktis, model ini dapat menjadi dasar bagi pengembangan sistem traceability MBG yang membutuhkan pencatatan *provenance* lintas aktor, terutama untuk menelusuri asal bahan, penggunaan bahan dalam produksi, distribusi, penerimaan, serta keterkaitan insiden dan *recall*.

4. Kesimpulan dan Saran

Penelitian ini menghasilkan MBGTraceability sebagai model *smart contract* untuk mendukung *food traceability* pada Program Makanan Bergizi Gratis. Model tersebut mengintegrasikan *batch lifecycle*, *ownership transfer*, *quantity-based traceability*, *production*, *delivery*, *incident*, dan *recall* dengan kontrol akses berbasis peran. Hasil validasi menunjukkan bahwa 17 dari 17 *test case* berhasil, sehingga fungsi yang diuji berjalan sesuai dengan perilaku yang diharapkan pada lingkungan *proof of concept*. Temuan utama menunjukkan bahwa integrasi hubungan *batch*, produksi, distribusi, dan insiden memungkinkan rekonstruksi *provenance* secara terstruktur, termasuk pada penggunaan sebagian *batch* dalam proses produksi. Kontribusi penelitian terletak pada penerjemahan kebutuhan traceability yang spesifik pada program pangan publik multiaktor ke dalam logika transaksi *smart contract* yang dapat diimplementasikan dan divalidasi secara fungsional. Secara praktis, model ini dapat menjadi dasar pengembangan mekanisme traceability MBG untuk menghubungkan informasi bahan, produksi, distribusi, penerimaan, insiden, dan *recall*. Namun, hasil penelitian belum menunjukkan kesiapan implementasi pada skala operasional nasional karena skalabilitas, biaya transaksi, performa jaringan, keamanan, dan privasi belum dievaluasi secara empiris. Penelitian selanjutnya perlu mengembangkan evaluasi pada skala yang lebih besar melalui *performance benchmarking*, analisis biaya transaksi, pengujian keamanan dan privasi, serta pengujian integrasi dengan sistem informasi MBG yang berjalan. Validasi dengan skenario dan data operasional dari aktor MBG juga diperlukan untuk menilai kesesuaian model dengan kondisi implementasi nyata.

5. Ucapan Terima Kasih

Penulis menyampaikan terima kasih kepada seluruh pihak yang telah memberikan dukungan dalam pelaksanaan penelitian dan pengembangan MBGTraceability, khususnya dalam proses

analisis kebutuhan, perancangan, implementasi, dan pengujian *smart contract*. Dukungan dan kontribusi yang diberikan telah membantu penyelesaian penelitian ini. Ucapan terima kasih disampaikan kepada Kementerian Pendidikan Tinggi, Sains, dan Teknologi (Kemdiktisaintek) atas dukungan finansial melalui pendanaan Program Penelitian BIMA (*Basis Informasi Penelitian dan Pengabdian kepada Masyarakat*) skema Penelitian Dasar (*Penelitian Dosen Pemula*) Tahun Anggaran 2026 dengan nomor kontrak induk: 283/C3/DT.05.00/PL-BARU/2026. Dalam proses penyusunan naskah ini, penulis menggunakan Google Gemini/NotebookLM sebagai alat bantu untuk meninjau dan mensintesis literatur referensi, serta ChatGPT untuk menyempurnakan tata bahasa dan keterbacaan naskah secara akademis. Seluruh hasil telaah dan teks telah diperiksa, diverifikasi, dan disunting kembali oleh penulis. Penulis bertanggung jawab penuh atas keabsahan data, keakuratan sitasi, dan keseluruhan isi artikel ini.

6. Daftar Pustaka

- Babo, D. H. P., Sidi, K. R., & Imtiyaz, H. (2026). Implementation of the Free Nutritious Meals (MBG) Program: A health policy perspective from Jakarta. *Jurnal Cakrawala Promkes*, 8(1), 50–59. <https://doi.org/10.12928/jcp.v8i1.15522>
- Badan Gizi Nasional. (2025). *Keputusan Deputy Bidang Pemantauan dan Pengawasan Nomor 3 Tahun 2025 tentang Petunjuk Teknis Pemantauan dan Pengawasan Program Makan Bergizi Gratis (MBG)*.
- Badan Gizi Nasional. (2026a). *Awal Januari 2026 BGN operasikan 19.188 SPPG*. <https://www.bgn.go.id/news/siaran-pers/awal-januari-2026-bgn-operasikan-19188-sppg>
- Badan Gizi Nasional. (2026b). *Awal 2026 program MBG jangkauan hampir 60 juta penerima manfaat*. <https://www.bgn.go.id/news/siaran-pers/awal-2026-program-mbg-jangkauan-hampir-60-juta-penerima-manfaat>
- Badan Gizi Nasional. (2026c). *BGN libatkan lebih dari 61 ribu supplier untuk program MBG*. <https://www.bgn.go.id/news/siaran-pers/bgn-libatkan-lebih-dari-61-ribu-supplier-untuk-program-mbg>
- Behnke, K., & Janssen, M. F. W. H. A. (2020). Boundary conditions for traceability in food supply chains using blockchain technology. *International Journal of Information Management*, 52, Article 101969. <https://doi.org/10.1016/j.ijinfomgt.2019.05.025>
- Casino, F., Kanakaris, V., Dasaklis, T. K., Moschuris, S., & Rachaniotis, N. P. (2019). Modeling food supply chain traceability based on blockchain technology. *IFAC-PapersOnLine*, 52(13), 2728–2733. <https://doi.org/10.1016/j.ifacol.2019.11.620>
- Darodjat, R., & Utarie, F. (2022). Optimalisasi penjualan bahan pangan oleh petani dan UMKM dengan memanfaatkan teknologi digital. *Dharmakarya: Jurnal Aplikasi Ipteks untuk Masyarakat*, 11(1), 1–4. <https://doi.org/10.24198/dharmakarya.v11i1.36093>
- Demestichas, K., Peppes, N., Alexakis, T., & Adamopoulou, E. (2020). Blockchain in agriculture traceability systems: A review. *Applied Sciences*, 10(12), Article 4113. <https://doi.org/10.3390/app10124113>
- Fernández-Iglesias, M. J., Delgado von Eitzen, C., & Anido-Rifón, L. (2024). Efficient traceability systems with smart contracts: Balancing on-chain and off-chain data storage for enhanced

scalability and privacy. *Applied Sciences*, 14(23), Article 11078. <https://doi.org/10.3390/app142311078>

Habibah, N., Aradytia, M. T., Izzah, K. N., Hawarie, K. A., & Maheswara, M. (2025). Reviewing the effectiveness of the Free Nutritious Meals (MBG) policy in promoting nutritional equity among school children. *Journal of Social, Policy, and Development Studies*, 2(2), 18–26. <https://doi.org/10.65119/jspds.v2i2.32>

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, 28(1), 75–105. <https://doi.org/10.2307/25148625>

Kechagias, E. P., Gayialis, S. P., Papadopoulos, G. A., & Papoutsis, G. (2023). An Ethereum-based distributed application for enhancing food supply chain traceability. *Foods*, 12(6), Article 1220. <https://doi.org/10.3390/foods12061220>

Keramati, A., Siau, B., Bellitto, T., Heydari, J., & Panchal, T. (2025). Blockchain's effects on responsiveness to recalls in the food and beverage industry. *Journal of Economy and Technology*, 3, 283–298. <https://doi.org/10.1016/j.ject.2025.05.001>

Lei, M., Xu, L., Liu, T., Liu, S., & Sun, C. (2022). Integration of privacy protection and blockchain-based food safety traceability: Potential and challenges. *Foods*, 11(15), Article 2262. <https://doi.org/10.3390/foods11152262>

Markovic, M., Edwards, P., Kollingbaum, M., & Rowe, A. (2020). Integrating Internet of Things, provenance, and blockchain to enhance trust in last mile food deliveries. *Frontiers in Sustainable Food Systems*, 4, Article 563424. <https://doi.org/10.3389/fsufs.2020.563424>

Patel, A. S., Brahmabhatt, M. N., Bariya, A. R., Nayak, J. B., & Singh, V. K. (2023). Blockchain technology in food safety and traceability concern to livestock products. *Heliyon*, 9, Article e16526. <https://doi.org/10.1016/j.heliyon.2023.e16526>

Peffer, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3), 45–77. <https://doi.org/10.2753/MIS0742-1222240302>

Perdana, T., Renaldi, E., Sadeli, A. H., Rahmawati, I. F., & Perdana, H. H. (2022). Pemberdayaan konsumen pangan untuk meningkatkan konsumsi pangan berkualitas dengan penciptaan nilai bersama melalui social commerce. *Dharmakarya: Jurnal Aplikasi Ipteks untuk Masyarakat*, 11(2), 165–171. <https://doi.org/10.24198/dharmakarya.v11i2.32289>

Subashini, B., & Hemavathi, D. (2023). Scalable blockchain technology for tracking the provenance of the agri-food. *Computers, Materials & Continua*, 75(2), 3339–3358. <https://doi.org/10.32604/cmc.2023.035074>

Vasileiou, M., Kyrgiakos, L. S., Kleisiari, C., Lappas, P. Z., Tsinopoulos, C., Klefodimos, G., Ntemou, A., Kateris, D., Moulogianni, C., & Vlontzos, G. (2026). Digital transformation of food supply chain management using blockchain: A systematic literature review towards food safety and traceability. *Business & Information Systems Engineering*, 68(4), 919–946. <https://doi.org/10.1007/s12599-025-00948-0>

Wang, L., He, Y., & Wu, Z. (2022). Design of a blockchain-enabled traceability system framework for food supply chains. *Foods*, 11(5), Article 744. <https://doi.org/10.3390/foods11050744>