

Model Reservasi Hotel Desentralisasi menggunakan Smart Contract ERC-1155 dan Escrow di Optimistic Rollups

Yobel Nathaniel Filipus¹, Mychael Maoeretz Engel^{1,*}, Elizabeth Nathania Witanto¹

¹ Program Studi Informatika, Universitas Ciputra Surabaya, Indonesia

* Correspondence: mychael.engel@ciputra.ac.id

Copyright: © 2025 by the authors

Received: 13 Oktober 2025 | Revised: 29 Oktober 2025 | Accepted: 30 November 2025 | Published: 9 Desember 2025

Abstrak

Sistem reservasi hotel digital masih rentan terhadap penipuan karena bergantung pada pihak ketiga dan otoritas terpusat. Penelitian ini bertujuan menilai efisiensi biaya transaksi melalui perilaku eksekusi fungsi *smart contract* yang mengintegrasikan ERC-1155, *escrow*, dan *Optimistic Rollups* dalam konteks reservasi hotel. Penelitian ini merupakan penelitian deskriptif-kuantitatif dengan menggunakan metode eksperimental. Empat fungsi *smart contract* diuji lima kali pada *Sepolia Testnet* dan *Optimism Sepolia Testnet*. Data konsumsi gas dan biaya transaksi dikumpulkan melalui *Blockscout*, dianalisis menggunakan nilai rata-rata, dan diuji signifikansinya menggunakan *t-test* dengan $\alpha = 0,05$. Hasil menunjukkan bahwa *Optimistic Rollups* memberikan efisiensi biaya sebesar 99,97–99,98%, dengan seluruh fungsi menghasilkan $p\text{-value} < 0,05$, sehingga *Layer 2* terbukti lebih efisien dan memiliki pola biaya lebih stabil. Integrasi ERC-1155 dan *escrow* juga tidak meningkatkan variabilitas biaya secara signifikan. Temuan penelitian ini memperkaya kajian mengenai *hybrid token* dan *trustless settlement* dalam *tokenomics* serta menunjukkan potensi implementasi komersial bagi OTA dan hotel karena biaya operasional yang sangat rendah. Namun, *challenge period* tujuh hari tetap menjadi batasan utama. Penelitian lanjutan diperlukan pada *mainnet*, skenario beban tinggi, serta pengembangan fitur pembatalan dan modifikasi reservasi.

Kata kunci: *erc-1155; escrow; optimistic rollups; reservasi hotel; smart contract*

Abstract

Digital hotel reservation systems remain vulnerable to fraud due to reliance on third parties and centralized authorities. This study aims to assess transaction cost efficiency through the execution behavior of smart contract functions that integrate ERC-1155, *escrow*, and *Optimistic Rollups* in the context of hotel reservations. This research employs a descriptive-quantitative approach using an experimental method. Four smart contract functions were tested five times on the *Sepolia Testnet* and the *Optimism Sepolia Testnet*. Gas usage and transaction fee data were collected through *Blockscout*, analyzed using mean values, and tested for significance using a *t-test* with $\alpha = 0.05$. The results show that *Optimistic Rollups* provide a cost efficiency of 99.97–99.98%, with all functions producing $p\text{-values} < 0.05$, indicating that *Layer 2* is more efficient and exhibits more stable cost patterns. The integration of ERC-1155 and *escrow* also does not significantly increase cost variability. These findings enrich the discussion on *hybrid tokens* and *trustless settlement* within *tokenomics* and demonstrate strong commercial potential for OTAs and hotels due to the extremely low operational costs. However, the seven-day *challenge period* remains a major limitation. Further research is required on *mainnet* environments, high-load scenarios, and the development of reservation cancellation and modification features.

Keywords: *erc-1155; escrow; hotel reservation; optimistic rollups; smart contract*



PENDAHULUAN

Industri perhotelan berkontribusi signifikan terhadap ekonomi melalui pendapatan dan lapangan kerja (Harji & Suharto, 2024; Maulina, 2023). Banyak hotel modern yang terus berupaya untuk meningkatkan pengalaman tamu dan efisiensi operasional melalui adopsi solusi digital yang inovatif, khususnya dalam pengelolaan reservasi berbasis digital (Ankor et al., 2025; Ramdhan & Suharto, 2024; Soklis et al., 2025). Hal ini dikarenakan solusi digital terbukti lebih cepat dan efisien dibandingkan sistem reservasi yang dilakukan secara manual (Putu & Pratiwi, 2024; Prastika & Suharto, 2024; Salameh et al., 2022).

Namun, efisiensi yang ditawarkan teknologi ini tidak menghilangkan kerentanan mendasar dalam operasional hotel. Literatur sebelumnya menemukan bahwa *internal fraud* merupakan bentuk *fraud* paling dominan di industri perhotelan, terutama karena proses pencatatan dan verifikasi dikendalikan oleh pihak yang memiliki otoritas tunggal (Kassem, 2024). Kerentanan tersebut semakin diperbesar oleh ketergantungan hotel pada *third-party dependency* yang merupakan akar dari berbagai bentuk *fraud* karena hotel tidak memiliki kontrol penuh terhadap mekanisme keamanan, *audit trail*, maupun integritas data dalam sistem yang dikelola vendor. Kondisi ini membuka ruang bagi berbagai bentuk manipulasi, termasuk *booking manipulation*, *payment fraud*, dan *chargeback fraud* ketika proses verifikasi pembayaran dan perubahan data melewati sistem yang tidak sepenuhnya dapat diaudit secara independen. Studi kasus Adi Rama Beach Hotel menunjukkan konsekuensi langsung dari kondisi tersebut, yaitu kelemahan kontrol internal dan kurangnya visibilitas pada sistem reservasi memungkinkan staf melakukan perubahan pemesanan tanpa terpantau, sehingga merugikan pelanggan dan menurunkan akurasi data (Gusti et al., 2025).

Arsitektur terpusat seperti ini menciptakan *single point of control*, di mana satu entitas memiliki kekuasaan penuh untuk memodifikasi transaksi. Kurangnya transparansi membuat manipulasi data sulit dideteksi *real-time* (Oyedotun et al., 2025). Kondisi ini menunjukkan bahwa akar masalah *fraud* berasal dari desain sistem yang sentralistik (Singh, 2025). Secara global, ACFE melaporkan kerugian median akibat penipuan di sektor perhotelan mencapai USD 579.000 (Kassem, 2024). Kondisi ini menunjukkan bahwa sistem reservasi digital masih rentan manipulasi, sehingga dibutuhkan mekanisme yang dapat menegakkan kepercayaan tanpa ketergantungan pada otoritas sentral atau perantara.

Salah satu teknologi yang menawarkan solusi terhadap permasalahan tersebut adalah *blockchain* beserta *smart contract* yang menyediakan mekanisme yang transparan, aman, dan terdesentralisasi. (Aprialim et al., 2021; Taherdoost, 2023; Zacky et al., 2023). Dalam konteks sistem reservasi hotel, teknologi tersebut berpotensi menggantikan proses verifikasi dan pembayaran berbasis pihak ketiga melalui mekanisme *escrow on-chain* yang otomatis dan tidak terpusat. Hal ini penting karena *escrow* konvensional masih mengandalkan pihak perantara, sehingga rentan terhadap ketidaksesuaian layanan, *moral hazard*, dan asimetri informasi (Dewi Kusumawati & Aulawi, 2021). Untuk mengoptimalkan implementasi *smart contract*, *Optimistic Rollups* dipilih karena menawarkan *throughput* transaksi yang lebih tinggi dan biaya yang lebih rendah dibandingkan *ZK-Rollups* maupun *Layer 1* (Artenie et al., 2025; Patel, 2023; Tortola et al., 2024). Optimalisasi ini penting karena jaringan utama sering mengalami kemacetan akibat keterbatasan *throughput* (Khan et al., 2021; Koutmos, 2023).

Penelitian yang dilakukan oleh (Patel, 2023) telah menilai efisiensi biaya *Optimistic Rollups*, tetapi belum mengkaji penerapannya pada arsitektur reservasi yang mengintegrasikan ERC-1155 dan *escrow*. Penelitian yang dilakukan oleh (Kuhn et al., 2021) menggunakan penerapan ERC-1155 untuk *traceability* produksi multi-hierarki, namun belum mengintegrasikan *escrow* untuk pengelolaan dana dan verifikasi transaksi secara otomatis. Penelitian yang dilakukan oleh (Agung et al., 2025) mengevaluasi mekanisme *escrow on-chain* pada tokenisasi tiket berbasis ERC-721, tetapi belum memanfaatkan kapabilitas token multiaset seperti ERC-1155 untuk mengelola aset *fungible* dan *non-fungible* dalam satu

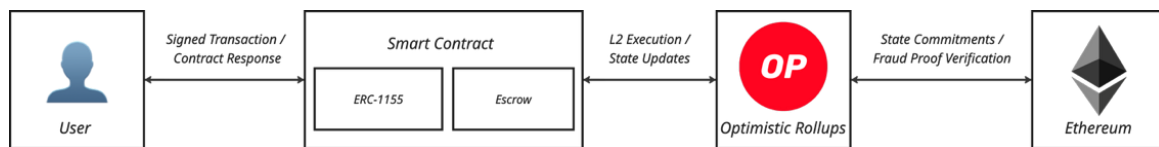
kerangka. Hingga kini, belum ada penelitian yang mengevaluasi performa empiris dari integrasi ketiganya dalam konteks reservasi hotel.

Penelitian ini memilih ERC-1155 karena secara fungsional mampu mengelola dua entitas yang saling bergantung, yaitu stok kamar (*fungible*) dan *invoice* reservasi (NFT) dalam satu *state machine*, sehingga status antara ketersediaan kamar, pesanan aktif, dan dana *escrow* tetap konsisten tanpa perlu sinkronisasi lintas-kontrak. Secara fungsional, struktur ini memperkecil resiko *mismatch* antara token ketersediaan kamar dan token bukti reservasi.

Tujuan penelitian ini adalah menganalisis efisiensi biaya transaksi melalui perilaku eksekusi fungsi pada *smart contract* yang mengintegrasikan arsitektur ERC-1155, mekanisme *escrow*, dan *Optimistic Rollups* dalam konteks reservasi hotel. Analisis dilakukan secara rinci per fungsi untuk menilai sejauh mana *Layer 2* mampu menurunkan biaya transaksi dibandingkan *Layer 1*. Kontribusi penelitian (*novelty*) ini terletak pada pemanfaatan token *hybrid* ERC-1155 yang mengelola stok kamar dan NFT *invoice* dan integrasi *escrow on-chain* dalam satu arsitektur, serta evaluasi empiris terhadap performa biaya transaksi di *Layer 2*.

METODE

Arsitektur konseptual penelitian ini ditunjukkan pada gambar 1. Dalam arsitektur tersebut, pengguna mengirim transaksi yang ditandatangani ke *smart contract* yang kemudian dieksekusi pada *Optimistic Rollups*. Hasil eksekusi menghasilkan *status*, *event log*, dan perubahan *state*, yang selanjutnya dikirim ke *Ethereum* sebagai *state commitment* untuk diverifikasi melalui mekanisme *fraud proof*. Arsitektur ini menjadi dasar pengujian performa dan analisis fungsi pada kedua jaringan.



Gambar 1. Arsitektur konseptual *hybrid* token dan *escrow* pada *optimistic rollups*

Penelitian ini merupakan penelitian deskriptif-kuantitatif dengan menggunakan metode eksperimental. Subjek penelitian merupakan eksekusi empat fungsi *smart contract* yang sebagaimana dijelaskan pada Tabel 1. Seluruh fungsi diuji pada dua jaringan, yaitu *Sepolia Testnet* sebagai representasi *Layer 1* dan *Optimism Sepolia Testnet* sebagai representasi *Layer 2* pada *Optimistic Rollups*.

Tabel 1. Deskripsi fungsi *smart contract*

Fungsi	Tujuan	Akses
<i>mintRoom</i>	Mencetak token <i>fungible</i> sebagai representasi tipe kamar	Hotel
<i>reserve</i>	Membuat reservasi dan menyeter <i>escrow</i> awal	Pelanggan
<i>approveOrder</i>	Menyeter <i>escrow</i> hotel, mencetak NFT <i>invoice</i> , dan membakar token <i>fungible</i> sesuai durasi menginap	Hotel
<i>checkIn</i>	Membakar NFT <i>invoice</i> dan melepaskan dana <i>escrow</i> ke hotel.	Pelanggan

Tahap pengembangan *smart contract* dilakukan menggunakan *foundry* dengan versi ^0.8.28 dengan *compiler* yang diatur dengan *optimizer ON*, menggunakan versi EVM *default*, yaitu *prague*, serta implementasi standar token ERC-1155 dari *openzeppelin*. Parameter jaringan untuk *deployment* terlampirkan pada tabel 2, sedangkan alamat kontrak disajikan pada tabel 3.

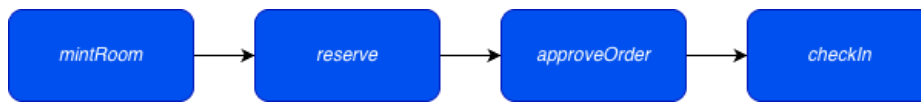
Tabel 2. Parameter jaringan *deployment*

Parameter	<i>Sepolia Testnet</i>	<i>Optimism Sepolia Testnet</i>
Estimasi Gas	<i>Default</i>	<i>Default</i>
RPC URL	https://eth-sepolia.g.alchemy.com/v2/nqsv0IJUSHf0byLbxCsHL	https://sepolia.optimism.io/
Verifier	<i>Blockscout</i>	<i>Blockscout</i>
Verifier URL	https://eth-sepolia.blockscout.com/api	https://testnet-explorer.optimism.io/api

Tabel 3. Alamat kontrak yang digunakan

Jaringan	Alamat
<i>Sepolia Testnet</i>	0x688c37cAdE89Ba325A6e02fe656439962e07be71
<i>Optimism Sepolia Testnet</i>	0xF66071A88469fB284b1Fd17A2e6bb9157BED5286

Tahapan pengujian dilakukan berurutan sebagaimana terlampirkan pada Gambar 2, dengan setiap fungsi dieksekusi lima kali pada kedua jaringan. Fungsi *reserve* hanya dapat dijalankan setelah *mintRoom*, karena perlu mengetahui *Token ID* yang akan dipesan. Fungsi *approveOrder* bergantung pada keberhasilan *reserve* agar mengetahui reservasi mana yang disetujui. Fungsi *checkIn* dapat dieksekusi setelah *approveOrder*, karena NFT *invoice* menjadi syarat untuk membakar token dan melepaskan dana *escrow* ke hotel.



Gambar 2. Alur pengujian

Teknik pengambilan data dilakukan dengan mencatat *Transaction fee* atau biaya transaksi serta *Gas usage* atau konsumsi gas suatu fungsi melalui *Blockscout*. Persamaan 1 menunjukkan struktur biaya transaksi pada *Optimism Sepolia Testnet*, yang terdiri dari *Operator fee* sebagai biaya operasional jaringan, perkalian *Gas price* sebagai biaya per satuan gas dengan *Gas usage*, serta *Layer 1 fee* sebagai biaya publikasi transaksi ke *Layer 1* (Tan-Vo et al., 2024). Persamaan 2 menunjukkan struktur biaya transaksi pada *Sepolia Testnet* yang hanya bergantung pada *Gas usage* dan *Gas price*, sehingga total biaya dihitung dari perkalian keduanya tanpa tambahan komponen lain (Zarir et al., 2021).

$$\text{Transaction fee} = \text{Operator fee} + (\text{Gas usage} * \text{Gas price}) + \text{Layer 1 fee} \quad (1)$$

$$\text{Transaction fee} = \text{Gas usage} * \text{Gas price} \quad (2)$$

Data yang diperoleh kemudian dianalisis menggunakan *mean* untuk menggambarkan efisiensi performa antar jaringan. Perbedaan *mean* biaya transaksi antar jaringan diuji dengan *t-test*, dengan $\alpha = 0,05$. Hipotesisnya: H_0 = tidak ada perbedaan signifikan biaya transaksi antara *Layer 1* dan *Layer 2*; H_1 = *Layer 2* lebih efisien dan memiliki biaya transaksi lebih rendah daripada *Layer 1*.

HASIL DAN PEMBAHASAN

Hasil

Hasil pada tabel 3 menunjukkan bahwa keempat fungsi *smart contract* memiliki biaya transaksi yang lebih rendah pada *Layer 2* dibandingkan *Layer 1*, dengan tingkat efisiensi mencapai 99,97%–99,98%. Fungsi *reserve* memiliki konsumsi gas (*Gas Usage*) tertinggi (130.208 gas). Namun, pada *Layer 2*, *mean* biaya transaksi tertinggi justru adalah *mintRoom* ($5,80 \times 10^{-8}$ ETH) yang bukan merupakan fungsi *escrow* dan konsumsi gasnya tertinggi kedua diantara fungsi lainnya. Kondisi ini berbeda dengan *Layer 1*, di mana *mean* biaya transaksi tertinggi terdapat pada fungsi *reserve* yang termasuk alur *escrow*. Di sisi lain, fungsi *checkIn* konsisten menunjukkan konsumsi gas serta *mean* biaya transaksi terendah pada kedua jaringan.

Tabel 3. Ringkasan *mean* biaya transaksi dan *gas usage* per fungsi di *layer 1* dan *layer 2*

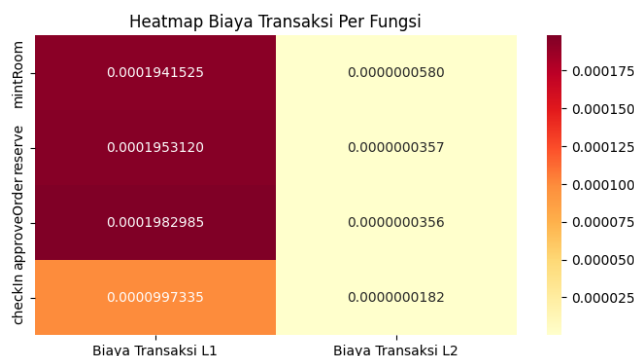
Fungsi	Gas Usage	Mean Biaya Transaksi Layer 1	Mean Biaya Transaksi Layer 2	Efisiensi Layer 2
<i>mintRoom</i>	129,435	$1,94 \times 10^{-4}$ ETH	$5,80 \times 10^{-8}$ ETH	99,97%
<i>reserve</i>	130,208	$1,95 \times 10^{-4}$ ETH	$3,57 \times 10^{-8}$ ETH	99,98%
<i>approveOrder</i>	128,219	$1,98 \times 10^{-4}$ ETH	$3,55 \times 10^{-8}$ ETH	99,98%
<i>checkIn</i>	66,489	$9,97 \times 10^{-5}$ ETH	$1,82 \times 10^{-8}$ ETH	99,98%

Hasil *t-test* pada tabel 4 menunjukkan bahwa semua fungsi memiliki nilai *P-value* di bawah 0,05 atau 5×10^{-2} , sehingga semua H_0 ditolak. Diantara empat fungsi, *approveOrder* memiliki nilai *T-statistic* yang lebih rendah dibandingkan lainnya (33,20964229157027). Sedangkan, nilai *T-statistic* pada ketiga fungsi lainnya berkisaran 70036–8620, yang menunjukkan perbedaan biaya yang jauh lebih ekstrem antara *Layer 1* dan *Layer 2*.

Tabel 4. Hasil *t-test* biaya transaksi antara *layer 1* dan *layer 2*

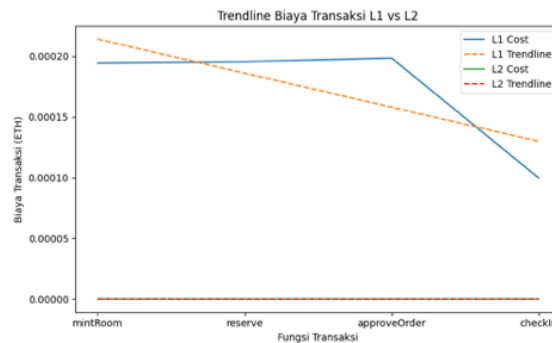
Fungsi	<i>T-statistic</i>	<i>P-value</i>	Keterangan
<i>mintRoom</i>	7036,66026565302	$2,45 \times 10^{-15}$	H_0 ditolak
<i>reserve</i>	8619,113561811748	$1,09 \times 10^{-15}$	H_0 ditolak
<i>approveOrder</i>	33,20964229157027	$4,90 \times 10^{-6}$	H_0 ditolak
<i>checkIn</i>	8619,11355745441	$1,09 \times 10^{-15}$	H_0 ditolak

Visualisasi berupa *heatmap* memperlihatkan perbandingan *mean* biaya transaksi setiap fungsi di *Layer 1* dan *Layer 2* sebagaimana terlampir pada Gambar 3. Warna merah pada *mintRoom*, *reserve*, dan *approveOrder* di *Layer 1* menyoroti fungsi dengan biaya rata-rata lebih tinggi dibandingkan *checkIn*. Warna krem yang seragam pada seluruh fungsi di *Layer 2* menunjukkan bahwa rata-rata biaya yang hampir setara pada semua fungsi.



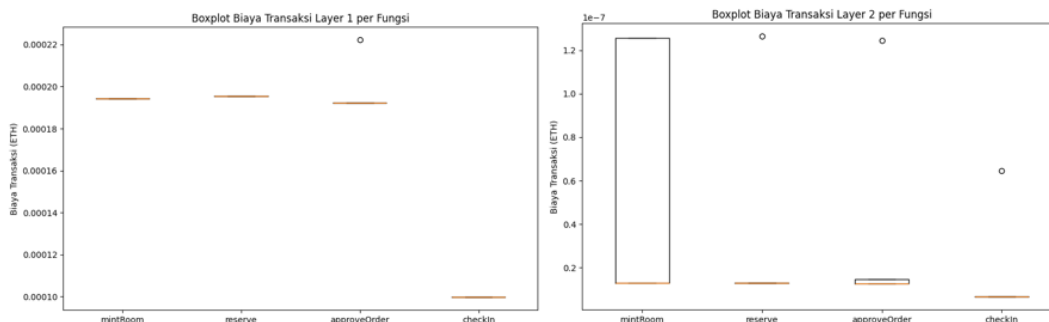
Gambar 3. *Heatmap* biaya transaksi tiap fungsi di *Layer 1* dan *Layer 2*

Visualisasi berupa *trendline* memperlihatkan biaya transaksi bersamaan dengan biaya aktual untuk *Layer 1* dan *Layer 2* sebagaimana terlampir pada Gambar 4. Pada *Layer 2*, *trendline* mengikuti biaya aktual, menandakan stabilitas antar fungsi. Sebaliknya, pada *Layer 1* cenderung menurun dan tidak sepenuhnya merepresentasikan biaya aktual, terutama karena biaya fungsi *checkIn* turun secara signifikan dibandingkan fungsi lainnya.



Gambar 4. Trendline biaya transaksi tiap fungsi di *Layer 1* dan *Layer 2*

Visualisasi berupa *boxplot* memperlihatkan distribusi biaya transaksi tiap fungsi di *Layer 1* dan *Layer 2* sebagaimana terlampirkan pada Gambar 5. Pada *Layer 1*, *outlier* hanya muncul di *approveOrder*, menunjukkan adanya iterasi dengan biaya lebih tinggi, sedangkan fungsi lain relatif stabil. Di *Layer 2*, *outlier* muncul pada *reserve*, *approveOrder*, dan *checkIn*, sementara distribusi biaya *mintRoom* memiliki rentang antarkuartil (IQR) yang paling besar dibandingkan fungsi lainnya.



Gambar 5. Boxplot biaya transaksi tiap fungsi di *layer 1* dan *layer 2*

Pembahasan

Hasil penelitian menunjukkan bahwa *smart contract* ERC-1155 dengan mekanisme *escrow* pada jaringan *Optimistic Rollups* memiliki efisiensi sebesar 99,97-99,98% pada aspek biaya transaksi dibandingkan *Layer 1*, dengan seluruh *t-test* menunjukkan penolakan terhadap H_0 . Hal ini dikarenakan *Optimistic Rollups* melakukan *batching* transaksi, eksekusi sebagian logika secara *off-chain*, dan kompresi data sebelum dipublikasikan ke *Layer 1*. Kombinasi mekanisme ini secara teknis mengurangi beban komputasi dan penyimpanan di *blockchain* utama, sehingga biaya gas per transaksi menurun drastis dibandingkan *Layer 1*.

Selain itu, *mean* biaya transaksi antar fungsi di *Layer 2* yang hampir setara pada semua fungsi sebagaimana terlihat pada Gambar 3 menunjukkan bahwa *Layer 2* lebih stabil dibandingkan *Layer 1*, karena struktur biaya *Optimistic Rollups* didominasi oleh *Layer 1 data fee*. Temuan ini selaras dengan Gambar 4, di mana *trendline* pada *Layer 2* yang mengikuti biaya aktual, menandakan bahwa *Layer 2* memiliki konsistensi dan keteraturan pola biaya antar

fungsi dibandingkan *Layer 1* yang fluktuatif antar fungsinya. Namun, setiap fungsi di *Layer 2* tetap menampilkan *outlier* antar iterasi sebagaimana terlihat pada Gambar 5, karena *Layer 1 data fee* mengikuti *base fee* Ethereum yang dapat berubah hingga 12,5% setiap blok. Pada skala biaya transaksi *Layer 2* yang sangat rendah, fluktuasi kecil pada *base fee* tampak signifikan sebagai variasi antar iterasi, meskipun tidak memengaruhi nilai *mean* antar fungsi.

Fungsi *approveOrder* memiliki *T-statistic* paling rendah karena selisih *mean* biaya transaksi antara *Layer 1* dan *Layer 2* tidak sebesar fungsi lainnya. Hal ini disebabkan oleh adanya *outlier* pada kedua jaringan sebagaimana terlampirkan pada Gambar 5, yang menaikkan varians sekaligus mengurangi jarak rata-rata biaya *Layer 1–Layer 2*. Sebaliknya, fungsi lainnya memiliki perbedaan biaya yang jauh lebih ekstrem dan varians yang lebih kecil, sehingga menghasilkan nilai *T-statistic* sangat besar.

Fungsi *reserve* memiliki konsumsi gas yang paling besar diantara lainnya. Hal ini dikarenakan *reserve* melakukan *update state* besar pada *storage* dibandingkan dengan fungsi lainnya, yaitu saat menambahkan data ke *struct* melalui *mapping*. Namun, *approveOrder* memiliki biaya transaksi tertinggi di *Layer 1* karena dieksekusi pada kondisi *base fee* yang lebih tinggi dibandingkan iterasi saat *reserve* dijalankan. Pada *Layer 1*, biaya transaksi sangat bergantung pada dinamika *base fee* yang dapat berubah tiap blok, sehingga fungsi dengan konsumsi gas yang tidak terbesar sekalipun dapat menghasilkan biaya akhir lebih tinggi jika dipanggil ketika *base fee* sedang meningkat. Pada *Layer 2*, *mean* biaya transaksi tertinggi di adalah *mintRoom*. Hal ini terjadi karena biaya transaksi di *Layer 2* tidak hanya dipengaruhi oleh konsumsi gas, tetapi juga oleh ukuran data (*call data*) yang dikirim ke *Layer 1* saat *batching*. Proses *mintRoom* menghasilkan *payload* yang lebih besar sehingga lebih sensitif terhadap perubahan biaya data *Layer 1*. Akibatnya, variabilitas biaya *mintRoom* menjadi lebih tinggi sebagaimana ditunjukkan oleh IQR yang lebih lebar pada Gambar 5. Kemudian, fungsi *checkIn* memiliki konsumsi gas dan biaya transaksi yang terendah, dikarenakan fungsi ini hanya *update state* sederhana, membakar NFT, dan *transfer* dana dari *escrow*.

Struktur ERC-1155 memang membuat fungsi berbasis *mint* seperti *mintRoom* dan *approveOrder* memiliki konsumsi gas lebih tinggi dibanding fungsi yang hanya melakukan *update state* sederhana, seperti *checkIn*. Namun, peningkatan gas akibat mekanisme ERC-1155 tersebut tidak sebesar *overhead* yang muncul pada fungsi *reserve*, karena *reserve* melakukan *update state* yang jauh lebih besar melalui penambahan data baru ke *struct* melalui *mapping*. Dengan demikian, beban konsumsi gas terbesar tetap berasal dari perubahan *storage* yang intensif, bukan dari karakteristik ERC-1155 itu sendiri.

Berbeda dari penelitian yang dilakukan oleh Patel (2023) yang hanya menilai efisiensi biaya *Optimistic Rollups* penelitian ini menunjukkan penerapan *Optimistic Rollups* pada *smart contract* ERC-1155 dengan mekanisme *escrow* dalam konteks reservasi hotel menghasilkan penghematan biaya hingga 99,97-99,98%, sehingga memberikan *insight* tambahan tentang kinerja *Layer 2* dalam konteks aplikasi nyata. Kemudian, berbeda dengan penelitian yang dilakukan oleh Kuhn et al. (2021) yang menggunakan ERC-1155 untuk *traceability* produksi multi-hierarki serta penelitian yang dilakukan oleh Agung et al. (2025) yang meneliti mekanisme *escrow on-chain* untuk tokenisasi tiket berbasis ERC-721, penelitian ini menunjukkan bagaimana gabungan keduanya dalam satu arsitektur di *Layer 2* tidak secara signifikan meningkatkan variabilitas biaya transaksi. Variasi biaya lebih banyak dipengaruhi oleh ukuran *update state* dan dinamika *base fee Layer 1* daripada karakteristik ERC-1155 atau logika *escrow* itu sendiri.

Penelitian ini memberikan kontribusi dengan memperluas pemahaman tentang *hybrid* token dengan mengintegrasikan ERC-1155, sehingga model representasi aset majemuk dalam teori *tokenomics* dapat diperluas. Implementasi ini menunjukkan bagaimana satu kontrak dapat mengelola *fungible token* untuk stok kamar hotel sekaligus *non-fungible token* untuk *invoice* sebagai bukti transaksi, memperlihatkan fleksibilitas token dalam merepresentasikan aset yang

heterogen. Selain itu, mekanisme *escrow* dua sisi memperkuat teori *trustless settlement* melalui insentif berbasis *game-theoretic*, di mana dana hanya dilepas ketika kondisi kontrak terpenuhi, sehingga risiko *fraud* berkurang tanpa membutuhkan pihak ketiga.

Efisiensi biaya transaksi hingga 99,97-99,98% pada *Layer 2* memberikan *insight* bagi industri OTA apabila mengevaluasi kemungkinan transisi ke sistem berbasis *blockchain*. Temuan ini menunjukkan bahwa biaya operasional bukan lagi faktor penghambat utama dalam mempertimbangkan penggunaan *smart contract* untuk mendukung proses reservasi. *Challenge period* tujuh hari pada *Optimistic Rollups* menunjukkan bahwa finalitas transaksi yang tertunda dapat mengganggu UX jika ditampilkan apa adanya, sehingga antarmuka perlu menandai reservasi sebagai aktif segera setelah transaksi dikonfirmasi di *Layer 2*. Bagi operasional hotel, periode ini menuntut pemisahan antara status operasional yang digunakan untuk *check-in* dan status finalitas *on-chain* yang baru definitif setelah tujuh hari, sehingga pelayanan tetap dapat berjalan tanpa harus menunggu finalitas *blockchain*. Selain itu, Sistem perlu *middleware* yang menerjemahkan *event on-chain* menjadi pembaruan di *Property Management System*, seperti ketersediaan kamar atau status pembayaran agar hotel bisa memanfaatkan transparansi *blockchain* tanpa mengubah alur operasional yang sudah ada.

Beberapa risiko komersial yang mungkin terjadi adalah potensi hambatan proses reservasi bagi pengguna yang masih belum terbiasa dengan UX *Web3*. Bagi hotel, penguncian dana selama *challenge period* dapat mengganggu likuiditas dan ketidaksinkronan antara *blockchain* dan PMS berpotensi mengacaukan data operasional. Mitigasinya meliputi implementasi antarmuka menyerupai *Web2* untuk menyembunyikan kompleksitas *blockchain*, konfirmasi *off-chain* untuk kebutuhan operasional sebelum finalitas, mekanisme *auto-resync* dengan PMS, serta opsi transisi *hybrid* agar hotel dapat mengadopsi sistem secara bertahap.

Keterbatasan penelitian ini meliputi penggunaan *testnet* yang belum sepenuhnya mencerminkan *mainnet*, belum diuji pada beban tinggi, dan belum menangani pembatalan atau perubahan pesanan, serta *trade-off Optimistic Rollups* berupa finalitas yang tidak instan akibat *challenge period* tujuh hari. Hal ini menimbulkan ketidakpastian meskipun mekanisme *fraud proof* menjaga keamanan dan efisiensi biaya. Secara praktis, hal ini menekankan perlunya pengujian lebih lanjut pada kondisi nyata agar efektivitas arsitektur berbasis *hybrid* token, *escrow* dan *Layer 2* dapat dipastikan. Secara teoretis, keterbatasan ini membuka peluang untuk memperkuat pemahaman mengenai mekanisme *escrow*, ERC-1155, dan *Optimistic Rollups*, serta bagaimana desain *hybrid* token memengaruhi cara transaksi dieksekusi di *Layer 2*.

SIMPULAN

Penelitian ini secara empiris membuktikan bahwa integrasi ERC-1155, mekanisme *escrow*, dan *Optimistic Rollups* meningkatkan efisiensi biaya transaksi dibandingkan *Layer 1* sebesar 99,97-99,98% melalui *benchmark* performa terhadap 4 fungsi yang terlampirkan di Tabel 1. Seluruh fungsi menghasilkan *p-value* < 0,05, sehingga hipotesis H_0 ditolak. Hal ini menegaskan bahwa *Layer 2* secara statistik terbukti lebih efisien daripada *Layer 1*. Sedangkan, integrasi ERC-1155 dan *escrow* dalam satu arsitektur di *Layer 2* tidak secara signifikan meningkatkan variabilitas biaya transaksi. Secara teoretis, penelitian ini memperluas pemahaman mengenai *hybrid* token dalam teori *tokenomics*, serta teori *trustless settlement* berbasis insentif *game-theoretic* dalam *escrow* dua sisi. Kontribusi metodologis penelitian ini terletak pada pengukuran biaya transaksi menggunakan *mean* dan *t-test*, yang menghasilkan kerangka evaluasi replikatif untuk menilai efisiensi biaya di berbagai platform *blockchain* maupun arsitektur *smart contract*. Meskipun demikian, penelitian ini memiliki batasan, seperti penggunaan *testnet*, belum adanya skenario beban tinggi, belum tersedianya fitur pembatalan atau perubahan pesanan, serta *trade-off* berupa *challenge period* tujuh hari yang menunda finalitas transaksi. Keterbatasan ini membuka peluang penelitian lanjutan untuk pengujian *mainnet*, desain *escrow* adaptif, serta analisis risiko dalam sistem reservasi berbasis *blockchain*.

REFERENSI

- Agung, A., Pratyaksa Nugraha, L., Wayan, N., Dewi, E. R., & Purnama, F. (2025). Implementation of Blockchain Smart Contract for Online Concert Ticket Transactions Based on NFTs. *Journal of Applied Informatics and Computing*, 9(4), 1115–1126. <https://doi.org/10.30871/jaic.v9i4.9598>
- Ankor, P., Kwame, N.-G., Dery, J., Asamoah, E., & Lardi Issaka, L. (2025). Forms and Benefits of Online Reservation Systems: Empirical Evidence from a Census of Star-Rated Hotels in Accra. *International Journal of Research and Innovation in Social Science*, 9(8), 368–384. <https://doi.org/10.47772/IJRISS>
- Aprialim, F., Adnan, & Paundu, A. W. (2021). Penerapan Blockchain dengan Integrasi Smart Contract pada Sistem Crowdfunding. *Jurnal Rekayasa Sistem Dan Teknologi Informasi*, 5(1), 148–154. <https://doi.org/10.29207/resti.v5i1.2613>
- Artenie, A. C., Silaghi, D. L., & Popescu, D. E. (2025). Exploring the Synergy Between Ethereum Layer 2 Solutions and Machine Learning to Improve Blockchain Scalability. *Journal Computers*, 14(9), 1–37. <https://doi.org/10.3390/computers14090359>
- Dewi Kusumawati, R., & Aulawi, H. (2021). Peran Perceived Effectiveness of Escrow Services terhadap Satisfaction dan Trust pada E-Commerce di Indonesia. *Jurnal Ekonomi Indonesia*, 10(1), 13–24. <https://doi.org/10.52813/jei.v10i1.67>
- Gusti, I., Indra, N., Utama, P., Made, I., Darmawan, E., Juliarta, W., & Suardana, D. G. (2025). The Role of Internal Control in Reducing the Risk of Room Reservation Fraud at Adi Rama Beach Hotel. *Journal of Entrepreneurial and Business Diversity*, 3(1), 398–401. <https://doi.org/10.38142/jebd.v3i1.286>
- Harji, L. putri, & Suharto, B. (2024). Analisis Peluang Dan Hambatan Penggunaan Sistem Digital Dalam Mengoptimalkan Operasional Bisnis Hotel. *Jurnal Ilmiah Hospitality Management*, 14(2), 193–212. <https://doi.org/10.22334/jihm.v14i2.280>
- I Putu, E. I., & Andita Dwi Pratiwi, K. (2024). Optimalisasi Sistem Reservasi Hotel dengan Teknologi Terkini: Studi Kasus pada Discovery Kartika Plaza Hotel. *Journal of Hotel Management*, 2(1), 72–77. <https://doi.org/10.52352/jhm.v2i1.1303>
- Kassem, R. (2024). Spotlight on fraud risk in hospitality a systematic literature review. *International Journal of Hospitality Management*, 116(1), 1–16. <https://doi.org/10.1016/j.ijhm.2023.103630>
- Khan, D., Jung, L. T., & Hashmani, M. A. (2021). Systematic literature review of challenges in blockchain scalability. *Journal Applied Sciences*, 11(20), 1–27. <https://doi.org/10.3390/app11209372>
- Koutmos, D. (2023). Network Activity and Ethereum Gas Prices. *Journal of Risk and Financial Management*, 16(10), 1–14. <https://doi.org/10.3390/jrfm16100431>
- Kuhn, M., Funk, F., Zhang, G., & Franke, J. (2021). Blockchain-based application for the traceability of complex assembly structures. *Journal of Manufacturing Systems*, 59(1), 617–630. <https://doi.org/10.1016/j.jmsy.2021.04.013>
- Maulina, L. (2023). Revitalisasi Industri Perhotelan dengan Inovasi Teknologi: Meningkatkan Keunggulan Bersaing dan Pengalaman Pelanggan. *Jurnal Ilmiah Manajemen, Ekonomi, Dan Akuntansi*, 7(1), 504–519. <https://doi.org/10.31955/mea.v7i1.2962>
- Oyedotun, S. A., Oise, G. P., Akilo, B. E., Nwabuokeyi, O. C., Ejenarhome, P. O., Fole, M., & Onwuzo, C. J. (2025). The Role of Internal Audit in Fraud Detection and Prevention: A Multi-Contextual Review and Research Agenda. *Journal of Science Research and Reviews*, 2(2), 76–85. <https://doi.org/10.70882/josrar.2025.v2i2>
- Patel, O. (2023). Blockchain Layer 2 Rollups, Optimistic vs zero knowledge. *International Journal of Advanced Research in Engineering and Technology*, 14(2), 13–29. <https://doi.org/10.34218/ijaret.14.2.002>

- Prastika, P. D. A., & Suharto, B. (2024). Analisis Efisiensi Reservasi Kamar Hotel Secara Online dari Prespektif Sistem Digital. *Jurnal Pendidikan Tambusai*, 7(3), 13807–13817.
- Ramdhan, K. Z., & Suharto, B. (2024). Transformasi Digital Di Bidang Hospitalitas: Menjelajahi Peran Teknologi Dalam Memperlancar Operasional Hotel. *Jurnal Ilmiah Hospitality Management*, 14(2), 181–192. <https://doi.org/10.22334/jihm.v14i2.282>
- Salameh, A. A., Al Mamun, A., Hayat, N., & Ali, M. H. (2022). Modelling the significance of website quality and online reviews to predict the intention and usage of online hotel booking platforms. *Journal Heliyon*, 8(9), 1–9. <https://doi.org/10.1016/j.heliyon.2022.e10735>
- Singh, H. (2025). Blockchain-enabled fraud detection technique for secure transactions. *International Journal of Computing and Artificial Intelligence*, 6(2), 33–36. <https://doi.org/10.33545/27076571.2025.v6.i2a.175>
- Soklis, G., Petrakos, G., & Panousi, S. (2025). The Contribution of the Hotel Industry to the Greek Economy. *Journal Tourism and Hospitality*, 6(1), 1–21. <https://doi.org/10.3390/tourhosp6010011>
- Taherdoost, H. (2023). Smart Contracts in Blockchain Technology: A Critical Review. *Journal Information*, 14(2), 1–19. <https://doi.org/10.3390/info14020117>
- Tan-Vo, K., Pham, K., Huynh, P., Thi, M. T. N., Ta, T. T., Nguyen, T., Nguyen-Hoang, T. A., Dinh, N. T., & Nguyen, H. T. (2024). Optimizing Academic Certificate Management With Blockchain and Machine Learning: A Novel Approach Using Optimistic Rollups and Fraud Detection. *Institute of Electrical and Electronics Engineers*, 12(1), 168135–168159. <https://doi.org/10.1109/ACCESS.2024.3486029>
- Tortola, D., Lisi, A., Mori, P., & Ricci, L. (2024). Tethering Layer 2 solutions to the blockchain: A survey on proving schemes. *Computer Communications*, 225(1), 289–310. <https://doi.org/10.1016/j.comcom.2024.07.017>
- Zacky, M. I., Helmi, S., & Della Cella, I. (2023). Smart Contracts on the Blockchain: Design, Use Cases, and Prospects. *Blockchain Frontier Technology*, 3(1), 54–73. <https://doi.org/10.34306/bfront.v3i1.363>
- Zarir, A. A., Oliva, G. A., Jiang, Z. M. J., & Hassan, A. E. (2021). Developing Cost-Effective Blockchain-Powered Applications: A Case Study of the Gas Usage of Smart Contract Transactions in the Ethereum Blockchain Platform. *Association for Computing Machinery*, 30(3), 1–38. <https://doi.org/10.1145/3431726>