

PRACTICAL BLOCKCHAIN DEPLOYMENT FOR TICKET MANAGEMENT: A CASE STUDY WITH SOLIDITY SMART CONTRACTS

Jan Mrákava¹, Radim Farana²

¹Department of Economics, Faculty of Business and Economics, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech Republic

²Department of Informatics, Faculty of Business and Economics, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech Republic

DOI: https://doi.org/10.11118/ecos2026_023

ABSTRACT

This article deals with the design and implementation of a web application for ticket management based on a local Ethereum-compatible blockchain. The work responds to the gap between marketing claims about blockchain as a universal document verification tool and limited knowledge of real transaction costs. The frontend uses React.js, the backend Node.js with Express and MongoDB, and the logic consists of two Solidity 0.8.10 smart contracts (EventContract, TicketContract) deployed through Truffle into the Ganache environment (Chain ID 1337). Measurements showed that creating a single event with a typical description costs 0.0078 ETH (~546 CZK at the 2024 exchange rate), while ticket purchase and resale require an order of magnitude lower fees. The main finding: transaction cost is dominated by the length of text strings, which for document verification in general means storing only hashes and short identifiers on the chain.

Keywords: blockchain, Ethereum, smart contracts, Solidity, document verification, gas cost
JEL Code: O33

1 INTRODUCTION

Two widely reported cases illustrate the same problem. In 2017, tickets for the Rammstein concert in Prague reappeared in secondary sale at several times the original price, and the organizer had no way to verify who had actually sold what to whom. In 2019, the forensic expert Jiří Závora presented a forged document to 66 offices in the Czech Republic; in 51 cases the officials accepted it as authentic. Both cases share the same root — there is no single source of truth about which document (a ticket, a diploma, a power of attorney) is genuine and to whom it currently belongs.

Blockchain — a distributed and cryptographically secured registry — appears as a candidate solution thanks to record immutability and smart contracts for automation. The aim of this work was to build a functional prototype that demonstrates this approach on a concrete use case (ticket management), to measure real transaction costs, and to generalize the results for document verification at large.



This article is licensed under CC BY-SA 4.0 Attribution-ShareAlike 4.0 International license.
<https://creativecommons.org/licenses/by-sa/4.0/deed.en>

2 BUILDING BLOCKS AND CONSENSUS

A cryptographically secured chain of timestamps for digital documents was described by Haber and Stornetta (1991); Merkle (1988) added a tree structure of hashes for efficient integrity verification. These concepts were combined by Nakamoto (2008) in the Bitcoin whitepaper. To reach consensus about network state, various algorithms are used: Proof-of-Work (high security at the cost of energy consumption comparable to the annual demand of Pakistan according to CCAF 2023), Proof-of-Stake (energy efficient), and Proof-of-Authority (for private networks with validated identity of validators, highly scalable). Ganache uses a simplified PoA variant — confirmation latency is eliminated and transactions confirm within milliseconds.

3 EXISTING DOCUMENT VERIFICATION SYSTEMS

Academic reviews (Catalini and Gans 2017; Houben and Snyers 2018) and commercial systems confirm the same pattern — the document is hashed and only the digest with metadata is stored on the chain. Dock uses decentralized identifiers DID with Issuer/Holder/Verifier roles. The Czech Certoo uses the Litecoin blockchain with a maximum file size of 500 MB. Diplomachain cooperates with the Czech Ministry of Industry and Trade, the Chamber of Commerce, CTU Prague and TBU Zlín. Blockademia runs on Cardano with the ACI token. None of these systems stores entire documents on the chain — only the hash, identifier, and metadata, which ensures a constant transaction cost regardless of document size.

4 METHODOLOGY

4.1 Technology Stack

For the prototype, a combination enabling rapid development and measurement of real transaction costs was chosen: React.js (frontend), Node.js with Express.js and MongoDB (backend with user data), Ganache (local development blockchain with ten preset accounts of 100 fictitious ETH each), Truffle (compilation, migration, and contract testing), Solidity 0.8.10 (with built-in overflow protection), and MetaMask (browser wallet). Ganache listens on `127.0.0.1:8545` with Chain ID 1337; MetaMask is configured with the network name “TicketBlock”.

4.2 Two Smart Contracts

The application architecture separates event management and ticket management into two independent contracts. EventContract owns the Event structure with the `eventID` field (bytes32), name, date, ticket count, price, description, category, and owner address; the key function `buyTicket(eventID)` delegates the call to TicketContract. TicketContract owns the Ticket structure with `ticketID` (bytes32), `eventID`, owner, purchase date, and flags `isRedeemed`, `isValid`, and `forSale`; key functions include `createNewTicket`, `transferTicket`, `setTicketForSale`, and `verifyTicket`. The separation has a practical reason — events and tickets have different lifecycles, which would lead to more complex logic and higher gas overhead in a monolithic contract.

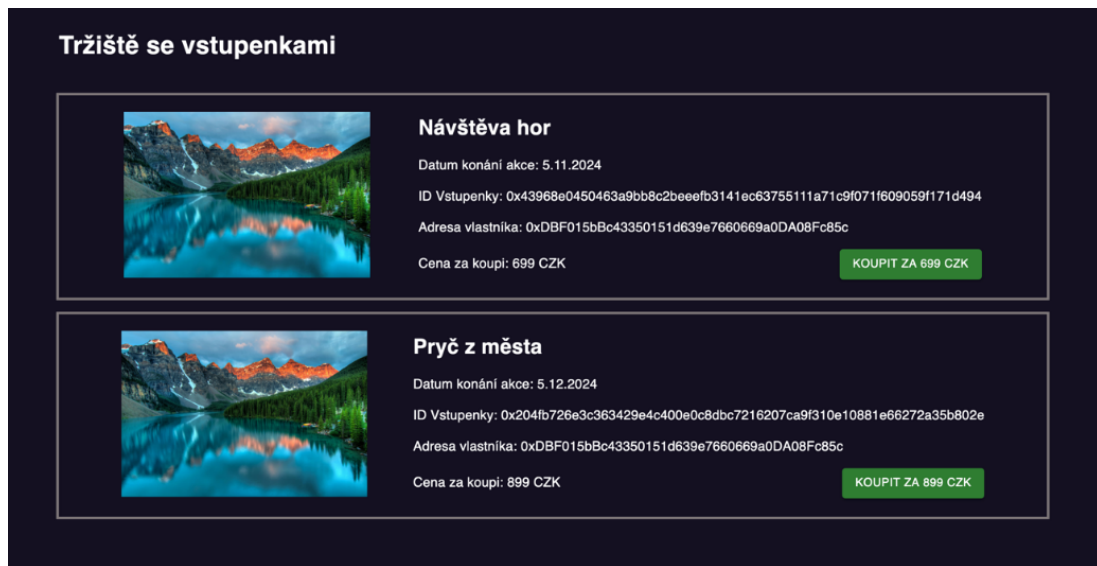


Fig. 1: Resale marketplace in the web application — each entry shows the event name, on-chain event ID, current owner address (Ethereum account) and asking price in CZK, which is converted to ETH/wei at confirmation time.

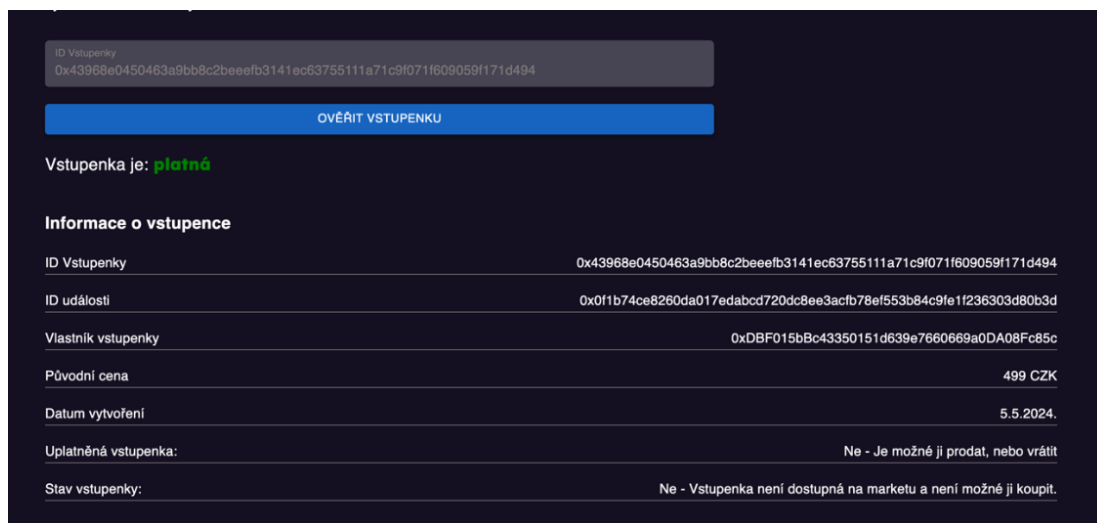


Fig. 2: Result of ticket verification after entering a valid ID — the system immediately returns the status (“valid”), event ID, owner address, original price, creation date and redemption status. Verification is a view function that consumes no gas and can be invoked by anyone.

4.3 Identifier Generation and Audit Trail

Identifiers are built by hashing the combination of attributes using the Keccak-256 function, which produces a 32-byte output (bytes32):

```
„eventID“=“keccak256“ („abi.encodePacked“ („eventName“ ,“block.timestamp“ )).
```

```
„ticketID“=“keccak256“ („abi.encodePacked“ („eventID“ ,“userAddress“ ,“block.timestamp“ )).
```

The collision risk at this size is negligible; before writing into the map, `require(!eventExists[_eventID])` is checked, which would formally catch a collision should it occur. Each contract emits structured events permanently written into the blockchain as an audit trail — TicketContract emits ten event types (`TicketCreated`, `TicketBought`, `TicketTransferred`, `TicketRedeemed`, and others). The frontend obtains the history by calling `getPastEvents(,allEvents‘, { filter: { ticketID }, fromBlock: 0, toBlock: ,latest‘ } )`.

4.4 Gas Cost Measurement

After truffle migrate, the addresses of deployed contracts and their ABI are generated. Web3.js initializes the blockchain connection; state-changing functions use `.send({ from: userAddress, gas: ,500000‘ } )`, view functions only `.call()`. For each key operation (event creation with various description lengths, purchase, resale, verification), gas usage, gas price, and total cost in ETH and CZK were measured according to the Coinbase 2024 exchange rate.

5 RESULTS

5.1 Transaction Costs

For the full workflow (event creation → purchase → resale → verification), the values in Tab. 1 were measured. The cost of event creation is dominated by description length, while purchase and resale have a relatively constant cost given by hashing and map update. In Solidity, each additional byte of a string represents about 17 gas units, which scales the cost linearly with the volume of text data (Mrákava, 2026).

Tab. 1 Measured transaction costs (Ganache, gas price 2.5 gwei, Coinbase 2024 exchange rate)

Operation	Cost (ETH)	Cost (CZK)	Note
Event creation — short description (30 chars)	~0.002	~140	minimal payload
Event creation — typical description (200 chars)	0.0078	~546	gas usage ~3,305,129
Event creation — long description (500 chars)	~0.015	~1,050	festival program
Ticket purchase	~0.001	~70	no payload
Ticket resale	~0.001	~70	no payload
Verification (view)	0	0	off-chain

5.2 Stability and Auditability

The application was tested on 30 tickets across 5 events; all transactions completed successfully and no identifier collision occurred. In a simulation of two concurrent purchases of the last available ticket, the second purchase reverted with “Not enough tickets left”. For a randomly selected ticket, four history records were retrieved through `getEventsForTicketID` (creation, two transfers, redemption) including transaction hashes, block hashes, and timestamps. Third-party verification without access to the UI was possible through a Web3.js call directly to Ganache — auditability without trusting the system operator is the main value blockchain adds over a centralized solution.

6 DISCUSSION

The measurements revealed a practical boundary for economic deployment: the size of data stored on chain must be as small as possible. For tickets this was not a problem — ID, owner, and state flags fit into tens of bytes. For event creation with a long description, the cost rises sharply because each byte of text is expensive on Ethereum. For document verification in general, a practical pattern emerges: the document is stored off chain (IPFS, server, archive), its hash (SHA-256 or keccak-256) is computed, and only the hash, identifier, issuer, and timestamp are stored on chain. Verification then means comparing the hash from the file with the hash on the chain.

Economics matters more than technology. The cost of 546 CZK per event may seem negligible, but for an organizer with 1,000 events per year this means 546,000 CZK only for the blockchain, and at tickets priced 50-200 CZK the viability is questionable. For high-value documents with long validity (diplomas), the situation differs — a one-off cost against the document’s value is reasonable, which explains the interest of CTU and TBU in Diplomachain. For production deployment, Polygon would be worth considering, as it preserves the Ethereum API while reducing fees by an order of magnitude. Permanent on-chain storage also conflicts with the right to be forgotten under GDPR; Czech law does not explicitly regulate blockchain (Hospodka 2020) and no Czech court has officially accepted blockchain as evidence (Hanák 2022).

7 CONCLUSION

The work showed that a functional blockchain application for ticket management can be built with a common stack (React, Node.js, MongoDB, Truffle, Solidity) in a reasonable time, and measured specific transaction costs that marketing materials rarely disclose. The cost of event creation ranges from ~140 CZK for a short description to ~1,050 CZK for a long one — text dominates the gas cost. Architectural separation into two contracts (EventContract, TicketContract) reduces overhead in repeated operations and simplifies code audit. For document verification at large the rule holds: store only hashes and metadata on chain, keep content off chain. Future extensions include deploying an identical contract on Polygon to measure real transaction costs under competitive operation and implementing a combined on-chain hash plus off-chain payload mechanism with deletion option for GDPR compliance.

Acknowledgements

This paper was supported by the project CZ.02.1.01/0.0/0.0/16_017/0002334 Research Infrastructure for Young Scientists, which is co-financed from the Operational Programme Research, Development and Education.

REFERENCES

- CATALINI, C., GANS, J. S. 2017. *Some Simple Economics of the Blockchain*. NBER Working Paper No. 22952. National Bureau of Economic Research. <https://doi.org/10.3386/w22952>
- CAMBRIDGE CENTRE FOR ALTERNATIVE FINANCE (CCAF). 2023. Cambridge Bitcoin Electricity Consumption Index. CCAF. <https://ccaf.io/cbnsi/cbeci>
- COINBASE. 2024. ETH/CZK: Convert Ethereum (ETH) to Czech Koruna (CZK). Coinbase. <https://www.coinbase.com/converter/eth/czk>
- HABER, S. and STORNETTA, W. S. 1991. How to Time-Stamp a Digital Document. *Journal of Cryptology*. 3(2), 99-111. ISSN 0933-2790. <https://doi.org/10.1007/BF00196791>
- HANÁK, O. 2022. What Use Does Blockchain Have in Law? [in Czech]. *Kroupa & Helan*. <https://kroupahelan.cz/blog/jake-vyuziti-ma-blockchain-v-pravu/136>
- HOSPODKA, J. 2020. *Legal Aspects of Cryptocurrencies* [in Czech]. Master's thesis. Prague: Charles University.
- HOUBEN, R., SNYERS, A. 2018. *Cryptocurrencies and Blockchain: Legal Context and Implications for Financial Crime, Money Laundering and Tax Evasion*. Brussels: European Parliament TAX3 Study.
- MERKLE, R. C. 1988. A Digital Signature Based on a Conventional Encryption Function. In: *Advances in Cryptology — CRYPTO '87*. Berlin: Springer, pp. 369-378. ISBN 978-3-540-18796-7. https://doi.org/10.1007/3-540-48184-2_32
- MRÁKAVA, J. 2026. *Practical Blockchain Deployment for Ticket Management with Smart Contracts in Solidity* [in Czech]. Master's thesis, head: FARANA, R. Brno: Mendel University in Brno.
- NAKAMOTO, S. 2008. *Bitcoin: A Peer-to-Peer Electronic Cash System*. <https://bitcoin.org/bitcoin.pdf>
- ZHAI, X., PANG, S., WANG, M., QIAO, S., LV, S. 2021. TVS: A Trusted Verification Scheme for Office Documents Based on Blockchain. *Complex & Intelligent Systems*. 8, 3941-3954. ISSN 2199-4536. <https://doi.org/10.1007/s40747-021-00617-1>

Contact information

Jan Mrákava: e-mail: janmrakava@gmail.com

Radim Farana: e-mail: radim.farana@mendelu.cz