Article

Not peer-reviewed version

Integrating Blockchain Technology into Mobility-as-a-Service Platforms for Smart Cities

Radu Miron *, Mihai Hulea, Vlad Muresan

Posted Date: 1 November 2024

doi: 10.20944/preprints202411.0046.v1

Keywords: blockchain technology; smart cities; Mobility as a Service (MaaS); Integrated Transportation Systems; urban mobility; Decentralized Payment Systems; Hyperledger Fabric; Sustainable Transportation Solutions

Article

Integrating Blockchain Technology into Mobility-as-a-Service Platforms for Smart Cities

Radu Miron *, Mihai Hulea and Vlad Muresan

Automation Department, Faculty of Automation and Computer Science, Technical University of Cluj-Napoca, Memoradumului no. 28, 400114, Romania; radu.miron@aut.utcluj.ro (R.M); mihai.hulea@aut.utcluj.ro (M.H); vlad.muresan@aut.utcluj.ro (V.M.)

* Correspondence: radu.miron@aut.utcluj.ro

Abstract: As cities evolve into smarter and more connected environments, there is a growing need for innovative solutions to improve urban mobility. This study examines the potential of integrating blockchain technology into passenger transportation systems within smart cities, with a particular emphasis on a blockchain-enabled Mobility as a Service (MaaS) solution. In contrast to traditional technologies, blockchain's decentralized structure improves data security and guarantees transaction transparency, thus reducing the risk of fraud and errors. The proposed MaaS framework enables seamless collaboration between key transportation stakeholders, promoting more efficient utilization of services like buses, trains, bike-sharing, and ride-hailing. By improving integrated payment and ticketing systems, the solution aims to create a smoother user experience while advancing urban goals of efficiency, environmental sustainability, and secure data handling. This research evaluates the feasibility of a Hyperledger Fabric-based solution, demonstrating its performance under various load conditions and proposing scalability adjustments based on pilot results. The conclusions indicate that blockchain-enabled MaaS systems have the potential to transform urban mobility. Further exploration into pilot projects and the expansion to freight transportation are needed for an integrated approach to city-wide transport solutions.

Keywords: blockchain technology; smart cities; Mobility as a Service (MaaS); Integrated Transportation Systems; urban mobility; Decentralized Payment Systems; Hyperledger Fabric; Sustainable Transportation Solutions

---

Highlights

What are the main findings?

- Developed a blockchain-enabled Mobility-as-a-Service (MaaS) platform using Hyperledger Fabric to enhance security, transparency, and operational efficiency in urban mobility systems.
- Demonstrated through load testing that the proposed system maintains consistent performance under varying data volumes and concurrent user requests, confirming its scalability for real-world applications.

What is the implication of the main finding?

- Integrating blockchain technology into urban transportation systems can improve stakeholder collaboration and trust, leading to more efficient and secure smart city mobility solutions.
- The successful implementation indicates potential for expanding blockchain-enabled MaaS platforms to include freight logistics, contributing to a unified and sustainable urban transportation ecosystem.

1. Introduction

In contemporary smart cities, incorporating innovative technologies into urban infrastructure is essential for enhancing transportation services, efficiently accommodating both passengers and goods [1]. An important aspect involves the development of multimodal transportation that

 doi:10.20944/preprints202411.0046.v1

combines different methods of transport into a unified system. This integration enhances mobility, traceability, and coordination. In this context, blockchain technology emerges as a solution offering a transparent platform that boosts cooperation among various service providers and users in the transportation industry.

The goal of this research is to explore the development and implementation of a blockchain-enabled MaaS system. This study specifically examines whether blockchain technology can improve the security, transparency, and operational efficiency of urban mobility services, with a particular focus on payment and ticketing systems. The core hypothesis is that blockchain's decentralized structure will enhance transaction integrity and stakeholder cooperation, creating a more efficient and secure urban transportation ecosystem. These systems use technology designed to transport people but can also handle shipping and moving goods. A key focus is on how blockchain technology can change payment and ticketing methods by enabling transactions between users and different service providers—such as buses, trains, bike-sharing programs, shipping companies, and ride-hailing services. By integrating this system into a city's infrastructure, it not only enhances user convenience but also helps support broader objectives like sustainability and better data security.

With the increasing need for secure, transparent, and efficient urban transport systems, various initiatives are exploring how blockchain can optimize MaaS platforms. A notable example is the European Union's DELPHI project, which is part of Horizon Europe's efforts to advance urban mobility and logistics, using innovative technologies. This initiative aims to build a unified transport system that integrates multiple modes of transport, both for passengers and freight. By incorporating blockchain, DELPHI seeks to boost data security, streamline collaboration among stakeholders, and refine transport logistics. The project's research focuses on establishing a blockchain-enabled platform that securely manages data across different transport modes. A key aspect is integrating payment and ticketing systems within urban mobility, which contributes to the development of the secure, transparent, and efficient MaaS framework that DELPHI envisions [2].

To fully understand blockchain's potential in MaaS, it is important to examine both the general landscape of blockchain technology and the current challenges facing its integration into mobility systems. Issues such as scalability, security, and interoperability remain critical barriers, yet blockchain's core features—decentralization, transparency, and immutability—position it as a promising solution for overcoming these obstacles.

The main part of the paper examines the design and structure of a MaaS system that uses blockchain technology. This section will delve into the technical aspects of the blockchain solution and its data model. We will also address challenges related to scalability to ensure the system's technical robustness.

MaaS systems face several challenges, including security vulnerabilities, scalability issues, and inefficiencies. Our proposed blockchain-based platform directly addresses these concerns by creating a decentralized, secure, and scalable solution for urban mobility. Utilizing Hyperledger Fabric's permissioned architecture and modular consensus mechanisms, the platform significantly boosts data security and transaction speeds, delivering reliable performance even under varying loads. Initially, the platform will focus on passenger transport, but future research will extend its application to freight logistics. This aligns with the DELPHI project's broader objective of developing a unified and efficient transportation ecosystem.

2. Related Work

2.1. Blockchain Technology

Blockchain started a proposal in 2008 [3] and was launched in early 2009. It operates as a public ledger, storing all committed transactions in a growing chain of blocks. Key traits of blockchain include decentralization, auditability, persistence, and anonymity. Blockchain uses advanced technologies: cryptographic hashes, asymmetric cryptography for digital signatures, and distributed consensus mechanisms. The use of these technologies supports the creation of decentralized transactions resulting in cost reductions and efficiency improvements [4].

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 November 2024 doi:10.20944/preprints202411.0046.v1

The blockchains' information is stored into a linked chain of blocks, where each block is linked to the previous one and secured through cryptographic methods. The peers in the network are linked through a P2P topology, thus ensuring the absence of a single point of failure. The network also uses a consensus mechanism to ensure universally agreed-upon transactions and blocks sequences. This maintains the consistency and the reliability of data across all nodes [5]. Every transaction on the blockchain is a discrete record in the blocks that make up this public ledger [6].

When smart contracts first emerged, as an integral part of blockchain technology, the second milestone in the blockchain began. This is referred to as Blockchain 2.0. Smart contracts are computer programs that run inside the blockchain network. These programs can express triggers, conditions, and business logic, enabling complex programmable transactions [7]. A smart contract was first mentioned under this name by computer scientist and cryptographer Nick Szabo long before the blockchain concept was invented, see [8].

As the most well-known application of blockchain technology, Bitcoin is often mistakenly considered synonymous with blockchain itself. Despite that, the utility of blockchain extends beyond cryptocurrencies. The main trait of blockchain technology is the ability to facilitate transactions without relying on intermediaries. This means it can be used across various financial services such as: online payments, digital assets, and remittances [9]. Smart contracts significantly expanded the range of industries that can benefit from this modern technology. Some blockchain new use cases beyond digital finance are logistics, supply chain management, IoT, healthcare, personal identity systems, music royalty processing, anti-money laundering, voting, agriculture, and many others [10–12].

Blockchain stands ready to revolutionize other applications also, such as art sales through digital media transfer, supply chains, delivery of remote services like tourism and travel, and innovative platforms shifting computing towards data sources and distributed credentialing [13].

2.2. Blockchain in Public Transportation

References [14,15] explore the idea of MaaS and its potential to reshape contemporary transportation. The emphasis is on transitioning from owning private vehicles to utilizing a range of transportation options offered as services. It is emphasized that such a system could also accommodate and streamline goods transportation. MaaS combines different transport services into one unified, user-friendly platform. Blockchain technology can improve MaaS by offering a secure and transparent way to manage transactions and data across various transport modes. Additionally, the use of smart contracts and tokenization facilitates smoother integration and incentivizes users within the MaaS framework.

Application of blockchain technology in implementing MaaS platform is explored in [16], with focus on performances under various load scenarios. Authors explore and analyze two well know and mature blockchain technologies: Ethereum platform based on permissionless technology and Hyperledger Fabric platform which is based on permissioned technology. Ethereum offers decentralized consensus and broad developer support, but its permissionless nature can lead to scalability issues and slower transaction speeds, making it less suitable for high-throughput environments like urban mobility systems. In contrast, Hyperledger Fabric provides a permissioned network with customizable consensus mechanisms, better suited for environments requiring higher control, privacy, and performance, as found in MaaS systems. While both platforms are capable of supporting MaaS, Hyperledger Fabric was chosen for its modularity, scalability, and enterprise-grade security features.

In [17], the authors propose a different method, utilizing the Hyperledger Indy [18] blockchain to create a multi-layered framework for a Smart Mobility Data Market (BSMD). This approach tackles major concerns like privacy, security, management, and scalability in the context of smart mobility. They point out the weaknesses of conventional data storage methods, such as central servers, which are prone to hacking and unauthorized data sharing. The BSMD framework offers a secure and transparent way for participants to share encrypted data on the blockchain, with transactions governed by mutually agreed-upon terms.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 November 2024 doi:10.20944/preprints202411.0046.v1

Enescu et al. [19] highlight the challenges and viable solutions related to the key subsystems of public transportation services. It is particularly emphasized the effects of these services in densely populated urban areas. The study focuses on exploring blockchain-based innovations to enhance existing management platforms and drive future developments in the sector.

Multiple academic research works explore and propose implementations of blockchain technologies in various aspects of public transportation, like ticketing systems, supply chain management, data sharing and integration, and identity management.

The ticketing system can be streamlined by enabling secure, peer-to-peer transactions. Ticket issuance and ticket validation can be automated by leveraging smart contracts, reducing fraud and operational costs [20].

Public transport operators can use blockchain for tracking their assets (like vehicle fleet, parts, and components) throughout their lifecycle, ensuring accountability and efficiency in maintenance and operations [21].

Secure data sharing is crucial for modern smart cities transportation systems. Blockchain can secure this process among different transportation agencies. resulting in better coordination and integration of services. This can lead to improved route planning and real-time updates for passengers [22].

Blockchain can provide secure digital means of user identification, allowing for seamless access to various transportation services without the need for multiple accounts or credentials [23].

There are several cities and organizations which have initiated pilot projects that intend to explore the benefits of blockchain in public transportation:

It is known that Dubai has the ambition to become a global leader in blockchain technology and smart transportation. The Dubai Roads and Transport Authority (RTA) is exploring blockchain for service efficiency and customer experience. It also tries to promote transparency in operations. RTA's aim is to streamline various processes, such as: vehicle registration, maintenance records, and driver licenses [24].

Singapore's Land Transport Authority (LTA) collaborated with TransitLink to introduce a ticketing and fare management system utilizing blockchain technology. The system was designed to improve both efficiency and transparency. Commuters were able to use a mobile application, or a contactless smart card powered by blockchain to pay for transportation across different modes. Fare validation becomes automated, enhancing the system’s connectivity, accountability, and transparency [25].

Estonia has implemented a smart ticketing system using blockchain technology to make public transportation smoother. Passengers can easily use the services of multiple transportation providers with a digital wallet, which securely keeps their tickets and payment information on a blockchain [26].

Denver's Regional Transportation District (RTD) implemented a pilot program for public mobility, which uses blockchain. "Go Denver", the application that they develop integrates various transit modes (buses, light rail, and ridesharing services) into one platform. Commuters can use it to plan, book, and pay for trips in one place [27].

The Seoul Metropolitan Government has incorporated blockchain technology into its public transportation systems as part of its ongoing smart city efforts. This technology boosts the security and transparency of fare payment systems by decentralizing and securing commuter data. Blockchain ensures that sensitive payment data is tamper-resistant and protected. By adopting blockchain, Seoul aims to increase trust and efficiency in public services, especially in transportation, where secure and transparent fare management is crucial. Seoul has also developed a cryptocurrency payment platform for public transportation (S-Coin), with plans to extend its use to other public services [28–30].

2.3. Blockchain Adoption Challenges

As blockchain technology continues to integrate into public transportation systems, several challenges must be addressed: scalability concerns, interoperability between different platforms, evolving regulatory landscapes, high initial costs, and the need for user education and acceptance.

Handling the high transaction volumes typical in public transportation systems can be a problem for most blockchain solutions [31]. The ever-increasing number of passengers requires speed in transaction processing that is above what is currently available across the existing blockchain networks.

At the same time, various transportation agencies can use different platforms that are not compatible (not necessarily blockchain-based solutions). This failure to standardize the systems can prohibit the easy sharing of information and services [32].

The regulations about blockchain technology are still evolving. Public transportation groups might have legal issues with data privacy, security, and following current laws [33]. Dealing with these rules can delay the adoption of these modern technologies.

Even though blockchain has the potential to save a lot in operational costs over time, the initial investment that it requires in the development of technology, upgrading infrastructure, and training are high. This could also deter smaller agencies from adopting blockchain solutions [34].

It is known that blockchain-based ticketing and transportation systems will only work successfully once users accept the recent technologies. For that reason, training and support will be much more necessary and overall obligatory, increasing the overall cost of implementation [35].

3. Platform Design and Implementation

3.1. Functionality Overview

The primary goal of the proposed MaaS platform is to consolidate payments made to transport companies in a city into one application. This integration also allows the platform to track ride details and generate useful statistics for both transport operators and passengers. Figure 1 illustrates the main use cases of this solution.

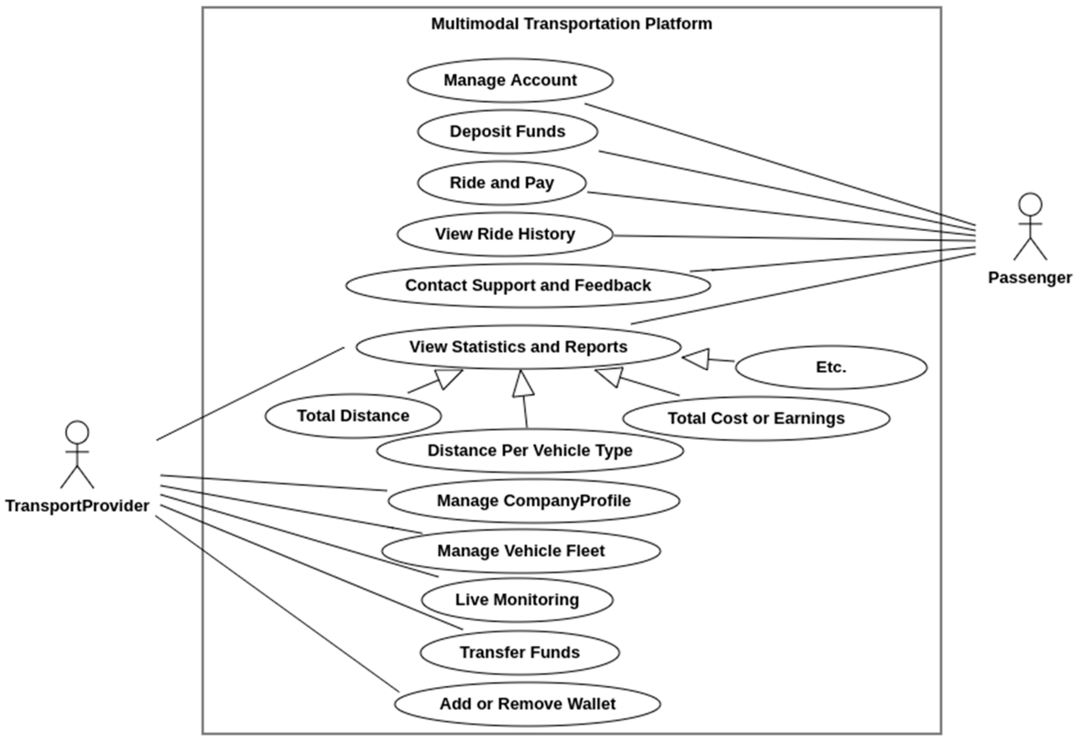

Figure 1. The use-case diagram of the proposed platform.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 November 2024 doi:10.20944/preprints202411.0046.v1

The proposed platform's actors include transportation providers or companies, as well as entities that use the network to commute, such as individuals or items that can be conveyed by public transportation. Because these two user classes have access to different features, they will interact with the platform via various interfaces or client applications.

The user's application employs a secure virtual wallet where passengers can deposit funds via external payment providers. This wallet facilitates internal payments within the platform, eliminating the need for cash or multiple tickets. Once the vehicle is boarded, the passenger will pay by transferring the due amount to the transport provider's wallet. External payment providers are used for processing deposits and withdrawals.

Passengers can view real-time information about their current rides, including route progress and estimated time of arrival. The app also maintains a history of past rides for reference. The user application also allows passengers to rate their experiences, provide feedback, and access customer support for any issues encountered.

Both types of users will be able to see statistics such as: distance per time period, earnings, or costs per time period, distance per vehicle, distance per vehicle type, etc. For example, providing the passengers with data regarding the vehicle type can encourage them to favor companies with a more environmentally friendly fleet for future rides. These insights can also assist providers in making informed decisions about fleet management and service improvements.

Like passengers, transport providers have one or multiple virtual wallets where they receive payments for services rendered. They can track earnings, process withdrawals, and manage financial transactions securely.

The providers can monitor active rides involving their vehicles, manage ride requests, and dispatch vehicles accordingly to optimize service delivery.

3.2. The System's Architecture

Figure 2 illustrates the system's architecture, highlighting its main components. These include the client applications, the platform's back-end, and the distributed ledger or blockchain network.

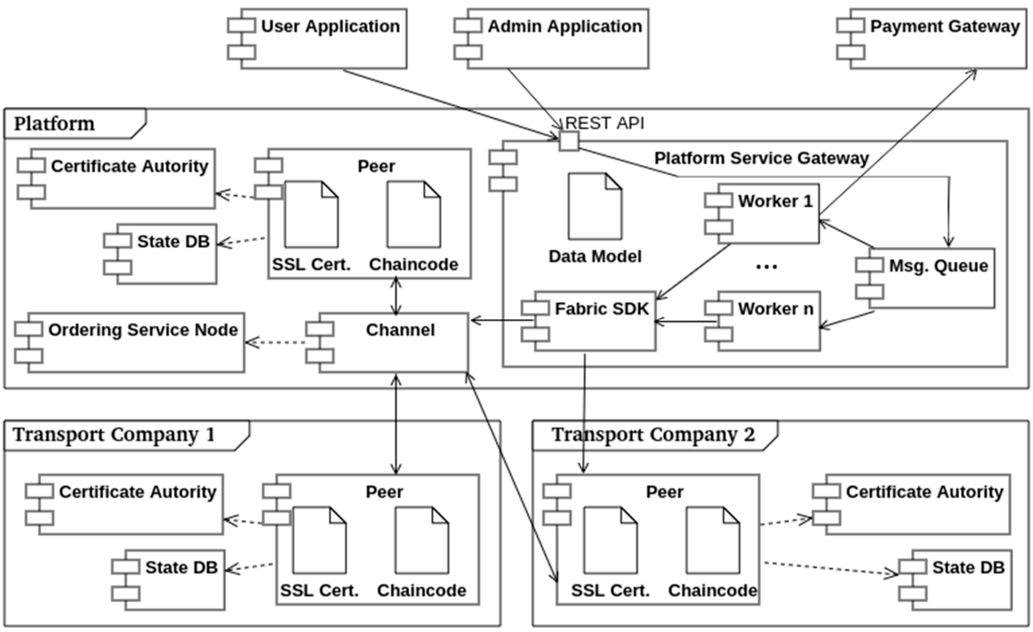

Figure 2. The architecture of the system.

3.2.1. Client Applications.

The system offers two distinct client applications: one designed for passengers and the other for transport providers. These apps enable users to engage with the platform's key features, as outlined in the previous section, such as managing virtual wallets, processing payments, and viewing ride-related statistics.

By offering these tailored applications, the system ensures that both passengers and transport providers have the tools they need to efficiently interact with the platform, improving the user experience and supporting more sustainable urban mobility practices.

3.2.2. Blockchain Network.

The platform's persistence layer is entirely built on blockchain. Multiple organizations maintain copies of both the state and the ledger databases, ensuring data immutability and security. Replicating the data across various network nodes enhances resilience against single points of failure. The blockchain solution used is Hyperledger Fabric.

The proposed architecture includes three participating organizations within the network. The *Platform* acts as the legal entity overseeing the system, while *Transport Company 1* and *Transport Company 2* serve as separate transport providers. Although participation is optional, it is strongly recommended that all transport providers actively engage in the network to help maintain the integrity and security of the shared data. As shown in the previous diagram, each organization operates its own peer, state database, and certificate authority.

The smart contracts or programs that define the business logic governing transactions on the blockchain network are called *Chaincode*. It is the code that specifies the rules and procedures for interactions within the network, enabling the execution of transactions and the manipulation of the ledger's state.

A *Peer* is a node in the network that holds a copy of the ledger and smart contracts (chaincode). Peers are vital to the network, as they endorse transactions, maintain the ledger, and take part in the consensus process.

The *Certified Authority (CA)* is responsible for user certificate management duties such as enrollment and revocation. Hyperledger Fabric is a permissioned blockchain network that uses X.509 standard certificates to establish rights, roles, and attributes for individual users. Users can execute queries or initiate transactions using channels that correspond to their permissions and roles.

The state database (*State DB*) refers to the database responsible for storing the current state of the ledger. The ledger in Fabric maintains a record of all transactions that have occurred, and the state database keeps track of the current state of all assets or data within the blockchain network. The state database functions as a key-value store that tracks the status of the ledger, storing the present values of all ledger assets and the active state of any deployed smart contracts on the platform. Fabric offers two options for the peer state database. Hyperledger Fabric supports two peer state database options. The default state database integrated into the peer node is LevelDB. This option stores data as simple key-value pairs and supports queries by key, key range, and composite keys. Alternatively, CouchDB can be configured as an optional state database, allowing data to be modeled as JSON on the ledger and enabling rich queries based on data values, rather than limiting retrieval to key-based searches. CouchDB allows indexes to be deployed within the chaincode package, improving query efficiency and enabling rapid searches across large datasets [36].

A *Channel* is a private and confidential communication line between specific network members, used to segregate transaction data and enable secure communication within a subset of the network.

The *Ordering Service Node*, or orderer, is essential for managing the process of grouping validated transactions into blocks, ensuring consensus, and adding these blocks to the distributed ledger.

3.2.3. Platform's Backend.

The system's backend is built using *Java* with the *Spring* framework, which was selected for its robust enterprise features and wide adoption in scalable applications [37]. Java's platform independence and extensive library support make it ideal for handling complex business logic and high-performance environments like MaaS. The Spring framework was chosen due to its flexibility, lightweight nature, and built-in support for creating secure, scalable REST APIs [38], which are essential for interfacing with both client applications and the Hyperledger Fabric blockchain. In

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 November 2024 doi:10.20944/preprints202411.0046.v1

Hyperledger Fabric, direct access to peers from the client application is restricted. The backend also utilizes a messaging queue and a payment gateway to support its operations.

The platform's backend exposes a *REST API* based on JSON, allowing communication with client applications.

The *Java Fabric SDK* [39] was used to link the API with the Blockchain network.

Basic operations, such as ride payments or balance inquiries, are directly managed by the API through the Fabric SDK. However, more complex tasks, like generating statistics, are offloaded using a *Messaging System* (RabbitMQ). When such a request is made, the API logs it into the messaging service, and a worker processes the message once available. The messaging queue distributes tasks among multiple workers, enabling load balancing and ensuring scalable, efficient resource use while avoiding bottlenecks [40].

3.2.4. Payment Gateway

The *Payment Gateway* is a vital component of the system architecture, allowing for easy connection with external payment providers. It connects the MaaS platform with financial institutions, allowing users to make safe fund deposits, payments, and withdrawals. The gateway enables real-time transaction processing, guaranteeing that all payments are confirmed and authorized using known financial procedures such as PCI-DSS (Payment Card Industry Data Security Standard).).

Furthermore, to ensure secure and smooth transactions, the gateway must encrypt sensitive payment data during transfer, minimizing the risk of fraud and unauthorized access. It also manages tokenization processes to replace sensitive card information with a unique identifier, ensuring user privacy and security standards compliance.

3.3. Data Model

Figure 3 provides a comprehensive illustration of the data model stored within the blockchain. This figure delves into the blockchain's data architecture.

TransportProvider and *Passenger* store the user details for transport companies and transportation users, respectively.

The *Vehicle* entity is responsible for storing data related to the companies' fleet. It includes three *VehicleTypes*: electric vehicle, internal combustion, and alternative fuel.

The *Wallet* entity is linked to an owner, which can be either a company or an individual user and manages its balance. When a *deposit* or a *withdrawal* takes place, the balance is updated, and a corresponding transaction record is created. Similarly, when a payment is made, the balances of both involved wallets are adjusted, and the transaction is documented.

The *Ride* entity records comprehensive details for each transportation instance, including the user utilizing the service, the vehicle employed, the date of the ride, the starting and ending locations, the cost incurred, and the distance traveled. Additionally, multiple connected *Ride* records are grouped together to create a *Trip*, which represents a series of linked rides within a single journey.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 November 2024 doi:10.20944/preprints202411.0046.v1

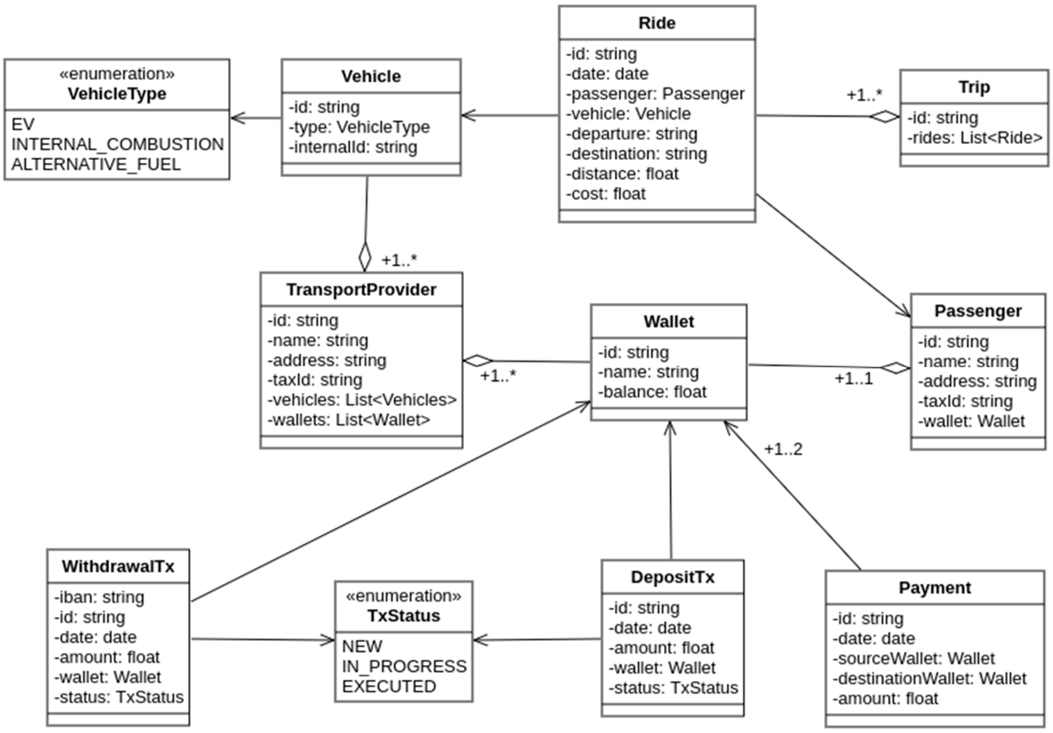

Figure 1. The data model of the platform.

3.4. Chaincode Implementation

Figure 4 presents the chaincode structure for the proposed solution, which was implemented in Node.js.

Hyperledger Fabric Blockchain was chosen to implement the proposed proof of concept. The key features of this technology include a modular architecture (pluggable components), a flexible consensus mechanism (supporting both Crash Fault Tolerance (CFT) and Byzantine Fault Tolerance (BFT)), and a permissioned network that allows only authorized participants to join. Additional advantages are its scalability, high performance, private transactions, and channel-based communication that enables confidential interactions between specific participants. Hyperledger Fabric also offers robust identity management and enterprise-focused features, addressing concerns related to compliance, regulatory requirements, and auditability. These features make Hyperledger Fabric a preferred choice for enterprises and consortiums deploying permissioned blockchain networks. It is particularly employed in industries like supply chain management, finance, and healthcare, where privacy, scalability, and customizability are essential [41].

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 November 2024 doi:10.20944/preprints202411.0046.v1

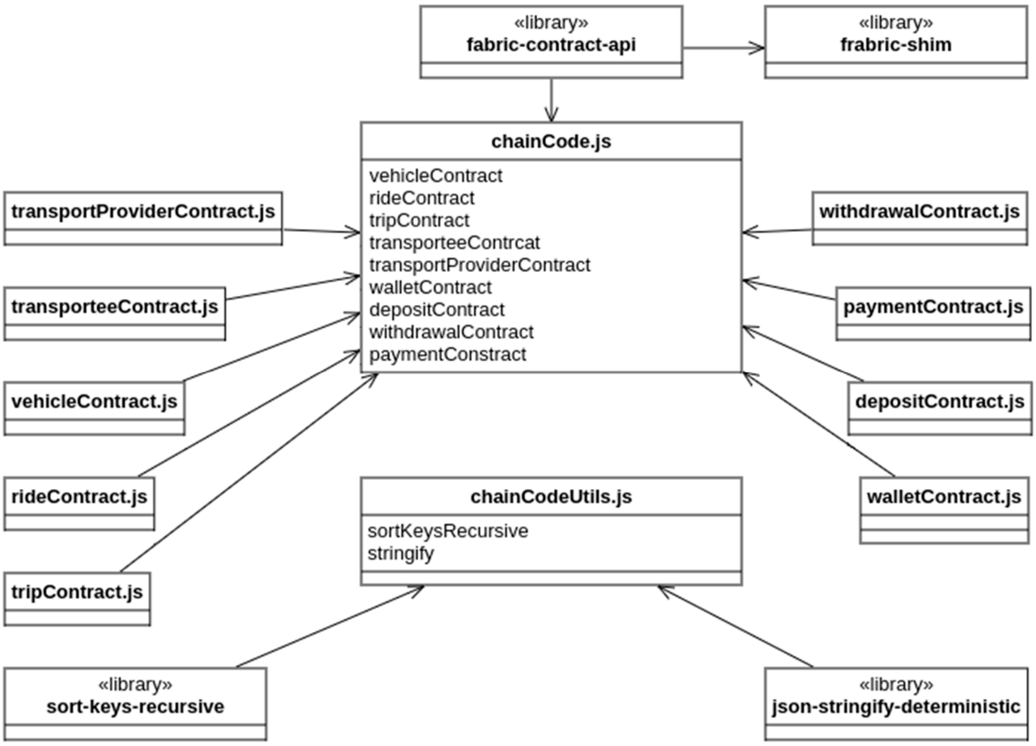

Figure 4. The chaincode module dependency diagram.

The Hyperledger Fabric chaincode is equivalent to smart contracts found in other blockchain platforms. It establishes the rules and logic that govern transactions within the network. Chaincode, which can be written in Go, Node.js, or Java, is a program that follows a predefined interface (API) provided by the *fabric-contract-api* library. Another key library for developing and deploying chaincode is *fabric-shim*, which provides the necessary implementation to enable communication between smart contracts created with *fabric-contract-api* and Hyperledger Fabric peers.

The business logic within the chaincode was implemented across multiple modules. Each module primarily implements the available operations for one of the entities defined in the data model (e.g., *vehicleContract.js* for *Vehicle*, *rideContract.js* for *Ride*, *walletContract.js* for *Wallet*, etc.). These are referred to as domain logic modules.

The *chaincodeUtils.js* module is utilized by the domain logic modules, as it offers a generic implementation for operations such as *create*, *read*, *update*, *delete*, *getAll*, *saveAll*, *exists*, and *getBySelector*, which is used to execute complex queries.

The complete business logic is made accessible externally through the facade module *chainCode.js*. This module consolidates all the public operations implemented by the domain logic modules and it is implemented as a class that inherits from the *Contract* class provided by the *fabric-contract-api* library.

4. Tests and Results

The test environment was deployed on the following configuration: Intel® Core™ i7-8565U CPU @ 1.80GHz × 8, 16 GB RAM, Ubuntu 24.04.1 LTS.

The application's backend was functionally tested using Open API v3.0.1, which was specifically set up to interface with the platform's REST API.

To understand how response times are influenced by the number of simultaneous requests and the volume of blockchain records, a comprehensive series of load tests were performed. It is important to note that during these evaluations, the entire system — including both the backend and the blockchain network — was hosted on a single machine. Consequently, the performance measurements obtained are relative and may vary once deployed in a production environment. These

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 November 2024 doi:10.20944/preprints202411.0046.v1

load tests were primarily aimed at assessing how response times fluctuate based on the intensity of concurrent requests and the size of the blockchain data. Apache JMeter 5.6.3 was used for implementing the load tests.

The tests assess two standard operations -getting the total monthly spending and retrieving the wallet by ID - that users can access through the client applications. The state database of Fabric is CouchDB, and an index was added for the second operation on its search criteria: [sourceWalletId, date]. Integrating the CouchDB index into the chaincode requires placing a JSON file in the chaincode's main directory at the path: META-INF/statedb/couchdb/indexes. The content of the index is shown below:

```
{
  "index": {
    "fields": [
      "sourceWalletId",
      "date"
    ]
  },
  "ddoc": "indexSourceWalletIdDate",
  "name": "indexSWalletDate",
  "type": "json"
}
```

Following the addition of 10,000, 100,000, and 1.000,000 fictitious data records to the blockchain, respectively, the tests have demonstrated that, provided the records are indexed by the search criteria, response times are independent of the volume of data stored. The average blockchain response time for each number of concurrent requests is shown in Table 1.

Table 1. Load test results.

Number of requests	Get wallet by ID time [ms]	Total cost per month time [ms]
1	5	12
10	13	29
50	65	103
100	131	160
200	370	410
300	497	640
400	716	843

At around 400 parallel requests, the response error rate (timeouts) averaged around 10% for both operations. This request limit can serve as a reference point for adjusting the number of workers in the messaging queue.

Table 2 shows the average response time (for 100 requests with 1 thread) for the primary operations supported by each entity in the data model, including *Vehicle*, *Passenger*, *Ride*, and others.

Table 2. The temporal performances of main operations.

Preexistent number of records	Get record by name - no index [ms]	Get record by ID [ms]	Update record [ms]	Delete record [ms]
1000	45	4	2123	2013
5000	152	4	2210	2101
10 000	312	5	2175	2175
50 000	1453	4	2187	2141
100 000	2645	4	2256	2123
1 000 000	24987	6	2187	2217

As shown in the table above, the performance of basic CRUD operations is not influenced by the number of existing blockchain records. Efficient retrieval by id is ensured through implicit indexing of assets. Create, update, and delete operations consistently take about 2 seconds each, as they involve blockchain state modifications, which include communication and validation processes. In contrast, retrieving an asset by a different attribute (e.g., name) encounters substantial delays, which are proportional to the size of the dataset. This issue can be mitigated by indexing assets by name, improving retrieval efficiency.

5. Conclusions

This study investigates the integration of blockchain technology into smart mobility systems, specifically focusing on Mobility-as-a-Service (MaaS). It addresses the effectiveness of blockchain in improving the efficiency, safety, and environmental performance of urban transport systems. Our results suggest that blockchain, as a decentralized, transparent, and secure solution, can resolve trust and integration issues between different transport technologies and stakeholders.

The Hyperledger Fabric-based design has proven to be highly effective at managing high request volumes, which makes it a viable option for networked transport systems. The study also highlights how blockchain might improve safety and transparency by streamlining ticketing and payment procedures. Additionally, we investigated the main technological features of a blockchain-based MaaS platform, paying particular attention to its data types and system architecture.

Although the single-machine setup used for pilot testing offered a cost-effective solution for initial testing, load tests exposed its limitations in scalability and resilience when managing higher concurrent requests. Performance began to decline once the system handled 400 simultaneous requests, making it clear that transitioning to a cloud-based, distributed architecture is essential for real-world applications. This change would allow the system to dynamically scale for larger user volumes, with load balancing and caching mechanisms enhancing response times during peak periods. Additionally, data management strategies such as pruning and sharding will ensure long-term efficiency as the system grows, making it suitable for large-scale deployments in urban settings.

To verify the scalability of these systems in distributed cloud environments, future research should give priority to pilot projects and field experiments. Additionally, it is necessary to investigate how blockchain technology might help integrate freight logistics into MaaS, guaranteeing safe and open coordination between passenger and freight services. With the further expansion of these systems, it will also be crucial to optimize blockchain transaction rates and minimize latency in real-time urban transit applications.

Author Contributions: Conceptualization, R.M. and M.H.; methodology, V.M.; software, R.M.; validation, V.M. and M.H.; formal analysis, M.H.; investigation, V.M.; writing—original draft preparation, R.M.; writing—review and editing, M.H and V.M.

Funding: This research received no external funding.

Data Availability Statement: The data from this study is available on request from the corresponding author.

Acknowledgments: This research has been conducted as part of the DELPHI project, which is funded by the European Union, under grant agreement No 101104263. Views and opinions expressed are only those of the author(s) and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Orecchini, F., Santiangeli, A., Zuccari, F., Pieroni, A., Suppa, T.: Blockchain Technology in Smart City: A New Opportunity for Smart Environment and Smart Mobility. In: Vasant, P., Zelinka, I., Weber, G.-W. (eds.) Intelligent Computing & Optimization, Advances in Intelligent Systems and Computing. Springer, Cham (2019).
2. DELPHI Homepage, https://delphi-project.eu/, Accessed on 23 Feb 2024.

3. Nakamoto, S.: Bitcoin: A Peer-to-Peer Electronic Cash System, https://bitcoin.org/bitcoin.pdf, Accessed on 1 Jan 2024.
4. Zheng, Z. et al.: Blockchain challenges and opportunities: a survey. Int. J. Web and Grid Services 14(4), 354–355 (2018).
5. Yuan, Y., Wang, F.Y.: Blockchain and cryptocurrencies: Model techniques and applications. IEEE Trans. Syst. Man Cybern. Syst. 48(9), 1421–1428 (2018).
6. Tschorsch, F., Scheuermann, B.: Bitcoin and beyond: A technical survey on decentralized digital currencies. IEEE Commun. Surveys Tuts. 18(3), 2084–2123 (2016).
7. Xu, X. et al.: The blockchain as a software connector. Proc. 13th Working IEEE/IFIP Conf. Softw. Archit. (WICSA), 182–191 (2016).
8. Szabo, N.: Smart Contracts: Building Blocks for Digital Markets, http://www.fon.hum.uva.nl/rob/Courses/Information-In-Speech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html, last accessed Accessed on 23 Feb 2024.
9. Peters, G., Panayi, E., Chapelle, A.: Trends in cryptocurrencies and blockchain technologies: A monetary theory and regulation perspective. J. Financial Perspect. 3(3), 1–25 (2015).
10. S. Manski, S., Building the blockchain world: Technological commonwealth or just more of the same?. Strategic Change. pp. 511– 522 (2017).
11. Christidis, K., Devetsikiotis, M., Blockchains and Smart Contracts for the Internet of Things, IEEE Access, vol. 4, pp. 2292-2303 (2016).
12. Monrat, A.A., Schelén, O., Andersson, K.: A Survey of Blockchain From the Perspectives of Applications, Challenges, and Opportunities. IEEE Access 7, 117134–117151 (2019).
13. Casino, F., Dasaklis, T.K., Patsakis, C.: A systematic literature review of blockchain-based applications: Current status classification and open issues. Telematics Inform. 36, 55–81 (2019).
14. Bothos, E., Magoutas, B., Arnaoutaki, K., Mentzas, G.: Leveraging Blockchain for Open Mobility-as-a-Service Ecosystems. In: IEEE/WIC/ACM International Conference on Web Intelligence - Companion Volume, ACM, Thessaloniki Greece (2019).
15. Nguyen, T. Partala, J., Pirttikangas, S., "Blockchain-Based Mobility-as-a-Service," 2019 28th International Conference on Computer Communication and Networks (ICCCN), Valencia, Spain, pp. 1-6 (2019)
16. Nguyen, T., Nguyen, H., Partala, J., Pirttikangas, S.: TrustedMaaS: Transforming trust and transparency Mobility-as-a-Service with blockchain. Future Generation Computer Systems 149, 606–621 (2023).
17. López, D., Farooq, B.: A multi-layered blockchain framework for smart mobility data-markets. Transportation Research Part C: Emerging Technologies 111, 588–615 (2020).
18. Indy Homepage, https://www.hyperledger.org/projects/hyperledger-indy, Accessed on 5 Jan 2024.
19. Enescu, F.M., Birleanu, F.G., Raboaca, M.S., Bizon, N., Thounthong, P.: A Review of the Public Transport Services Based on the Blockchain Technology. Sustainability 14(20):13027 (2022).
20. Sree, V., et. al., Ticketing System Using Blockchain Technology . International Journal of Innovative Research in Science, Engineering and Technology, 13(4), 5315-5321 (2024).
21. Terekhov, V., How to Implement Blockchain for Secure Transportation Transactions (2024). https://attractgroup.com/blog/how-to-implement-blockchain-for-secure-transportation-transactions. Accessed on 14 Sept 2024.
22. Jiang, S., et. al., Privacy-preserving and efficient data sharing for blockchain-based intelligent transportation systems, Information Sciences, 635, 72-85 (2023).
23. Stockburger, L.,et. al., Blockchain-enabled decentralized identity management: The case of self-sovereign identity in public transportation, Blockchain: Research and Applications, 2(2),100014 (2021).
24. Roads and Transport Authority (RTA), Al Masar Magazine, 142 (2020). https://www.rta.ae/links/magazine/masar/Al_Masar_142_Eng.pdf. Accessed on 14 Sept 2024.
25. Valavan, A. Blockchain-Powered Transit: Transforming Public Mobility Across Metro, MRTS, Railways and Buses. Metro Rail Today (2024). Accessed on 14 Sept 2024.
26. Semenzin, S., Rozas, D., Hassan, S., Blockchain-based application at a governmental level: disruption or illusion? The case of Estonia, Policy and Society 41(3), 386-401 (2022).
27. Denver's RTD, Transit Partner to Simplify Trip Planning, Fare Payment, Metro Magazine (2024), https://www.metro-magazine.com/10219415/denvers-rtd-transit-partner-to-simplify-trip-planning-fare-payment, Accessed on 14 Sept 2024.
28. Kim, S. et al. Sampling Blockchain-enabled Smart City Applications Among South Korea, the United States and China. Journal of Smart Cities and Society, 1(1), 53-70 (2022).
29. Yoo, Y. Toward Sustainable Governance: Strategic Analysis of the Smart City Seoul Portal in Korea. Sustainability. 13(11), 5886 (2021).
30. Hong, J., Tamakloe, R., Lee, S., Park, D., Exploring the Topological Characteristics of Complex Public Transportation Networks: Focus on Variations in Both Single and Integrated Systems in the Seoul Metropolitan Area. Sustainability. 13(16), 9087 (2021).

31. Zheng, Z., et al., Blockchain Challenges and Opportunities: A Survey. International Journal of Web and Grid Services, 14(4), 352-375 (2018).
32. Čuš-Babič, N.; Guerra De Oliveira, S.F.; Tibaut, A. Interoperability of Infrastructure and Transportation Information Models: A Public Transport Case Study. Appl. Sci. 2022, 12, 6234.
33. Böhme, R., et. al., Bitcoin: Economics, Technology, and Governance, Journal of Economic Perspectives 29(2), 213-238 (2015).
34. Upadhyay, N., Demystifying blockchain: A critical analysis of challenges, applications and opportunities, International Journal of Information Management 54, 102120 (2020).
35. Makhdoom, I., et. al., Blockchain for IoT: The Challenges and a Way Forward. IEEE Communications Surveys & Tutorials, 21(3), 2526-2552.
36. "Hyperledger Fabric Docs – Using CouchDB". https://hyperledger-fabric.readthedocs.io/en/latest/couchdb_tutorial.html. Accessed on 10 Sept 2024.
37. Blinowski, G., Ojdowska, A., Przybyłek, A., Monolithic vs. Microservice Architecture: A Performance and Scalability Evaluation. IEEE Access, vol. 10, pp. 20357-20374 (2022)
38. Walls, C. Spring in Action (6th ed.). Manning Publications. ISBN-13: 978-1617297571. (2022).
39. Hyperledger Fabric Gateway SDK for Java Homepage, https://hyperledger.github.io/fabric-gateway-java/, Accessed on 22 Feb 2024.
40. Pathak, A., Kalaiarasan, C.: RabbitMQ Queuing Mechanism of Publish Subscribe model for better Throughput and Response. In: 2021 Fourth International Conference on Electrical, Computer and Communication Technologies (ICECCT), Erode, India (2021).
41. Hyperledger Fabric Docs – Introduction, https://hyperledger-fabric.readthedocs.io/en/latest/whatis.html. Accessed on 10 Sept 2024.