International Research Journal of Multidisciplinary Sciences (Double-BlindPeerReviewJournal)VOL-2
ISSUE-4 April 2026 PP:60-80 ISSN:3107-930X(ONLINE)

A Comprehensive Blockchain Framework for Privacy-Preserving Shared Mobility Platforms

[1]Dr.UmarKhalidFarooqui,[2]RizwanAkhtar
[1]AssistantProfessor,AIIT,AmityUniversity,Lucknow,[2]Asst.Professor,Dept.ofComputer Application, Integral University
[1]uk.farooqui@gmail.com,[2]rizwanakhtar360@gmail.com

ABSTRACT-Volatility and speculation in the digital asset ecosystem have made capital preservation an urgent challenge for modern traders. While decentralized protocols challenge traditional financial structuresthroughamixofcryptographyanddistributedcom puting, their systemic vulnerabilities require a more rigorous analytical approach. This paper bridges the gap between blockchain architectural security and market technical analysis by establishing a novel trust-verification framework. Through a comparative structural examination of decentralized networks—focusing on Bitcoin's disintermediated consensus model—we isolate critical cryptographic risk vectors. We then operationalize these insights via an integrated valuation model that pairs asset macro-metrics, such as circulating market capitalization, with high-frequency time-series data. This approach tracks high/low timestamp anomalies, orderbook liquidity distribution, and moving average convergences normalized against the Relative Strength Index (RSI). Ultimately, our findings demonstrate that syncing underlying cryptographic validation with quantitative market indicators offers an optimized, risk-adjusted strategy for navigating speculative asset markets.

I. INTRODUCTION

1. Overview of Cryptocurrency and Risk Factors in Ride-Sharing

App-based transit systems have fundamentally reorganized urban transport by providing daily commuters with efficient alternatives to traditional car ownership.Sharedmobilitymodelsalleviatetraffic congestion by grouping passengers traveling along identical spatial corridors, thereby lowering individual transportation costs and curbing vehicular emissions. Critically, researchers distinguish between "ride-hailing"—where a passenger hires a dedicated vehicle for a direct, point-to-point trip—and "ride-sharing," which utilizes a dynamic, multi-passenger pooling structure requiring intermediate stops for co-riders. While leading Mobility-as-a-Service (MaaS) platforms like Uberand Lyft offer bothconfigurations,optimizing these pooled ride-sharing networks holds significant macroeconomic value. Indeed, advanced routing architectures can improve systemic productivity by reducing commuter travel times by up to 40% (Lanctot, 2017).

2ResearchQuestionsandObjectives

At its core, this study aims to rethink how ride-sharing platforms operate so that both drivers and passengersget a fairer, better experience. By looking closely athowtheseplatformsarebuilt,theresearchexploreshow modern decentralized tech might fix common industry pain points.

Tomapthisout,theprojectfocusesonfourkey areas:

- **Demystifying Blockchain in Practice:**Moving past the hype to analyze how blockchain actually functions day-to-daywithin real organizations, rather than just in theory.

- **Assessing Tech to Business Alignment:** Examining how different companies select and deploy specific technologies based on their unique operational demands.
- **Weighing Security and PrivacyFrameworks:** Comparing the architectural setups of major ride-sharing platforms to pinpoint which models do the best job of protecting sensitive user and driver data.
- **Measuring Real-World Impact on Ride-Hailing:** Investigating exactly how blockchain integration is currently shifting the landscape for ride-hailing applications, noting where it succeeds and where it falls short.

3MovingBeyondCentralizedRide-Sharing

The dominance of centralized ride-sharing platforms stems largely from their sheer convenience. Usersenjoy immense flexibility, whether they are booking a ride days in advance or hailing a car on demand. But this convenience comes at a steep cost to user trust. Centralizedarchitectures routinely failto guaranteedata privacy, meaning personal tracking information can be shared or commercialized entirely behind the scenes. Furthermore, these platforms operate as a "black box" regarding pricing; passengers and drivers are often leftin the dark about the exact breakdown of service fees, surges, taxes, and toll calculations.

Resolving these structural flaws requires a shift toward decentralization. To eliminate the middleman, the industry must look toward blockchain technology.

TheFoundationofDecentralizedArchitecture

The conceptual framework for blockchain originally emergedfromSatoshiNakamoto'sfoundationalworkon peer-to-peer electronic cash. While Nakamoto designed this distributed network specifically to power Bitcoin, the underlying mechanics proved capable of far more than just managing digital currency. At its core, a blockchain functions as an append-only distributed ledger,allowingeveryconnectedparticipanttoaccess and verify system data securely without a central authority.

While the majority of blockchain projects still focus on cryptocurrency, this peer-to-peer architecture isuniquely suited for logistical networks. By removing third-party intermediaries, a decentralized ledger can directly connect drivers and riders—and eventually, autonomous vehicle fleets—creating a fundamentally transparent and secure marketplace.

II. LITERATUREREVIEW

1. UnderstandingBlockchainTechnology

At its core, a blockchain functions as a specialized, chronological database organized as an unbroken sequence of cryptographically linked data blocks maintained across a distributed peer-to-peer network [16]. When new transaction data is generated, it is compiled into a fresh block; once that block reaches its storage capacity, it is permanently appended to the precedingblockusingacryptographichashpointer[24].
While a blockchain can theoretically house any digital asset, its most prominent real-world application is serving as a decentralized transaction ledger [10, 24]. Crucially, blockchain networks are immutable; oncedata has been verified and committed to a block, modifying or deleting historical entries becomes computationally impossible without invalidating the network's mathematical logic [16].

This decentralized architecture stands in stark contrastto traditional centralized client-server models. In a classic web setup, a central server houses all data in a single logical location. While this framework makes updating and managing data highly efficient for authorized administrators, it introduces critical single points of failure, administrative monopolies, and severe system vulnerabilities to external exploits [14].

2. NetworkArchitectureandOperationalGoals

In a blockchain framework, structural control is democratic and distributed rather than centralized.Every network participant, or node, has the authority to view,validate,andupdatedataaccordingtostrict

consensus rules [16, 24]. Because the responsibility of maintaining accurate records is shared across the entire collective, the network achieves robust security without relying on a trusted middleman or clearinghouse [10]. For modern organizations and enterprise systems, integrating thisdecentralizedarchitectureachievesthree primary strategic goals:

- **Auditable Data Provenance:** Unlike a centralized database that only displays a snapshot of current data states, a blockchain functions as an ever-expanding historical archive [24]. The complete lifecycle of any asset or transaction remains fully auditable at any point in time.
- **Operational Cost Reduction:** Maintaining massive, centrally owned databases—such as those run by traditional financial institutions or centralized corporate networks—requires capital-intensive security infrastructures to defend against targeted cyber threats. Decentralization naturally distributes this systemic risk across independent nodes, lowering administrative overhead and eliminating intermediary gatekeeper fees [10].
- **Data Integrity Over Raw Speed:** Because transaction validation relies on independent node networks solving cryptographic puzzles rather than a single fast server, processingtimes can be slower [24]. However, this architectural trade-off deliberately prioritizes absolute data security, authenticity, andtamper-resistance over raw execution speed.

3. ResearchBackground

a. EvolutionandTypologiesofDistributedLedgers

The practical era of distributed ledgers began in late 2008whenananonymous individual orgroup operating under the pseudonym Satoshi Nakamoto published the seminal whitepaper, *Bitcoin: A Peer-to-Peer Electronic Cash System* [16]. Bitcoin served as the first working proof-of-concept for blockchain technology, demonstrating that trust could be engineered cryptographically without institutional backing.

Blockchain Typology	**Access Control**	**Primary Architectural Benefits**	**KeyTrade-offs & Limitations**
Public [16, 24]	Fully Open/ Permissionless	High security,full decentralization,complete user anonymity.	Slower processing speeds;heavy computational landenergy demands.
Private[5,8]	Restricted / Permissioned	Extremelyfast transactions, highly scalable,stricti nternal data control.	Lacks true decentralization;acentral entitymakes final validation calls.
Consortium [24]	Multi-Organization/ Federated	Balanced transparency, highlysecure, optimized transaction time efficiency.	Less anonymous due to collective oversightby member entities.
Hybrid[11, 22]	Mixed Permissioned & Public	Flexibledata privacy, granular accesscontrol, tamper-proof logs.	Complex architectureto integrate; restrictednode ecosystem.

Table 1:Comparative Analysis of Blockchain Typologies

Structurally, blockchains rely on a combination oflinked lists and cryptographic hash pointers to organize data sequentially [24]. A hash pointer contains the cryptographicaddress of anotherpieceofdata, actingas a secure link. In a blockchain's linked list, each block holds a hash pointer referencing the exact block before it.Theveryfirstblockinthechain—knownasthe

genesis block—contains no previous hash. Data flows forward, block by block, until the most recent transaction is reached, where the final pointer remains open until a new block is mined.

As the underlying technology matured, different blockchain variations emerged to satisfy specific operational requirements, shifting from entirely public networks to private or hybrid models:

PublicBlockchains

Thesearecompletelypermissionlessnetworksopento any participant with an internet connection [16]. Networks like Bitcoin or Ethereum allow users to transact pseudonymously. To secure the ledger withouta central coordinator, public blockchains utilize global consensus mechanisms like Proof of Work (PoW) or Proof of Stake (PoS).The primary drawbackof a public chain is latency; validating transactions via global consensus requires significant time and computational power [24].

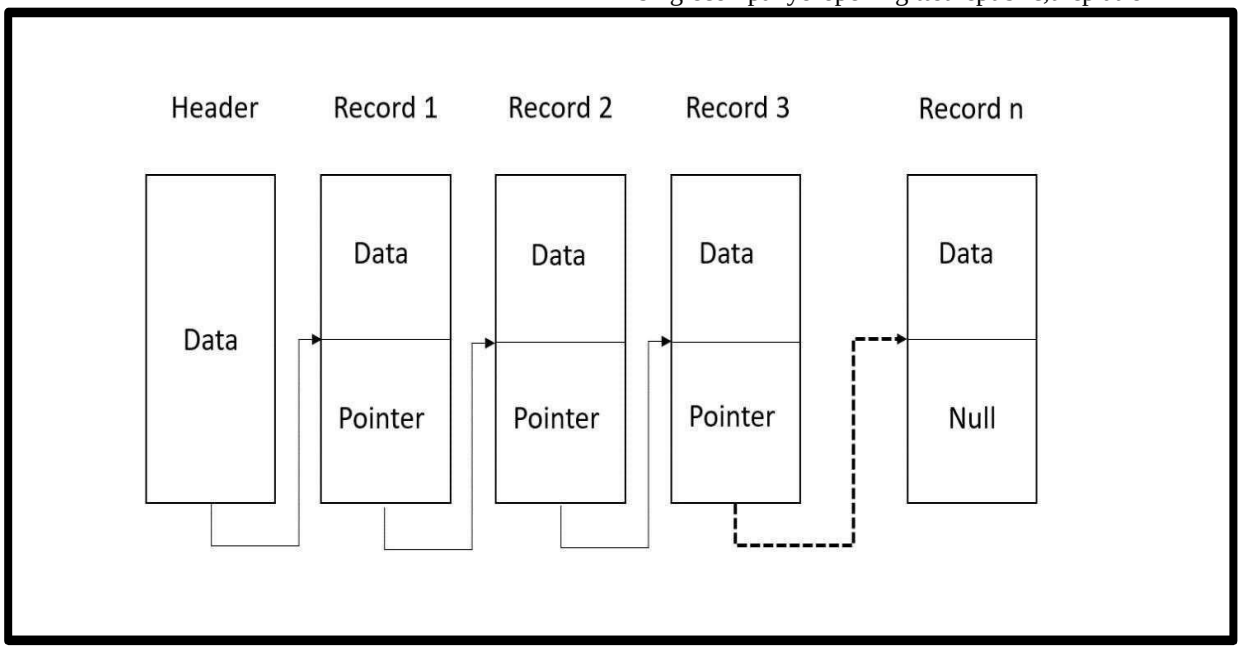


Figure1:BlockchainBasicStructure

PrivateBlockchains

Operating as restricted, permissioned networks, private blockchains require explicit authorization to join [8]. Theyareideallysuitedforinternalcorporate environments where a single entity must control access while utilizing ledger technology for immutable record-keeping. Because the number of participants is small, private chains process transactions rapidly. However, they lack true decentralization, meaning trust remains tiedtoacentralauthority,andfewernodescanmakethe network more vulnerable to targeted security breaches.

Consortium(Federated)Blockchains

A consortium blockchain bridges the gap betweenpublic and private models.Instead ofgiving control to a singlecompanyoropeningittothepublic,theplatform is governed by a select group of independent organizations [24]. Validator nodes within a consortium can both initiate and verify transactions. This structure offers excellent transparency and processing efficiency while maintaining decentralized governance, though it reduces user anonymity because the participating organizations control network access.

HybridBlockchains

Hybrid networks allow companies to run a private, permissioned ecosystem alongside a public, permissionless ledger [22]. By utilizing smart contracts to bridge the two environments, organizations can strictly control who sees sensitive internal data while anchoringtransactionproofstoapublicchainfor

immutable verification [11]. Hybrid systems are secure, scalable,andcost-effective,ensuringthatdatacannotbe altered even if the internal corporate environment is compromised.

b. TheEvolutionoftheSharingEconomy

While community asset-sharing has existed for centuries, the rise of the internet, mobile networks, and big data analytics transformed it into a global economic force [10]. Variously termed the peer economy, collaborativeconsumption,orthesharing economy, this modelallowsindividualstomonetize underutilized,idle assets [13].

The most prominent early success stories include Airbnb, which turned spare bedrooms into hospitality assets, and traditional ride-hailing platforms, which turned empty passenger seats into transport services [6, 7]. Today, this collaborative framework extends across bothconsumerand business-to-business(B2B)markets, driven primarily by three main pillars:

- **Co-working Platforms:** Decentralized, shared office spaces that provide flexibleinfrastructure for freelancers, startups, and remote corporate employees.
- **Peer-to-Peer (P2P) Lending:** Financial networks that connect borrowers directly with individual investors, offering competitive interest rates by cutting out commercial banks.
- **Collaborative Fashion Platforms:** Digital marketplacesthatfacilitate the resale,rental, or cyclical sharing of apparel and accessories.

c. SmartContracts

Smart contracts are self-executing lines of code deployed directly onto a blockchain network [24]. They are programmed to automatically execute specific actions the moment predefined criteria are satisfied. By shifting enforcement from human legal structures or corporate policies to deterministic code, smart contracts eliminatetheneedforthird-partyintermediaries, reducing transaction friction and administrative delays [10].

In a ride-sharing context, a smart contract can governthe entire transactional logic between a driver and arider [19]. The code is replicated across all network nodes;whena user initiatesatriprequest(triggering the required condition), the contract automates the workflow, escrows the fare, and handles the payout [1, 11]. If the conditions are not met, the script remainsidle. In modern blockchain development, the vast majority of these decentralized applications are built using Solidity, an object-oriented programminglanguage designed for execution on the Ethereum Virtual Machine.

d. TheEthereumProtocol

While Bitcoin was built strictly as a peer-to-peer digital payment system [16], Ethereum was engineered as a decentralized, programmable software platform. Developed by Vitalik Buterin, Ethereum allows developers to build and deploy complex decentralized applications (DApps) using smart contracts [24]. The ecosystem is powered by its native cryptographic token, Ether(ETH),whichis used topayforthecomputational resources—quantified as "gas"—required to execute transactions and smart contracts across its global node network.

e. CryptocurrenciesasDigitalAssets

Cryptocurrencies are digital, decentralized financial assets secured by cryptographic protocols [16]. By removing central banking institutions from the transaction pipeline, they enable direct, secure, and borderless transactions. Their adoption has grown rapidly, moving from experimental tech spaces into mainstreamcommerce,withmajorenterprisesaccepting digital currencies as legitimate payment instruments for consumer goods and services [10, 13].

f. StructuralMechanics:HowBlockchainsOperate

Every new interaction or record on a blockchainrequiresthe creationandcryptographicsigningof anew block. Before this block can be permanently appendedtotheledger,amajorityofthenetwork'sindependent

nodes must audit and validate its contents through consensus.The security of this process relies heavily on cryptographichashalgorithms,whichgenerateaunique, fixed-lengthstring of characters(a hash)foreach block. This hash acts as a digital fingerprint. Because any minor change to the underlying transaction data completely alters the resulting hash, tampering is instantly detectable [24].

Block Sequence	Data Context	CryptographicLinkage Details	Integrity Function
Block 1 (Genesis)	Initial TransactionLogs	Previous Hash: 000000 Current Hash: A78B9C	Serves as the mathematical rootfor all future nodes.
Block2	Subsequent TransactionLogs	Previous Hash: A78B9C Current Hash: F2D4E6	Links directly toBlock 1; any root tamper breaks this connection.
Block3	Latest Transaction Logs	Previous Hash: F2D4E6 Current Hash: 9B1C3A	Continues the sequentialchain; open to point to Block4

Table2:CryptographicInterconnectionandLinkage

ModelofSequentialBlocks

Crucially, every block contains the hash of the blockthat came before it, creating an unbroken mathematical chain. If amalicious actor attempts to alter a transaction inanolderblock, that block'shashchanges.Thisbreaks the link to all subsequent blocks, rendering the entire chain invalid in the eyes of the network.

When a new block is broadcast to the peer-to-peer network,eachnodecross-checksitagainstitslocalcopy of the ledger [16]. If the block complies with the network's self-enforcing consensus rules, the nodes add ittotheirlocaldatabases.Tosuccessfullycompromisea blockchain, an attacker would have to alter an entry, recalculate the hash properties for every subsequent block, and simultaneously seize control of more than 51% of the network's total computing power—a featthat is computationally and financially unfeasible for established networks.

DecentralizedApplications(Dapps)

DApps are softwareapplications that operate on a distributed peer-to-peer network rather than a centralized corporate server [14]. Their source code is typically open, transparent, and governed by consensus; any structural upgrades require agreement from the network community. DApps leverage blockchain backend logic to execute global, intermediary-free transactions safely, securely, and autonomously [10]. Network validators are incentivized to maintain this infrastructure by earning native cryptographic tokens for their work.

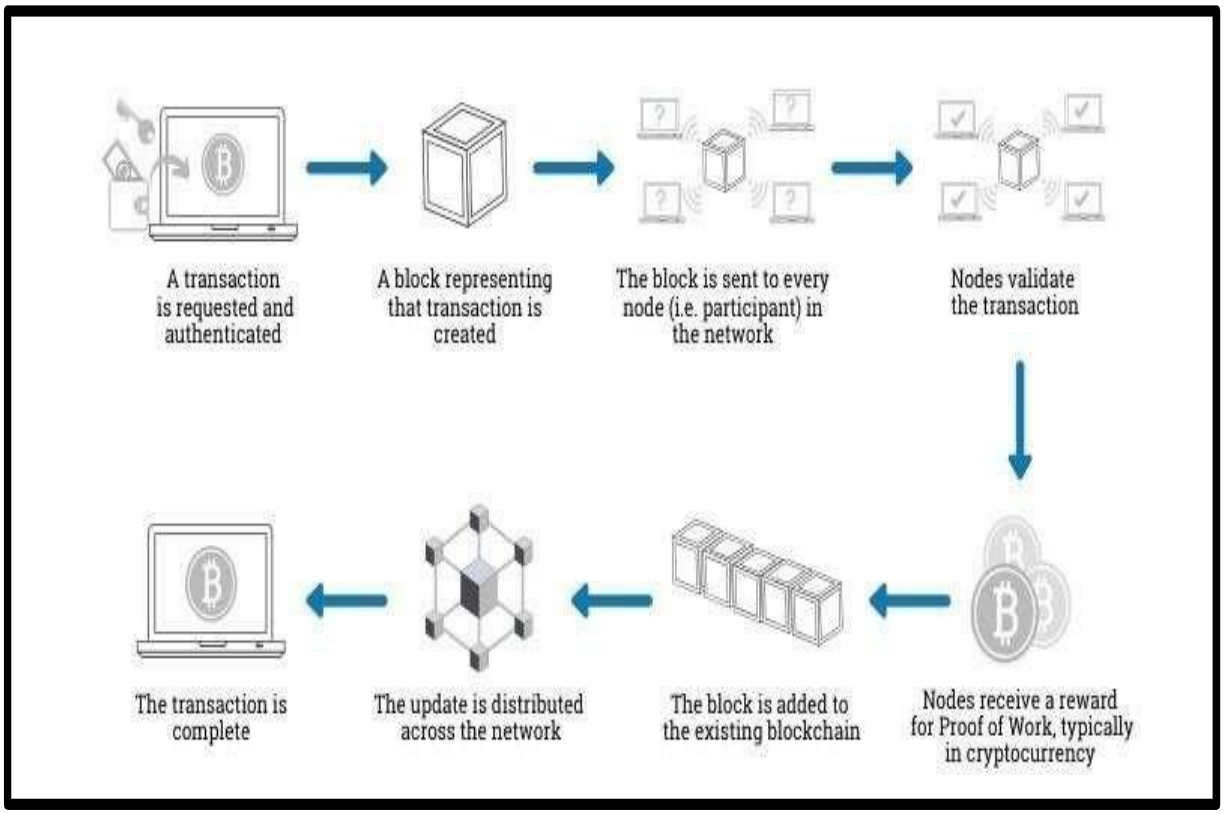


Figure 2: How does a transaction is done on Blockchain

g. EnterpriseFrameworks:HyperledgerFabric

Unlike open, public networks like Ethereum, Hyperledger Fabric is a permissioned blockchain framework designed explicitly for enterprise environments [5, 8]. In high-stakes industries like financeorhealthcare,dataprivacyisregulated,meaning participants must have a verified identity before interacting with the system. Hyperledger Fabric addresses this by replacing anonymous access with a robust identity management layer managed by Membership Service Providers (MSPs).

The platform executes smart contracts via "chaincode," which outlines the explicit business logic and rules governingnetworkinteractions[5].Tooptimize performance and privacy, access to specific chaincode functions can be restricted to designated peer groups. Furthermore, Hyperledger Fabric uses a modular architecture that breaks down transaction processinginto a strict three-phase pipeline:

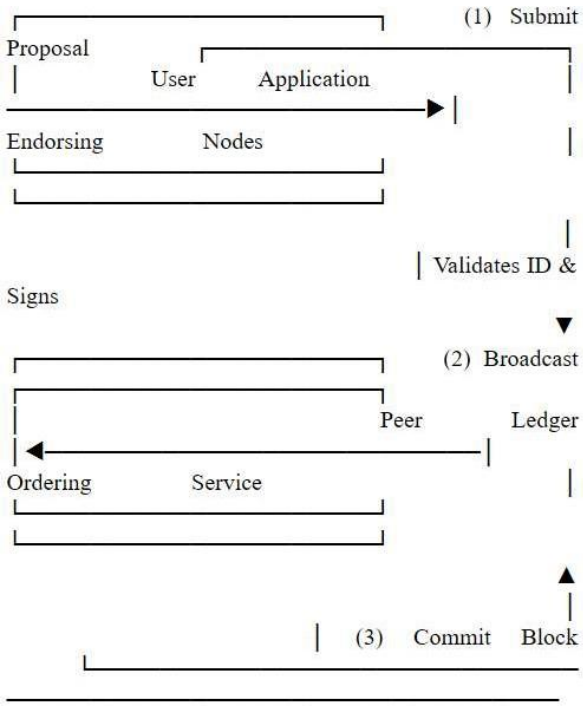


Fig3:ThePermissionedEnterpriseTransaction Pipeline

- **1.Phase 1 — Endorsement:**Execution and Identity Check.
- A user application submits a transactionrequest directlyto designated Endorsing Nodes [5]. These nodes verify the participant's cryptographic identity, execute the transaction simulateously against the chaincode, and sign off on the proposal.
- **2.Phase 2 — Ordering:**Chronological Sequencing.
- These endorsed transactions are packaged and routed to the Ordering Service [20]. This layer sequences the transactions chronologically and bundles them into clean data blocks without needing to run heavy, slow cryptographic mining puzzles.
- **3.Phase 3 — Validation & Commitment:**Ledger State Update.
- The ordered blocks are distributed to the broader pool of Peer Nodes [5]. These nodes perform a final check to ensure all preset endorsementpolicieswеremet andthatnodata conflicts exist before permanently committing the block to the local ledger.
- This streamlined workflow significantly improves transaction speeds and data privacy, ensuring that sensitive transaction details are only visible to the parties directly involved in the transaction.

h. AnalysisofCentralizedRide-SharingModels

- Traditional ride-sharing platforms rely entirely on a centralized broker architecture to manage interactions between riders and drivers [6, 15]. While these applications provide convenient on-demand matching, algorithmic route plotting, and digital payment handling, they introduce significant market inefficiencies by acting as a powerful digital middleman [11].

Pipeline Stage	**Active Participant**	**Process Actions & System Behavior**	**Economic / Data Output**
1. Trip Request	Rider App$ \rightarrow$ Server	Passenger submitspickup coordinatesand destinationtoa centralcorporate server [4].	Platform logs private user location data.
2. Dispatch	Server $ \rightarrow$ Driver App	Proprietary matching algorithmsscan nearby coordinatesand isolate a designated	Directs vehicle via corporate routing logic.

Pipeline Stage	Active Participant	Process Actions & System Behavior	Economic / Data Output
		driver[18].	
3. Processing	Rider App$ \rightarrow$ Server	Trip finishes; passengerpays throughtheapp. Centralauthority interceptsgross funds.	Platform extractsa 10% to 20%+ booking fee [6].
4. Net Payout	Server $ \rightarrow$ Driver App	Centralauthority processes the payment split, releasing the diluted remainder.	Driver receives net income after delayed processing [15].

Table 3:Structural Analysis of Transaction Flows, Participant Actions, and Data Outputs inCentralized Ride-Hailing

- This centralized structure introduces several core challenges:
- **Asymmetric Pricing:** The fare algorithms operate as an unverified "black box." Passengers face unpredictable surge pricing, and drivers have little visibility into how fees, taxes, and toll surcharges are split [6].
- **Driver Disenfranchisement:** High corporate commissions significantly suppress driver earnings. Because the central platform dictates terms unilaterally, drivers cannot negotiatefares or build independent client bases [15].
- **Data Vulnerabilities:** Centralized storage of user and driver data—including real-time location histories, home addresses, and financial credentials—creates an attractive target for bad actors, increasing the risk of data breaches and privacy violations [4, 22].
- **Safety Accountability:** Despite corporate vetting procedures, centralized platforms still struggle with safety issues. Because users have no direct waytoaudit background checks,trust rests entirely on corporate assurances.

i. Mechanics of a DecentralizedRide-Sharing Ecosystem

- By replacing centralized servers with blockchain architectures and smart contracts, decentralized ride-sharing platforms allow drivers and riders to connect directly [19]. This peer-to-peer approach offers a transparent, fair alternative to traditional services.

1. IdentityVerificationandProfileCreation

- The process begins with identity verification. When a driver registers on the platform, their credentials, vehicle registration, and insurance papers are uploaded and cryptographically hashed onto the blockchain [11]. Smart contractsautomaticallyalertverifiedthird-party background check services to audit the documentation.
- Once approved, the driver's current license status and safety rating are permanently maintained via smart contracts [19]. If an insurance policy or license is within 30 days of expiring, the system automatically sends a renewal alert. If the driver fails to update their documents, the smart contract lowers their system visibility, protecting market safety without requiring a corporate supervisor.
- Passengers go through a similar verification process[22].Ridersmust uploada validdigital ID alongside basic contact information. This data is written securely to the ledger, allowing

smart contracts to run background checks on passengers as well. This creates a balanced ecosystem where both drivers and riders can view each other's verified, system-generated safety scores before agreeing to a trip.

2. DecentralizedRideCreationandTransparent Pricing

- To start a shift, a driver publishes an available route, specifying their starting point, destination, departure time, and open seats [18, 23]. This information is broadcast instantly across the peer network.

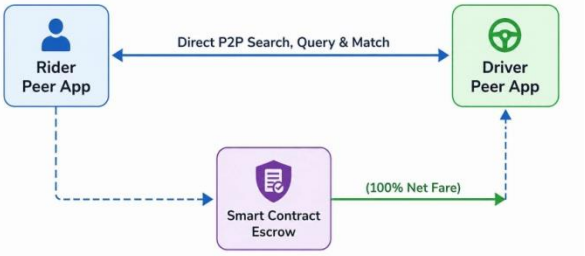


Fig 4:Decentralized Ride Creation and Transparent Pricing

Fares are calculated openly via smart contracts using transparent data points: current traffic data from open mapping APIs, vehicle class, fuel or energy efficiency metrics [7, 9], and a base per-mile rate set by the driver [2].Thisremovesalgorithmicmanipulation.Driverscan customize their rates based on the unique services they offer, and passengers can review a transparent breakdown of distance costs, toll fees, and service charges before booking:

- **Driver-Controlled Tariff Rates:** Rates are definedexplicitly by the operator per kilometer or mile, allowing open-market competition [2].
- **Verifiable Add-On Fees:** Toll taxes,municipal utility tariffs, and extra service facilities are structured natively before trip initiation [11].
- **Elimination of Intermediary Surges:** Third-partyalgorithmscannotartificiallymanipulate supply-demand lines to force inflated platform fees.

3. Peer-to-Peer Matching and Autonomous Settlement

When a passenger searches for a ride, smart contracts find nearby matches thatfit their criteriausing localized search indexes [21] and dynamic routing frameworks [18]. The application displays the driver's verified profile, vehicle details, cost breakdown, and historical rating [19]. If the passenger is satisfied, they send a direct ride request to the driver. The driver reviews the passenger's verified rating and can freely accept or decline the trip based on their own safety preferences.

Once a ride is accepted, smart contracts track the vehicle's coordinates via decentralized locationprotocols [4]. When the vehicle reaches its destination, the smart contract triggers the payment logic [24]. The agreedfareisautomaticallytransferreddirectlyfromthe rider's digital wallet into the driver's crypto wallet [1]. By removing corporate intermediaries, the driver keeps the full value of the fare minus a minimal network transaction fee, creating a fairer, highly secure, and transparent ecosystem [11, 15].

III.RELATEDWORK

1ImplementationsofDecentralizedRide-Sharing

This section evaluates the academic literature and real-world rollouts of peer-to-peer, ledger-based transit networks. Over the last decade, multiple cryptographic mobility ecosystems have entered the commercialspace. Early developers leaned heavily toward public, Ethereum-basedinfrastructures[16,24].However,these open ledgers introduce severe bottlenecks when executing smartcontracts thathandle sensitive user data or real-time location metrics. To bypass these constraints, modern implementations increasingly favor permissioned, private, or hybrid ledger frameworks [5, 22]. Understanding the core trade-offs between these architecturalstylesisessentialforidentifyingtheideal

infrastructure for next-generation ride-sharing applications.

SnagRideservesasaprominentexampleofanintercity

shared-mobility network that pairs distributed smart contractswithmachinelearning[20].Poweredbyits

proprietary utility asset, the MILE token, the system feeds data archived on the ledger into an artificial intelligence engine. This bot parses historical travel behaviors and recurring route paths to optimize matching,minimizingtransittimeswhiletailoringrides to individual passenger preferences [20].

Arcade City offers a stark contrast by operating as a completely decentralized ecosystem built directly on Ethereum's public layer [2]. Instead of treating driversas independent contractors, the platform uses self-executing smart contracts to route fares directly to the operators, turning them into primary stakeholders. A core innovation of Arcade City is its open-market pricing paradigm. Drivers bypass centralized corporate algorithms by organizing into localized co-ops called "guilds" [2]. These self-governing groups establish localizedbylawsandfixedmileagerates (e.g.,$1.50per mile). To maintain service quality within the peer-to-peer network, guild members face expulsion by their peers if they violate these community standards [2].

GreenRide shifts toward an eco-friendly, semi-decentralized model that relies on the Ridecoin asset framework [11, 17]. By removing corporate intermediaries, the platform allows riders and drivers to negotiate fares dynamically, using GreenRide Tokens (GRT) specifically to incentivize high-occupancy carpooling and curb vehicle emissions [11]. Architecturally, GreenRide utilizes a hybrid ledger where consensus relies on pre-authorized validation nodes—a strategy that largely resolves the transaction delaysandscalinglimitsseeninopennetworks[11].To mitigatetheprivacy risks tied to real-timetelemetry [4], theframeworksplitsitsdataarchitecture,keeping sensitive employeerecords and corporatedatabasessafe behind a secure Google Firebase wrapper. Thislocalized design [8] helps GreenRide sidestep the regulatory friction of public crypto-assets, making it highly effective for closed campus environments like universities or corporate complexes [11].

Finally, **Chasyr** represents an early, driver-led push to buildapublicpeer-to-peerridesharingalternativeaimed squarely at breaking the market monopoly held by centralized ride-hailing networks [3].

2. Resolving Latency and Spatial Privacy Bottlenecks

While early applicationsproved that peer-to-peer transit wasviable,theyexposeddeepsystemicissuesregarding searchindexingdelays[21]andvehicleroutingvulnerabilities[18,23].Recentresearchhighlightsa clearshiftfrombasicdecentralizedmatchingtohardened,decoupledarchitectures.Forinstance,modern frameworksnowuseDecentralizedIdentifiers(DIDs)to completelyseparateapassenger'sphysicalroutecoordinates fromtheir blockchain identityprofile [22]. Additionally,advancedpre-matchingdataschemesare being used to handle complex vehicle routing before a smart contract ever touches the ledger [12], while dedicatedpublicsmartwallets,suchasRideChain,successfully eliminatetheheavytransactionfeesthat undermined early open platforms like Arcade City [1].

IV. METHODOLOGY

While traditional ride-sharing architectures rely on centralized clearinghouse intermediaries to manage driver-passenger coordination [6], a decentralized framework shifts this operational oversight toimmutable distributed ledger technology [10]. In centralized platforms like
Uberand Lyft, a corporate third party dictatesoperationalbylaws, establishespricing frameworks, and acts as a trusted escrow broker to guaranteetransactionalsettlementsandusermatching[15]. By

eliminating this centralized broker, a blockchain-based ride-sharing model distributes regulatory and transactional enforcement directly to deterministic protocols and smart contracts [19].

MethodologyA:EthereumPublicArchitecture

This model focuses on a **public, trustless user-matching system** with financial penalties for bad behavior.

1. **Request & Privacy:** A rider requests a trip,and their location is passed through a **Spatial Cloaking** mechanism to hide exactcoordinates.

2. **Smart Contract Execution:** The obfuscated data is sent to an **Ethereum Smart Contract**, which triggers **Dynamic MatchingAlgorithms** to find nearby drivers.

3. **Verification:** A match is made and verified securely using Zero-Knowledge Proofs (**ZKP Verification**).

4. **TheHandshake(Escrow&Settlement):**

- If the **driver fails** to show up, their deposit is forfeited to the rider.

- If the **rider fails** to show up, their deposit is forfeited to the driver.

1) EthereumBasedRide-SharingFramework

This framework automates decentralized transactions between passengers and drivers through self-executing smart contracts and cryptographic protocol layers deployed on a public ledger [16, 24]. Smart contracts serve as the core structural mechanism to bypass third-party broker interventions [10].

On a valid trip, funds are progressively released via micropayments using a secure wallet architecture.Methodology

B: Hyperledger Fabric Permissioned Enterprise Pipeline

This model operates as a **private, enterprise-grade transaction pipeline** focused on data consensus and security.

1. **Proposal Submission:** Drivers or riders (End-User Clients) submit a signed transaction proposal verified by their cryptographic **MSP Identity**.

2. **Endorsement Simulation: Endorsing Nodes** simulate the transaction chaincode (smart contract)toverifyitandsignoffonitsvalidity.

3. **Chronological Ordering:** Validated endorsementsare sent to the**Ordering Service** (using consensus protocols like Raft or Kafka), which sequences the transactions into chronological order.

4. **Ledger Update: Committing Peers bundle these clean transactions into state blocks, permanently updating the ledger and distributing thedata across specific network channels.**

Business logic is written directly into the distributed systemtoorchestratetrustlesstransactionsbetweenpeer nodes. Within this ecosystem, operations executed on-chain occur natively inside the state machine of the blockchain, whereas off-chain workflowsrun externally to reduce storage overhead [24]. In this approach, smart contract deployment and state changes are executed strictly on-chain.

To preserve operational security and userconfidentiality, the framework integrates Zero-Knowledge Proofs (ZKPs)—a cryptographic protocol wherein a proving node mathematically demonstrates to a verifying node that aspecific statementistruewithout revealing any supplementary underlying data attributes [24]. The operational pipeline of this decentralized public network is structured through the following progressive phases:

Phase1:RequestandSpatialCloaking

The passenger initiates a ride request via a smartphone client interacting with an Ethereum smart contract [24]. To mitigate location tracking vulnerabilities, the user's explicit pickup and drop-off coordinates are masked usingspatialcloaking protocols[4].Thisapproachblurs exact coordinate data into a broader geographic region, protecting rider privacy while retaining sufficient structural utility for macro vehicle-routing [4].

Apublic-ledgermatchingalgorithmthenevaluatesopen driver profiles to see which routes intersect with the cloaked spatial region [18, 23]. Matched drivers receive the trip parameters and can post binding travel offers[2]. Because the pricing metrics are written into the shared ledger logic, all initial matching drivers receive the same base price, preventing artificial surge manipulation. The driver's precise itinerary is encrypted usingthepassenger'spubliccryptographickey,ensuring that only the matched rider can decrypt and view the real-time physical approach of the vehicle [4].

Phase2:HandshakeandSmartContractEscrow

The passenger reviews the available offers, selects a driver, and deploys a ZKP-based smart contract containing the calculated fare budget alongside a security deposit [24]. To accept the match, the driver must match this commitment by posting an identical securitydepositto thecontractescrow[11].Thismutual deposit architecture establishes cryptographic economic trustwithoutrequiringanintermediary.Thesmart contract acts as an automated arbiter, utilizing ZKP verification to determine the outcome of the agreement based on physical presence and compliance [24]:

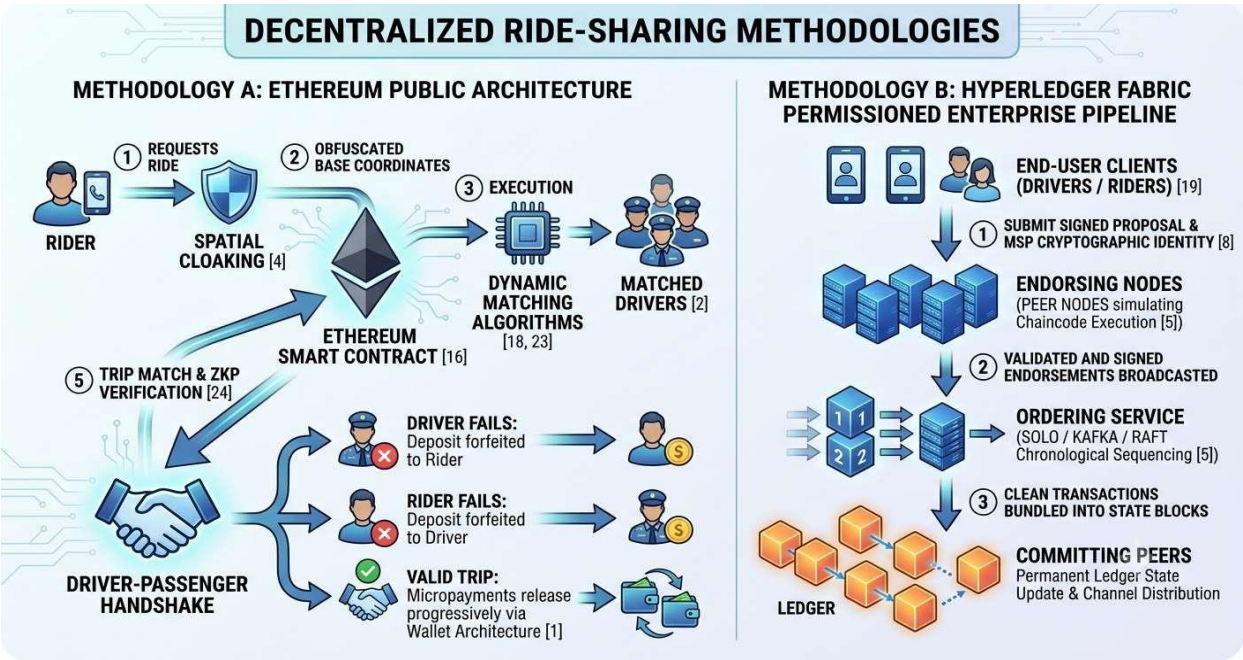
- Ifadriveracceptsatripbutfails toarriveatthe designated origin or cancels the dispatch, the smart contract automatically forfeits thedriver's deposit, transferring it to the rider's wallet.
- If a passenger confirms a ride but fails toappear at the origin point within the scheduled window, their deposit is forfeited and automatically routed directly to the driver's account.
- If both parties arrive successfully at the coordinated coordinates, the contract validates their presence, returns the driver's deposit intact,and converts thepassenger'sdepositinto an initial partial payment for the journey.

Phase3:TelemetryandIncrementalSettlement

Drivers and passengers maintain real-time connectivity to theblockchain network using GPS and wireless peer-to-peer data links on their mobile hardware [19]. Verified location-based service providers feed authenticated telemetry data into the ledger to verifyTo mitigate cash-flow risk and driver non-performance, the smart contract handles payments progressively, using specializedwallet architectures to stream micro-payouts to the driver at fixed 5-to-10 minute intervalsduring the trip [1].Upon reachingthedestination,the contractruns a final verification check to confirm successful trip completion before releasing the remaining service fee [24]. Because this model relies on open participation, transparent audits, and cross-border token settlement, a public blockchain structure like Ethereum provides the necessary ecosystem to handle these transactions globally [16, 24]. that the vehicle has arrived at the origin before the trip state transitions to active.

a. MatchingProcessoftheDriverandPassengers

To optimize matching, multiple algorithmic paradigms are utilized to balance vehicle passenger capacities, maximum passenger wait-time tolerances, and optimal path selection. While classic routing configurations rely on Dijkstra's Algorithm to find the shortest path from a single starting node across static networks, it evaluates routes purely from initial static location inputs,rendering it suboptimal for dynamic, real-time ridesharing.



To overcome the challenges of real-time matching, greedy heuristic models are integrated into the system:

- **TimeGreedyHeuristic:**Thisapproachtargets the vehicle pairing that minimizes cumulative travel delays and passenger wait times [18].

- **Distance Greedy Heuristic:** This method optimizesmatchingbasedstrictlyonproximity, selectingtheclosestavailablevehicletothetrip origin [23].

- **Advanced Data Structure Node Mapping:** This approach builds a structured network of independent ride-offer nodes, evaluating and compressing inverted travel indexes to rapidly calculate optimized routes across complex, overlapping routes [21].

b. PaymentsthroughSmartContract

Integratingsmartcontractsbringscomplete transparencytothepaymentpipeline,preventing common fraud vectors like route padding, where adriver records inflated mileage metrics to overcharge a user. Under this framework, the passenger's client application uses its private cryptographic key to periodically sign off on the actual traveled distance recorded by the phone's GPS [4]. The smart contract processes payment distributions only after receiving these authenticated updates, preventing unilateral fare inflation.

2) HyperledgerBasedRide-SharingFramework

In contrast to public networks, an enterprise-grade ride-hailing framework can be deployed using Hyperledger Fabric, a permissioned, private blockchain architecture designed for regulated environments [5, 8]. This framework relies on a federated consortium layout supported by multiple verified organizations, endorsing peernodes,channelstructures,andachaincodeprotocol layer[19].Theecosystemisgovernedbyfourcorenode types: client nodes, peer nodes, orderer nodes, and Certificate Authorities (CAs).

a. ArchitectureandComponentsofHyperledgerFabric

Clients

Clients are network applications that proposetransaction invocations on behalf of specific network users (drivers or riders) [19]. Utilizing the Fabric Software Development Kit (SDK), client applications connect directly to the development frameworks to query or update the ledger state. Each client mustpresent a valid cryptographic certificate issued by the Certificate Authority to ensure that every network transaction can be traced to a verified user profile [5].

MembershipServiceProvider (MSP)

The MSP acts as the primary identity management and access control layer for the permissionednetwork[8]. It authenticates, validates, and manages user identities, ensuring that only authorized participants can submit transaction proposals. The MSP interfaces with pluggable Certificate Authorities to issue, manage, and revoke digital x.509 certificates. Because Hyperledger Fabric supports multi-MSP integration, different fleet operations or municipal transit organizations can connect their own identity systems to a shared ledger network [5].

OrdererNode

The ordering service provides chronological transaction sequencing for the distributed ledger [5]. Unlike public chains that rely on slow, resource-heavy cryptographic mining puzzles (PoW) to establish blocks [16, 24], Hyperledger Fabric decouples consensus from execution. The network can utilize plug-and-play ordering mechanisms like SOLO (for development), Kafka, or Raft (crash fault-tolerant protocols) to sequence transactions cleanly into validated blocks [5]. This architectural separation minimizes processing latency, making it highly suitable for high-throughput commercial mobility applications.

PeerNodes

Peer nodesform the foundationalcomputing core of the organization and are split into two operational roles: **Endorsing Peers** and **Committing Peers** [5].

- **Endorsing Peers** receive incoming transaction proposals from driver or rider client apps, simulate the chaincode execution against their local state database, and return a cryptographically signed endorsement to the client. The client bundles these signatures and forwards them to the ordering service.

- **Committing Peers** receive the ordered blocks from the consensus service, perform a final validation check on the endorsement policies, and permanently commit the transactions to their local ledger copy [5, 19].

Channels

Channels provide private communication lanes within a shared Hyperledger Fabric network, allowing for complete data isolation. Each channel maintains a separate ledger instance that is invisible to unauthorized entitiesoutsidethatchannel.Whilethisspecificmodel

manages access control through granular chaincode permissions rather than maintaining multiple separate channels, the channel architecture allows peers to scale easily across separate regional transit markets or fleet subdivisions [5].

CertificateAuthority(CA)

The CA handles identity generation, issuing root and operationaldigitalcertificatestoallparticipatingorderer nodes, peer nodes, and end-user client apps [19]. These cryptographic identities are maintained within localMSP instances on each entity's system. Because each organization has access to a public certificate verification root, nodes can instantly verify that anentity invoking a specific chaincode function has been properly vetted and authorized by an allied organization [5].

b. ChaincodeProtocol

To protect consumer data and comply with location privacy regulations, the platform's enterprise business logic (chaincode) must be engineered to restrict telemetry access exclusively to the specific rider and driver involved in an active transaction [19, 22]. Individual participants are tracked using a unique User ID, generated by combining the user's localized MSP identifier with the organization's global MSP root key. This binding links the user's operational identity to their cryptographic certificate, making identity theft orprofile spoofing unfeasible without compromising the underlying private keys.

To manage the mobility lifecycle, client applications invoke the following core functions written into the Hyperledger Fabric chaincode layer [19]:

- RegisterUser:Createsauniqueprofileentryon the ledger, binding the salted cryptographic hash of the user's credentials to an array of emptyridestructures.Whenregisteringa driver profile, the system requires supplementary verification parameters, such as age, gender, and background checks, before granting system access.

- Request Ride: Generates a temporary state object on the ledger representing an active trip request.Thisfunctiontriggersanopennetwork event that alert-listening driver client applications can receive.

- Accept Ride: Modifies the temporary tripobject to lock in the matched driver's ID, changing the trip state to accepted. This function triggers a real-time event that notifies the passenger that a driver is en route.

- Set Ride Destination: Updates the coordinate state within the temporary trip object. The passenger's exact pickup coordinates remain hidden until this method confirms the match, protecting the user's location privacy [4, 22].

- Pickup Rider: Invoked by the driver upon arrival at the pickup point. The chaincode cross-checks the vehicle's GPS coordinates to confirm proximity before changing the ride status to active.

- Set Co-rider Information: Used in shared or carpooling models to dynamically add or remove passenger information within the temporary ride object during an active trip.

- Drop-off Rider: Triggered by the driver when arriving at the destination coordinates. This function extracts the accumulated telemetry datafromthetemporarystateobjecttogenerate a permanent ride history record for the driver.

- Leave Driver: Triggered automatically by the drop-offevent.Itgeneratesapermanenthistory log for the passenger, transfers the required token assets, and deletes the temporary tripstate object from the world state database.

- Get User Info: Allows verified users to safely query their own historical records, retrieving their credential hashesandassociatedhistorical trip logs.

V. DISCUSSION AND COMPARATIVE ANALYSIS

The architectural shift from centralized legacy frameworks to public and permissioned blockchains represents a fundamental evolution in consumer privacy and economic equity. To contextualize the performance of the proposed architecture, it must be evaluatedagainst existing operational models across key operational dimensions.

ComparativeEvaluationMatrix

Architectural Metric	**CentralizedModels(e.g.,LegacyRide-Hailing)**	**PublicBlockchain(MethodologyA:Ethereum)**	**ProposedEnterprisePipeline(MethodologyB:HyperledgerFabric + ZKP)**
Middleman Commission	High($20\%\text{--}30\%$)	Ultra-low(Gas fees only)	None(Internalenterprise/consortiumdistribution)
DataPrivacy	Poor(Storedoncorporateservers)	Publiclyexposedonledger	High (Isoratedchannels andencryptedprofiles)
IdentityVerification	Centralizedcheck	Cryptographic pseudo-anonymity	MandatorycryptographicMSP+ZKP validation
TransactionSecurity	Prone to single-pointfailure	Fraud-resistantbutpublic	Tamper-proofby design (Chaincode +zero-leakage)
SystemLatency	Instantaneous	Variable(Congestiondependent)	Highlyoptimized(High-throughputpermissionedblocks)

Table 4: Comparative Evaluation of Centralized Models, Public Blockchains, and the Proposed Enterprise Pipeline

PerformanceTrade-offsandAnalyticalChart

While public networks like Ethereum successfully decentralize the economic ledger, they struggle with high data exposure. The proposed Hyperledger Fabric The chart below illustrates how the proposed system optimally balances privacy, throughput, and security compared to public blockchains and legacy platforms: By decoupling operations from a central corporate intermediary and using permissioned data streams, driver syndicates and independent transport networks gain sovereign control over their operations. This approach eliminates security vulnerabilities while reliably scaling up their user base. pipeline solves this dilemma by walling off transaction states into private, permissioned channels. By adding ZKPs to the handshake layer, drivers and passengers canauthenticate their credentials and fulfill trip escrows withouteverrevealingactualidentityparametersor

base coordinates to unauthorizedthirdparties.Belowwe presentPerformanceComparisonandCumulativeIndex Rating of Legacy, Public Blockchain, and the Proposed HLF+ZKP Framework The chart below illustrates how the proposed system optimally balances privacy, throughput, and security compared to publicblockchains and legacy platforms: By decoupling operations from a central corporate intermediary and using permissioned data streams, driver syndicates and independent transport networks gain sovereign control over their operations. This approach eliminates security vulnerabilities while reliably scaling up their user base.

Architectural Variant	**Privacy Preservation(α)**	**Transaction Throughput (β)**	**System Security (γ)**	**Cumulative Index (∑)**
Centralized Legacy Platforms	1.0	3.0	2.0	$6.0/9.0$
Public Blockchain (Ethereum)	1.0	1.0	3.0	$5.0/9.0$
Proposed Framework (HLF+ZKP)	$\mathbf{3.0}$	$\mathbf{3.0}$	$\mathbf{3.0}$	$\mathbf{9.0/9.0}$

Table 5 : Evaluation Matrix for Privacy, Throughput, and Security across Different Frameworks

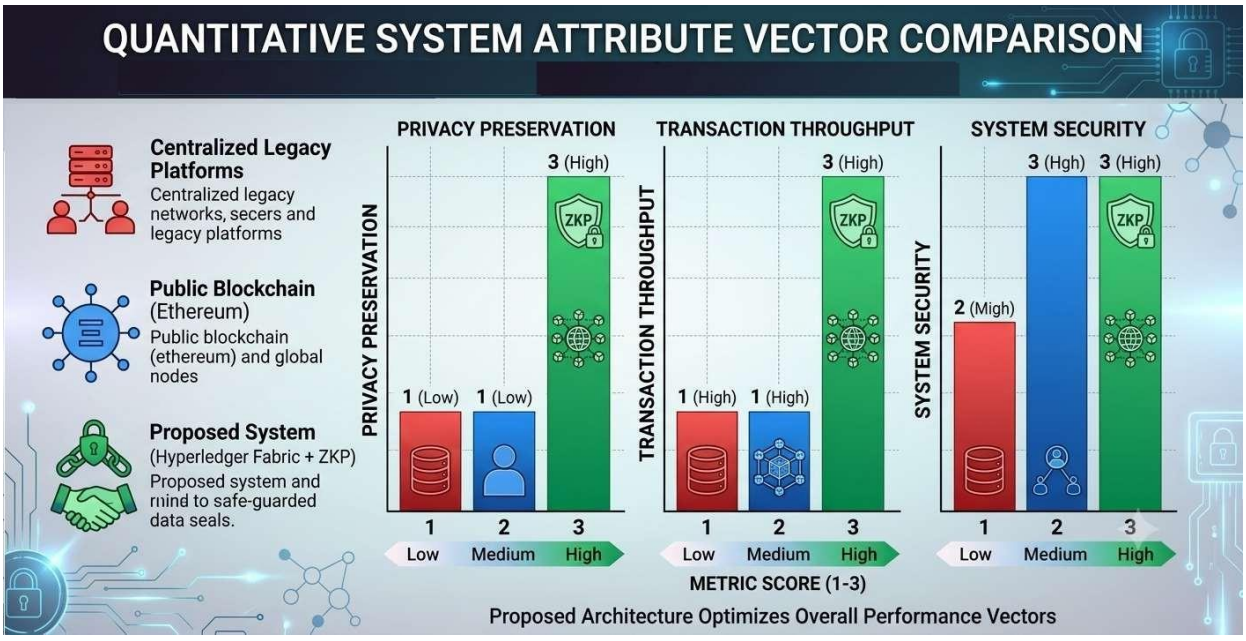


Fig 6: Comaprision of proposed solution with existingmethods

VI. CONCLUSION

Traditional centralized ride-hailing platforms increasingly conflict with the modern digital economy duetohighintermediaryfees,opaquepricingstructures, and heavy corporate commissions that diminish driver earnings. Transitioning to a blockchain-driven decentralized model offers a fair, transparent alternative for both short- and long-distance transport.

WhileEthereum-based frameworks have broken ground inreal-world applications,theirpublicnatureintroduces significant data exposure risks; anyone on the network can view and harvest sensitive user metadata, leaving participant information vulnerable to exploitationdespite the ledger's immutability.

Conversely, Hyperledger Fabric remains largely experimental in the ride-sharing domain, yet it introduces vital security advantages. By restricting access to authenticated, background-checked users within a permissioned environment, it delivers robust privacy controls. Its native chaincode protocols and implicitcertificatesisolateuserdatabydesign, ensuring that transactional data is non-discoverable by unrelated network participants.

Ultimately, integrating Zero-Knowledge Proofs (ZKPs) alongside Hyperledger Fabric eliminates cheating and fraud risks without sacrificing data privacy. Future iterations can scale this transparency further by introducing a "web of trust" peer-to-peer public key infrastructure to robustly shelter users against evolving online threats.

REFERENCES

1. Alhogail, A., Alshahrani, M., Alsheddi, A., Almadi, D., & Alfaris, N. (2024). RideChain:A blockchain-based decentralized public transportation smart wallet. *Mathematics*, *12*(19),3033. https://doi.org/10.3390/math12193033
2. Arcade City. (2018). *The future of ridesharing is decentralized*. Foundation for Economic Education (FEE). https://fee.org/articles/arcade-city-the-future-of-ridesharing-is-decentralized/

3. Bitcoinist. (2017). *CHASYR is emerging as the ultimate ridesharing alternative*. Bitcoinist.com.
4. Chow, C.-Y. (2008). Cloaking algorithms for location privacy. In *Encyclopedia of GIS* (pp. 93–97). Springer. https://doi.org/10.1007/978-0-387-35973-1_143
5. Clairvoyantsoft.(2019).*HyperledgerFabric componentsandarchitecture*.Clairvoyant Blog. https://blog.clairvoyantsoft.com/hyperledger-fabric-components-and-architecture-b874b36c4af5
6. Clewlow, R. R., & Mishra, G. S. (2017). *Disruptive transportation: The adoption, utilization, and impacts of ride-hailing in the UnitedStates*(ResearchReportUCD-ITS-RR-17-07, pp. 1-30). Institute ofTransportation Studies, University of California, Davis.
7. Fagnant, D. J., & Kockelman, K. M. (2014). The travel and environmental implications of sharedautonomousvehicles,usingagent-based modelscenarios.*TransportationResearchPart C: Emerging Technologies*, *40*, 1-13. https://doi.org/10.1016/j.trc.2013.12.001
8. Frankenfield, J. (2021). *Hyperledger Fabric in blockchain*. Investopedia. https://www.investopedia.com/terms/h/hyperledger-fabric.asp
9. Güneralp, B., Zhou, Y., Ürge-Vorsatz, D., Gupta, M., Yu, S., Patel, L., Fragkias, M., Li, X., & Seto, K. (2017). Global scenarios of urban density and its impacts on building energy use through 2050. *Proceedings of the National Academy of Sciences*, *114*(34), 8945-8950. http://dx.doi.org/10.1073/pnas.1606035114
10. Huckle, S., Bhattacharya, R., White, M., & Beloff, N. (2016). Internet of things,blockchainandsharedeconomyapplications. *Procedia Computer Science*, *98*, 461-466. https://doi.org/10.1016/j.procs.2016.09.074
11. Khanji, S., & Assaf, S. (2019). Boosting ridesharing efficiency through blockchain: GreenRideapplicationcasestudy.In*201910th International Conference on Information and Communication Systems (ICICS)* (pp. 1-5). IEEE. http://dx.doi.org/10.1109/IACS.2019.8809108
12. Kumar,L.,Kaur,I., Rana,D., &Srivastava,A. (2024). A novel approach for online B-ride-hailingservicebyTan-Tumusingpre-matching information scheme. *AIP Conference Proceedings*, *3168*(1), 020002. https://doi.org/10.1063/5.0221125
13. Lanctot, R. (2017). *Accelerating the future:The economic impact of the emerging passenger economy* (Strategy Analytics Executive Report, pp. 6-30). Intel Corporation.
14. Lastovetska, A. (2021). *Howtobuild yourown blockchain architecture*. MLSDev. https://mlsdev.com/blog/156-how-to-build-your-own-blockchain-architecture
15. LeewayHertz.(n.d.).*Blockchaintodisrupt Uber: Entering ridesharing industry*. LeewayHertzCustomSoftwareDevelopment. https://www.leewayhertz.com/blockchain-disrupting-uber-platform/
16. Nakamoto, S. (2008). *Bitcoin: A peer-to-peer electronic cash system* (Whitepaper, pp. 1-9). Bitcoin.org.
17. Ridecoin.(2018). *Ridecoin whitepaper*. https://static1.squarespace.com/static/5aa1f1de0dbda36d6bc88b4e/t/5ab19b17f950b7dcfa8e0450/1521589015691/Ridecoin+Whitepaper.pdf
18. Santos, D., & Xavier, E. (2013). Dynamic taxi and ridesharing - a framework and heuristics for the optimization problem. *Proceedings of the 2013 Winter Simulation Conference*, 2885-2891. https://doi.org/10.1109/WSC.2013.6721635

19. Shivers, R., Rahman, M., & Shahriar, H. (2019). Toward a secure and decentralized blockchain-based ride-hailing platform for autonomous vehicles. In *2019 IEEE International Conference on Big Data (Big Data)*(pp.4972-4977).IEEE. https://doi.org/10.1109/BigData47090.2019.9006001
20. SnagRide. (n.d.). *Snagride (MILE) - ICOrating and overview*. ICOmarks. https://icomarks.com/ico/snagride
21. Song, X., Yang, Y., & Jiang, K. (2019). Optimizing partitioning strategies for faster inverted index compression. *Information Processing & Management*, *56*(2), 343-356. https://doi.org/10.1016/j.ipm.2018.10.012
22. Sun,N.,Liu,Y.,Zhang,Y.,&Liu,Y.(2024). Decoupling online ride-hailing services: A privacy protection scheme based on decentralized identity. *Electronics*, *13*(20), 4060. https://doi.org/10.3390/electronics13204060
23. Tao, C., & Chen, C. (2007). Heuristic algorithms for the dynamic taxipoolingproblem based on intelligent transportation system technologies. *Proceedings of the 2007 International Conference on Fuzzy Systemsand Knowledge Discovery*, *3*, 509-513. https://doi.org/10.1109/FSKD.2007.346
24. Zheng, Z., Xie, S., Dai, H., & Wang, H.(2016). Blockchain challenges and opportunities: A survey. *International Journal of Web and Grid Services*, *14*(4), 352-375. https://doi.org/10.1504/IJWGS.2018.09564