Indonesian Journal of Electrical Engineering and Computer Science
Vol. 42, No. 3, June 2026, pp. 892∼901
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v42.i3.pp892-901

Smart contracts and a dual blockchain structure for collaborative tourism

Zohra Temmar[1], Asmaa Boughrara[2]
[1]Department of Computer Science, University of Science and Technology of Oran-Mohamed Boudiaf, Oran, Algeria
[2]LSSD Laboratory, Department of Computer Science, University of Science and Technology of Oran-Mohamed Boudiaf, Oran, Algeria

Article Info

Article history:

Received Jul 6, 2025
Revised May 4, 2026
Accepted May 26, 2026

Keywords:

Blockchain
Consensus
Distributed system
Smart contract
Tourism

ABSTRACT

This article optimizes a decentralized system for collaborative tourism in Algeria using blockchain, smart contracts, and proof of reputation (PoR) consensus. The system matches services into organized trips, manages reservations, and automates payments to ensure transparency and autonomy without centralized authority. This work opens the door to exploring dual-blockchain architectures. Building on a previous work, we enhanced node interactions, automated contract execution, and introduced a dual-blockchain structure to reduce latency while improving scalability and security.

Corresponding Author:

Zohra Temmar
Department of mathematics and Computer Science, Universiti of Science and Technology Oran
El Mnaouar, BP 1505, Bir El Djir Oran, Algeria
Email: zahratemmar@gmail.com

1. INTRODUCTION

Collaborative tourism links travelers directly with local service providers, which creates personalized and authentic experiences [1]. The traditional tourism platforms that function using a web2 structure often suffer from issues [2] such as fraud risks [3], easy manipulation and hacking of the cental authority, and single point failure for servers [4], it also lacks transparency payments wise which might complicate tracking.

This has sparked multiple suggestions to adopt Web3 technologies to address these issues [5], [6]. Indeed, several companies have already implemented blockchain solutions to manage payments, reviews, and insurance claims [7]. The work presented in [8] proposes adopting Web 3.0 principles, and more specifically blockchain, to manage a collaborative tourism outside of traditional platforms. To ensure fairness and trust, it also optimized the proof of reputation (PoR) consensus mechanism. While promising, that approach lacked support for smart contracts and a complete decentralization which defeats the purpose of the web3 (eliminating centralized authority).

To address these gaps, our study proposes a decentralized, blockchain-based platform tailored for collaborative tourism. This work extends previous research by integrating smart contracts into the framework while introducing a dual-blockchain data structure. This architecture is designed to lower latency and enhance user experience, while also enabling secure, transparent, and fully automated tourism management.

2. CONTEXT

WEB3: or decentralized web, is a web space that allows people to interact with a mutual trust that is set by the technology of blockchain itself. An ideal Web3 infrastructure should not include a middleman that

has any power over the network, and now since 2020 multiple fields had started emerging their services into the blockchain, chasing the security and integrity it offers [9].

Blockchain: is a decentralized technology that stores data across multiple nodes, enabling real-time information exchange without a central authority [10]. It ensures transparency, security, and tamper-resistant record keeping through cryptography. Blockchain is based on key principles including distributed data sharing, peer-to-peer communication, transparency with pseudonymity, irreversibility of records, smart contract execution, and consensus among participants [11].

Consensus: is an algorithm and a technique to ensure the public nodes in a blockchain netwok are trusted and responsible for data updates by giving a form of validity to the new data. There are multiple types of consensus algorithms, such as Proof of Work and Proof of Stake, [12], which differ in how nodes are selected to validate and add new blocks to the blockchain, and also Proof of Reputation which is the consensus used in this work.

Proof of reputation: a consensus mechanism that uses reputation, rather than digital tokens as the main incentive for participation. In PoR, nodes with strong reputations are selected to validate and publish blocks [13]. This mechanism uses the behavior of nodes within themselves over a period of time to determine their reputation value . It then uses that value to determine a set of nodes responsible for creating data and updating and preforming tasks within the network [14].

3. BACKGROUND

3.1. Related work

In the context of integrating blockchain in the tourism field there are some remarkable work in the past recent years such as Travala, XcelTrip, and LockTrip, [15] that employ blockchain technology for decentralized cryptocurrency payments, transparent booking transactions, and token-based loyalty and reward systems, aiming to reduce intermediary fees, increase trust between users and organizations, and enhance payment security. Still, their use of blockchain is limited to the financial layer rather than trip planning and providing the service.

Beyond tourism, blockchain has been extensively studied in other application areas, such as secure biological data sharing through federated learning frameworks (BIOCHAIN [16]), provenance tracking and integrity verification of digital textual content (CERVANTES [17]), and supply chain management to improve transparency, traceability, and resistance to fraud (Chowdhury [18]).

3.2. Overview of the work to optimize

The approach proposed in [8] aims to help service providers (guides, housing providers, and transporters) offer their services, enabling travelers to easily consult and book trips. When the service providers submit their services, they get matched together to form a trip package. This process happens when their information match in date and location. The matching happens in a centralized server, and then the new data gets transferred to the blockchain network that is responsible for the reservation for the clients. Once a reservation is made, the network initiates a node election process to identify a leader for block creation. The selected leader adds the reservation data into a new block, utilizing PoR as the underlying consensus protocol.

The PoR consensus was proposed, where voting a leader depends on the reputation of nodes that is determined by two parameters: score (the number of times a node has been elected as the leader), and weight (a normalized value of the Score to prevent nodes from dominating the selection process), to help promoting decentralization. While executing the consensus, all nodes in the network are concerned of the voting process to insure a fair selection. Just like proof of work consensus, this consensus focuses on security (ensuring fair selection) and decentralization (all nodes are concerned with the voting process) while sacrificing a good amount of scalability (according to the Three-Dimensional Tradeoffs for Consensus Algorithms [19])

3.3. Limits of the approach to optimize

Complexity: the multi-step voting process for consensus introduces additional complexity, which can lead to increased latency, especially as the network grows. Reputation system exploitation: the PoR mechanism is vulnerable to collusion. Malicious nodes can coordinate to artificially inflate their reputation through false validations, eventually leading to a centralized takeover of the network. Centralization: this work was not fully decentralized when coming to generating trips since the organization server was the entity cross matching the trips and sending it to the nodes, which is what we are looking to avoid when choosing to decentralize the system. Smart contracts and payment automation: this work's conception and execution didn't include

the smart contracts concept and the automation of payments as it was left unfinished due to the lack of time. And for these reasons and problems we have decided to restructure this approach to get an improved refined approach that will be explained in the next section

4. PROPOSED APPROACH

Our proposed approach decentralized the services matching and trips management and integrated the concept of smart contracts to manage the payments within the nodes automatically, this was done without retouching the consensus method, only re-implementing it.

4.1. The architecture of the proposed approach

Our approach was structured into 3 different layers:

Application layer: the interface given to the user to interact with (login, offer services, and participate in trips)

Network gateway: this layer manages the communication between the blockchain network and the application layer, it is a gateway between the users environment and the blockchain network, it also ensures the well management of the platform. Blockachain layer: this layer includes 2 seperate modules:

- Network module: this module manages the synchronization of the nodes (managing the consensus, the reputation system, broadcasting the results and processing what has been received).
- Smart Contract module: this module is what manages the trips part (deploying and executing the smart contracts, updating the trip blocks, verifying the finished trips and executing the payments).

4.2. Actors and components

The user : We have two types of users:

a) Simple user: upon registration, the user can select to interact as a simple user (someone looking for reservations). A normal user can browse available trips and reserve spots as desired, complete the reservation process by making the payment , view a list of trips they have successfully reserved, and access and edit their personal profile.

b) Service provider: during registration, a user may also choose to interact as a service provider (e.g., guide, transporter, or host). A service provider can submit a service offering to be matched within a trip package, view a list of their submitted services and see which trips they have been matched to, and access and view the profile.

The node: In this system, the node has two states of functioning

a) The node in a normal state : the node can: participate in a consensus and broadcast the results, organize the consensus when randomly chosen by the network gateway, and validate then accept or reject received data.

b) The node as a leader: When an election is done (as shown in the previous work) a node can possibly be chosen to be a leader which gives it access to executing and deploying smart contracts, generating new data and broadcasting it, distributing payments, and registering a new node in the network.

4.3. The smart contracts

4.3.1. The trip planning smart contract:

This smart contract automates the creation of new trip packages and secure their execution, while conserving the rights of service providers and applying a ticketing system for the participators. The template of the smart contract is stored in the genesis block of the trip planning blockchain in decimal format.

4.3.2. The payment smart contract

This smart contract ensures and secures payments for service providers. When a reservation is made, the payments get collected. Then, after the trip is over, the service providers' shares are distributed to their accounts. This smart contract acts as a legal contract for this agreement and is what ensures this process.

4.4. The proposed blockchain format

Our approach modified the conventional blockchain model by introducing a dual-chain architecture, where two completely separate but interactive blockchains were used to manage different functions within the system. Both blockchains exist in every node in the system, and when the node is active and interacting, the blockchains are triggered to write new data, update, or fetch in a completely separate way.

4.4.1. Trip planning blockchain

Genesis: the genesis is generated automatically when a new node is first launched, and it will contain the same data for all nodes(the template of the trip planning smart contract).

Trip information block: this type of block contains the trips data including:the indexing and timestamp of the block, the new trip data (location, start date, end date, price, spots, available spots, etc.), including the services data,an object that stores and manages spots(IDs and spots reserved),a flag determining the state of the trip (if payed or not), and the current and previous block hashes for security.

4.4.2. The payments blockchain

Genesis: the genesis is generated automatically when a new node is first launched, and it will contain the same data for all nodes(the template of the payment smart contract).

Payment smart contract block: this type of block contains the payment contract data including: the index, creation timestamp of the block, the trip ID related to this payment, the execution date, the data required for payment distribution (service IDs, bank accounts, and requested amounts), the smart contract execution arguments (a string of args for the system), and the current and previous block hashes for security.

4.5. The dual blockchain

The trip blockchain focuses solely on trip planning, while the payments blockchain ensures secure financial operations. We chose this architecture to reduce complexity for simple tasks. When a user browses the platform, trips are fetched from nodes with limited resources. By isolating trip data from financial transactions (smart contracts), these nodes avoid processing a big chunk of data for simple operations like displaying trips. Also, this separation allows each blockchain to evolve independently. The payments blockchain can prioritize security, while the trip blockchain can be optimized for lower latency and a better user experience.

4.6. The proposed approach process

The general process of the approach go through multiple steps:

4.6.1. Registering a new node

To register a new node, the system must assign it a representative. As shown in Figure 1, the gateway receives the join request, saves it, and triggers a network consensus. Once a representative is selected, the gateway sends its address to the new node to continue the registration.

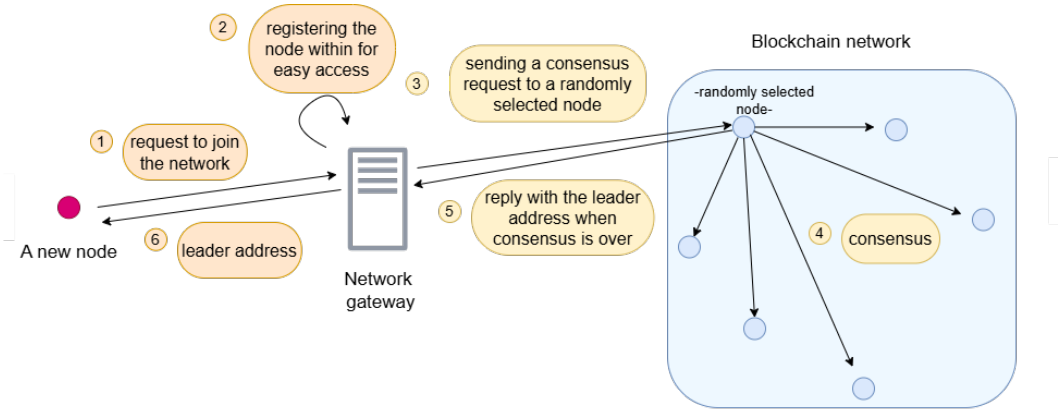

Figure 1. Assigning a representative for the registration

After assigning the representative, the new node sends the registration request to the elected representative. The representative next registers the node within its own database, then broadcasts a request in the network to ensure the new node is registered in all parts of the network, while also sharing the network data and the database with the new node.

4.6.2. Submitting a service

As shown in Figure 2, this process starts when a service provider submits a service to the network gateway, the gateway next sends a consensus request to a randomly selected node, that node will launch and manage the consensus until it is finished, then it will share the result (the leader) with the gateway, the gateway then will send the request to the leader, the leader will process the request, broadcast the results, and reply with the results to the gateway, the gateway will share the results with the service provider.

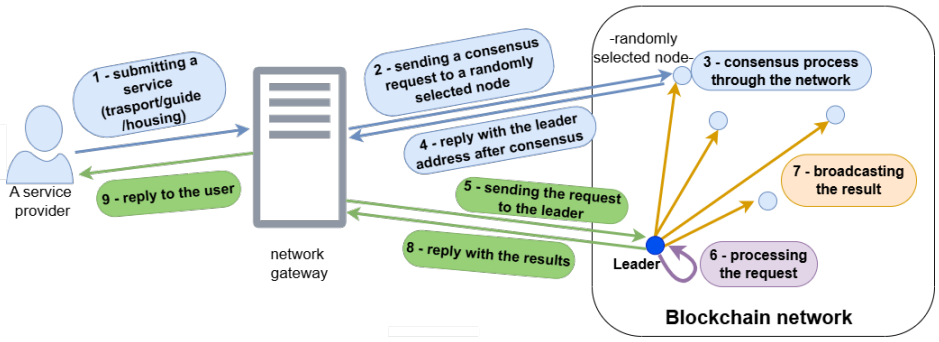

Figure 2. Submitting a service through the system

When the leader receives the request to add a service. it starts by extracting the smart contract template from the genisis and translating it from byte to code format. When the template is ready, the code gets executed so it saves the new service into the database and goes through all the available services looking to match them into a new generated trip. If a match is found it generates a trip, if not it saves the service back into the database. When the execution is over the new produced data (either a new trip or a new service) gets broadcasted to the rest of the network and shared with the gateway.

4.6.3. Participating in a trip

The process of adding a participation through the network is the same as adding a new service (that has been explained earlier) only with one step added, which is verifying the payment, this step is done by the network gateway through PayPal right after the submission of the request by the user. As shown in Figure 3, when the leader receives the participation request, it proceeds with deploying a new block of the payment smart contract that contains the data needed for later execution. Then it will create a new trip block updating the trip data to include this new participation, both these blocks will be next shared with the network.

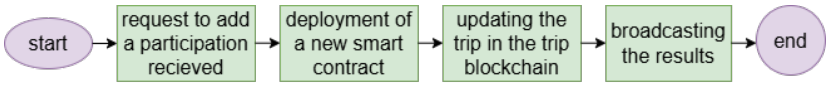

Figure 3. Flow diagram for the leader processing a participation request

4.6.4. Verifying and distributing payment

After a trip, the service providers expect a payment for thier services, which are done this process :

This process starts when the time trigger goes off in the gateway, this time trigger is important to make sure all payments are delivered on time, its result will be a consensus request sent to a randomly selected node, the node will launch a consensus through the network and when done, the result (the leader) is shared with the gateway, the gateway next randomly selects another node and contacts it to verify the validity of the leader address (this step is for extra security since the leader will have access to funds).

After the leader is selected and the environment is ready, as shown in Figure 4, the verification process starts when the gateway sends the verification request along with a PayPal token to the leader. The leader then will process the request then distribute the payments, broadcast the results, and share the results with the gateway all at once, the gateway next will notify the users.

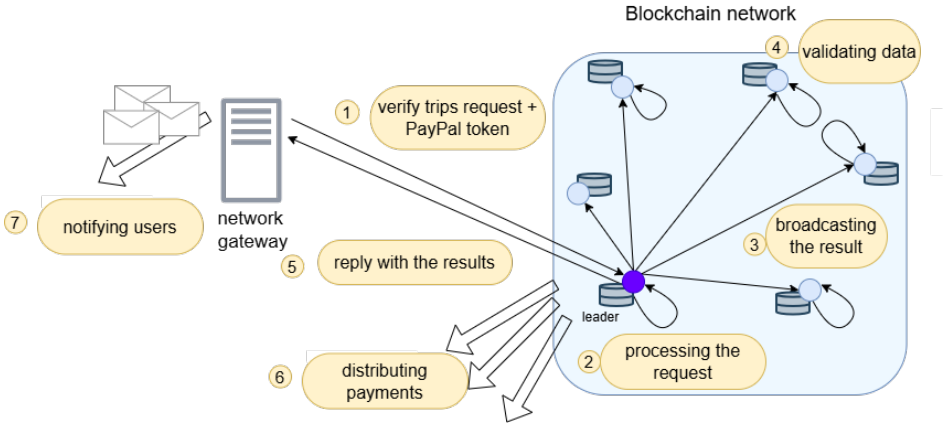

Figure 4. Verifying the trips through the network

As shown in Figure 5, when the leader receives the request, it will start by searching through the trips blockchain looking for trips that are over but not yet payed, when done, it again goes through the payment blockchain looking the payments related to those trips, when the search is done the leader executes the smart contracts and produces a new trip block for each trip to update it as payed, next the leader will distribute the payments using the received token, broadcast the results, and notify the gateway.

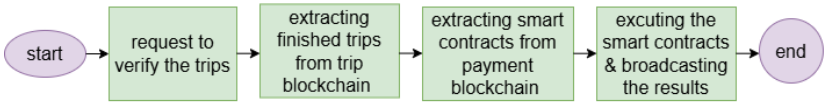

Figure 5. Flow diagram of verifying the trips within the leader

5. EXPERIMENTING

5.1. Development environment

To implement the approach we have used a lenovo thinkpad laptop with an i5 - 3600U cpu, 8gb ddr4 Ram and 512gb solid state drive. for software we utilized Express , VueJs, HTML, CSS, Postgress v16, and paypal sdks [20]. The implementation required 3449 lines of code distributed on 36 different scripts.

5.2. Testing and evaluating

We evaluated several platforms before deciding to build from scratch to ensure full customization of the node, network, and consensus. Our testing included Hyperledger Fabric [21] (Golang), which we found too enterprise-oriented, as well as Ethereum [22] and Solana [23]. Although we implemented smart contracts in Solidity, Vyper, and Rust, these platforms were ultimately unsuitable due to their predetermined consensus mechanisms. A precise comparison is not feasible, as our custom development was driven specifically by the lack of an existing platform meeting our requirements. We considered while testing that the platform is limits the trips creations in order to lower competition (there cannot be multiple trips with the same time and location), the evaluation was done by testing:

5.2.1. The resources usage

The payment verifications resources usage Payment verification occurs when a node is selected to verify and payout trips as shown in Figure 6. This includes both verification and result broadcasting. Each trip

requires the processing about 40 blocks per 10 trips. The following data illustrates the resource usage of this process. As shown in Table 1 the growth rate develops in low values.

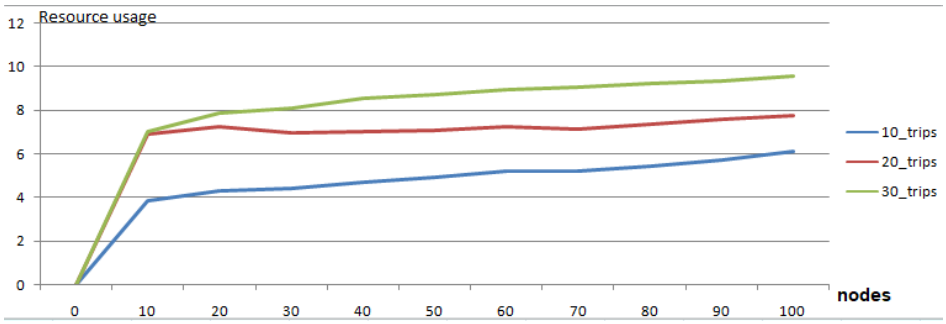

Figure 6. The payment verifications resources usage (percentage of CPU usage)

To evaluate the results we have used the growth rate:

$$\text{Growth Rate} = \frac{\text{Resource Usage}_{100\text{nodes}} - \text{Resource Usage}_{10\text{nodes}}}{90} \quad (1)$$

Table 1. Growth rate per node for different workloads

Metric	10 trips	20 trips	30 trips	Average
Growth Rate (units/node)	0.025	0.0096	0.028	0.021

Receiving and validating the payment verifications resources usage In this case the resources usage depends solely on the number of blocks received,each trip has 4 reservations which means validating about 40 blocks for each 10 trips. As shown in Figure 7, although the resource usage grows consistently with node count, the growth rate of R=0.22 is a high rate and shows that this process may require optimization at larger scales.

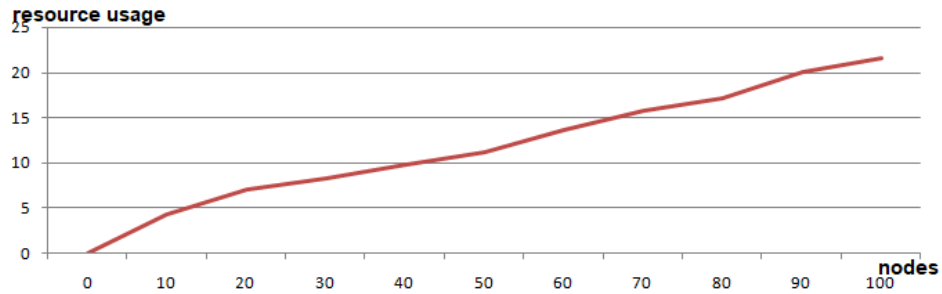

Figure 7. Receiving and validating the payment verifications (% of resources usage per number of nodes)

Comparison with the previous work: the previous wark of [8] evaluated the results using the resources usage of the developed proof of reputation consensus. To grasp our progress we will be comparing our implementation results with the results shown in that work. Shown next is the 2 results compared by the resources usage of the first and second round of the consensus.

Scaling the previous work to 35 nodes caused network jamming and errors during result exchange. By restructuring the architecture and re-implementing the consensus, we resolved these issues and achieved smoother execution. As shown in Figure 8, our implementation significantly reduced CPU usage, reaching a 37.2% improvement over the original work.

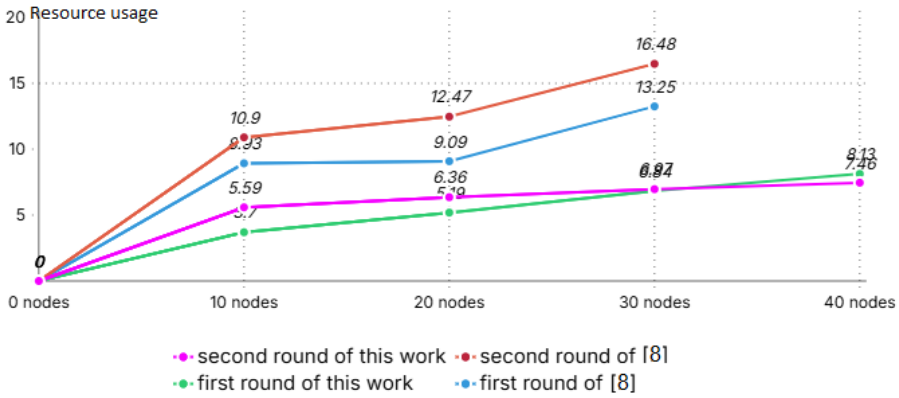

Figure 8. Comparison with the work proposed in [8]

$$\text{Improvement } (\%) = \frac{\text{Resource usage of [6]} - \text{Resource usage of our work}}{\text{Resource usage of [6]}} \times 100 = 37.2\% \quad (2)$$

5.3. Evaluation of the security

We have only applied superficial simple security tests to ensure the architecture is functional,we are aiming to apply more complex detailed tests in the future after developing the work more.

5.3.1. A man in the middle attack (tempering with created data)

In a 10-node network with one malicious actor, an elected leader's broadcast is intercepted before reaching the nodes. The attacker tampers with the data and rebroadcasts it. However, the nodes reject these blocks as invalid. After confirming the leader's identity, each node verifies the block's hash; since the tampered hash does not match, the block is discarded.

5.3.2. A malicious node broadcasting fake blocks

In a 10-node network with one malicious actor, the attacker broadcasts a block containing seemingly valid data and hashes signed with its own private key. The network consistently rejects these blocks. Before data validation, nodes verify the creatorNodeUrl and consult the "leaders array" to ensure the sender is the authorized consensus leader. Since the malicious node is not the designated leader, the block is discarded.

5.4. Limitations

The project is limited by the lack of further exploration into scalability, the work could also use more detailed testing to test the security. Also since we did not optimize the consensus itself it would still struggle from Reputation System Exploitation issues we have mentioned before. Future work could optimize the work by integrating reward mechanisms, optimizing the two blockchains separately since they serve different purposes, and enhancing node behavior to detect and isolate malicious actors.

6. CONCLUSION

In this work, we improved and optimized a Decentralized solution for Collaborative Tourism. A blockchain and smart contract based system that enables trip organization, reservations, and automated payments. We have managed to enhance scalability and improve the decentralization compared to the previous work. This was done by proposing a dual blockchain architecture. Overall, this work demonstrates clear improvements and provides a solid foundation for future developments, particularly in expanding node behavior and improving the request handling to include rewarding data-creating nodes.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Zohra Temmar	✓	✓	✓		✓	✓	✓	✓	✓		✓		✓	
Asmaa Boughrara				✓			✓	✓	✓	✓		✓	✓	✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal Analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject Administration
Fu : **Fu**nding Acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study will be available in: https://github.com/zahratemmar/tripPlanner.git

REFERENCES

[1] D. Getz and T. B. Jamal, "The environment–community symbiosis: A case for collaborative tourism planning," *Journal of Sustainable Tourism*, vol. 2, no. 3, pp. 152–173, 1994.

[2] M. E. Noti, "Web 2.0 and its influence in the tourism sector," *European Scientific Journal*, vol. 9, no. 20, 2013.

[3] S. A. Ansar, J. Yadav, S. K. Dwivedi, and A. Pandey, "A critical analysis of fraud cases on the internet," *Turkish Journal of Computer and Mathematics Education*, vol. 12, no. 1, pp. 424–445, 2021.

[4] N. A. Saqib, A. A. Salam, A.-U. Rahman, and S. Dash, "Reviewing risks and vulnerabilities in web 2.0 for matching security considerations in web 3.0," *Journal of Discrete Mathematical Sciences and Cryptography*, 2021, doi: 10.1080/09720529.2020.1857903.

[5] D. Duziak, "Blockchain for Hospitality and Tourism,"Apress LP, 2023.

[6] Y. Fan, B. Lin, and Y. Lin, "Web3-enabled technologies in tourism businesses: a multi-method study of user trust, loyalty, and tangible economic outcomes," *Tourism Review*, Aug. 2025. doi: 10.1108/TR-03-2025-0293.

[7] I. Önder and U. Gunter, "Blockchain: Is it the future for the tourism and hospitality industry?" *Journal of Vacation Marketing*, vol. 28, no. 2, pp. 135–150, 2022, doi: 10.1177/1354816620961707.

[8] A. Boughrara, N. A. Daoud, and N. L. Harkati, "Automated organized tour planning with blockchain: An innovative solution," *International Journal of Intelligent Information and Database Systems*, vol. 18, no. 2, 2026, doi:10.1504/IJIIDS.2025.10074152.

[9] Q. Wang, R. Li, Q. Wang, S. Chen, M. Ryan, and T. Hardjono, "Exploring Web3 from the view of blockchain," 2022.

[10] A. A. Talib, M. H. Abdulkareem, S. N. Selman, and S. A. Talib, "Impact blockchain technology on traditional electronic payment system," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 32, no. 3, pp. 1703–1711, Dec. 2023, doi: 10.11591/ijeecs.v32.i3.pp1703-1711.

[11] M. Iansiti and K. R. Lakhani, "The Truth About Blockchain," *Harvard Business Review*, vol. 95, no. 1, pp. 119–127, 2017.

[12] C. Menkel-Meadow, ed., "Multi-party dispute resolution, democracy and decision-making," *Complex Dispute Resolution*, vol. 2, Routledge, 2012, p. 602. ISBN: 978-0-7546-2799-9.

[13] F. Gai, B. Wang, W. Deng, and W. Peng, "Proof of reputation: A reputation-based consensus protocol for peer-to-peer network," 2018.

[14] O. Aluko and A. Kolonin, "Proof-of-reputation: An alternative consensus mechanism for blockchain systems," *International Journal of Network Security & Its Applications (IJNSA)*,, vol. 13, no. 4, Jul. 2021, doi: 10.5121/ijnsa.2021.13403.

[15] J. M. Luo and Z. W. Hu, "Blockchain technology-based business model: A case study of travel sites," *Journal of Tourism Quarterly*, vol. 5, no. 3–4, pp. 59–76, 2023.

[16] V. Bonnici, V. Arceri, A. Diana, F. Bertini, E. Iotti, A. Levante, V. Bernini, E. Neviani, and A. Dal Palù, "BIOCHAIN: Towards a platform for securely sharing microbiological data," in *Proc. Int. Database Engineered Applications Symposium (IDEAS 2023)*, Heraklion, Crete, Greece, May 05–07, 2023, pp. 1–5, ACM, doi: 10.1145/3589462.3589501.

[17] F. Bertini, A. Benetton, and D. Montesi, "Ensuring news integrity against online information disorder through text watermarking and blockchain," *Blockchain: Research and Applications*, in press, available online 31 October 2025, article 100414, doi: 10.1016/j.bcra.2025.100414.

[18] R. H. Chowdhury, "Automating supply chain management with blockchain technology," *World Journal of Advanced Research and Reviews*, vol. 22, no. 3, pp. 1568–1574, 2024, doi: 10.30574/wjarr.2024.22.3.1895.

[19] L. Li, P. Shi, X. Fu, P. Chen, T. Zhong, and J. Kong, "Three-Dimensional Tradeoffs for Consensus Algorithms: A Review," *IEEE Transactions on Network and Service Management*, vol. 19, no. 2, pp. 1121–1140, 2022, doi: 10.1109/TNSM.2021.3133933.

[20] D. S. Kumar and K. Jayasurya, "Fintech's Hidden Hand: How PayPal Reshaped the Financial World," 2025.

[21] E. Androulaki *et al.*, "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," in *Proceedings of the Thirteenth EuroSys Conference (EuroSys '18)*, Article No. 30, pp. 1–15, ACM, 2018, doi: 10.1145/3190508.3190538.
[22] D. Vujičić, D. Jagodić, and S. Randić, "Blockchain Technology, Bitcoin, and Ethereum: A Brief Overview," in *Proceedings of the 17th International Symposium INFOTEH-JAHORINA*, East Sarajevo, Bosnia and Herzegovina, pp. 1–6, 2018, doi: 10.1109/INFOTEH.2018.8345547.
[23] A. Yakovenko, "Solana: A New Architecture for a High Performance Blockchain," white paper, version 0.8.13, 2018.

BIOGRAPHIES OF AUTHORS

Zohra Temmar obtained from the University of Science and Technology of Oran, Mohamed Boudiaf (USTO-MB), Algeria, her License degree in computer science in 2023 and her Master's degree in Networks and Distributed Systems in 2025. In 2025, she joined the Department of Computer Science at USTO-MB, where she currently serves as a temporary teacher. Her main research areas are distributed systems, blockchain technology, consensus mechanisms, and smart-contract–based collaborative tourism systems, which aligns with her Master's thesis on optimizing a decentralized tourism platform using blockchain. She can be contacted at: zahratemmar@gmail.com.

Asmaa Boughrara obtained from the University of Science and Technology of Oran, Mohamed Boudiaf (USTO-MB), Algeria, her License degree in computer science in 2007 and MSc degree in Computer Systems and Networks in 2009. She pursued her PhD degree in Computer Systems and Networks from USTO-MB, in 2015. In 2016, she joined the Department of Computer Science at USTO-MB and became lecture class B in 2017. Her main research areas are distributed systems, blockchain, and routing and switching. She can be contacted at: asmaa.boughrara@univ-usto.dz.