Proceedings of the International Conference IMCS-60, MSU
October 10–13, 2024, Chisinau, Republic of Moldova

TransShare: A Peer-to-Peer Car Sharing Platform

Amalia-Nicola Harton, Cosmin Irimia, Madalin Matei

Abstract

This paper presents the implementation details and technological framework of "TransShare", a distributed web application designed to facilitate the temporary sharing of private vehicles between users. The platform supports both the listing and rental of cars within the territory of Romania. The implementation has been reimagined using a peer-to-peer (P2P) mechanism to decentralize operations, improving system efficiency, scalability, and reducing reliance on central servers. Additionally, the platform integrates OpenAI for enhanced interaction but introduces an optimization layer to mitigate potential operational costs associated with AI queries. The system was developed using modern web technologies, namely the MERN (MongoDB, Express.js, React, and Node.js) stack, and adheres to the Model-View-Controller (MVC) architecture to ensure scalability, maintainability, and modularity.

Keywords: Peer-to-peer car sharing, AI-driven optimization, cost-efficient operations, MERN stack architecture.

1 Introduction

The rise of shared mobility solutions and decentralized architectures has revolutionized how users access transportation. "TransShare" seeks to capitalize on this trend by offering a web-based peer-to-peer (P2P) [1] system for car sharing. The platform allows private car owners to list their vehicles for temporary rental, while users seeking cars can

©2024 by Amalia-Nicola Harton et al.

browse and filter based on various criteria. This approach fosters a more decentralized and efficient method for sharing resources, reducing dependency on central server infrastructures and increasing scalability.

Moreover, the platform leverages OpenAI [6] to provide users with real-time assistance. However, recognizing the cost implications of frequent interactions with AI, we introduce an additional optimization layer to manage and minimize the operational costs related to OpenAI queries, ensuring a balance between AI functionality and affordability.

2 Proposed system

Traditional client-server models in web applications often face limitations in scalability and centralization of control. In contrast, peer-to-peer (P2P) systems decentralize operations [5], enabling direct interactions between users. This paradigm shift has significant implications for platforms such as "TransShare".

2.1 Peer to peer model

The P2P model within "TransShare" allows users (both renters and car owners) to communicate and exchange information directly, bypassing the need for a central server to manage all transactions. This is achieved by employing distributed technologies where each user acts as both a client and a server, managing their data locally while sharing only the necessary information for transaction processing.

The P2P system in "TransShare" functions by allowing users to upload, store, and access vehicle data locally. A distributed ledger or decentralized database can be used to maintain records of all transactions and agreements. Once a car rental request is accepted, direct peer-to-peer communication is established through the platform's secure messaging service, allowing the renter and owner to finalize logistics without the need for continuous server mediation.

2.2 Cost optimizations

A major feature of "TransShare" is the integration of a virtual assistant powered by OpenAI, designed to provide real-time assistance to users. This AI-driven feature can answer inquiries about vehicle details, rental processes, and other relevant information, significantly improving the user experience.

However, the frequent use of OpenAI-based queries can lead to high operational costs [3]. To address this, an optimization layer is introduced to manage and reduce the frequency of unnecessary AI interactions while maintaining responsiveness.

The optimization layer functions by filtering user queries through a tiered system:

- Local Cache of Frequent Queries: Commonly asked questions and standard responses are cached locally. The system retrieves answers from this local database before making a call to the OpenAI service, thereby reducing the need for frequent API requests.

- Contextual Understanding: The system leverages natural language processing (NLP) to determine whether a query requires a fresh response from OpenAI or can be answered using previous interactions. This approach minimizes redundant calls to the API.

- Usage Monitoring and Rate Limiting: The platform monitors AI query usage patterns and can enforce rate-limiting measures to avoid excessive API calls during periods of heavy usage.

- Batching and Delayed Responses: Non-urgent queries can be batched and processed during off-peak hours or combined into a single OpenAI request, reducing the overall number of API interactions.

By implementing this optimization layer, "TransShare" ensures that users benefit from the real-time intelligence provided by OpenAI

without incurring prohibitive costs. This solution balances the need for advanced AI-driven features with the economic realities of maintaining a cost-effective service.

The system leverages the MERN stack [4] (MongoDB, Express.js, React, and Node.js), which supports both the P2P architecture and the integration of OpenAI. MongoDB, a NoSQL database, serves as the decentralized database for storing vehicle listings and transaction histories, ensuring scalability and efficient data retrieval.

The application's frontend, developed using React, provides an intuitive user interface for listing and searching vehicles. Node.js serves as the runtime environment for the server-side code, enabling the seamless handling of network requests. Express.js manages the server infrastructure and facilitates secure communication between users in the P2P environment.

2.3 Features and Functionality

"TransShare" provides a comprehensive set of features for both car owners and renters. Users listing a vehicle can input detailed specifications, accompanied by high-quality images, while renters can search for vehicles based on a wide array of criteria such as location, model, price, and date. Once a rental request is accepted, secure direct messaging is facilitated through the P2P network, allowing users to finalize logistics without requiring continuous server-side management.

Additionally, OpenAI assists users by answering questions related to the platform, enhancing overall usability. The introduction of an optimization layer ensures that the costs associated with this AI interaction are kept in check.

3 Conclusions

This paper has outlined the implementation of "TransShare," a distributed P2P web application designed to facilitate car sharing in Romania. The transition to a peer-to-peer architecture enhances scalabil-

ity, efficiency, and reliability, while the integration of OpenAI, complemented by an optimization layer, provides intelligent real-time interaction at a manageable cost. Through the use of the MERN stack and adherence to the MVC architecture, "TransShare" achieves a balance between performance, scalability, and user experience, establishing itself as a forward-thinking solution for peer-to-peer car sharing.

References

[1] Zsolt Nyéki, Miklós Kozlovszky, Gábor Hermann, László Kovács, and Miklós Arvay. "Cost-based optimization of scientific workflows in cloud and grid environments," *Concurrency and Computation: Practice and Experience*, vol. 18, no. 13, pp. 1741–1751, 2006. DOI: 10.1002/cpe.1189.

[2] Zahra Ashktorab and Mashael D. Alrashidi. "The Role of Cloud Computing Technology in Business Organizations," *Journal of International Technology and Information Management*, vol. 22, no. 3, Article 6, 2013. Available at: https://scholarworks.lib.csusb.edu/jitim/vol22/iss3/6.

[3] Krishnadas Nanath and Radhakrishna Pillai. "A Model for Cost-Benefit Analysis of Cloud Computing". `https://scholarworks.lib.csusb.edu/cgi/viewcontent.cgi?article=1017&context=jitim`

[4] Sanchit Aggarwal and Jyoti Verma. "Comparative analysis of MEAN stack and MERN stack". Available at: https://www.ijrra.net/Vol5issue1/IJRRA-05-01-26.pdf.

[5] N. El Ioini and C. Pahl. "A Review of Distributed Ledger Technologies," in *Confederated International Conferences: CoopIS, C&TC, and ODBASE 2018, Valletta, Malta, October 22-26, 2018, Proceedings, Part II*, 2018, pp. 277–288. DOI: 10.1007/978-3-030-02671-4_16. Available at: https://link.springer.com/chapter/10.1007/978-3-030-02671-4_16.

[6] Konstantinos I. Roumeliotis and Nikolaos D. Tselikas. "ChatGPT and Open-AI Models: A Preliminary Review," *Future Internet*, vol. 15, no. 6, Article ID. 192, 2023. DOI: https://doi.org/10.3390/fi15060192. Available at: https://www.mdpi.com/1999-5903/15/6/192.

[7] Souhaila Serbout and Amine El Malki. "API Rate Limit Adoption – A pattern collection". Available at: https://dl.acm.org/doi/abs/10.1145/3628034.3628039.

Amalia-Nicola Harton[1], Cosmin Irimia[2], Madalin Matei[3]

[1 2 3]Faculty of Computer Science, University Alexandru Ioan Cuza Iasi

[1]ORCID: https://orcid.org/0009-0006-4316-0065
E–mail: hartonamalia99@yahoo.com
[2]ORCID: https://orcid.org/0009-0002-7536-1161
E–mail: cosmin.irimia@info.uaic.ro
[3]ORCID: https://orcid.org/0009-0003-5629-9857
E–mail: bmatei.madalin@info.uaic.ro