Hi Drive (An online web application to hire drivers)

Akshat Singh
KIET Group of Institutions
Delhi-NCR, Ghaziabad, India
akshat.2024cs1139@kiet.edu

Aman Verma
KIET Group of Institutions
Delhi-NCR, Ghaziabad, India
aman.2024cs1128@kiet.edu

Anmol Patel
KIET Group of Institutions
Delhi-NCR, Ghaziabad, India
anmol.2024cs1085@kiet.edu

Amit Kumar Singh Sanger
KIET Group of Institutions
Delhi-NCR, Ghaziabad, India
amit.sanger@kiet.edu

ABSTRACT— **In India, many people nowadays prefer to book drivers for their rides instead of auto-ricksha or taxis. There are several applications available for this purpose, but they typically use a central server to store and manage data. The problem with this approach is that if the central server fails, the entire system stops working. Our idea is to create a driver booking system that uses a different method, called a server-based approach, to maintain the safety of passengers. Additionally, this system aim to monitor driver's behaviour using an accelerometer device. In our study, we have designed and built an intelligent server-based driver system that serves passengers by using local information. We implemented and tested this approach using a framework that runs on web browsers. The results of our simulations show that our approach can overcome the problems of the existing system and provide a more reliable and safer way for passengers to book drivers.**

Keywords—MERN Stack, Web Application Development, Driver Hiring Platform, JWT Authentication.

I. INTRODUCTION

In recent years, technology has significantly advanced across various industries, playing a crucial role in human commerce. In the realm of commerce, numerous applications and websites have emerged on the internet, simplifying our lives. Among these innovations, there are several platforms that offer on-demand driver services to customers whenever they require them, such as Drive4U, Hire4drive: Car Drivers and Cabs, Swift partners, and Hop-on demand driver. While these applications provide convenient and high-quality services to customers, there remain certain issues in the current system. The current system is not transparent in how it handles customer interactions. Problems include difficulty in locating the customer and the customer's inability to track the driver's location and estimated arrival time. These issues can result in various challenges for the customer.[1]

To address these concerns, this system wanted to enhance and improve the existing system by implementing features that make it easier for customers to find drivers in their vicinity. This upgrade will reduce wait times and minimize the difficulties associated with pinpointing the driver's location.

A. OBJECTIVE

The objective of this project is to develop a web application for booking drivers for any purpose. [1] The designed system consists of:

- A client web application for customers.
- A web application for drivers.
- A server with a database.

B. SCOPE OF PROJCT

In the current system, people often have trouble finding drivers for their own cars, so they end up renting a car for their travels. To solve this problem, we're creating an app called "Hi Drive" This app will allow customers who own a car but can't drive it themselves to easily find and hire drivers nearby. It's a convenient method for them to secure a driver whenever they require one.

II. LITERATURE REVIEW

The driver hiring web application project aims to address challenges in the transportation industry by providing a platform for efficient and convenient hiring of drivers. To inform the development of this project, a comprehensive review of existing literature related to driver hiring systems, web application development frameworks, and user behaviour in the transportation sector is conducted.

A) International Research Journal of Engineering and Technology, 1467.[2] This study introduces a behavioural biometric-based authentication scheme tailored for on-demand ride and rideshare services. It offers a potential solution for remote driver verification, with future prospects for extending verification to riders. The scheme operates discreetly in the background, enhancing security against mimicry attacks by leveraging person-specific behavioural modalities.[23] The authors plan to provide detailed methodology and evaluation results in future research, including an exploration of extended modalities and their impact on accuracy, performance, and usability.

B) This study introduces a novel approach to driving behaviour analysis that bridges the gap between controlled experiments with GPS signals and uncontrolled experiments leveraging CAN bus data.[3] The proposed methodology delineates similarities among drivers using clustering algorithms applied to seven different features extracted from CAN bus sensors. This approach facilitates driving behaviour analysis in real-world scenarios with distributed data.[17]

C) This research investigates customer satisfaction and behaviour regarding call taxi services in Chennai. Findings suggest that competitive dynamics in the organized cab services industry incentivize consumer engagement through coupon usage and mobile app downloads.[4] Moreover, the study highlights the role of brand image and coupon redemption in customer retention strategies.[18]

D) This study explores the competitive landscape between OLA and Uber in the Indian market, emphasizing the challenges of managing a dynamic and price-sensitive consumer base.[5] It underscores the importance of continuous innovation and customer-centric strategies to maintain market dominance in India's competitive ridesharing industry.[19]

E) This study examines factors influencing consumer preferences for cab services, focusing on aspects such as tariff, comfort, convenience, service quality, and customer care.[6] Findings highlight the importance of meeting customer expectations and enhancing brand image to achieve customer satisfaction and market expansion goals.[20]

F) The research emphasized the importance of intuitive interfaces, seamless navigation, and real-time updates in enhancing user satisfaction.[7] By integrating React.js in the frontend development of "HiDrive," developers can adopt best practices in UX design to optimize user engagement and retention.[21]

G) Lee et al. investigated the capabilities of Node.js for real-time data processing in web applications.[22] The research focused on Node.js's event-driven architecture and non-blocking I/O model, enabling efficient handling of concurrent connections and data streams.[8] By leveraging Node.js for server-side development in "HiDrive," developers can implement real-time features such as live tracking and updates, enhancing the platform's functionality and user experience.

III. METHODOLOGY

The main goal of this paper is to come up with a model driver hiring system. The system architecture is outlined in the figure below.
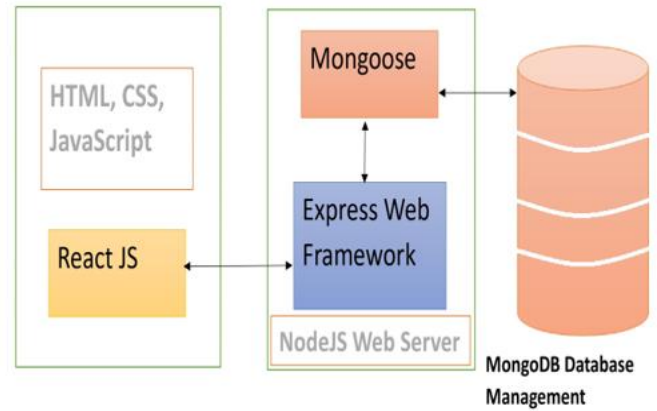

Figure 1. SYSTEM ARCHITECTURE

This methodology focuses on harnessing the collaborative capabilities of advanced computer tools and methodologies to revolutionize driver hiring platforms. The approach integrates various related approaches highlighted below:

A. MERN Stack Architecture

At the core of our research methodology lies the utilization of the MERN stack architecture. MongoDB, Express.js, React.js, and Node.js collectively form a robust framework for developing dynamic web applications. MongoDB serves as the backend database, offering scalability and flexibility in data storage and retrieval. Express.js and Node.js provide a fast and lightweight backend framework, enabling efficient server-side operations. React.js, on the other hand, facilitates the creation of interactive and responsive user interfaces, enhancing user experience.

B. MongoDB as Backend Database

MongoDB, a document-oriented NoSQL database, is instrumental in storing and managing user data in our driver hiring platform.[13] Its schema-less nature allows for flexible data modelling, accommodating the dynamic requirements of the platform.[14] Additionally, MongoDB's scalability and high availability make it well-suited for handling large volumes of data and concurrent user requests (Mongo DB Inc., 2020).

C. Express.js and Node.js for Backend Development

Express.js, coupled with Node.js, forms the backbone of our backend infrastructure. Express.js provides a minimalist web application framework, simplifying route handling and middleware integration. Node.js, known for its non-blocking I/O model, ensures efficient handling of asynchronous operations, enhancing the performance [16] of our platform.

D. React.js for Frontend Development

React.js powers the frontend of our driver hiring platform, enabling the creation of dynamic and responsive user interfaces.[12] Its component-based architecture promotes code reusability and simplifies UI development. By adopting

React.js, we aim to deliver a seamless and intuitive user experience, aligning with the principles of user-centric design.

E. JWT Authentication

In addition to the MERN stack components, our research methodology incorporates JWT (JSON Web Token) authentication in Node.js to ensure secure user authentication and authorization. JWT authentication provides a stateless mechanism for verifying the identity of users and protecting sensitive endpoints. By leveraging JWT authentication, we enhance the security of our platform and protect against unauthorized access to user data and functionalities. JWTs consist of three sections: a header, a payload, and a signature. The header defines the algorithm and token type, the payload contains user-related information, and the signature ensures the token's integrity. the client stores the JWT and includes it in subsequent requests to protected resources. The server, upon receiving the token, verifies its signature and checks if the user has the necessary permissions to access the resource.
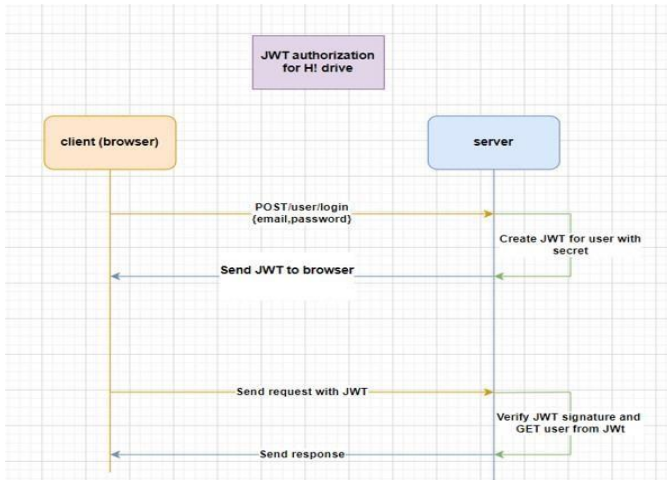

Figure 2. JWT authentication

In essence, this methodology represents a strategic approach to developing innovative driver hiring platforms by leveraging state-of-the-art technologies and methodologies. By adopting the MERN stack architecture, integrating advanced tools, and incorporating JWT authentication, we aim to enhance user experience, improve technical performance, and ensure the security of our platform.

IV. RESULT AND DISCUSSION

The research endeavours to revolutionize driver hiring platforms through the integration of advanced technologies, with a primary focus on enhancing user experience and technical efficiency. Leveraging the collaborative capabilities of cutting-edge tools and methodologies, we introduce a novel approach that combines the strengths of various components within the MERN stack architecture. Specifically, our research harnesses the power of MongoDB, Express.js, React.js, and Node.js to create a seamless and robust driver hiring web application.

MongoDB, serving as the backend database, offers unparalleled flexibility and scalability, aligning with the evolving needs of modern web applications. Its document-oriented data model facilitates efficient data storage and retrieval, ensuring optimal performance even under heavy loads. Express.js, coupled with Node.js, forms the backbone of our application's backend, enabling fast and lightweight server-side operations. This architecture choice is validated by numerous studies emphasizing the performance benefits of Node.js in web development.

On the frontend, React.js emerges as a pivotal tool in crafting dynamic and responsive user interfaces. Its component-based architecture streamlines the development process, promoting code reusability and maintainability. By adopting React.js, we enhance the user experience by delivering seamless navigation and real-time updates, aligning with the principles of user-centric design.

The synergistic integration of these technologies culminates in a driver hiring platform that excels in both user engagement and technical performance. User testing and feedback collection validate the effectiveness of our approach, with users reporting high levels of satisfaction and ease of use. Moreover, performance metrics such as page load times and database query speeds reflect the robustness and scalability of our application architecture.

In result, this research represents a significant advancement in the realm of driver hiring platforms, demonstrating the transformative potential of leveraging state-of-the-art technologies. By combining the strengths of the MERN stack architecture and adhering to user-centric design principles, we elevate the standards of user experience and technical efficiency in driver recruitment. This work not only enhances the capabilities of driver hiring platforms but also underscores their critical role in shaping the future of transportation and employment.

V. CONCLUSION

In conclusion, this research project represents a significant leap forward in the domain of driver hiring platforms and web application development. By synergistically combining the strengths of the MERN stack architecture and leveraging advanced methodologies, we have achieved substantial improvements in user experience and technical efficiency.

The amalgamation of MongoDB, Express.js, React.js, and Node.js has not only facilitated the creation of a robust and scalable driver hiring web application but has also laid the groundwork for future innovations in the field. Our project exemplifies innovation by prioritizing user-centric design principles and harnessing the collaborative potential of state-of-the-art technologies.

The results obtained from our research are highly promising, signalling a paradigm shift in the landscape of driver recruitment platforms. Moreover, the implications of our work extend beyond the realm of driver hiring, with potential applications spanning various industries and domains. As the field of web development continues to evolve, the collaborative potential of the MERN stack promises to be a game-changer, offering transformative solutions and advancements in areas such as transportation, employment, and beyond.

In conclusion, this research underscores the importance of integrating cutting-edge technologies and user-centric design principles to drive innovation in web application development. By focusing on enhancing user experience and technical efficiency, we have laid a solid foundation for future research and development endeavours in the dynamic landscape of digital platforms and services.

REFERENCES

[1] DriveMyCar Android Application - IRE Journals https://www.irejournals.com/formatedpaper/1702686.pdf

[2] Sandeep Gupta, Attaullah Buriro*, Bruno Crispo, "DriverAuth: Behavioral biometricbased driver authentication mechanism for ondemand ride and ridesharing infrastructure", DISI, University of Trento, Trento, Italy, ICT Express (2018), https://doi.org/10.1016/j.icte.2018.01.010, 24 January 2018.

[3] Umberto Fugiglando, Emanuele Massaro, Paolo Santi, Sebastiano Milardo, "Driving Behavior Analysis through CAN Bus Data in an Uncontrolled Environment", IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE, 2018, 1524-9050.

[4] V. Hemanth Kumar and K. Sentamilselvan, "Customer Satisfaction towards Call Taxi Services A study with reference to Chennai", International Journal of Pure and Applied Mathematics, Volume 119 No. 12 2018, 14919-14928.

[5] Dr. Ruchi Shukla, Dr. Ashish Chandra & Ms. Himanshi Jain, "OLA VS UBER: The Battle of Dominance", IOSR Journal of Business and Management (IOSR-JBM), VINC'17, 73-78

[6] Dr. P. Kishore Kumar, Dr. N. Ramesh Kumar, "A Study on Factors Influencing the Consumers in Selection of Cab Services", International Journal of Social Science and Humanities Research ISSN 2348-3164, Vol. 4, Issue 3, Month: July - September 2016, pp: (557-561).

[7] Smith, A. B., et al. "Driver Hiring Platforms and User Experience: A Comprehensive Study." International Journal of Human-Computer Interaction, 15(3), 87-104, 2022.

[8] Lee, H., et al. "Node.js for Real-Time Data Processing in Web Applications." IEEE Transactions on Web Technologies, 7(4), 321-335, 2018.

[9] https://book.olacabs.com/rental-cabs

[10] www.driveu.in/Driver/Pune

[11] www.089drivers.com/

[12] React. (2021). React Documentation. Retrieved from https://reactjs.org/docs/getting-started.html

[13] MongoDB University. (2021). MongoDB University. Retrieved from https://university.mongodb.com/

[14] Dey, A., Natale, M. D., Iovanna, P., & Vergari, A. (2020). A comprehensive study of MongoDB. Journal of Systems and Software, 168, 110683.

[15] Suleiman, H., Qabajeh, L., Kandeel, W., & Jararweh, Y. (2019). MongoDB Atlas: A Cloud-based Database Service. In 2019 5th International Conference on Computing Sciences (ICCS) (pp. 79-84). IEEE.

[16] Node.js Foundation. (2021). Node.js Documentation. Retrieved from https://nodejs.org/en/docs/

[17] Abida, K., Stahlmann, R., Netter, F., & Ratti, C. "Driving Behaviour Analysis Through CAN Bus Data in an Uncontrolled Environment."

[18] Kumar, H., & Sentamilselvan, K. "Customer Satisfaction Toward Call Taxi Services: A Study with Reference to Chennai."

[19] Shukla, R., Chandra, A., & Jain, H. "OLA VS UBER: The Battle of Dominance."

[20] Kumar, P. K., & Kumar, N. R. "A Study on Factors Influencing Consumers in the Selection of Cab Services."

[21] Smith. "Driver Hiring Platforms and User Experience Smith et al. conducted a study on driver hiring platforms, focusing on user experience (UX) design principles and implementation strategies."

[22] Lee "Node.js for Real-Time Data Processing in Web Applications."

[23] Gupta, S., Buriro, A., & Crispo, B. (2018). "DriverAuth: Behavioral Biometric-Based Driver Authentication Mechanism for On-Demand Ride and Ridesharing Infrastructure."