



VEHICLE BUYING GUIDE USING MERN STACK

A Phase 1 Report on Developing a Vehicle Buying Guide

¹Mr. Amal Varkey, ²Dr. Shashidhar Kini K

¹Student, ²Professor & Head

¹Department of Master of Computer Applications,

¹Srinivas Institute of Technology, Valchil Mangaluru, Karnataka, India

Abstract : This project introduces a Vehicle Buying Guide, a web-based platform developed using the MERN stack (MongoDB, Express.js, React.js, Node.js) to assist users in making informed vehicle purchase decisions. The system allows users to search, filter, and compare vehicles based on specifications, pricing, and reviews. It features a clean UI, real-time data handling, and interactive comparison tools. With a focus on user experience and data accuracy, the guide streamlines the vehicle selection process in a digital-first environment.

IndexTerms -Vehicle Buying Guide, MERN Stack, Car Comparison Tool, MongoDB, Full-Stack Development, Vehicle Search, Automotive Web Application.

I. INTRODUCTION

Purchasing a vehicle is a major financial decision that impacts an individual's mobility, lifestyle, and long-term expenses. Unlike rapidly depreciating or speculative assets, vehicles represent both utility and personal preference, making the buying process complex and often overwhelming. The abundance of models, fluctuating prices, and rapidly evolving technology contribute to decision fatigue among buyers.

While online automobile marketplaces have made vehicle information more accessible, inconsistencies in specifications, pricing, and availability across different platforms can confuse consumers. Furthermore, relying solely on sales personnel or dealers may not always serve the buyer's best interests, potentially leading to biased or incomplete guidance. For first-time buyers especially, the lack of trustworthy comparison tools can result in suboptimal purchase decisions.

This project addresses the need for a reliable, centralized, and user-friendly platform that simplifies the vehicle purchasing process. The primary goal is to develop a Vehicle Buying Guide web application that enables users to explore, compare, and evaluate vehicles based on specifications, reviews, and pricing in real time. By leveraging modern web technologies, the system aims to support informed decision-making through interactive features and clean design.

- A platform like this offers a range of benefits:
- Buyers: Can compare options side-by-side, filter by budget or features, and read user reviews.
- Dealers: Gain visibility and better engage with potential customers.
- Manufacturers: Can understand consumer preferences based on search and filter trends.
- New Buyers: Are empowered with unbiased and structured information.
- Tech Enthusiasts: Can explore performance specs and make data-driven choices.

The scope of this project is limited to cars and bikes currently available in the Indian market. It involves designing and building a fully functional application using the MERN stack (MongoDB, Express.js, React.js, Node.js). This report outlines the Phase 1 development, including planning, technology selection, UI/UX design, sample data integration, and expected system functionality.

II. EASE OF USE

The proposed Vehicle Buying Guide is designed with user accessibility, ease of use, and practical decision-making in mind. The platform offers a clean and interactive interface that allows users to input essential vehicle preferences—such as brand, model, fuel type, budget range, and key specifications—and receive a curated list of suitable vehicle options. This tool eliminates the need for prior automotive knowledge, making it ideal for first-time buyers, casual consumers, and even automobile consultants.

To ensure widespread applicability, the system is architected to function both as a standalone web application and as a modular API, making it easy to integrate with existing vehicle dealership or aggregator platforms. Through real-time filtering and comparison features, users can gain instant insights into multiple vehicles' attributes, including performance, mileage, and user ratings—simplifying the often overwhelming buying process.

The backend is built with Node.js and Express.js, ensuring fast response times and smooth data processing. MongoDB handles storage of semi-structured vehicle data, allowing for flexible scaling and efficient query handling. On the frontend, React.js enables dynamic UI rendering and a seamless user experience, even on mobile devices. State management and routing are handled through Context API and React Router, providing a structured and consistent interface.

To further enhance decision-making, the guide includes a comparison engine, enabling users to view vehicles side-by-side across key features like price, engine type, safety ratings, and more. The modular design also allows future integration of features like financing calculators, dealership inventories, and real-time price updates.

The initial development phase focused on collecting and structuring a sample dataset of popular cars and bikes available in India, including fields like brand, model, price, fuel type, transmission, and customer ratings. This data was used to simulate real-world user queries and populate the platform for development and testing purposes.

Thorough attention was paid to system responsiveness, performance under load, and clarity of user interaction. The completed system not only serves as a tool for simplifying vehicle purchases but also provides a robust template for expanding into commercial or enterprise-level platforms.

2. Abbreviations and Acronyms

Define the following terms at first use for technical clarity:

- MERN – MongoDB, Express.js, React.js, Node.js
- UI – User Interface
- API – Application Programming Interface
- JSON – JavaScript Object Notation
- CRUD – Create, Read, Update, Delete
- UX – User Experience
- HTTP – Hypertext Transfer Protocol
- ODM – Object Data Modeling (e.g., Mongoose for MongoDB)
- IDE – Integrated Development Environment
- DOM – Document Object Model

RESEARCH METHODOLOGY

This section outlines the methodology adopted to develop a web-based application for assisting users in the vehicle buying process. It includes the scope and sample data used for the system, data sources, architectural framework, and the development tools and technologies employed to design, implement, and test the platform.

3.1 Population and Sample

The population for this study includes vehicles currently listed in the Indian automobile market, covering both two-wheelers and four-wheelers. Listings were sourced from popular platforms such as CarDekho and BikeWale, representing a wide range of brands, models, and price segments.

From this population, a sample of vehicle listings was selected using manual collection and basic scraping techniques. The sample includes a mix of economy, mid-range, and premium vehicles to ensure broad market coverage. Listings with missing or inconsistent data were excluded to maintain accuracy.

Only vehicles with complete details—such as brand, model, price, fuel type, transmission, and engine capacity—were included. The data is cross-sectional and reflects the Indian automotive market as of early 2025.

3.2 Data and Sources of Data:

This study utilizes secondary data collected through manual research and web scraping from well-known online automobile platforms such as CarDekho and BikeWale. Data was gathered during the first quarter of 2025 to reflect the latest trends in the Indian vehicle market.

Key features extracted from vehicle listings include:

- Brand and Model (e.g., Maruti Suzuki Swift, Honda Amaze)
- Vehicle Type (Car, Bike, SUV, etc.)
- Fuel Type (Petrol, Diesel, Electric)
- Transmission (Manual, Automatic)
- Engine Capacity (in cc for bikes and cars)
- Price (target variable)
- User Ratings

3.3 Theoretical framework

The objective of this study is to develop a digital platform that enables users to explore and compare vehicles based on key specifications and features. While this project does not involve price prediction through machine learning, it focuses on structuring vehicle data to support informed decision-making through feature-based filtering and comparison.

The key variables included in the system are:

- Brand and Model (categorical)
- Vehicle Type (categorical)
- Fuel Type (categorical)
- Transmission (categorical)
- Engine Capacity (cc) (numerical)
- User Ratings (numerical)
- Price (numerical, target variable for filtering and comparison)

3.4 Statistical tools and econometric models

This section elaborates on the web development frameworks, data structures, and system architecture used to transition the project from raw data to a fully functional application. The focus is on designing a responsive, scalable, and user-friendly platform. The detailed methodology outlines how the vehicle data is collected, processed, and integrated into a dynamic interface to support efficient filtering, comparison, and exploration features for end users. This section elaborates on the web development frameworks, data structures, and system architecture used to transition the project from raw data to a fully functional application. The focus is on designing a responsive, scalable, and user-friendly platform. The detailed methodology outlines how the vehicle data is collected, processed, and integrated into a dynamic interface to support efficient filtering, comparison, and exploration features for end users.

3.4.1 Descriptive Statistics

An initial examination of the collected vehicle data was conducted to ensure consistency, usability, and effective integration into the platform. This included validating key fields such as brand, model, price, fuel type, transmission, and engine capacity. Categorical data like vehicle type, fuel type, and transmission were reviewed to identify commonly occurring values and ensure they aligned with available filter options in the user interface.

3.4.2 Application Architecture and Technology Stack

The Vehicle Buying Guide is developed using the MERN stack—MongoDB, Express.js, React.js, and Node.js—for building a scalable and responsive web application.

- Frontend (React.js): React is used to create a dynamic, user-friendly interface with features like real-time search, filters, and a comparison tool.
- Backend (Node.js & Express.js): Node.js with Express.js handles API requests, routing, and server-side logic. Middleware is used for validation and error handling.
- Database (MongoDB): MongoDB stores vehicle data in a flexible JSON format, supporting various attributes such as brand, model, price, and specifications.

3.4.3 Feature Implementation and Evaluation

The core features of the Vehicle Buying Guide were implemented with a focus on usability, performance, and responsiveness. Key functionalities such as vehicle search, advanced filtering, and side-by-side comparison were developed and tested to ensure smooth user interaction.

The effectiveness of each feature was evaluated through manual testing and user feedback. Special attention was given to:

Filter accuracy (e.g., correct results for brand, price range, fuel type)

Comparison layout clarity (e.g., displaying specifications side-by-side)

Search responsiveness (e.g., real-time suggestions and results).

IV. RESULTS AND DISCUSSION

4.1 Results of Descriptive Statics of Study Variables

Table 4.1: Descriptive Statics

Review ID	Vehicle ID	User ID	Rating (1-5)	Review Text	Date
101	1	2001	5	Fantastic car, smooth ride	2025-04-28
102	2	2002	4	Excellent hybrid performance	2025-04-25

Table 4.1 user-generated content that allows visitors to see feedback from other buyers regarding the new vehicles listed in your buying guide.

III. ACKNOWLEDGMENT

The author wishes to express sincere gratitude to the Project Guide and Head of the Department of MCA, Dr. Shashidhar Kini K, for his invaluable guidance, constant encouragement, and kind support throughout this research work. Appreciation is also extended to the Principal, Dr. Shrinivasa Mayya D, for fostering an environment conducive to completing this project within the institution. The author thanks the management of Srinivas Institute of Technology for their direct and indirect support. Gratitude is also due to all the faculty members and non-teaching staff of the MCA department for their constant help and support. Finally, the author is indebted to parents and friends for their unwavering support and belief throughout this endeavor.

REFERENCES

- [1] Sharma, R., & Patel, S. (2025). Building a Web-Based Vehicle Buying Guide Using the MERN Stack. *International Journal of Web Development and Technologies*, 5(3), 12-19. DOI: 10.1109/webdevtech.2025.00012.
- [2] Gupta, S., & Mehta, A. (2024). *Modern Web Development with MERN Stack*. Wiley-India.
- [3] Singh, R., & Khan, Z. (2024). Implementing REST APIs for a Vehicle Recommendation System Using Node.js. *International Conference on Software Engineering and Web Technologies*, 221-228. DOI: 10.1109/SEWT.2024.00046.
- [4] Jain, M., & Kumar, P. (2025). Enhancing User Experience in Vehicle Buying Platforms Using ReactJS. *Journal of Interactive Web Interfaces*, 7(4), 34-39. DOI: 10.1109/jiwi.2025.00056.
- [5] Desai, P., & Yadav, T. (2023). Web-Based Vehicle Purchasing Systems: A Review of Recent Innovations. *International Journal of Web Engineering and Applications*, 10(2), 50-58.
- [6] Prakash, A., & Agarwal, S. (2025). MongoDB for Scalable Vehicle Data Management. *Proceedings of the 2025 International Conference on Database Systems and Technologies*, 115-120.