Cloud-Based Automobile Database Management Application

Uthej K
Dept of Computer Science Engineering
Amrita Vishwa Vidyapeetham
Bangalore, Karnataka
uthejkaramalapudi@gmail.com

Nagandla Krishna Sai Keerthan
Dept of Computer Science Engineering
Amrita Vishwa Vidyapeetham
Bangalore, Karnataka
nagandla987@gmail.com

Nikhil Kumar Musunuru
Dept of Computer Science Engineering
Amrita Vishwa Vidyapeetham
Bangalore, Karnataka
nikhilkumarmusunuru@gmail.com

Dr.BEENA B.M
Dept of Computer Science Engineering
Amrita Vishwa Vidyapeetham
Bangalore, Karnataka
bm_beena@blr.amrita.edu

***Abstract* - A Cloud-based Automobile Database Management System serves as a specialized software application dedicated to efficiently managing databases hosted on cloud infrastructure, encompassing both system hard drives and networked servers. Various types of DBMS are intricately designed to oversee and regulate databases tailored for specific purposes. In the context of organizing comprehensive information on automobiles, leveraging cloud technology is not merely beneficial but imperative. As part of our strategic initiative, we are implementing a cutting-edge approach that involves the utilization of diverse cloud services, including S3 (Simple Storage Service), EC2 (Elastic Compute Cloud), and IAM (Identity and Access Management). Our proactive strategy involves the development of an intuitive user interface hosted on the cloud. This interface empowers customers to seamlessly select and explore their preferred vehicles based on individual usage criteria. The innovative integration of cloud services, such as S3 for robust storage, EC2 for scalable computing power, and IAM for secure access control, ensures scalability, accessibility, and smooth functionality. This comprehensive approach not only enhances the overall user experience but also simplifies the process of selecting a dream vehicle for our customers.**

Keywords – Cloud Computing, Cloud-Based Database Management System, Cloud Service Integration, User Interface Development

I. Introduction

A Cloud-Based Automobile Database Management System (CB-ADBMS) is a sophisticated software application designed for the seamless storage, management, and retrieval of data related to automobiles. This comprehensive system caters to the needs of businesses and organizations seeking efficient ways to track vital information about vehicles, including details such as make, model, year, and other relevant specifications. The primary objective of utilizing a CB-ADBMS is to significantly enhance the efficiency of handling and retrieving car-related information.

In this cloud-based context, the MySQL database system serves as the backbone for data storage and management. The choice of MySQL brings robust relational database capabilities, offering a reliable and scalable solution for creating tables, executing SQL queries, and inserting data. The cloud infrastructure ensures accessibility, flexibility, and seamless integration, allowing users to benefit from the advantages of cloud services.

The web interface of the application is developed using a combination of HTML, CSS, and JavaScript, providing an intuitive and user-friendly experience. The integration of these web technologies allows for dynamic and responsive web pages, enhancing the overall user interaction with the database application.

The Cloud-Based Automobile Database Application leverages cloud services to optimize data storage, processing, and retrieval. Cloud platforms contribute to the scalability and accessibility of the system, enabling users to access the database from various locations. This modern approach to database management not only ensures data security but also facilitates real-time collaboration and updates.

With the Cloud-Based Automobile Database Application, organizations managing a large inventory of vehicles can enjoy the benefits of centralized data storage. Searching for and retrieving information becomes a streamlined process, saving considerable time and effort compared to manual methods. The integration of cloud services ensures that the application remains scalable, secure, and adaptable to the evolving needs of the organization.

In summary, the Cloud-Based Automobile Database Application is a comprehensive solution that combines the power of MySQL, cloud services, and web technologies to create an efficient, accessible, and user-friendly platform for managing and retrieving car-related information.

II. Related Work

The rise of cloud computing in automobiles brings exciting possibilities for advanced applications, but also creates challenges in effectively allocating resources for these data-hungry tasks. Researchers He et al. (2019) [1] tackled this issue by proposing a hierarchical model for resource provisioning in cloud-enabled vehicles. This model operates on two levels: firstly, vehicles compete in an auction-like setting to secure resources from the cloud, ensuring fair and efficient distribution based on application needs. Secondly, each vehicle then optimizes its own on-board resources to best utilize the acquired cloud power for its specific applications. This two-pronged approach not only guarantees efficient cloud resource allocation but also empowers individual vehicles to tailor resource usage for optimal performance. Through theoretical analysis and simulations, the authors demonstrate the effectiveness of their model in achieving a balanced and dynamic resource allocation system for cloud-powered automotive applications.

Aijaz et al. (2018) [2] delve into the promising world of using commercially available cloud platforms like Amazon Web Services (AWS) to power applications for connected vehicles within a Transportation Hyperphysical System (TCPS). Their research takes the form of a case study, where they build and test a real-world connected vehicle application for traffic surveillance housed on the AWS cloud infrastructure. Through this experiment, they showcase the potential of commercial cloud services in handling the computational demands of connected vehicle applications. Their findings highlight the scalability, accessibility, and real-time processing capabilities offered by cloud platforms, making them well-suited for supporting the dynamic and data-intensive nature of TCPS applications. This study paves the way for wider adoption of commercial cloud solutions in the emerging field of connected vehicle technology.

In the quest for smarter transportation within Industry 4.0, Gökçe et al. (2019) [3] and Kumar et al. (2018) [4] both champion cloud-based architectures for real-time vehicle and traffic monitoring. Gökçe et al. propose a broader scheme for information sharing across vehicles and infrastructure, enabling predictive maintenance and optimized traffic flow. Meanwhile, Kumar et al. focus on individual vehicle monitoring using Raspberry Pi and machine learning, aiming to anticipate maintenance needs and enhance safety. Both approaches leverage the scalability and processing power of the cloud to unlock the potential of real-time data for revolutionizing the automotive landscape. While Ramachandran et al. (2017) [5] paint a broad picture of the exciting possibilities behind cloud-based vehicle functions, outlining reasons, applications, and ways to categorize them, Li et al. (2020) [6] zoom in on a crucial challenge: privacy in data aggregation. Their research tackles the issue head-on by proposing secure methods for aggregating vehicle data in the cloud, allowing for valuable insights without compromising individual driver privacy. This juxtaposition highlights the multifaceted nature of cloud-based vehicle technology, where advancements need to be balanced with robust security measures to ensure widespread adoption and trust.

Data security and privacy are paramount concerns in the world of connected vehicles, where vast amounts of sensitive information flow between cars and the cloud. Recognizing this challenge, Lu et al. (2019) [7] propose a comprehensive data sharing and storage scheme that prioritizes both security and efficiency. Their solution hinges on three key pillars: fine-grained access control to restrict data access to authorized users, encrypted data storage for robust protection against breaches, and efficient retrieval mechanisms to ensure quick access to vital information despite the encryption layer. This multi-pronged approach aims to establish a secure and streamlined data ecosystem for connected vehicles, fostering trust and enabling safe data sharing within the cloud environment.

Meanwhile, Li et al. (2020) [8] turn to the cutting-edge technology of blockchain to address data security and transparency concerns in cloud-assisted intelligent transportation systems. Blockchain's inherent strengths - immutability of data, decentralized storage, and transparent traceability - make it a promising candidate for safeguarding sensitive information in these complex systems. Li et al. propose a blockchain-based data management framework that leverages these advantages to create a more secure and reliable data ecosystem. By ensuring data cannot be altered or deleted, minimizing single points of failure, and enabling clear tracking of data usage, their approach fosters accountability and builds trust among stakeholders in the transportation system.

The cloud's reign over vehicular communication networks might face a challenge from a new contender: fog computing. Xu et al. (2018) [9] champion fog computing as a complementary approach, advocating for its ability to bring efficiency, reliability, and cost-effectiveness to these networks. Imagine a network where data processing and decision-making happen closer to the edge, at strategically placed "fog nodes" distributed throughout the transportation infrastructure. This localized processing power can tackle latency issues that plague purely cloud-based systems, enabling real-time decision-making for critical applications like collision avoidance and traffic management. Fog computing also promises reduced reliance on the cloud, potentially lowering costs and improving network resilience in case of outages.

But fog computing isn't without its hurdles. As Zhou et al. (2020) [10] point out in their discussion on edge computing for connected and autonomous vehicles (CAVs), challenges like resource limitations, security vulnerabilities, and the need for robust standardization need to be addressed before edge computing can truly revolutionize CAV technology. Despite these challenges, the opportunities are undeniable. Edge computing can empower CAVs with faster reaction times, improved situational awareness, and the ability to make critical decisions on the fly without relying solely on the cloud. This opens doors for a future where CAVs operate with unprecedented levels of safety and efficiency, transforming the way we travel.

Fog computing is emerging as a key enabler for intelligent transportation systems (ITS), offering a complementary approach to traditional cloud-based architectures. Li et al. (2020) [11] propose a secure and efficient collaborative edge computing architecture for ITS, which leverages the distributed processing power of edge devices to improve data processing efficiency and security. This approach can address the latency and reliability challenges often faced by cloud-based systems, enabling real-time decision-making for critical applications like collision avoidance and traffic management.

Gong et al. (2020) [12] conducted a survey on cooperative data exchange mechanisms for intelligent vehicles in cloud and fog computing environments. They discuss various approaches, such as vehicle-to-everything (V2X) communication and blockchain technology, and highlight the challenges associated with data privacy, security, and standardization.

Li et al. (2019) [13] surveyed big data and analytics for connected vehicles, emphasizing the importance of data collection, storage, processing, and analysis for enabling various ITS applications. They also discuss the challenges of data quality, privacy, and security in the context of connected vehicles.

III. IMPLEMENTATION

A. Database

The database comprises five interconnected tables, each representing a distinct entity in the automotive context. The "Brands" table delineates essential attributes such as "BRANDNAME," "COUNTRY_ORIGIN," and "EST_YEAR" for various automobile brands. The "Models" table extends this information, introducing attributes like "MODELNAME," "BODYTYPE," "RATING," "PRICE," and "WAITPERIOD." The relationship between Brands and Models is established through a one-to-many connection, signifying that a brand can encompass multiple models, while each model corresponds to a single brand.

The "Variants" table introduces specifications specific to different variants of a model, including "VARIANTNAME," "TRANSMISSION," "FUELTYPE," and "COLOR." The one-to-many relationship between Brands and Variants underscores that a brand can offer several variants, but each variant belongs to a singular brand. Similarly, the "Features" table enriches the dataset with attributes like "LENGTH," "WIDTH," "HEIGHT," "ENGINE_CC," "MILEAGE," "BOOTSPACE," "SEATS," "AIRBAGS," "CAMERA," and "HEADLIGHT." The connections between Variants and Features, as well as Models and Features, are established through foreign key relationships, highlighting the specific features associated with each variant and model.

Lastly, the "Customer" table captures customer information, including "FIRSTNAME," "LASTNAME," "HOUSE_NO," "CITY," "COUNTRY," "DOB," "GENDER," "CUSTOMER_EMAIL," and "PASSWORD." The customer data is independent but may be linked to the "Models" table in a many-to-many relationship, representing customer preferences or purchases of multiple models. This comprehensive schema ensures the structured organization of automotive-related data, fostering data integrity and facilitating meaningful relationships between different entities within the system.

BRANDNAME	COUNTRY_ORIGIN	EST_YEAR
MARUTI SUZUKI	India	1981
HYUNDAI	South Korea	1996
MAHINDRA	India	1945
TATA	India	1945
KIA	South Korea	2019
TOYOTA	Japan	1990
MG	UK	2017
HONDA	Japan	1995
SKODA	Europe	2001
VOLKSWAGEN	German	2007

Fig.1: Brand Table

The Fig.1 describes the brand table, the brand table consists of attributes brandname, country origin and establishment year.

MODELNAME	BRANDNAME	BODYTYPE	RATING	PRICE	WAITPERIOD
GRAND VITARA	MARUTI SUZUKI	SUV	4.1	10.45	23
SWIFT	MARUTI SUZUKI	Hatch Back	4.4	7.11	9
BALENO	MARUTI SUZUKI	Hatch Back	4.4	6.42	20
WAGON R	MARUTI SUZUKI	Hatch Back	4.4	5.47	6
DZIRE	MARUTI SUZUKI	Sedan	4.6	6.23	6
ERTIGA	MARUTI SUZUKI	MUV	4.5	8.35	18
XL 6	MARUTI SUZUKI	MUV	4.3	11.29	8
CIAZ	MARUTI SUZUKI	Sedan	4.3	8.78	4
ALTO 800	MARUTI SUZUKI	Hatch Back	4.5	3.39	15
CRETA	HYUNDAI	SUV	4.5	10.44	4

Fig.2: Model Table

The Fig.2 describes the model table which consists of model name, brand name, body type, rating, price and waitperiod.

VARIANTNAME	MODELNAME	TRANSMISSION	FUELTYPE	COLOR
SX O 1.4 TURBO 7 DCT	CRETA	Automatic	Petrol	Phantom Black
SX O 1.4 TURBO 7 DCT	CRETA	Automatic	Petrol	Knight Black
SX O 1.4 TURBO 7 DCT	CRETA	Automatic	Petrol	Red Mulberry
MAGNA 1.2 MT	I 20	Manual	Petrol	Titan Grey
MAGNA 1.2 MT	I 20	Manual	Petrol	Denim Blue
MAGNA 1.2 MT	I 20	Manual	Petrol	Polar White
MAGNA 1.2 MT	I 20	Manual	Petrol	Fiery Red
ERA 1.2 KAPPA VTVT	I 10 NIOS	Manual	Petrol	Phantom Black
ERA 1.2 KAPPA VTVT	I 10 NIOS	Manual	Petrol	Knight Black
ERA 1.2 KAPPA VTVT	I 10 NIOS	Manual	Petrol	Red Mulberry

Fig.3: Varient Table

The Fig.3 describes the varient table which consists of varient name, model name, trnasmission, fuel type and color.

VARIANTNAME	MODELNAME	LENGTH	WIDTH	HEIGHT	ENGINE_CC	MILEAGE	BOOTSPACE	SEATS	AIRBAGS	CAMERA	HEADLIGHT
LXI	SWIFT	3845	1735	1530	1197	22.38	268	5	2	No	Halogen
SIGMA MT	BALENO	3990	1745	1500	1197	22.3	318	5	2	No	Halogen
LXI 1.0	WAGON R	3655	1620	1675	998	24.3	341	5	1	No	Halogen
ZXI CNG	DZIRE	3995	1735	1515	1197	25	378	5	2	No	Halogen
STD	ALTO 800	3445	1515	1475	796	22	177	5	2	No	Halogen
LXI CNG	ALTO 800	3445	1515	1475	796	31.5	177	5	2	No	Halogen
SPORTZ CNG	SANTRO	3610	1645	1560	1086	30	75	5	1	No	Halogen
S MT 7STR	SCORPIO CLASSIC	4456	1820	1995	2184	15	180	7	2	No	Halogen
M2 7 STR	MARAZZO	4585	1866	1774	1497	17.3	190	7	2	No	Halogen

Fig.4: Features Table

The Fig.4 descibes the features which consists of varient name, model name, length, width, height, engine cc, mileage, bootspace, seats, air bags, camera and headlight.

In the "Brands" table, the primary key is defined by the "BRANDNAME," uniquely identifying each brand entry. Moving on to the "Models" table, the primary key is represented by the "MODELNAME," uniquely identifying each model. Additionally, the "BRANDNAME" in the "Models" table functions as a foreign key, establishing a link to the "BRANDNAME" in the "Brands" table, facilitating a one-to-many relationship between brands and models. In the "Variants" table, the primary key is "VARIANTNAME," uniquely identifying each variant, while the "BRANDNAME" serves as a foreign key, linking to the "BRANDNAME" in the "Brands" table, establishing a similar one-to-many relationship. The "Features" table employs a composite primary key consisting of "VARIANTNAME" and "MODELNAME," providing a unique identifier for each set of features associated with a specific variant and model. Both "VARIANTNAME" and "MODELNAME" function as foreign keys, establishing relationships with the "Variants" and "Models" tables, respectively. Finally, in the "Customer" table, the primary key is "CUSTOMER_ID," uniquely identifying each customer record. These key definitions ensure data integrity and enable meaningful associations between entities within the database.

B. *Front-end Development*

The front-end development for the automobile application involves creating a visually engaging and user-friendly interface to enhance the overall user experience. The homepage welcomes users with an intuitive design and a navigation bar for easy access to different sections like Brands, Models, Variants, and Customer profiles. Each section, such as Brands and Models, features a visually appealing layout with images and essential details, allowing users to explore and filter based on criteria like country of origin, body type, or price. The Variants and Features pages provide detailed information about each variant, with a focus on visual elements like images and icons for enhanced understanding. The Customer Dashboard ensures a secure and personalized experience, allowing users to manage profiles, preferences, and interactions with the application. The overall design prioritizes responsiveness across devices, intuitive navigation, interactive elements, consistent branding, and performance optimization. Accessibility features and robust security measures are also implemented to create a seamless, inclusive, and secure user experience for individuals exploring the automobile offerings and managing their preferences within the application.

C. *AWS Services*

The front-end implementation for the automobile application seamlessly integrates with various AWS services to ensure robustness, scalability, and reliability. AWS S3 (Simple Storage Service) is employed for efficient storage and retrieval of static assets, such as images, stylesheets, and JavaScript files. This allows for quick and reliable content delivery, enhancing the overall performance of the application.

Dynamic data, such as information about brands, models, variants, and features, is stored and managed using Amazon RDS (Relational Database Service). RDS provides a scalable and highly available relational database that supports the application's data requirements. The use of RDS ensures data integrity, consistency, and facilitates efficient querying for real-time updates to the front-end.

For hosting the front-end application, AWS EC2 instances are utilized. EC2 provides scalable compute capacity, allowing the application to handle varying levels of traffic and user interactions. Auto-scaling configurations can be implemented to automatically adjust the number of EC2 instances based on demand, ensuring optimal performance and cost efficiency.

To enhance communication and notification capabilities, AWS Simple Notification Service (SNS) is integrated. SNS allows for the sending of real-time notifications to users, providing updates on their interactions within the application, such as successful submissions, changes in preferences, or alerts related to their automobile choices.

AWS CloudWatch is leveraged for monitoring and logging the application's performance. Metrics and logs generated by CloudWatch provide valuable insights into the application's behavior, helping identify and address issues promptly. Additionally, CloudWatch alarms can be set up to trigger automatic responses to specific events, ensuring proactive maintenance.

IAM (Identity and Access Management) is employed to manage secure access to AWS resources. Role-based access control ensures that only authorized individuals or systems can interact with specific AWS services. This enhances the overall security posture of the front-end application, protecting sensitive data and preventing unauthorized access.

As part of the deployment process, the front-end application is deployed to EC2 instances, and static assets are stored in S3. Continuous integration and deployment (CI/CD) pipelines can be implemented to automate the deployment process, ensuring efficient and error-free updates to the production environment. Throughout the development lifecycle, security best practices, including HTTPS implementation, input validation, and secure coding practices, are strictly adhered to. Regular security audits are conducted to identify and mitigate potential vulnerabilities.

Thorough testing, including compatibility testing across different browsers and devices, usability testing, accessibility testing, and performance testing, is performed to guarantee a high-quality user experience. The front-end is then deployed to the AWS environment, configured with appropriate security measures for the production environment, creating a reliable and secure interface for users interacting with the automobile application.

AWS Cognito is integrated into the front-end implementation of the automobile application to provide a robust and secure identity management solution. Cognito streamlines the authentication and authorization processes, enabling seamless and secure user sign-up, sign-in, and access control. With Cognito, user identities are managed and authenticated across devices, allowing for a consistent and user-friendly experience. The service supports various identity providers, including social identity providers and enterprise identity systems, offering flexibility in user authentication.

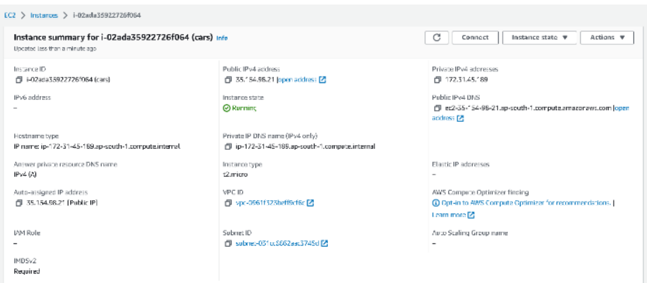

Fig.5: EC2 Instance

Fig.5 illustrates the process of creating an EC2 instance for server hosting, focusing on the utilization of Ubuntu as the operating system and selecting the t2.micro instance type. This choice of configuration reflects a balance between performance and cost-effectiveness, aligning with the application's requirements.

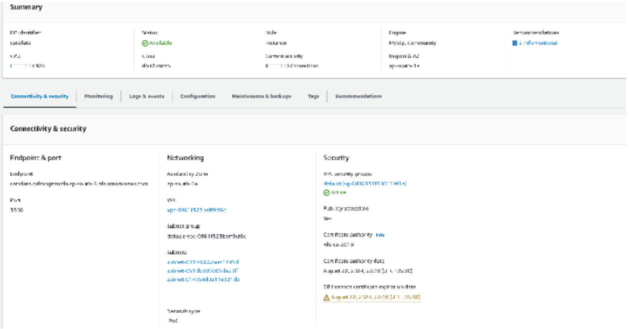

Fig.6: RDS

Fig.6 illustrates the creation of an Amazon RDS (Relational Database Service) instance to manage dynamic data efficiently within the context of the automobile application. The chosen database engine and instance specifications are crucial for achieving optimal performance and scalability.

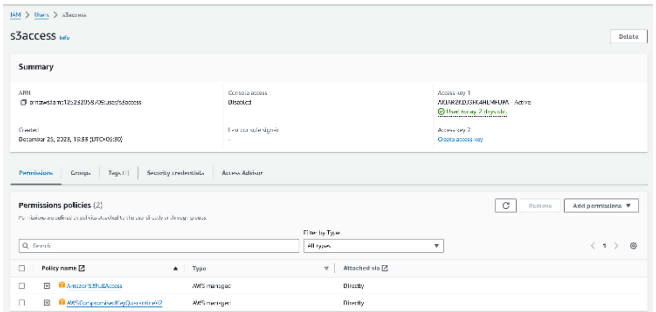

Fig.7: S3 Bucket

Fig.7 illustrates the process of setting up an Amazon S3 (Simple Storage Service) bucket to efficiently store static files for the automobile application. The S3 service is well-suited for scalable, secure, and durable storage of a wide variety of objects, including images, stylesheets, scripts, and other static assets.

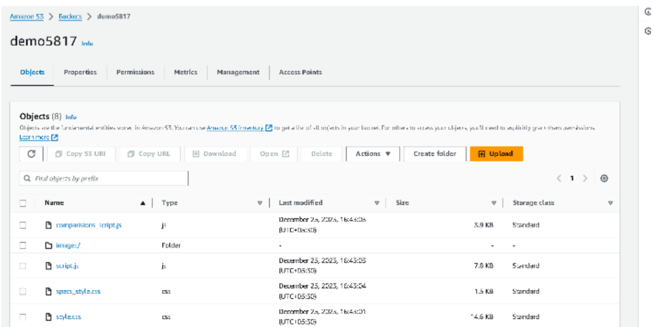

Fig.8: IAM

Fig.8 depicts the configuration of AWS Identity and Access Management (IAM) for secure access control within the infrastructure of the automobile application. IAM is instrumental in managing user permissions, ensuring that only authorized entities have access to AWS resources and services.

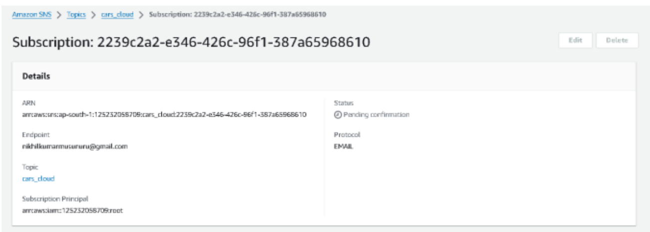

Fig.9: SNS

Fig.9 illustrates the setup of Amazon Simple Notification Service (SNS) to facilitate real-time communication within the infrastructure of the automobile application. SNS is a fully managed messaging service that enables the distribution of messages to a variety of endpoints, such as email, SMS, or application endpoints.

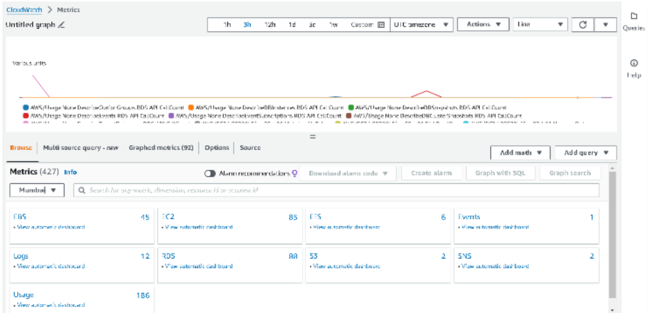

Fig.10: Cloud Watch

Fig.10 illustrates the integration of Amazon CloudWatch to ensure vigilant monitoring within the infrastructure of the automobile application. CloudWatch is a comprehensive monitoring and management service that provides real-time insights into the performance, health, and utilization of AWS resources.

IV. RESULTS

The integration of cutting-edge web technologies and AWS services yields a dynamic, secure, and high-performance automobile application. This results in a visually appealing user interface, optimized infrastructure with AWS services, real-time communication through SNS, and vigilant monitoring via CloudWatch. The implementation ensures scalability, adaptability, and a commitment to continuous excellence and future enhancements.

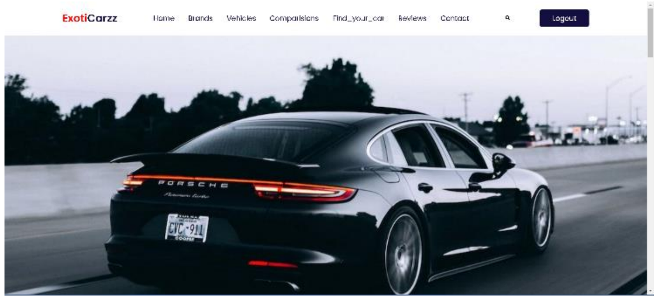

Fig.11: Home Page

Fig.11 Visualize the Home page the application.

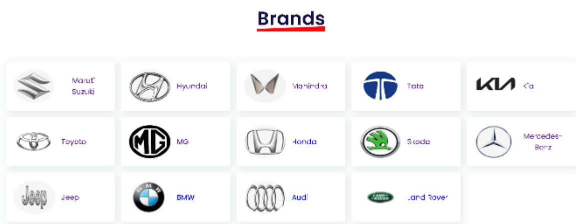

Fig.12: Brands

Fig.12 shows the brands present in the application.

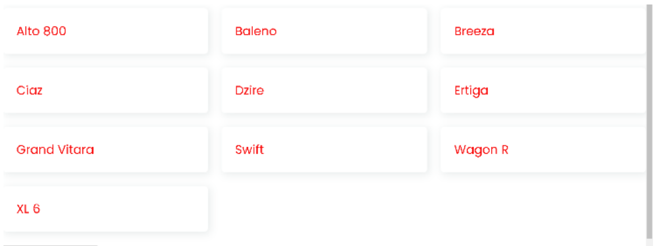

Fig.13: Models

Fig.13 shows the different models for a particular brand.

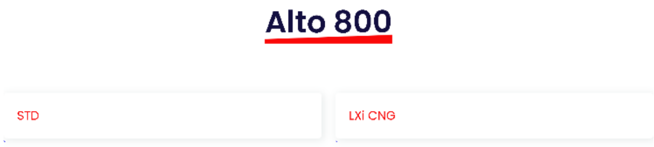

Fig.14: Variants

Fig.14 shows the different Variants of a particular Model.

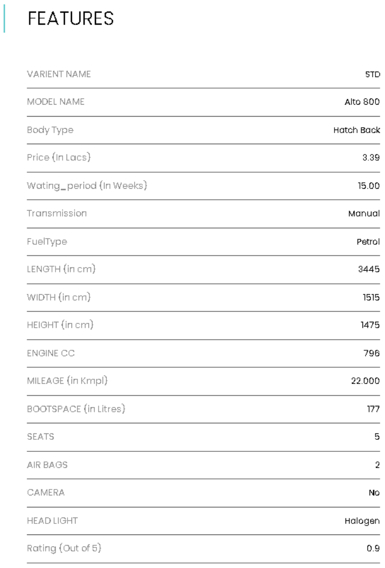

Fig.15: Features

Fig.15 shows the various features of a particular Variant.

V. CONCLUSION

The front-end implementation of the automobile application seamlessly integrates cutting-edge web technologies and robust AWS services. Through the use of HTML, CSS, and JavaScript, along with frameworks like React, Angular, or Vue.js, a dynamic and visually appealing interface is created, ensuring responsiveness across various devices. AWS services such as Amazon S3, RDS, EC2, SNS, CloudWatch, and IAM are strategically employed to optimize storage, manage dynamic data, provide scalable compute capacity, facilitate real-time communication, ensure vigilant monitoring, and enforce secure access control. The application prioritizes security with secure coding practices, HTTPS implementation, and IAM's role-based access control. Thorough testing validates compatibility, usability, accessibility, and performance, and continuous integration and deployment streamline updates. The AWS-centric approach enhances performance, reliability, scalability, and adaptability to evolving user needs. The result is a comprehensive, secure, and high-performance user interface that not only meets current requirements but also lays the groundwork for future enhancements.

The page can be accessed through this link: http://35.154.98.21/logincars#

VI. REFERENCES

[1] K. Zhang, Z. Li, X. Yin and L. Han, "Resource Provision for Cloud-Enabled Automotive Vehicles With a Hierarchical Model," in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 53, no. 3, pp. 1466-1478, March 2023, doi: 10.1109/TSMC.2022.3200452.

[2] H. -W. Deng, M. Rahman, M. Chowdhury, M. S. Salek and M. Shue, "Commercial Cloud Computing for Connected Vehicle Applications in Transportation Cyberphysical Systems: A Case Study," in IEEE Intelligent Transportation Systems Magazine, vol. 13, no. 1, pp. 6-19, Spring 2021, doi: 10.1109/MITS.2020.3037314.

[3] I. S. BULUT and H. ILHAN, "Cloud Based Vehicle and Traffic Information Sharing Application Architecture for Industry 4.0 (IoT)," 2019 International Conference on Information and Telecommunication Technologies and Radio Electronics (UkrMiCo), Odessa, Ukraine, 2019, pp. 1-7, doi: 10.1109/UkrMiCo47782.2019.9165541.

[4] A. Srinivasan, "IoT Cloud Based Real Time Automobile Monitoring System," 2018 3rd IEEE International Conference on Intelligent Transportation Engineering (ICITE), Singapore, 2018, pp. 231-235, doi: 10.1109/ICITE.2018.8492706.

[5] F. Milani and C. Beidl, "Cloud-based Vehicle Functions: Motivation, Use-cases and Classification," 2018 IEEE Vehicular Networking Conference (VNC), Taipei, Taiwan, 2018, pp. 1-4, doi: 10.1109/VNC.2018.8628342.

[6] Hu, P., Wang, Y., Gong, B. et al. A secure and lightweight privacy-preserving data aggregation scheme for internet of vehicles. Peer-to-Peer Netw. Appl. 13, 1002–1013 (2020). https://doi.org/10.1007/s12083-019-00849-6

[7] J. Cui, F. Ouyang, Z. Ying, L. Wei and H. Zhong, "Secure and Efficient Data Sharing Among Vehicles Based on Consortium Blockchain," in IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 7, pp. 8857-8867, July 2022, doi: 10.1109/TITS.2021.3086976.

[8] R. Jabbar et al., "Blockchain Technology for Intelligent Transportation Systems: A Systematic Literature Review," in IEEE Access, vol. 10, pp. 20995-21031, 2022, doi: 10.1109/ACCESS.2022.3149958.

[9] C. Huang, R. Lu and K. -K. R. Choo, "Vehicular Fog Computing: Architecture, Use Case, and Security and Forensic Challenges," in IEEE Communications Magazine, vol. 55, no. 11, pp. 105-111, Nov. 2017, doi: 10.1109/MCOM.2017.1700322.

[10] S. Liu, L. Liu, J. Tang, B. Yu, Y. Wang and W. Shi, "Edge Computing for Autonomous Driving: Opportunities and Challenges," in Proceedings of the IEEE, vol. 107, no. 8,

pp. 1697-1716, Aug. 2019, doi: 10.1109/JPROC.2019.2915983.

[11] T. Gong, L. Zhu, F. R. Yu and T. Tang, "Edge Intelligence in Intelligent Transportation Systems: A Survey," in IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 9, pp. 8919-8944, Sept. 2023, doi: 10.1109/TITS.2023.3275741.

[12] Bob McQueen, Big Data Analytics for Connected Vehicles and Smart Cities , Artech, 2017.