Modernizing Car Rentals with the Wheels on Demand System: Design and Implementation

Submitted in partial fulfilment of the requirements for the award of degree of

MASTER OF ENGINEERING

IN

COMPUTER SCIENCE & ENGINEERING

Submitted to:

Kriti Gupta (E17555)

Assistant Professor

Submitted by:

Inder Dev Singh

UID: 24MAI10043

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Chandigarh University, Gharuan

Aug 2024 – Dec 2024

Table of Content

List of Figures

List of Abbreviations

JSON : JavaScript Object Notation

JSONCpp: JavaScript Object Notation C plus plus

GUI: Graphical User Interface

WOD: Wheels On Demand

ABSTRACT

The **Wheels on Demand System** is an innovative solution designed to optimize the management of car rental operations, aiming to automate and streamline processes for both rental staff and customers. One of the most notable features of the system is its use of **JSON (JavaScript Object Notation)** format for data storage, which provides significant flexibility and ease in data manipulation. JSON is a lightweight, text-based data interchange format that is easy for both humans and machines to read and write. This format is not only simple but also highly effective in representing complex data structures such as the details of vehicles, rental transactions, and customer records. By employing JSON, the system ensures that car rental businesses can easily adapt to changing needs while maintaining an efficient database management structure.

The **Json Cpp** library plays a critical role in the **Wheels on Demand System**. JsonCpp is a C++ library that provides tools for parsing, manipulating, and writing JSON data. This library simplifies the process of interacting with JSON files, making it easier to handle large volumes of data while ensuring that updates, deletions, and retrievals of car and rental information are smooth and efficient. The system's use of JsonCpp allows for real-time updates to key data points, such as vehicle availability, rent prices, customer details, and rental durations. For example, as cars are added to the fleet or removed, their details are automatically updated in the JSON file, reducing the likelihood of errors and inconsistencies that are common with manual data entry systems.

One of the key advantages of using JSON and JsonCpp in this system is the **flexibility** it offers in terms of data storage. Car rental businesses, whether small or large, often deal with a diverse set of vehicles, rental types, customer categories, and other varying factors. The flexibility of JSON allows for easy expansion of the system's capabilities without having to overhaul the underlying structure. For instance, as businesses grow and expand their fleets, new car models or features can be added seamlessly to the system's data repository. The system can also handle complex data relationships, such as linking a rental record to a specific vehicle, customer, and rental period, all through a simple JSON structure.

In addition to its flexibility, the **scalability** of the Wheels on Demand System is another important feature. As the business grows and the volume of cars and rental records increases, the system is designed to handle large datasets without compromising performance. This is achieved through the use of efficient data structures and file handling processes in C++ that are optimized for speed and memory usage. The ability to scale the system ensures that it remains effective for both small, single-location rental businesses and large, multi-location rental operations.

Automating critical rental processes is a central goal of the Wheels on Demand System. By replacing manual record-keeping with an automated solution, the system improves both **efficiency** and **accuracy**. With automated car availability tracking, rental pricing updates, and customer history management, rental staff can save significant amounts of time that would otherwise be spent on administrative tasks. Furthermore, automation reduces human error, ensuring that customers receive accurate information and that no double bookings or outdated car details are displayed.

Ultimately, the **Wheels on Demand System** provides a **user-friendly interface** that allows rental staff to easily interact with the system. With a focus on simplicity and functionality, the system ensures that both experienced users and novices can quickly perform key operations such as adding a car, updating availability, or checking rental history. This ease of use contributes to a seamless experience for both rental staff and customers, improving customer satisfaction and operational efficiency.

The system's core value lies in its ability to handle the **complexities** of car rental management, offering a comprehensive, modern solution that aligns with the needs of businesses aiming for growth and enhanced service delivery. By using JSON for data management and leveraging C++ for optimal performance, the Wheels on Demand System is a scalable, efficient, and adaptable solution for the evolving car rental industry.

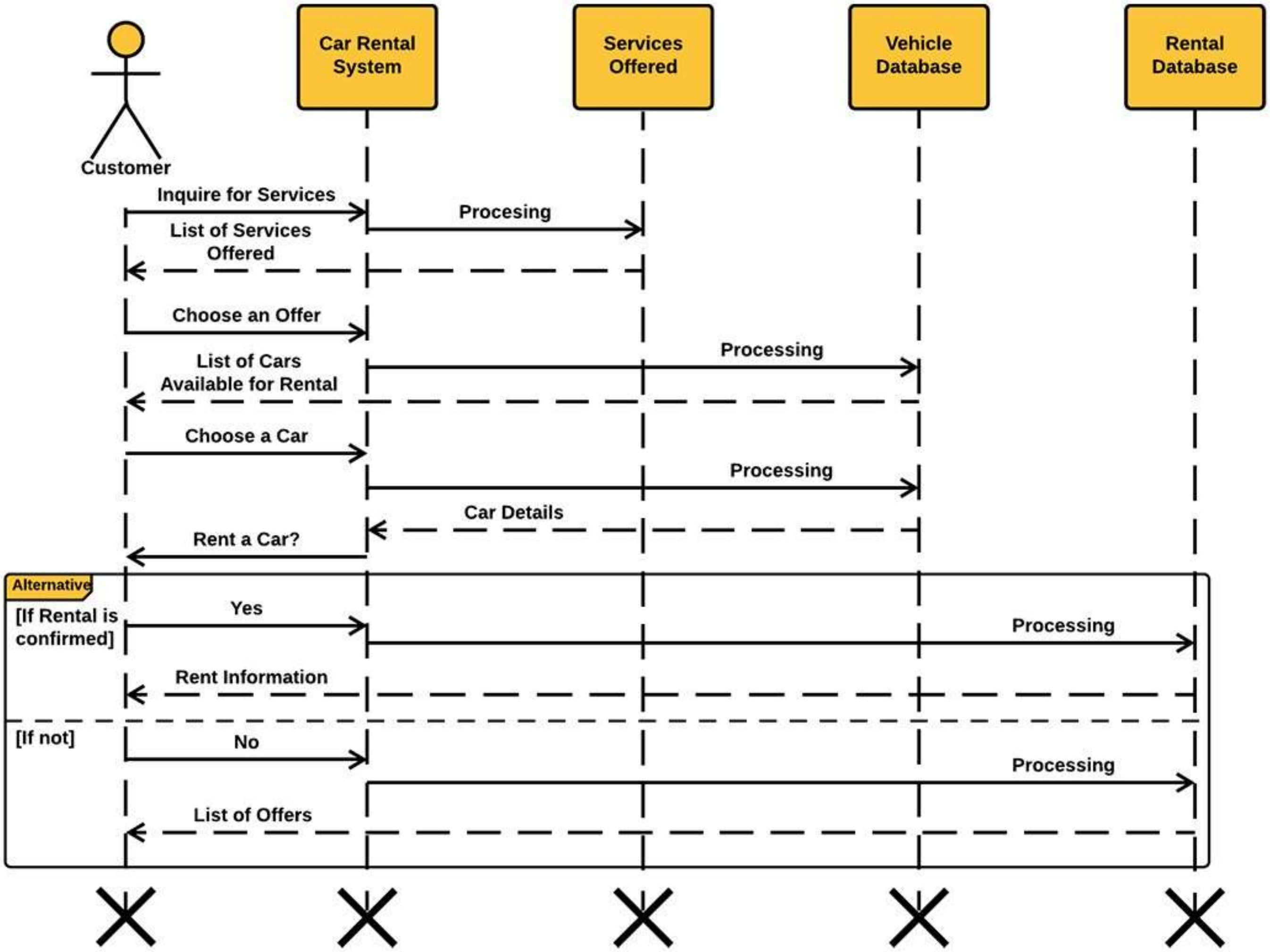

Fig 1. Sequence Diagram for Wheels On Demand System

INTRODUCTION

WOD businesses require efficient management tools to handle the complexities of renting cars, managing records, and maintaining up-to-date information. Traditional methods of record-keeping can be cumbersome, especially as the number of vehicles and rentals increases. To address these challenges, this project develops a Wheels on Demand System using C++ to automate day-to-day rental operations. The system leverages the JsonCpp library for effective data handling, making it possible to store, retrieve, and update car and rental details using JSON files. The primary goal is to reduce manual effort, improve accuracy, and ensure that the staff can quickly access and modify records.

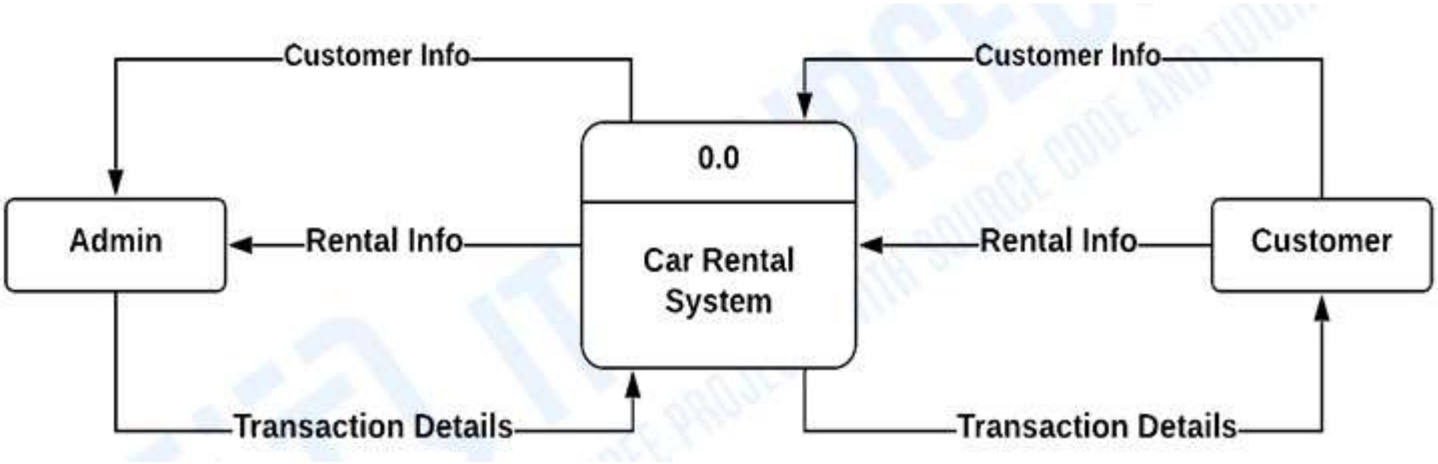

Fig 2. Data Flow Diagram (Level 0)

The DFD Level 0 is also known as the context diagram of Wheels On Demand System. Wheels On Demand System context diagram presents the main idea to be the basis for the proceeding levels. The basic idea is represented by a single process consisting the main process, users and data.

This context diagram portrays the general process done in Wheels On Demand System. The general process is based on the concept of the proposed project and illustrated using the DFD symbols to visualize the idea. This will also serve as a guide as you go through the deeper processes of the car rental system data flow di agrams.

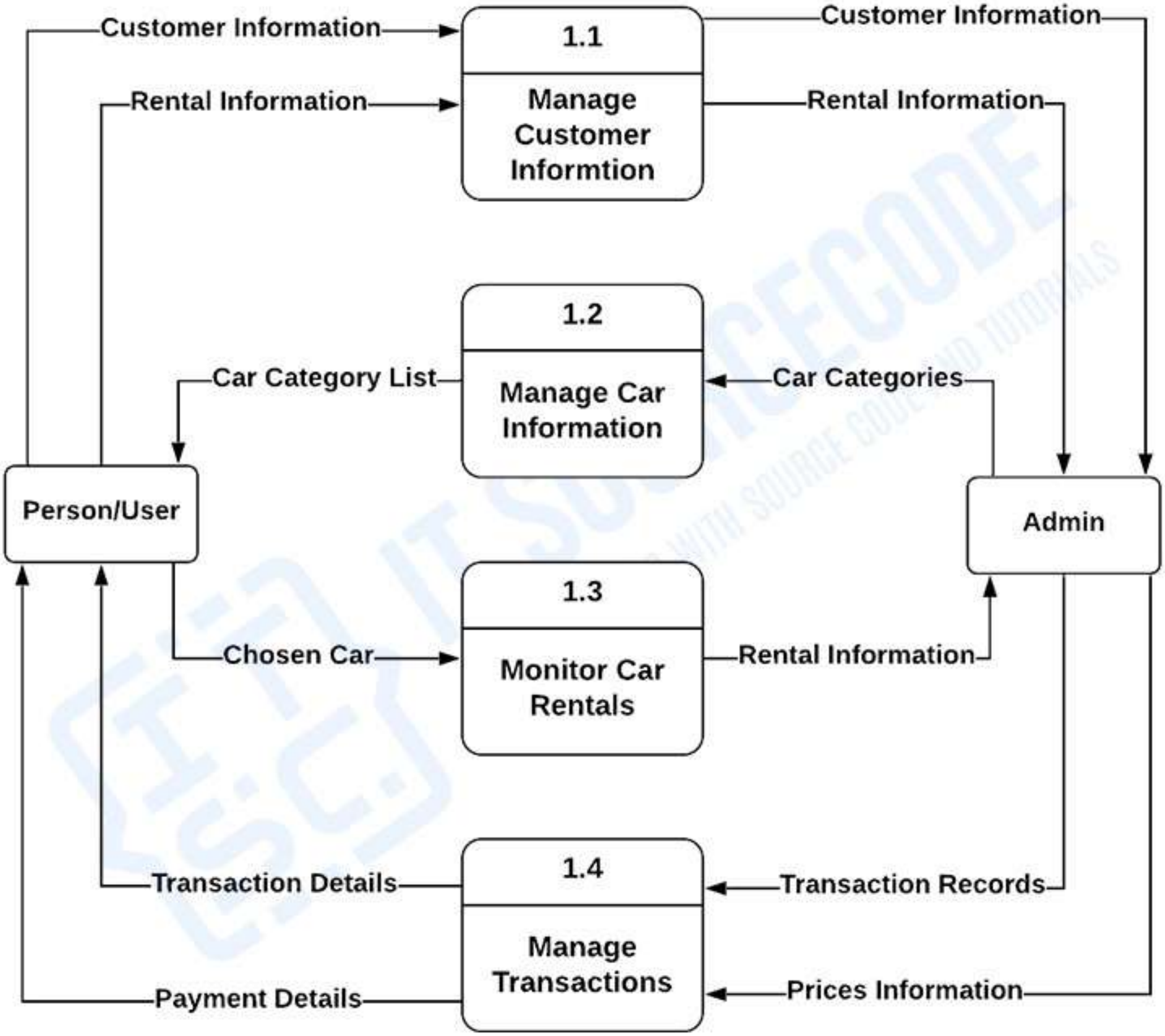

Fig 3. Data Flow Diagram (Level 1)

Next to the context diagram is the level 1 data flow diagram. It contains the broadened concept of DFD Level 0. It'll be done by determining the included processes to complete the basic idea from the context diagram. So as the processes were determined, data inputs will be specified as well as the outputs.

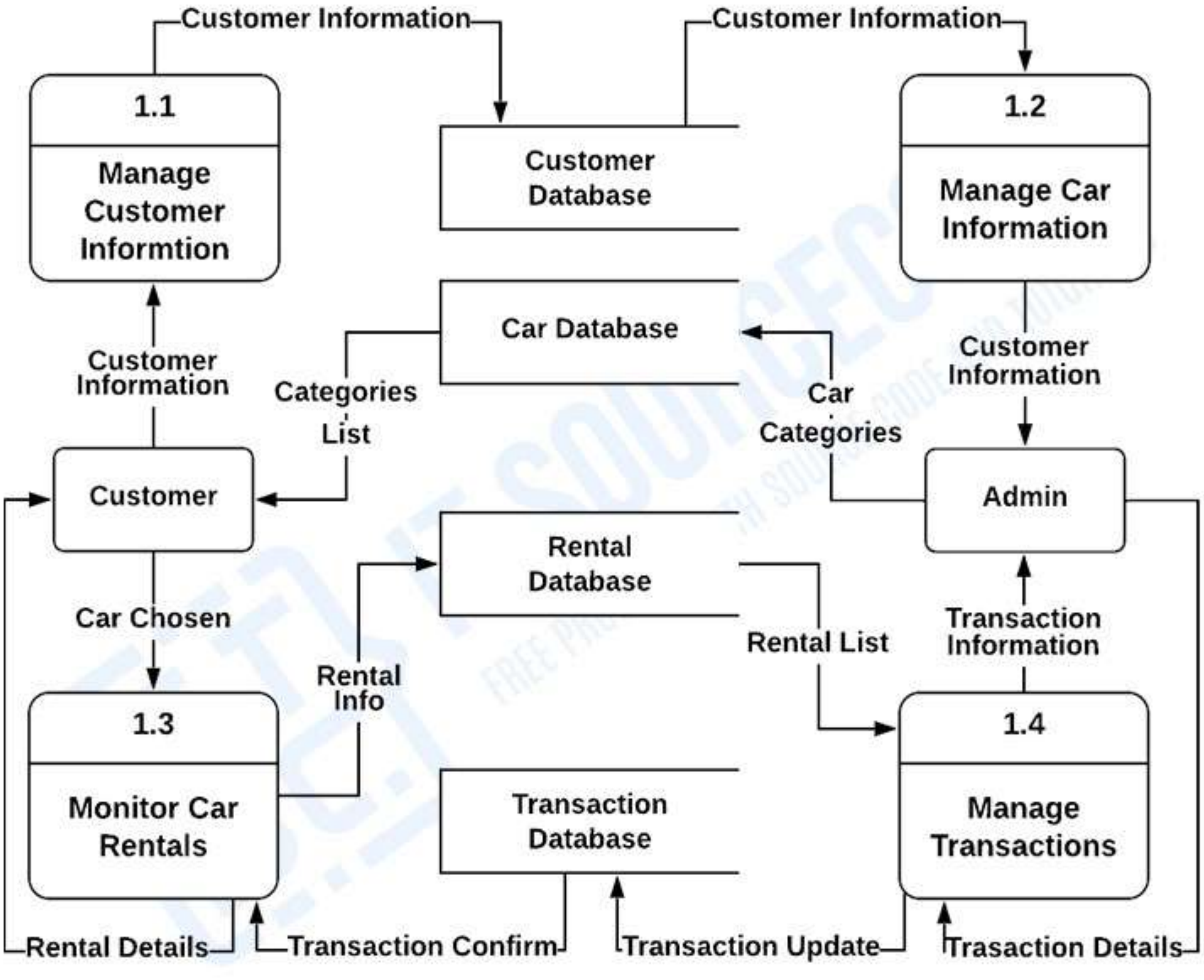

Fig 4: Data Flow Diagram (Level 2)

After presenting the Car Rental System DFD levels 0 and 1, next to that is level 2. The DFD Level 2 was considered as the highest abstraction of Car Rental System Data Flow Diagram. This level is expected to have the complete and detailed illustration of the project. As the former Levels were discussed, the DFD Level0 shows the main concept, the level 1 broadens it then the level 2 placed the database used to store all the data that flows in Car

Rental System. Data Flow Diagram level 2 let’s you know the ideas on where does the data inputs goes and outputs comes within the project. The presented level not only shows you the detailed processes of system, but also gives you precise destination of the data that flows in the system. This DFD Levels will also be your references as you make your own Car Rental System DFD levels 0, 1 and 2.

LITERATURE REVIEW

Various car rental systems have been developed over time, with many focusing on automating processes such as booking, tracking vehicle availability, and generating invoices. Early systems were largely paper-based, and as technology progressed, these evolved into database management systems using software like Microsoft Access and Excel. With the rise of object-oriented programming, languages like C++ became widely adopted for developing more robust systems. JSON has become a popular choice for data storage due to its flexibility and compatibility with various programming languages. Libraries like JsonCpp allow developers to efficiently handle JSON data, making it ideal for systems requiring frequent updates and complex record management. This project builds on the existing knowledge of car rental system development, focusing on automating tasks and improving performance through modern data handling techniques.

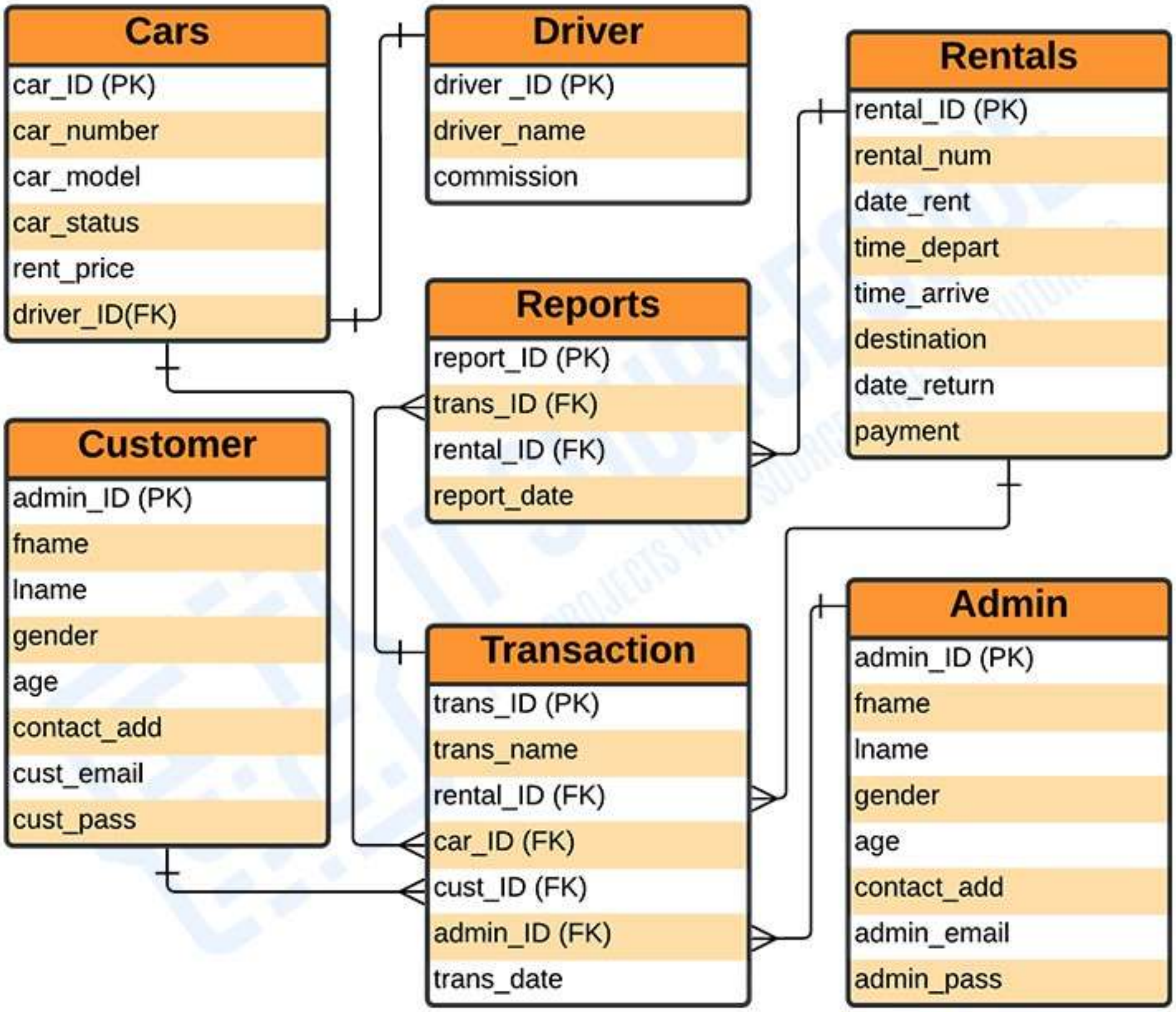

Fig 5. ER Diagram for Wheels On Demand System

METHODOLOGY

The Wheels on Demand System is built using C++ and integrates JsonCpp for managing car records and rental data. The methodology involves several steps:

1. **Requirement Analysis**: Identifying the primary features needed by the car rental business, such as car management, rental tracking, and data modification.
2. **System Design**: The system is designed to allow users to perform tasks such as adding new cars, updating car details, checking car availability, modifying rental details, and removing cars from the database.
3. **Development**: The system is implemented in C++, and the JsonCpp library is used for handling car and rental data in JSON format. Various classes are defined to manage the cars, rentals, and records, ensuring the system is modular and scalable.
4. **Data Storage and Retrieval**: All data, including car and rental records, are stored in JSON format, ensuring easy retrieval and modification. JsonCpp handles parsing, creating, and updating JSON files.
5. **Testing**: The system undergoes rigorous testing to ensure that all functionalities—adding, updating, and removing car details and rental records—work as expected. The system is tested for accuracy, speed, and data consistency.

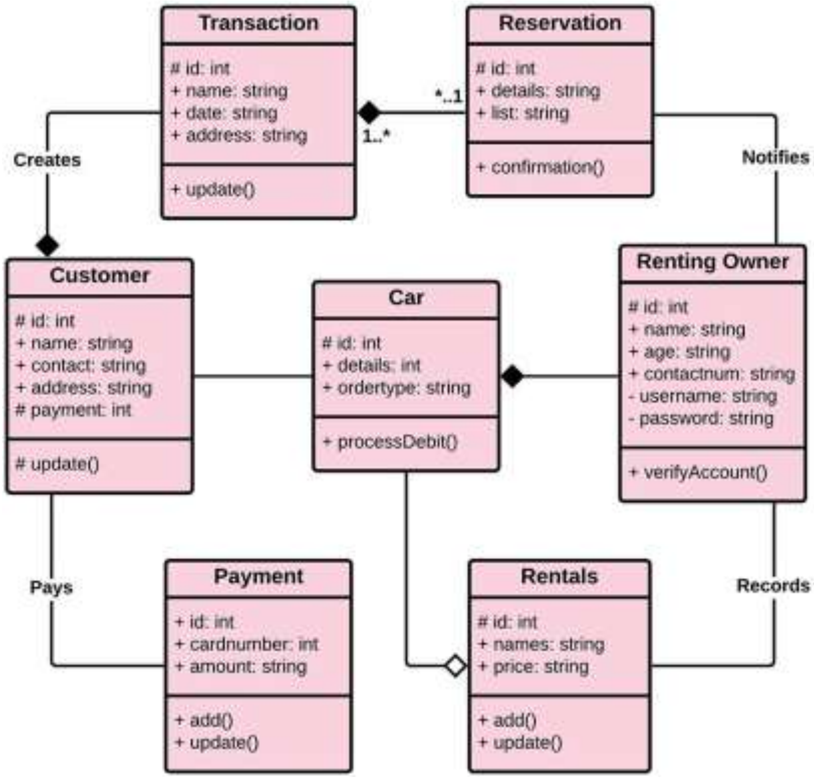

Fig 6. UML Class Diagram for Wheels On Demand System

DISCUSSION

The **Wheels on Demand System** successfully addresses the challenges faced by car rental businesses in managing their rental operations. Traditional methods of record-keeping, such as spreadsheets or paper-based systems, are not only time-consuming but prone to errors, especially as the fleet size grows. This project offers a streamlined solution by automating these operations using C++ and the JsonCpp library.

A key strength of the system is its use of JSON for data management, which makes it highly scalable and adaptable. JSON's lightweight format allows for easy storage and manipulation of complex data structures like car and rental records. The JsonCpp library simplifies reading and writing JSON files, enabling smooth operations such as adding new cars, retrieving and modifying rental details, and updating records with minimal user effort.

Another advantage of the **Wheels on Demand System** is its potential to handle a large volume of data. As car rental companies expand, they require systems that can scale with the business, and this project is designed with flexibility and growth in mind. The system is capable of managing an expanding fleet and growing customer base without compromising performance or ease of use.

However, there are potential areas for improvement. While the system effectively handles core functionalities, future versions could include features like integration with payment gateways, real-time vehicle tracking, and automated notifications for maintenance or rental expirations. Additionally, providing a more intuitive graphical user interface (GUI) would enhance the user experience, particularly for non-technical staff.

Overall, the **Wheels on Demand System** serves as a valuable tool for automating and improving the efficiency of car rental management, offering a scalable solution that reduces manual labor and improves data accuracy. Future developments could further enhance its functionality and ease of use.

CONCLUSION

The **Wheels on Demand System** provides an effective solution for automating the essential tasks of managing car rentals, such as maintaining car details, rental records, and modifying information. By leveraging C++ and the JsonCpp library for JSON data handling, the system successfully streamlines the operations of car rental businesses, reducing manual labor, improving accuracy, and making data management more efficient.

The system's ability to scale and manage a large number of vehicles and rental records makes it a versatile tool for both small and large rental companies. It simplifies data storage and retrieval, allowing staff to quickly add, update, and retrieve car and rental details as needed. This significantly improves the day-to-day efficiency of rental operations. While the system effectively automates core functionalities, there is room for future development, such as enhancing the user interface and adding advanced features like payment integrations and vehicle tracking. Overall, the **Wheels on Demand System** represents a solid foundation for further innovations, providing a modern, scalable, and user-friendly approach to car rental management.

REFERENCES

1. System, Online. 'Online Car Rental System'. Academia.edu. N.p., 2015. Web. 9 June2015.
2. Scribd.com, Online. '49930505 Car Rental System Project Report'. N.p., 2015. Web. 9June 2015.
3. Scribd.com, Online. 'Car Rental System Documentation'. N.p., 2015. Web. 9 June 2015.
4. Freelancer, Online. 'Project Documentation Car Rental Company SoftwareDevelopment Freelancers and Jobs - Freelancer'. N.p., 2015. Web. 9 June 2015.
5. Slideshare.net, Online. 'Zook Car Rental System Project'. N.p., 2015. Web. 9 June 2015.
6. Kaewman, Sasitorn. 'Online Decision Support System of Used Car Selection using K-Nearest Neighbor Technique'. IJFCC (2012): 164-166. Web.
7. Wikipedia, Online. 'Use Case Diagram'. N.p., 2015. Web. 9 June 2015.
8. Wikipedia, Online. 'Activity Diagram'. N.p., 2015. Web. 9 June 2015.
9. Tutorialspoint.com, Online. 'UML - Activity Diagrams'. N.p., 2015. Web. 9 June 2015.
10. Wikipedia, online. 'Swim Lane'. N.p., 2015. Web. 9 June 2015.
11. Mindtools.com, Online. 'Swim Lane/Rummler-Brache Diagrams: Mapping andImproving Processes in Your Organization'. N.p., 2015. Web. 9 June 2015.
12. Laudon, Kenneth C, and Jane Price Laudon. Management Information Systems. UpperSaddle River, NJ: Prentice Hall, 2000. Print.
13. Menkus, Belden. 'Car Rental Chain Former Owners Charged With Computer Frauds'.Computer Fraud & Security Bulletin 1993.3 (1993): 3-4. Web.
14. Li, Zhang. 'Design And Realization Of Car Rental Management System Based OnAJAX+SSH'.

Information Technology J. 12.14 (2013): 2756-2761. Web.