INF 1343 Data Modeling and Database Design

Assignment 1: Data Modeling, March 3rd, 2023

Instructor: Prof. Maher Elshakankiri

Group Members:

Jiaqi Han, #1005160530
Rohit Pandit, #1009704465
Zeqing Yu, #1009729233

Academic Department and Program:
Faculty of information, Master of information

Word count: 3328 words

Executive Summary:

This project is to analyze and model the car rental process of the Rentalcars company and make preparations for the database modeling. Renralcars.com is a large-scale online car rental service provider that serves as an intermediary between customer and car rental companies. Through conducting interview calls with the branch vice-president and different roles involved in the car rental process, we get a deep understanding of the whole process and identify the current information system of the business, especially the database management system. Also, we conducted secondary research to figure out the detailed attributes for each entity and their relationships y in the Rentalcars.com. After collecting all the data, we sketches three major conceptual data models for the whole business, namely ER Diagram, EER Diagram, as well as the UML class diagram to clearly demonstrate the information entities and relationships in the domain settings. A detailed analysis of each conceptual data model is provided under each diagram. The differences between the models compared to the other two are stated as well. The pros and cons for each technique are highlighted in the project. At the end, we generated a data catalog which is prepared for our future database design.

Section 1. Context for the Study

Rentalcars.com is a large-scale online platform that offers car rental services from a range of car rental companies. The business began in 2004 as TravelJigsaw and is headquartered in Manchester, UK. The company now operates in over 160 countries all over the world, including the United States, Canada, Europe, and Australia. Based on our investigation and interviews, as of 2021, Rentalcar.com has more than 4,000 employees working across various locations around the world, offering clients access to rental cars from more than 900 car rental companies. Each year, millions of users visit their website to book rental cars successfully.

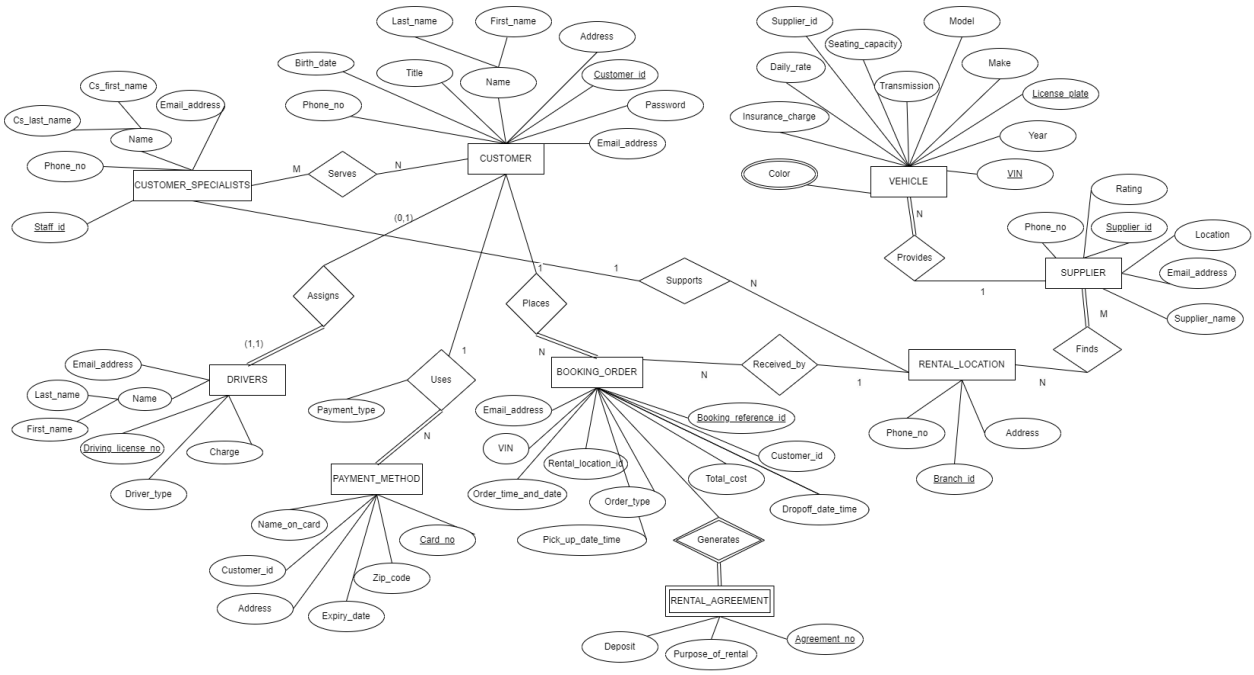
The external objective of the RentalCars company is to connect customers who have the demand of renting a car with car rental companies, letting customers to search for the available cars and compare rental car options from multiple suppliers on the website, and finding the appropriate vehicles for the customer based on their location and budgets. And then allowing customers to book the cars and pay the bill on the website. Finally, Rentalcars.com will provide an order summary for both the customer and the car rental company. The internal objective for the RentalCar company is to find reliable car suppliers and expand their network with the rental cars supplier, ensuring the maintenance of the website, and improving the user experience for both online and offline services. The organization objectives of Rentalcars.com is to expand their market share, explore new regions, simplify the rental process and enhance their customer services. The scopes of the Rentalcars.com are to demonstrate how their database provides a variety of results based on the customers search, providing an order summary for both the customer and the car rental company, storing incoming data from the suppliers, retrieving data which is relevant to the customers search query and managing transactions from customers and to external companies.

The major work processes for Rentalcars.com is to allow customers to browse the rental cars from various suppliers, initiating the booking orders on the website, processing payment on the website, and providing offline instructions after placing the order. After that, the offline rental locations will assign a customer specialist to assist the customer for the check-in and check-out matters. For the searching process, the automated system will take customers' requests for car rentals into consideration, checking the availability of the requested car and dates. For booking orders, given that this is an online platform, customers are expected to register an account and enter the personal information, then account holders can place the booking orders. Booking orders consist of the customer's registered information, booking reference, pick-up time and date, drop-off information,etc. A rental agreement will be generated from the booking order and customers are supposed to agree with certain conditions to place the order successfully. The Rentalcars.com will send order summaries to both their customers and suppliers. As for the payment, the online bill will accept card payments only and the card information such as the card number, name on the card will be recorded under the customer's private account. Once the payment completes, the rental locations will find a car provider to find the car. The offline rental location will assign a customer specialist to serve the customer when doing the check-in and check-out process. This process involves verifying the customer's identity and driver's license, confirming their bookings, and providing them with the rental agreement and keys to the vehicle. When the rental period is over, the customer drops-off the car to the rental company and the customer specialist checks the vehicle condition for any damages before refunding the deposit.

The existing information systems that the RentalCars company has are user interface system, database management system, communication system and analytics system. User interface is the front-end component of the system that users interact with when searching information for car rentals on the Rentalcars.com. In general, the user interface includes a search engine, filters, and sorting functions that enable customers to find the most suitable car rental options. The database management system is the back-end component of the system that records and manages all data related to car rentals, such as pricing, availability, and location. The database management system ensures that the data is accurate,

up-to-date, and easily accessible to users. The communication system facilitates communication between Rentalcars.com and car rental companies, as well as between Rentalcars.com and users. The communication system has practices such as email notifications, customer services. The analytics system collects and analyzes data such as user behavior and preferences on the platform. The analytics system can provide insights into user trends, popular rental car options, and areas for improvement. In this assignment, we are gonna focus on the database system of Rentalcars.com to dig deeper into its eight entities as well as and their attributes and relationships.

Section 2. Modelling using ERD diagram

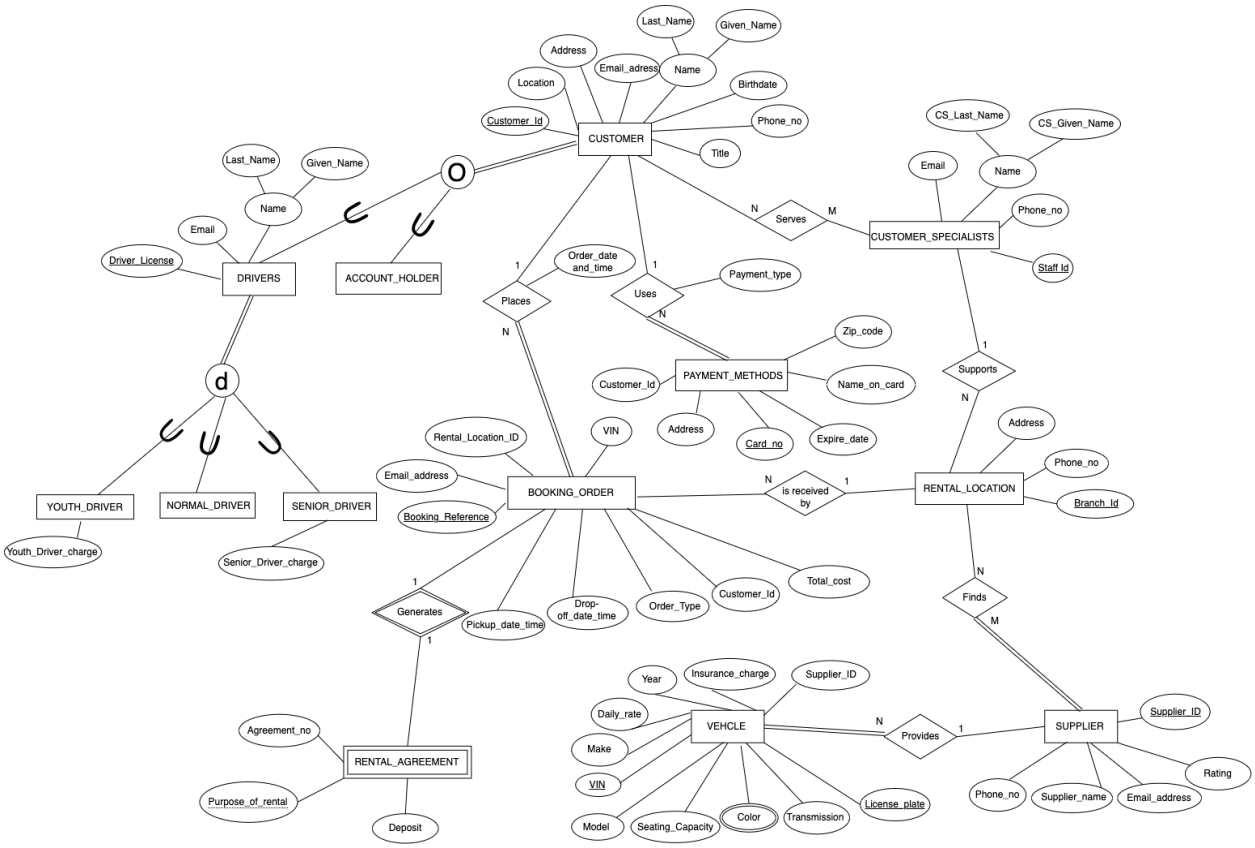


The above ERD diagram represents entities, their attributes and relationship between different entities along with the cardinality values for a Car Rental system. Customer entity represents the individuals who rent a car from the company. Customer specialists entity represents the staff members of the company that assist customers with their queries and requests. Vehicle entity lists all the available cars to be rented along with their specifications. Driver entity represents the individuals who would drive the rented car. Payment method entity lists the modes of payment along with the customer's transaction details. Booking order entity represents the booking details summary of the rented car. Rental agreement entity represents the agreement details between the customer and the company. Supplier entity represents the details of the supplier that provided that vehicle to the company. Rental location entity represents the details about the car rental company. The relationship between these entities include-

1. One customer can place zero or many booking orders, and one booking order can only be placed by one and only one customer.
2. One customer specialist can serve many customers, and one customer should be served by atleast one customer specialist.
3. One customer specialist can provide support to one or many rental locations, and one rental location should have at least three customer specialists.
4. One supplier can provide at least one vehicle, and one vehicle can only be provided by one and only one supplier.

5. One rental location can find many suppliers, and one supplier can work with many rental locations.
6. One booking order can be received by one and only one rental location, and one rental location may receive zero or many booking orders.
7. One customer can drive the rented car himself or assign at least one driver, and one driver can be assigned by a customer or could be the customer himself.
8. One customer should have at least one payment method, and each payment method should only be held by one customer.
9. One booking order generates one rental agreement, and each rental agreement is generated by one booking order.

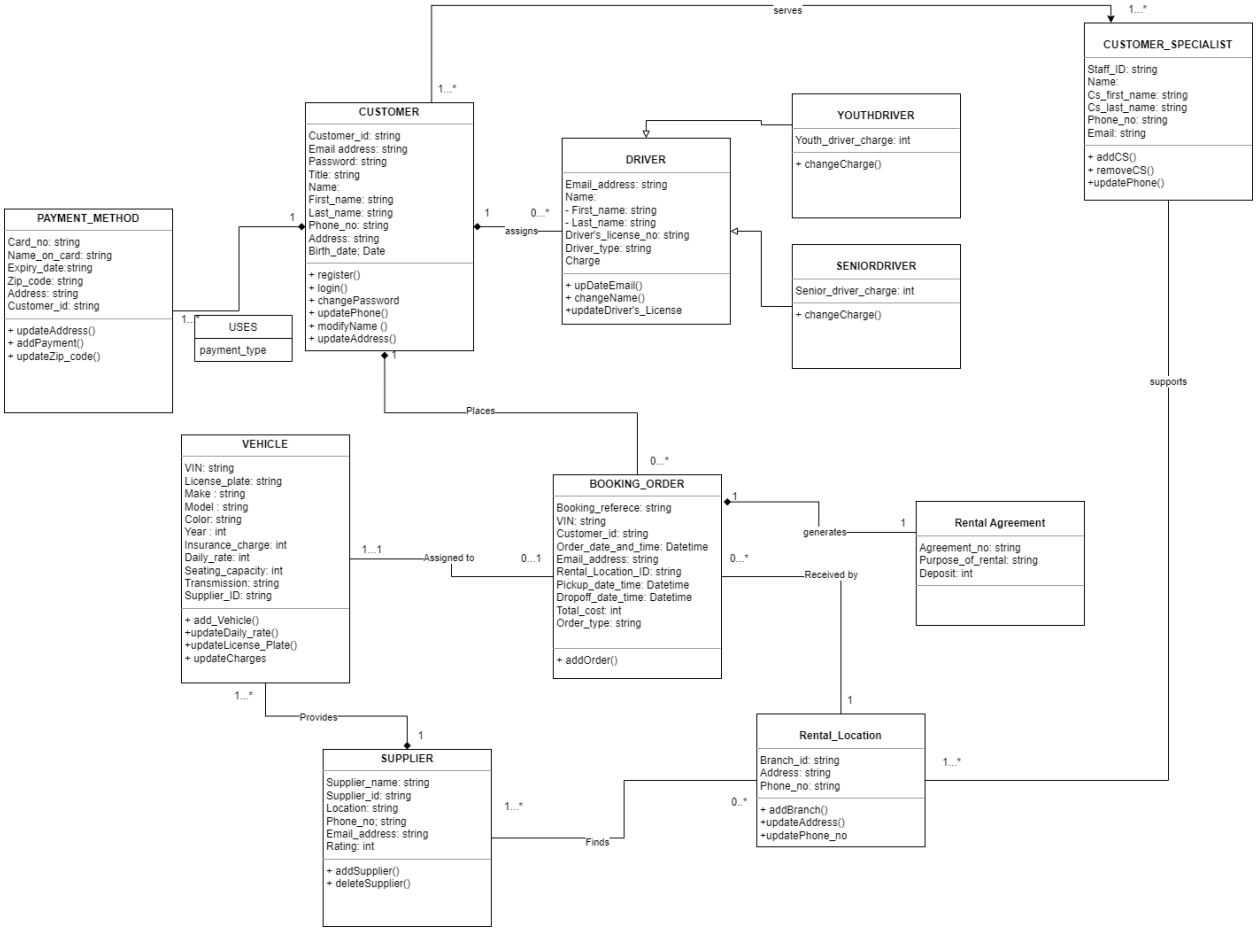
Section 3. Modelling using EERD diagram



The EER diagram vividly shows all the entities, attributes as well as their relationships involved in the business of Rentalcars company. In general, the business has main entities in the EER diagram, namely customer, booking order, payment method, rental location, supplier, vehicle and customer specialist. Also, there's newly added entities named driver and account holder under the customer entity. The relationship of the whole business can be summarized as the customer places the booking order on the website successfully on condition that they agree on the rental agreement and choose at least one payment method to complete the payment. Then the rental location will find the reserved vehicle from the supplier and assign a customer specialist to serve the customer for the check-in and check-out process. EER diagram outlines the subclass and superclass relationship in the entities. Taking the CUSTOMER entity for example, a customer who places the booking order also drives the car. But, ideally a customer who placed the order through their account is not supposed to be the driver themselves. Here, the customer is not only the driver but the account holder. Thus, there's an overlapping relationship for the CUSTOMER entity

between DRIVER and ACCOUNT HOLDER. Looking at the DRIVER entity, the business divides drivers into three groups: Youth drivers aged below 21; Normal drivers aged between 21 to 69; Senior drivers aged 70 and above. Driving licence is mandatory for all drivers, there's youth driver charge and senior driver charge for youth driver and senior driver respectively. Hence, the DRIVER ENTITY is decomposed into three entities: YOUTH DRIVER, NORMAL DRIVER and SENIOR DRIVER. This is the specialization for the DRIVER entity along with attributes of youth driver charge and senior driver charge.

Section 4. Modelling using UML diagram



The UML class diagram shows all the entities, attributes contained by each entity, datatype of each attribute, functions of each table, the relationship among the attributes, and the cardinality of the relationships. Let's take the entity CUSTOMER as an example. The entity CUSTOMER has attributes about information of the customer who registered the account, such as the customer's ID of the account (with string as its datatype), password (datatype: string), email address (datatype: string), first and last name, (datatype: string) phone number (datatype: string), etc. Although some attributes, such as password and phone number may consist of numbers, their datatype is still string, because these numbers will not be involved in any kind of calculations. Some functions of the table include changing the password, updating the customer's phone number, updating customer's address, etc. Moreover, this entity has a relationship called "Places" with another entity BOOKING_ORDER. The diagram shows "0…*" on the booking order side, which means one customer can place zero to many orders. The number on the customer side is

one, which means one booking order can only be placed by one customer. In addition, in this relationship, the end of the relationship line on CUSTOMER's side is a black diamond. It means it's a composition relationship. The entity BOOKING_ORDER can not exist individually if the entity CUSTOMER is removed. Another special relationship is inheritance, which appears between the YOUTHDRIVER entity and DRIVER, and SENIORDRIVER entity and DRIVER. It means the two entities inherit the attributes in DRIVER, such as the names and email address. These information can be the main points of interest in our UML diagram, because they can not be found in ERD or EERD.

Section 5. Comparison of the Three Modeling Techniques

5.1 Compare experiences and exam the differences to other two models

ERD:
Compared with EERD and UML, the ERD diagram focuses on representing the entities, attributes and the relationship that exists between them. It also represents the primary key, foreign key and several other attributes of the entities. Our ERD diagram explains the cardinality values over the relationships between different entities involved. For instance the relationship between the customers and drivers is explained concisely. A CUSTOMER can assign a DRIVER for the rented car or the CUSTOMER can drive the rented car on his own. Overall, the ERD diagram describes the workflow of the car rental system in a simple and clear manner which is required during the initial stages of development. However, it cannot represent complex relationships between entities with the superclass, subclass concepts as implemented in the EERD model. Moreover, as compared to the UML, the information about the datatype of the attributes and the functions for each table cannot be explained in the ERD model.

EERD:
Compared with ERD and UML, the EERD diagram is the most comprehensive data modeling technique. Firstly, EERD presents more detailed information than ERD. Our ERD and EERD look similar at the first glance, both of them show the main entities and their attributes as well as their relationships. However, the EERD diagram contains the subclass, superclass relationships and specialization. In our cases, the CUSTOMER entity has subclasses of DRIVER and ACCOUNT_HOLDER, meaning that a customer could be an account holder or both account holder and driver themselves. Also, there's a specialization for the DRIVER entity based on the driver charge. YOUTH_DRIVER has to pay a youth driver fee, NORMAL_DRIVER is free from driver charge and SENIOR_DRIVER has to pay a senior driver charge. These three entities don't exist in the ERD diagram given that the ERD diagram doesn't show the specialization. As for the comparison for the EERD and UML, the EERD provides more details about data modeling than UML. Some subclass entities such as ACCOUNT_HOLDER and NORMAL_DRIVER don't exist in the UML because they don't have any attributes. However, the UML diagram is technique-oriented and provides extra information such as behavioral and structural modeling to help us model the database. This is special for the UML class diagram.

UML:
Compared with ERD and EERD, the UML class diagram provides more information about the data types of the attributes in each table, which can be helpful and convenient when we need to build a real database. Moreover, it can also show the operations of each table, which can help us to understand the interactions between this entity and others in the system. However, the UML diagram did not provide a way to represent some special elements, such as weak entities and relationships, and multivalued attributes. In our case the weak entity is RENTAL_AGREEMENT, and the multivalued attribute is Color in the VEHICLE entity. In ERD and EERD diagrams there are symbols that are used specifically for weak entities and multivalued attributes.

5.2 Lists of pros and cons for each model

Data model	Pros	Cons
ERD	● ERD shows all necessary elements for designing database schema, such as entities and attribute, as well as their relationships ● ERD is easy to graph and easy to understand the data model for the designer ● ERD has a good visualization ● ERD is a straightforward representation	● ERD has limitations when describing complex relationships among entities ● ERD can not show the complete information in a real database ● ERD may fail to show the business rules with the data model
EERD	● EERD provides in-depth modeling capacities. With added information of subclass and superclass relationships and specialization. ● EERD shows the database schema in a more comprehensive way which enhances data accuracy ● EERD enhances data consistency	● EERD is somewhat hard to understand the relationships than ERD ● EERD is relatively hard to model for a complexed business ● EERD is a data model but can't be used directly to modeling the database

UML	• UML class diagram helps to develop the database as it mostly fits the programming language • UML class diagram has the standard notation for database development. • UML class diagram contains additional information such as such as behavioral and structural modelling.	• UML class diagram has limited support for database implementation. Developers should translate the diagram into the database schema manually. • UML class diagram is relatively hard to model for a complexed business as well • UML class diagram sometimes may not include the foreign keys,

Section 6. Data Catalog

Table name: CUSTOMER

Entity: CUSTOMER

Attribute Name	**Attribute Type**	**Attribute Domain**	**Attribute Constraints**
Customer_id	VARCHAR(10)	10-digit characters	Primary Key
Email_address	VARCHAR(254)	0 to 254-digit characters	Not Null
Password	VARCHAR(45)	8 to 45-digit characters	Not Null
Title	VARCHAR(45)	0 to 45-digit characters	Not Null
First_name	VARCHAR(45)	0 to 45-digit characters	Not Null
Last_name	VARCHAR(45)	0 to 45-digit characters	Not Null
Phone_no	VARCHAR(45)	0 to 45-digit characters	Not Null
Address	VARCHAR(254)	0 to 254-digit characters	Not Null
Birth_date	DATE	Jan 1, 1900 to current date	Not Null

Table name: DRIVER
Entity: DRIVER

Attribute Name	Attribute Type	Attribute Domain	Attribute Constraints
Email_address	VARCHAR(45)	0 to 254-digit characters	Not Null
First_name	VARCHAR(45)	0 to 45-digit characters	Not Null
Last_name	VARCHAR(45)	0 to 45-digit characters	Not Null
Driver's_license_no	VARCHAR(45)	0 to 45-digit characters	Primary Key
Driver_type	VARCHAR(45)	0 to 45-digit characters	Not Null
Charge	DECIMAL(10,2)	0 to 10-digit decimal number, 2 after decimal point	Not Null

Table name: BOOKING_ORDER
Entity: BOOKING_ORDER

Attribute Name	Attribute Type	Attribute Domain	Attribute Constraints
Booking_reference	VARCHAR(45)	0 to 45-digit characters	Primary Key
VIN	VARCHAR(17)	17-digit characters	Foreign Key
Customer_id	VARCHAR(10)	10-digit characters	Foreign Key
Order_date_and_time	DATETIME	Jan 1, 1900 to Dec 31, 9999	Not Null
Email_address	VARCHAR(45)	0 to 254-digit characters	Not Null
Rental_location_id	VARCHAR(10)	10-digit characters	Not Null
Pickup_date_time	DATETIME	Current date and time to Dec 31, 9999 23:59:59	Not Null
Dropoff_date_time	DATETIME	Current date and time to Dec 31, 9999 23:59:59	Not Null

Total_cost	DECIMAL (10,2)	0 to 10-digit decimal number, 2 after decimal point	Not Null
Order_type	VARCHAR(45)	0 to 45-digit characters	Not Null

Table name: PAYMENT_METHOD
Entity: PAYMENT_METHOD

Attribute Name	Attribute Type	Attribute Domain	Attribute Constraints
Card_no	VARCHAR(16)	16-digit numbers	Primary Key
Name_on_card	VARCHAR(45)	0 to 45-digit characters	Not Null
Expiry_date	VARCHAR(45)	0 to 45-digit characters	Not Null
Zip_code	VARCHAR(10)	0 to 10-digit characters	Not Null
Address	VARCHAR(254)	0 to 254-digit characters	Not Null
Customer_id	VARCHAR(10)	10-digit characters	Foreign Key

Table name: VEHICLE
Entity: VEHICLE

Attribute Name	Attribute Type	Attribute Domain	Attribute Constraints
VIN	VARCHAR(17)	17-digit characters	Primary Key
License_plate	VARCHAR(45)	0 to 45-digit characters	Not Null, Unique
Make	VARCHAR(45)	0 to 45-digit characters	Not Null
Model	VARCHAR(45)	0 to 45-digit characters	Not Null
Color	VARCHAR(45)	0 to 45-digit characters	Not Null
Year	INT	0000 to 9999	Not Null
Insurance_charge	DECIMAL(10, 2)	0 to 10-digit decimal number, 2 after decimal point	Not Null
Daily_rate	DECIMAL(10,2)	0 to 10-digit decimal number, 2 after decimal point	Not Null

Seating_capacity	INT	0 to 10	Not Null
Transmission	VARCHAR(45)	{Manual, Automatic}	Not Null
Supplier_id	VARCHAR(45)	0 to 45-digit characters	Foreign Key

Table name: SUPPLIER
Entity: SUPPLIER

Attribute Name	Attribute Type	Attribute Domain	Attribute Constraints
Supplier_name	VARCHAR(45)	0 to 45-digit characters	Not Null
Supplier_id	VARCHAR(45)	0 to 45-digit characters	Primary Key
Location	VARCHAR(254)	0 to 254-digit characters	Not Null
Phone_no	VARCHAR(45)	0 to 45-digit characters	Not Null
Email_address	VARCHAR(254)	0 to 254-digit characters	Not Null
Rating	DECIMAL (4,1)	0 to 4-digit decimal number, 2 after decimal point	Not Null

Table name: RENTAL_LOCATION
Entity: RENTAL_LOCATION

Attribute Name	Attribute Type	Attribute Domain	Attribute Constraints
Branch_id	VARCHAR(45)	0 to 45-digit characters	Primary Key
Address	VARCHAR(254)	0 to 254-digit characters	Not Null
Phone_no	VARCHAR(45)	0 to 45-digit characters	Not Null

Table name: RENTAL_AGREEMENT
Entity: RENTAL_AGREEMENT

Attribute Name	Attribute Type	Attribute Domain	Attribute Constraints
Agreement_no	VARCHAR(45)	0 to 45-digit characters	Weak Key Attribute

Purpose_of_rental	VARCHAR(45)	0 to 45-digit characters	Not Null
Deposit	DECIMAL(10,2)	0 to 10-digit decimal number, 2 after decimal point	Not Null

Table name: CUSTOMER_SPECIALIST
Entity: CUSTOMER_SPECIALIST

Attribute Name	Attribute Type	Attribute Domain	Attribute Constraints
Staff_ID	VARCHAR(45)	0 to 45-digit characters	Primary Key
Cs_first_name	VARCHAR(45)	0 to 45-digit characters	Not Null
Cs_last_name	VARCHAR(45)	0 to 45-digit characters	Not Null
Phone_no	VARCHAR(45)	0 to 45-digit characters	Not Null
Email_address	VARCHAR(254)	0 to 254-digit characters	Not Null

Section 7. References

RentalCar official website:https://www.rentalcars.com/us

Rozenblit, M., Rozenblit, A. M., & Wit, R. de. (2022, December 27). Rentalcars.com review: The best way to hire a Car. The World Was Here First. Retrieved March 1, 2023, from https://www.theworldwasherefirst.com/rentalcars-com-review/

Statement of individual contributions:

Jiaqi Han: UML class diagram, Data Catalog, Comparison of models.

Rohit Pandit: ERD diagram, Comparison of models, Report formatting.

Zeqing Yu: Executive summary, Context of the study, EERD diagram, Comparison of models, Pros and Cons list