EZ Rental Inc.

Customer Documentation

Table of Contents

1. **Executive Summary.....2**
2. **Problem Statement & Objectives.....3**
3. **Project Management Methodology.....4-5**
4. **Application Business requirements 6-16**
5. **Application Development & Technical Requirements.....17-23**
6. **Application Physical Technical Architecture.....24-25**
7. **Database Management System Development Environment & Physical Architecture.....26**
8. **Project Roles & Responsibilities.....27-28**
9. **Database Design Deliverable #1 – ER/EER Conceptual Model Diagram.....29**
10. **Database Design Deliverable #2 – Normalized Logical Model Diagram.....30-31**
11. **Database Design Deliverable #3 – Physical Model Data Dictionary.....32-38**
12. **Database Design Deliverable #4 – Physical Model Schema Design Diagram.....39**
13. **Database Implementation Deliverable #5 – Development & Implementation.....40-43**
14. **Database Implementation Deliverable #6 – Implemented Physical Schema Diagram.....44**
15. **Database Implementation Deliverable #7 – Database Validation Unit Testing.....45-49**
16. **Conclusion.....50**

PROJECT 1 – EZRental Auto Rental POS Management System Database Design and Implementation

Executive Summary

EZRental needs a ***DBMS*** (Database Management System) to store and preserve all the information affiliated with our business model. The aim of this project is to transform all needs of **EZRental** business requirement and create a working database abiding all aspects of its business that will preserve, maintain, and present all the customer, corporate and other requirements needed to operate Rental's day-to-day activity without any obstacle.

This documentation provides all the business requirements, methods implemented, models developed, and database formulated for running the **EZRental** operations.

Problem Statement & Objectives

EZRental Inc., has hired **NYC-Tech Solutions Inc,** With the goal of designing & implementing suite of Auto Rental Point-of-Sales Management System Applications**,** This application has included **EZRental POS** intended for ***Customer service representative*** and other ***employees*** in the ***rental agencies***, in addition to an ***INTRANET Corporate Website*** named **EZRentalCorp.com** intended for business employees in the corporate offices, and finally, an e-commerce ***INTERNET Website*** name **EZRental.com** intended for customers via the internet to make and manage reservations.

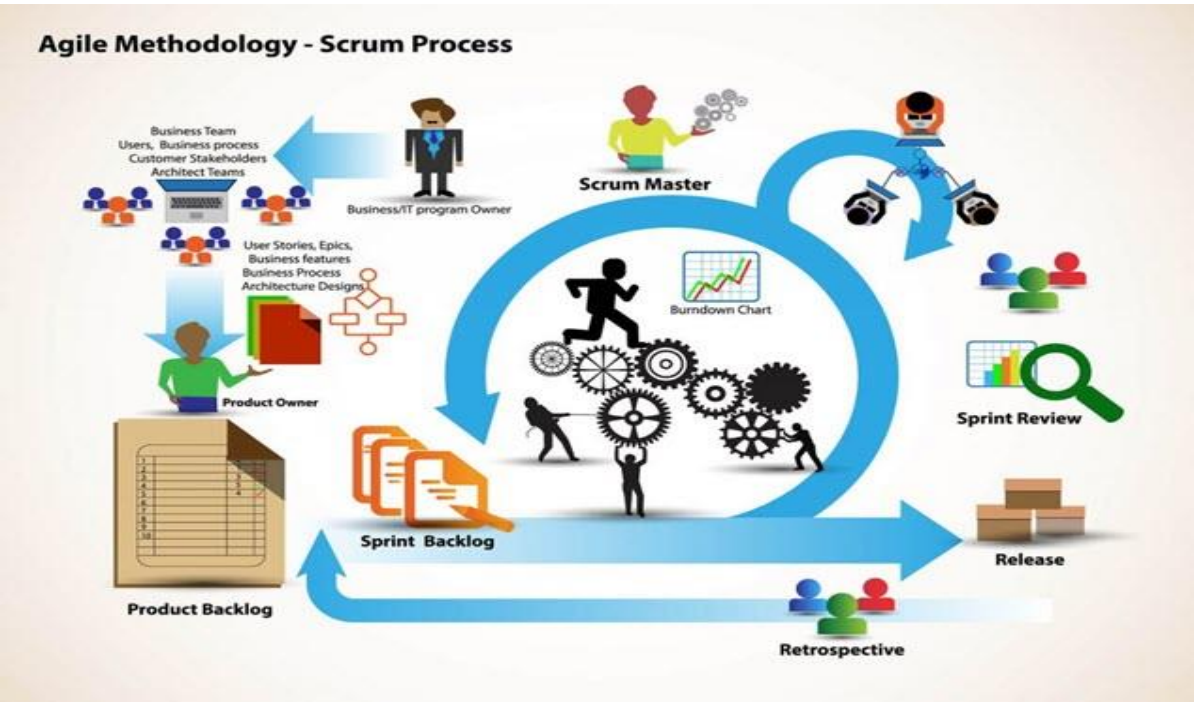
The application has implemented code and features that can allow customers, both in retail and corporate like ***Avis***, ***Hertz***, ***Budget*** etc. to reserve vehicles. The application has been designed with capacity to support dozens of major cities around the world, along with providing best experience and best pricing in the market by **EZRental Inc.,** it has also implemented functionality for ***Customer service representative*** and other front-line workers at **EZRental Inc**. The company has also targeted markets in *Asia, Africa, and the Mediterranean* to expand globally, currently having branches in US, Canada, Mexico, UK, Japan & Australia.

Project Management Methodology

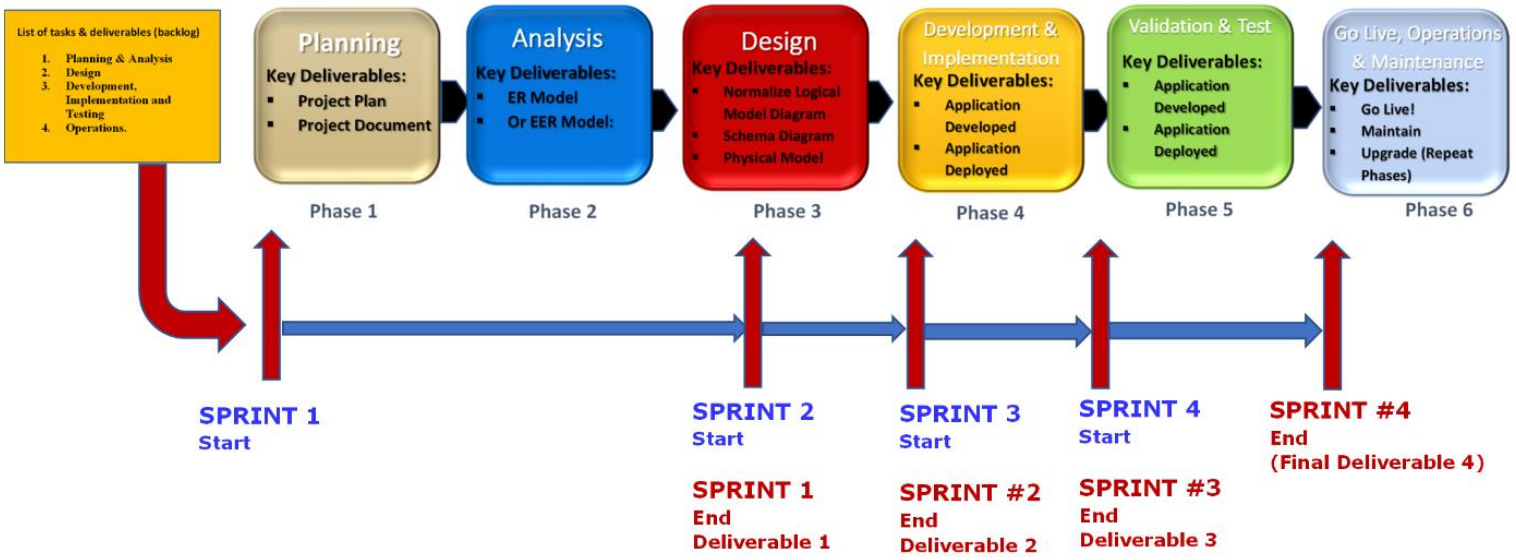
The **Waterfall Project Management** combined with **Agile** has been used to design, develop, and implement the ***DBMS Server Application*** for the Auto Rental Management System ER-Rental POS (Point of Sales), Two-Tiered Client/Server & Three-Tiered Web Client/Server Applications.

The Waterfall Project Management Methodology decided on by the project architects and project manager, contains 6-phase, illustrated below:

Agile methodology is a results-focused approach, which can be modified according to the need of the rapid changing world. Demo of agile method is shown in the diagram below.



Agile and Waterfall methodologies has been combined by leveraging idea of ***Sprints***. Database development and waterfall implementation phases has been divided into sprints (Agile approach) deliverable. The demonstration of these two methods has been illustrated in the figure below and the phases are elaborated in the preceding table:



AGILE SPRINT #	WATERFALL PHASE	Output Deliverable
SPRINT #1	Planning	1. Create Project Document. 2. Format and populate as per requirements.
	Analysis	1. ER/EER Conceptual Model Diagram
	Design Phase (Part 1)	1. Normalized Logical Model Diagram
SPRINT #2	Design Phase (Part 2)	1. Data Dictionary matrix 2. Physical Schema Diagram (Normalize Model + Data Dictionary combination)
SPRINT #3	Development & Implementation	1. Database application developed & implemented. 2. Database application installed, setup and configured.
SPRINT #4	Validation & Testing	1. Unit testing. 2. Integration testing.
	Operations	1. Operations or keep running. 2. Keeping the lights on!

Application Business requirements

Interviews were held with ***EZRental Inc., Project & Business Stakeholders*** to gather & compile a list of **Business Requirements** which are the foundation of the database design.

The following ***pages below***, contain **the Business Requirements** captured by the ***Business Analyst***.

Business Requirements

About Us:

EZ-Car Rental is an auto rental company that rents vehicles such as cars, SUVs, minivans & cargo vans to customers. In addition, other specialized vehicles such as trucks, motorcycles, boats, mobile homes, etc. We operate in several countries with rental agency locations in the US, Canada, Mexico, UK, Japan & Australia. Within each country we own and operate rental agencies located in cities, regions and state. For example, New York City has 2 rental agencies in Manhattan, one in Brooklyn and two in Queens located at each airport. With multiple rental agencies in cities, states etc., a customer can pick up a vehicle in one location and drop it off at another.

Current Challenges:

Our current rental system is outdated, with a poor user-experience, inefficient (breaks often thus expensive to operate), does not meet our business requirements, and is not scalable (cannot be easily updated with new features). Another very important shortcoming of the current system, is the lack of elasticity since it does not give us the flexibility to scale-up or scale-down resources during business trends and seasonal changes in the market.

We want to invest in modernizing our business with a new vehicle management system that can meet these challenges and delivers a great user-experience, meet our new business requirements, scalable, and elastic to adopt to business trends and seasonal market changes. Elasticity is very important since recently we have been faced with a new type of competition; small rental companies that are nimble and can quickly adopt to market changes thus able to provide new offerings that are appealing to customers thus affecting our profits. These smaller competitors are using new technologies that enable them to be nimble and elastic. Figurative speaking "*they are eating our lunch*".

We look forward to your proposed architecture & implementation of this new system. Below are our business requirements.

Our Agencies:

A **rental agency** is identified by a unique ***rental agency ID*** number, ***agency name***, ***address*** that is composed of the following elements: ***address line1***, ***address line 2 (***which is optional and used for apartment number, suite or any additional address information required), **city**, ***state code*** (which is the two-character code for a state in the US), ***zip code*** & ***country***. In addition, we also need to capture the agency's ***phone number,*** and ***email*** which is unique for all agencies as all emails are.

Our Customers:

EZ-Car Rental offer their services to two types of **Customers**: **Corporate Customers** & **Retail Customers**. **Corporate Customers** are individuals whose corporation have a contract with us to use our services with special corporate rate for their employee's rental services. On the other hand, **Retail Customers** are consumers not associated with a company and engaging in personal rental.

All Customers

To run our business, the application must store the following customer information for both types of **customer** (retail & corporate) so this data is common to both types of customers:

- A ***Customer ID*** number which uniquely identifies the customer, ***customer name*** which is composed of: ***first name***, ***last name***.
- ***Birth date***, ***Age***, ***Address*** which includes the elements: ***address line1***, ***address line 2 (***which is optional and used for apartment number, suite or any additional address information required), ***city***, ***state code*** (which is the two-character code for a state in the US), ***zip code*** & ***country***.
- Customer ***phone number*** & ***email*** (unique like all emails and required to rent).
- In addition, a driver license is required to reserve and rent a vehicle. Therefore, we need to capture the unique ***driver license number (an alpha numeric character string containing numbers & characters)***, ***driver license expiration Date*** and ***driver license state***. In addition, note the following business rule on the business importance of the ***driver license number***:

1. ***The driver license number is used throughout the business to identify a customer for searching, reporting etc.***

2. ***Therefore, the driver license number is the unique ID for a customer to be identified and managed from a business perspective.***

Business Requirements

Our Customers (Cont.):

- A very important attribute we need to capture for every customer is the **credit card**. A credit card includes the following attributes: ***credit card number*** that uniquely identifies the credit card and is a 16-character number digits, ***credit card owner name, credit card company issuing company name*** (such as American Express, Visa, MasterCard, Capital One, etc.), ***merchant name*** which is the credit card payment processing company that acts as an intermediary between our business and the customers' credit card companies or bank. The merchant handles the interaction between the purchase of a rental and the credit card company etc., validating credit card transaction. This merchant name attribute has business meaning and used throughout the business using a digit code number and the name of the merchant associated with the code. We currently use the following merchant code and merchant names throughout the world to handle our credit card processing:

Merchant Code	*Merchant Name*
1	Stax by Fattmerchant
2	Helcim
3	Dharma Merchant Services
4	Payment Depot
5	National Processing
6	Block
7	Intuit Quickbooks
8	PayPal
9	Stripe
10	Flagship Merchant Services
11	Clover

- Other attributes of credit card are ***expiration date, billing address*** composed of ***address line1, address line 2 (***which is optional and used for apartment number, suite or any additional address information required), ***city, state code*** (which is the two-character code for a state in the US), ***zip code*** & ***country***.
- In addition, ***credit card limit, credit card balance*** & ***activation status*** which is true if the credit card is active and can be used or false when disabled.
- During the interview with business steakholders we captured the following ***Business Rules*** related to a credit card:

1. ***You cannot reserve or rent one of our vehicles without a credit card***
2. ***A customer can have many credit cards they can use to pay for rental transactions.***
3. ***A credit card can be owned by the one customer or co-owned by other individuals such a family member or corporate entity the customer works for. Therefore, many customers can own the same credit card and a credit card can be owned by many customers.***

Business Requirements

Our Customers (Cont.):

Corporate Customers

Corporate Customers are customers who are renting vehicle during business travel and their company have a contract with **EZRental Inc**. These companies get special corporate rate for their employee's rental services. Therefore, for our **corporate customers only**, we must store the following attributes/properties: unique ***company ID*** (we have a unique ID number for each company doing business with us), ***company name***, ***company address*** which contains the elements: ***address line1***, ***address line 2*** (which is optional and used for apartment number, suite or any additional address information required), ***city***, ***state code***, ***zip code*** (which is the two-character code for a state in the US) & ***country***, in addition, ***company contact*** which is composed of ***company representative name***, ***contact phone number*** & ***contact email*** (unique as all email addresses). And finally, we need to store the ***company discount percentage rate*** which is the discounted percentage applied to a corporate customers rental. The company Discount percentage rate is store in the database as a decimal percentage value, for example 20% is stored as 0.02, 30% as 0.03. 50% as 0.05 etc. This discount percentage (0.0x) is applied to the **Vehicle Rental Categories** which determines the price of each category to determine the total discount. Therefore, when a corporate customer rents a vehicle from a vehicle category (such as economic, compact, standard etc.), this discount percentage is applied to each of the categories during the rental/reservation process. Note that every company has a different percentage rating depending on their contract with **EZ-Rentals Inc**. For example, some companies have 20% discount towards their rentals, which would be stored as 0.20 in the database, some have 30% (0.03) etc. Vehicle Rental Categories are discussed in more details later in these requirements.

Retail Customers

Retail Customers can (but don't have to) leverage promotional **discounts** or coupons obtain from other businesses, internet, magazine, organizations, etc., to save money on their rentals. Therefore, data unique to a retail customer that we need to capture for the promotional discount is unique random number ***discount ID*** which uniquely identifies a discount, a unique ***discount code*** or the coupon code itself used to redeem the coupon, which is an alphanumeric code **10-characters** long. This code is generated by our marketing team and published to magazines, newspapers, internet e-commerce sites, etc. Finally, the last attribute is ***discount code description*** or description of the discount. Examples of currently used ***discount ID***, ***discount code***, ***discount code description*** are shown in table below:

Discount ID	*Discount Code*	*Discount Code Description*
1234..	AAA9970054	**AAA Membership Discount - 25% off base rate plus 10% donated for breast cancer research.**
5678..	GOV8756921	**Government Employee Discount - 30% off base rate**
9101..	*STA3415632*	**State Employee Discount for 25% off base rate**
1213..	VET2055179	**Veteran Discount 35% off base rate Plus 10% donation to veteran's family fund.**
Etc..	Etc..	**Etc..**

Retail customers can opt-in to enrolled in the **EZPlus Rewards Program** where they earn points every time they rent and these points can be redeemed for future rentals. Note that the **EZPlus Rewards Program** is **optional** for retail customers & points are earned only when they rent vehicles. For the **EZPlus Rewards Program** we need to store unique random number ***EZPlus ID***, the unique ***Ezplus rewards code*** which is the code used in the business when managing the **EZPlus Rewards Program**. This random code is generated and assigned to a **Retail Customer** by the client application. The number starts with the 3-characters EZP and a 10-digit number e.g., **EZP9999999999**, and the final attribute is the ***EZPlus rewards earned points***, which is an integer that indicates the number of rewards points earned that accumulated after all the rentals and can be used to save on future rentals. **Examples of currently used *EZPlus ID*, *EZPlus rewards Code* and *EZPlus earned points* that we currently use are**:

EZPlus ID	*EZPlus Rewards Code*	*EZPlus Rewards Earned Points*
1234..	EZP9009854637	**10000**
5678..	EZP1000192461	**500**
9101..	EZP6493238865	**159000**
1213..	EZP2005135627	**23000**
	Etc..	**Etc..**

In this business, we have the following rules for our customers:

1. ***We only have two types of customers retail customer or corporate customers. No other type of customer exists.***

2. ***A customer cannot be a retail & corporate customer at the same time. A customer can only rent as a retail customer or as a corporate and these transactions must be separate. We don't want our customers to be able to combine both retail customer discounts, rewards program and corporate rates at the same time.***

Business Requirements (Cont.)

Our Vehicles:

EZ-Car Rental needs a system to manage their vehicles for renting, maintenance, selling, etc. Vehicles are classified into 4 main types: **CAR**, **SUV**, **MINIVAN**, and **CARGO VAN**. These are the vehicles most rented and available at every rental agency. Nevertheless, there are other categories of vehicles available only certain rental agency locations such as **RECREATIONAL VEHICLES**, **MOTORCYCLES**, **MOBILE HOMES**, etc. No matter what type of vehicle being rented, all vehicle types share the following common characteristics:

- Each vehicle is identified by the random number ***vehicle ID***. In addition, each vehicle is also identified by the alpha-numeric ***vehicle VIN number***. Note the following business rule on a ***vehicle VIN number***.

1. ***The vehicle VIN number is used throughout the business to identify a vehicle for searching, reporting etc.***
2. ***Therefore, the vehicle VIN number is the unique ID for a vehicle to be identified and managed from a business perspective.***

- Other attributes include the ***vehicle name*** composed of ***make***, ***model*** & **year**. Additional attributes are ***color***, also the ***license plate*** composed of the following components: ***license plate number***, ***license plate state***.
- More attributes are ***mileage***, ***transmission type*** of the vehicle. The Transmission Type attribute has business value thus used in reports and in the business processes. The values used for transmission type is a combination of a transmission code and a transmission code description as follows: fro transmission type we use are, , **Semi-automatic** & **dual-clutch**) and

Transmission Code	*Transmission Code Description*
1	Manual Transmission
2	Automatic Transmission
3	Continuously Variable Transmission (e.g., CVT).
4	Semi-automatic Transmission
5	Dual-clutch Transmission
6	Transaxle Transmission

- ***seat capacity*** which is the number of seats in the vehicle. Vehicles such as *cars* have a seat capacity of 5 passengers (2 in front and 3 in the back), *SUVs* have 7 or 8 passengers. Cargo Vans have only 2 passenger seat capacity, Minivan have 8 to 9 passengers, special vehicles such as passenger van hold 12 passenger seat capacity, a shuttles bus can hold 16 to 20 passengers, mini-buses 30 to 40 passengers and large busses can hold 70 passengers.
- All vehicles also have a special code and description that we use to track the vehicle status named ***vehicle status ID***. This is a unique number that identifies the status of a vehicle, which works in conjunction with ***vehicle status description*** which describes the status represented by the ***Vehicle Status ID***, such as **reserved**, **rented**, **available**, **maintenance**, **not available**, **transferred**, etc. Below Is the list of vehicle status IDs we are currently using and their descriptions:

Vehicle Status ID	*Vehicle Status Description*
1	Available
2	Reserved
3	Rented
4	Not available
5	Maintenance (Not available)
6	Dropped off and located at another agency
7	In Transport to Owning Agency
8	No Longer available for rental

Business Requirements (Cont.)

Our Vehicles (Cont.):

In addition to these attributes shared by all vehicles, there are 4 main categories of vehicle which share unique characteristics than the other types of vehicles found in our agencies. These 4 types are as follows:

- A **Car** is a vehicle whose ***trunk capacity*** (measured in cubic feet volume) is advertised to our customers. Customers can decide which vehicles better fits their needs based on the trunk capacity and number of luggage they are carrying etc. For example, a *luxury Mercedes E class* car has a trunk capacity of 18.5 cubic ft., which has a large trunk capacity.
- An **SUV** is a vehicle with a ***towing capacity*** attribute in pounds. Towing capacity is a single number in pound or could also be a decimal number in pounds. For example, some of our SUV have a maximum towing capacity of 3,000 pounds etc. Another attribute of SUV is an attribute classification if the SUV is ***All-Wheel-Drive***, which stores a Boolean value of **YES/NO** or **TRUE/FALSE**.
- A **Minivan** has the option of ***having a disability package***, which is also a Boolean value of **YES/NO** or **TRUE/FALSE**.
- Finally, a **Cargo Van**, has a ***cargo capacity*** in cubic feet volume. For example, the typical volume of our Vans is 245 cubic feet (cu.ft.). Cargo Vans also have a ***maximum payload*** attribute that determines how much weight in pound it can hold. Our cargo vans have typically a maximum payload of 3,880 lbs.

- As stated previously, there are other types of vehicles of interest that in some location we may want to store data on other than car, SUV minivans and cargo van.
- Note that the following Business Rules were identified by the business stakeholders on the vehicles:

1. ***A reservation/rental can only be for one of these four categories of Vehicles or other vehicle types, not a combination.***

2. ***This means, you can only rent either a car, SUV minivans, cargo van or other for a reservation or rental, not a combination such as a car & SUV at the same time. Each reservation is unique to one vehicle.***

Below are additional business rules for our vehicles and agency ownership:

1. ***Every vehicle is owned by one agency. The vehicle can be pick-up and dropped-off at any agency, but only one agency is the vehicle's owning agency. An agency can own many vehicles, but a vehicle can only be owned by one agency.***

2. ***A vehicle can currently be located at any agency depending on where it was dropped-off after a rental. We need to track the current agency where the vehicle is located, to arrange a transfer or a rental that will ultimately direct the vehicle to the owning agency.***

Reservation Process:

A vehicle must be reserved if a customer wants to guarantee the vehicle will be available for rental. There is a distinction between a reservation and a rental. A reservation guarantees a vehicle will be ready for you to be pick-up and rented. A rental means a customer complied with the reservation and rented the vehicle. On the other hand, a customer can walk into an agency and rent without reservation but only vehicles that are available at the time and not reserved.

We have the following business rules for reserving a vehicle reservation:

1. ***A reservation is NOT made for a specific vehicle, but to a vehicle rental category. Rental category examples are economy, intermediate, full size, luxury.***

2. ***Thus, a customer makes a reservation of a vehicle rental category at a rental agency. Therefore, the reservation process involves a customer a vehicle rental category and the rental agency*** where the vehicle will be picked up.

Business Requirements (Cont.)

Reservation Process (Cont.):

A **Vehicle Rental Category** contains a list of vehicles depending on the vehicle type: Car (economy, intermediate, full size, luxury), SUV (standard, full size etc.), or Cargo Van etc. Each of these categories have a different price range. Therefore, for a vehicle rental category we need to capture the unique ***vehicle rental category ID*** that identifies the category of the vehicle being reserved or rented, ***category name*** and finally ***category daily rental rate*** for the category. We used a specific code for our vehicle rental category ID, category name & daily rental rate. The table below shows the ID, category names and rate we currently using in our business:

Vehicle Rental Category ID	*Vehicle Rental Category Name*	*Category Daily Rental Rate*
1	Car-Economic	$113.99
2	Car-Compact	$115.99
3	Car-Intermediate	$116.67
4	Car-Standard	$119.99
5	Car-Full Size	$121.99
6	Car-Premium	$127.79
7	Car-Luxury	$139.99
8	SUV-Intermediate	$127.99
9	SUV-Standard	$128.99
10	SUV-Standard Elite	$135.99
11	SUV-Full Size	$148.99
12	SUV-Premium	$157.99
13	Minivan-Standard	$152.99
14	Van-Cargo Van	$19.95
15	Pick Up-Mid Size	$69.95
16	Pick Up-Full Size	$105.99
17	Motorcycle-Touring	$19.95
18	Motorcycle-Cruiser	$199.99
19	Motorcycle-Scooter	$79.95
20	Passenger Van (12 passengers)	$161.00
21	Passenger Shuttle (16 passengers)	$180.00
22	Passenger Shuttle (20 passengers)	$220.00
23	Passenger Mini-Bus (30 passengers)	$250.00
24	Passenger Mini-Bus (40 passengers)	$280.00
25	Passenger Large-Bus (80 passengers)	$300.00

We have the following business rule relate to a vehicle and a vehicle rental category:

1. ***A vehicle is a member of a vehicle rental category.***
2. ***A vehicle rental category can have one, none or many vehicles belonging to that category at any given time, nevertheless, a vehicle can only belong to one vehicle rental category.***

As stated previously, **a customer makes a reservation of a vehicle rental category at a rental agency**. Therefore, the reservation process requires the **customer**, **vehicle rental category** & **rental agency** for a reservation to be made. The following business rules apply to a reservation:

1. ***A vehicle can be reserved to be picked up at the INDICATED rental agency and dropped off at the SAME rental agency.***
2. ***A vehicle can be reserved to be picked up at the INDICATED rental agency and dropped off at a DIFFERENT rental agency.***
3. ***A reservation is made only for one pick-up rental agency, but a rental agency can have many reservations for pick-ups taking place.***
4. ***A reservation can only be for one drop-off rental agency, but a rental agency can have many reservations drop-offs taking place.***

When a customer reserves a vehicle rental category for a specific rental agency, we wish to capture the following:

- A unique ***reservation ID*** which is used by the business to manage and track reservations, the ***rental agency ID*** where the vehicle will be picked up, and the target ***reservation drop-off rental agency***.
- In addition, we need ***reservation pick up date, reservation pick up time, reservation drop off date*** and ***reservation drop off time***, also the ***reservation estimated rental cost***.

Business Requirements (Cont.)

Reservation Process (Cont.):

- Finally, we need to store the unique ***reservation status ID*** which is a unique number we use to indicate the status of a reservation and ***reservation status description*** which describe each of the status such as: **confirmed, cancelled, completed** etc. Below is an example of the ***reservation status ID*** and ***status description*** we currently use in our business.

Reservation Status ID	*Reservation Status Description*
1	Confirmed
2	Modified & reconfirmed
3	Cancelled
4	Fulfilled & closed
Etc..	Etc..

For a reservation we must adhere to the following business rules:

1. ***A customer can make none, one or many reservations for a vehicle rental category at a rental agency.***
2. ***A rental category can be reserved by none, one or many customers at a rental agency.***
3. ***A rental agency can get many or no reservations for a vehicle rental category by a customer.***
4. ***A reservation can only have one pick-up rental agency location, but a rental agency can have many reservation pick-ups happening.***
5. ***Each reservation has a drop-off rental agency (may be different than pick-up rental agency). A reservation can only have one drop-off rental agency location, but a rental agency can have many reservation drop-offs taking place.***

The Rental Process:

Once a vehicle has been reserved, the vehicle can be rented (picked up/dropped off) as per the scheduled of the reservation agreement. **A rental means a customer complied and fulfilled the reservation and rented the vehicle**.

For the rental process, the following business rules apply:

1. ***A customer rents a vehicle Rental Category at a rental agency. This means the rental process requires the customer, vehicle rental category, and & rental agency for a rental to be complete.***
2. ***A Rental includes a specific Vehicle of the vehicle rental category. A vehicle can be rented many times, but a rental is only for one vehicle only. You cannot rent multiple vehicles in one rental contract.***
3. ***During the rental process we may have any of the following business rules/scenarios:***

    1) ***A vehicle can be picked up at the SAME rental agency as indicated by the reservation and dropped off at the SAME rental agency.***
    2) ***Or a vehicle can be picked up at the SAME rental agency as indicated by the reservation and dropped off at ANOTHER rental agency.***
    3) ***Or a vehicle can be picked up at ANOTHER rental agency other than what was indicated by the reservation and dropped off at SAME rental agency of the reservation.***
    4) ***Finally, a vehicle can be picked up at ANOTHER rental agency other than what was indicated by the reservation and dropped off at ANOTHER rental agency of the reservation.***

    - Note that for scenarios 3 & 4, we cannot guarantee that the vehicle rental category of the reservation will be available at the agency other than what was agreed in the reservation. We will do our best to accommodate the change during these scenarios or find another vehicle that will be closed to the original reservation.

For the rental process, the following business rules also apply:

1. ***A rental can only be for one pick-up rental agency, but a rental agency can have many rental pick-ups taking place.***
2. ***A rental can only be to one drop-off rental agency, but a rental agency can have many rental drop-offs taking place.***

When a customer rents a vehicle at the rental agency, we need to capture the following information about the rental:

- The ***rental agreement ID*** that uniquely identifies the rental transaction, ***rental pick up date, rental pick up time, rental drop off date*** and ***rental drop off time, rental pick up odometer value*** and ***rental drop off odometer value***.

Business Requirements (Cont.)

The Rental Process (Cont.):

- In addition, a customers receive a vehicle with a full tank of gas and customers are expected to return the car on a full tank of gas otherwise they must pay a penalty upon return. Since we understand our customers are busy and may forget to return the car with a full tank of gas, we offer our customers with the option to pay in advance for a full tank of gas at our rates and don't have to worry about returning the vehicle with a full tank of gas. Therefore, we need to capture the unique ***rental fuel option ID*** or option chosen by the customer, ***rental fuel option description*** and ***rental fuel option additional cost***. We currently use the following fuel option IDs, descriptions, and example of each of the additional cost for the fuel option:

Rental Fuel Option ID	*Rental Fuel Option Description*	*Rental Fuel Option Additional Cost*
1	Return with a full tank or on return, pay for gas that is missing.	$13.97 (Important, this Decimal value of $13.97 is just an example, since the value is calculated during car return process and is based on the current price for a gallon of gas etc. therefore price will vary.)
2	Pay for full tank in advanced at time of rental, return car empty. No refund for unused gas.	$45.99 (Important, this Decimal value of $45.99 is just an example, since the value is calculated during car return process and is based on the current price for a gallon of gas etc. therefore price will vary.)

- Also, we give customer options for car insurance & protection, therefore we need to capture the unique ***insurance option ID, insurance option description*** and ***insurance option additional cost***. We currently use the following insurance option IDs, descriptions, and cost:

Rental Insurance Option ID	*Rental Insurance Option Description*	*Rental Insurance Option Additional Cost per Day*
1	No insurance. Opt-out.	$0.00
2	Collision Damage Waiver Max - Agency will pay for damage, lost or stolen vehicle.	$49.99
3	Collision Damage Waiver 3000 - Agency will pay for first $3,000 of loss or damage, renter pays all loss & damage after $3,000.	$39.99
4	Lability Extended Protection – Agency provides renter with third party liability protection up to $1 Million per accident for bodily injury or death or property damage to others.	$89.99
5	Roadside Assistance Plus – 24/7 roadside assistance, replacement for lost keys, flat tire service, fuel delivery, etc.	$15.99

- Other attributes required for the rental that we need to capture are the unique ***rental status ID & rental status description***. We currently use the following rental status IDs & descriptions:

Rental Status ID	*Rental Status Description*
1	Picked up as scheduled.
2	Dropped off as scheduled.
3	Returned late
4	In progress.
5	Roadside assistance in progress.
7	Unknown

Business Requirements (Cont.)

Vehicle Transportation:

We need to know where our vehicles are located at all times, such as at the Rental Agency that owns the vehicle, another Rental Agency that does not own the vehicle, being transported from one Rental Agency to another as a result of a vehicle transfer after a rental to the owning rental agency, being transported as a new delivery to a Rental Agency from our distribution center, being transported for maintenance, or currently being rented by a customer. Vehicles need to be tracked or location status known. At this time, we are only interested in tracking when a vehicle is transported from one Rental Agency to another Rental Agency under the following scenarios:

- Vehicle can be located at a Rental Agency that does not own the vehicle after a rental dropping off at a different location than the picked up owning Rental Agency, thus vehicle eventually needs to be transported and delivered to the owning agency.
- Another non-owning Rental Agency requests support from other Rental Agency(s) for loans of vehicle(s) to borrow due to an unexpected busy period and requesting agency is short on inventory. After the first agency is done with the loaner vehicles, these vehicles need to be returned to the borrowed owning Rental Agency(s).
- In our current process & systems we currently use the following reason IDs and reason descriptions:

Transport Reason ID	*Transport Reason Description*
1	Rental Drop off at different location
2	Vehicle Loaned to another Agency
3	Pick up from Distribution Center
4	Drop off to Distribution Center
5	Vehicle sent for maintenance
7	Unknown

Note that transportation to and from Rental Agency is executed by an employee who is part of a transportation team or drivers. Therefore, when an employee executes a transport request of a vehicle to and from Rental Agencies, we need to capture the following information:

- ***Transport pickup agency ID, Transport drop-off agency ID, Driver departure date, driver departure time, vehicle pick up date, vehicle pick up time, transport completed arrival date, transport completed arrival time, estimated arrival date, estimated arrival time,*** & actual ***transport time to completion***.

- In addition, we need to know at any time the transport status and transport status description of the transfer, such as: transfer completed, on route to pick up location, on route from pick up location, etc. Therefore, we need to capture the ***Transport Status ID*** or unique number that identifies a status and the ***Transport Status Description***, or description of each status ID. Currently we track a transportation event using the following ID and description:

Transport Status ID	*Transport Status Description*
1	Transport completed
2	On route to pick up location.
3	On route from pick up location
4	At pickup location. In progress (Loading etc.)
5	Pickup location delay
7	Unknown

The goal again is to be able to know where our vehicles are located at any time and their status.

Conclusion:

The business data listed in this business requirements document is what we need to capture for our business to operate. As our business evolve, additional data will be required in the future. We will address these new requirements in future versions of the application. For example, invoice processing & employee management at our rental agencies are features on our roadmap. Therefore, our expectations are that the design is modular and scalable for future growth.

Business Requirements (Cont.)

The Rental Process (Cont.):

- Other attribute we need to capture the ***rental deposit*** for a rental. The rental deposit value is calculated based on the **rental period + 25% of the rental period** and for any damage or other charges that were incurred during the rental period. This deposit is refunded to the customer's credit card when the vehicle is returned in the condition in which it was rented.
- Finally another attribute we need to capture is the ***rental total cost*** or total cost that needs to be paid by the customer. This value is calculated based on selected *fuel option, insurance option, vehicle rental category* price and other factor such as such as duration of the rental etc.

We need to be able to associate a reservation to a rental and vice versa, therefore we maintain the following additional business rules for our rental & reservation:

1. ***A reservation is made for a rental and the opposite holds true; a rental is based on a reservation.***
2. ***But NOT all rentals are based on a reservation. We allow a customer to walk into a rental agency and rent a vehicle without a reservation.***
3. ***When a reservation is made for a rental, then it must be for only one rental, and a rental can be for a reservation but not mandatory since a customer can walk into an agency and rent a vehicle without a reservation.***

Our Employees:

EZ-Car Rental employees consist of customer service agents who interact with our customer to reserve and rent vehicles. In addition, we have auto specialists who work in our services centers servicing our vehicles. In addition, drivers to transport our vehicles from one agency to another and maintenance personnel who maintain our agencies and finally our business team that handles the day-to-day business activities in our agencies and other roles. For now, we are only interested in storing the following data for all these types of employees:

- An ***Employee ID*** which uniquely identifies the employee, ***employee name*** which is composed of: ***first name, last name***, also ***employee address*** which includes the components: ***address line1, address line 2, city, state code, zip code & country***. Also, ***employee phone, employee job title*** and ***employee email***. In addition, we need to capture the employee ***social security number***. Below are some business rules and usage for the ***EmployeeID*** and the ***social security number***.

1. The employee ***social security number*** needs to be protected and secured as per federal regulations. All security measures such as encryption, etc., need to be taken to protect the ***social security number***; therefore, the full ***social security number*** ***cannot*** be seen by employees, reports, and other business processes.
2. In special cases where the ***social security number*** needs to be displayed, only the last 4 digits will be shown using the following format ****_**_1234**. Nevertheless, the goal is NOT to display the ***social security number*** as much as possible, and it should only be used internally within the application for processing but not displaying.
3. The ***EmployeeID*** number is what is used throughout the business to identify an employee for searching, reporting. business processing, etc.
4. Therefore, the ***EmployeeID*** is the unique ID for an employee to be identified and managed from a business perspective.

Security & Application Access:

To access our systems proper security and authentication is required. Only authorized users can have access our agencies Point-0f-Sales & Back-End Management systems. In addition to our **EZRental.com** portal by our customers. Therefore, due to security and regulatory compliance purpose, we want to separate the employee access data from the customer access data by using two separate user accounts:

- Employee user accounts
- Customer user accounts

Security Access for Employees to Computer Systems in our Agencies (Employee User Accounts):
For our authorized employees & customer service employees to access the agencies Point-0f-Sales & Back-End Management systems they need to log in by entering a username & password for access to the application. This means every employee owns an employee user account.

An employee user account should store the user ***employee user account ID*** a unique identifier alpha-numeric string that identifies the employee user account, ***employee username*** another unique alpha-numeric that identifies each individual user, and finally the ***employee password*** alpha-numeric that is known only to the user, An employee can own one employee user account only, and an employee user account can only be owned by one employee only since the user account represents the identify of that one employee.

Security Access for our Customers who register for our EZ-CarRental.com web site (Customer User Accounts):
Customer who accesses our online portal to reserve and rent our vehicles also need a username and password to access our system, therefore each customer owns a customer user account.

A customer user account should store the user ***customer user account ID*** a unique alpha-numeric string identifier that identifies the customer user account, ***customer username*** another unique alpha-numeric value that identifies each customer, and finally, the ***customer password*** that is an alpha-numeric known only to the customer. A customer can own one customer user account only, and a customer user account can only be owned by one customer. For a period, we will need to register customers into our business but the **EZRental.com** web portal may be incomplete, therefore creating a customer user account for a new customer can be optional. We will force the creation of customer user accounts when they login to our portal for the first time.

Application Development & Technical Requirements

Interviews were held with project's ***Business Decision Makers (BDMs)***, ***stakeholders and Information System Technology Decision Makers (TDMs)*** to make a list of **Application Development & Technical Requirements to design & develop the application**.

The **Application Development & Technical Requirements** are highlighted in the following pages below:

Application Development & Technical Requirements

Introduction & Current Challenges

As described in the Business Requirements, the current rental system is outdated, with a poor user-experience, breaks often thus expensive to operate, does not meet our business requirements, and is not scalable so it cannot be easily updated with new features etc. Also, not elastic since it does not give us the flexibility to scale-up or scale-down based on business trends and seasonal changes in the market. We want to invest in modernizing our business with a new vehicle management system that can meet these challenges and give us a great user-experience, meet new business requirements, scalable, and elastic to adopt to business trends and seasonal market changes.

We have an outdated IT infrastructure in our datacenter and there is a current initiative to modernize our datacenter and also leverage cloud technology in a hybrid environment to save on cost, streamline our operations and drive innovation.

We look forward to your proposed architecture & implementation of this new system that will meet these requirements. Next sections contain the results of our application development & technical requirements.

Rental Agencies Application & Technical Requirements:

The rental agencies are location where customers both Retail & Corporate will engage our *Customer Service Representatives* to engage in rental/return activities in addition to other transactions such as registering, searching & updating customer information etc. Therefore, the application in the rental agencies is vital to the user-experience for both our *Customer Service Representatives* as well as our *customers*.

We are forecasting that is some locations such as major city centers and airports, there will be many customers engaging throughout the day thus increasing the risk of a poor customer experience in addition to the work overload and poor experience for our *Customer Service Representatives*. We want our *Customers* to be serviced quickly and efficiently with a great experience, and our *Customer Service Representatives* to be able to process each *Customer* easily and effectively. With these criteria in mind, the application at our rental agencies must adhere to the following requirements:

Rental Agency Application Architecture Requirements:

Below are the requirements for the application used in our rental agencies by our customer service representatives, inventory team, service personnel and other employees working in our agencies:

1. Client application processing, transaction and response must be fast to minimize service time for a customer.
2. All transaction processing should be done in the user's computer or desktop for fast processing and response.
3. Application Architecture must be reusable and scalable to support future updates and new feature enhancements, without a long development lifecycle.
4. Depending on the architecture NYC-Tech Solutions Inc., decides for the application in the rental agencies (Desktop client or Web client), the primary Application Development Platform we use is ***C# & .NET technologies***. For any Web related development, we support JavaScript, React, NodeJs and other standard Web Technologies. We have aligned ***C#.NET*** & **ASP.NET Web developers** that have been assigned to assist, support and update the application once NYCTech consultants complete the project and development of this system.
5. Rental Agency Desktop Application Security Authentication System – Proper security and authentication must be implemented to make sure only authorized customer service representative and other rental office employees can access the Point-0f-Sales with appropriate conditional access.

Application Development & Technical Requirements (Cont.)

Rental Agency Application Features and Functionalities Requirements:

The list of features and functionalities that we have compiled for the rental agencies' application are listed in the table below:

No.	Feature	Functionalities
1	EZRental Rental Agency Point-of-Sales (POS) System	▪ Car Rental, Car Return, New Customer Registration & Search/Print Customer Information, Customer Update, Customer Deletion, Customer Listing operations etc.
2	EZRental Rental Agency Back-Office Vehicle Inventory Management System	▪ Back-office system meant for employees to perform bulk IN-MEMORY inventory processing or management tasks on vehicles such as adding vehicles to the system, searching for vehicles, updating vehicles etc. ▪ This system is NOT meant for Point-of-Sales, but for the inventory management employees who need to search, add, remove etc., a large/bulk number of vehicles or employees during a session. ▪ Back-office vehicle Management features – Allows inventory personnel and employees to bulk-manage Cars, SUVs, Mini-Vans, Cargo Vans to be searched, added, removed, printed, listed etc.
3	EZRental Rental Agency Back-Office Credit Card Management System	▪ The EZRental Credit Card Management System is a Back-office system meant for the Credit Card Department Employees to manage Credit Card Information. These uses can Search/Print, Add, Edit & Delete credit card information in the database
4	EZRental Rental Agency Back-Office Employee & Customer User Account Management System	▪ The EZRental Customer & Employee User Account Management System is a Back-end system meant for IT ADMINISTRATOR Employees to manage both Employee & Customer USER ACCOUNTS.
5	EZRental Rental Agency Desktop Application Security Authentication System	▪ Proper security and authentication must be implemented to make sure only authorized employees can access the Point-0f-Sales, Back-End Management system or any other access to the applications.

Rental Agency Application Graphical User Interface Requirements:

- Graphical User-Interface should be fast rendering and user-friendly workflow.
- Visual screens or forms should be rich in color and appearance and navigation flow should be flexible and easy.
- The following UI controls or data field need to be pre-populated in GUI Screens:

  - **Addresses**

    - Any forms/UI which contains addresses, the STATE & COUNTRY fields should be automatically populated with a list of STATES or COUNTRIES, so the user does not have to manually enter a state or a country and simply select from drop-down list etc.

  - **Discount Codes:**

    - UI screens with customer's DISCOUNT CODE fields should be prepopulated with discount codes. The idea is the user should be able to select the discount to apply to a customer entry from a drop-down list/Combo Box etc. Note that this may or may not include the Discount Code Description on the UI screen as well.
    - Also note that the DISCOUNT CODE VALUES are generated by our Marketing Team and need to be pre-populated in the database before a code can be used. Therefore, the discount codes are prepopulated in the database.
    - Currently, when the Marketing Team generates a new code, they make the request to the database administrator to manually enter an update any new Discount Codes.
    - In the future, we want the application to have the necessary features for the Marketing Team to be able to manage the discount codes. This is not an immediate requirement out of the gate but should be targeted as part of a future upgrade.

Application Development & Technical Requirements

Rental Agency Application Graphical User Interface Requirements (Cont.):

- **EZPlus Rewards Codes:**
  - The EZPlus Reward UI screens with customer's EZPLUS REWARDS CODE fields should be prepopulated with the EZPlus Rewards code for the customer is being applied to. The idea is the user should be able to select the EZPLUS REWARD CODE to apply to a customer entry from a drop-down list/Combo Box etc. or be handled by the back-end database.
  - **Important**: The EZPLUS REWARDS CODE VALUES are NOT generated by a business entity in our organization, but AUTOMATICALLY GENERATED by the application on the fly when registering a new customer. This is a different approach compared to the DISCOUNT CODE which are generated by Marketing Team. In this case, the EZPlus Rewards Code values are generated by the application and available via the UI screen to be used or some other method of generation.
  - To finalize this requirement, the idea is the EZPlus Rewards Code should be automatically generated and either appear in the UI Screen or automatically generated in the database.

- **Company Name:**
  - UI screens with corporate customer's COMPANY NAME fields should be prepopulated with the list of corporations that are members of our corporate program, which enables our users to avoid having to manually enter the company name. Note that this may or may not include the Company ID in the UI Screen which is a unique number with business value that we assign to each company.
  - Note that the company names. Company ids and other company data are managed by our Corporate Sales Team and need to be pre-populated in the database before any corporate customer processing can be made. Therefore, the company information is prepopulated in the database.
  - Currently, when the Corporate Sales Team adds a new corporation or company into the program, they make the request to the database administrator to manually enter and add the new company to the database.
  - In the future we want the application to have the necessary features for the Corporate Sales Team to have the functionality to manage the data of our corporate companies via the application. This is not an immediate requirement out of the gate but should be targeted as part of a future upgrade.

- **Vehicle Status:**
  - UI screens for vehicle inventory management, VEHICLE STATUS field should be prepopulated with the list of vehicle status. Based on the business requirements, the current list of vehicle status is listed in table below.

Vehicle Status ID	*Vehicle Status Description*
1	Reserved.
2	Rented.
3	Available.
4	Not available
5	Maintenance
6	Transferred to another agency

  - Currently populating the database with a vehicle status record is handled manually by the database administrator. In the future we would like the application to have the necessary features for our business to be able to manage the vehicle status data. This is not an immediate requirement out of the gate but should be targeted as part of a future upgrade.

- **Rental Agency:**
  - UI screens that required adding or managing a RENTAL AGENCY field should be prepopulated with the list of rental agencies in our company.
  - Currently populating the database with a rental agency record is handled manually by the database administrator. In the future we would like the application to have the necessary features for our business to be able to manage the rental agency data. This is not an immediate requirement out of the gate but should be targeted as part of a future upgrade.

Application Development & Technical Requirements

Rental Agency Application Graphical User Interface Requirements (Cont.):

- **Vehicle Rental Category:**

  - UI screens that require the use of the VEHICLE RENTAL CATEGORY fields, must be prepopulated with the list of vehicle rental categories. Based on the business requirements, the current list of vehicle rental categories is as follows:

Vehicle Rental Category ID	*Vehicle Rental Category Name*	*Category Daily Rental Rate*
1	Car-Economic	$113.99
2	Car-Compact	$115.99
3	Car-Intermediate	$116.67
4	Car-Standard	$119.99
5	Car-Full Size	$121.99
6	Car-Premium	$127.79
7	Car-Luxury	$139.99
8	SUV-Intermediate	$127.99
9	SUV-Standard	$128.99
10	SUV-Standard Elite	$135.99
11	SUV-Full Size	$148.99
12	SUV-Premium	$157.99
13	Minivan-Standard	$152.99
14	Van-Passenger Van (12 passengers)	$161.00
15	Van-Cargo Van	$19.95
16	Pick Up-Mid Size	$69.95
17	Pick Up-Full Size	$105.99
18	Motorcycle-Touring	$19.95
19	Motorcycle-Cruiser	$199.99
20	Motorcycle-Scooter	$79.95

  - Currently populating the database with vehicle rental category records is handled manually by the database administrator. In the future we would like the application to have the necessary features for our business to be able to manage the vehicle rental categories data. This is not an immediate requirement out of the gate but should be targeted as part of a future upgrade.

- **Reservation Status:**

  - UI screens that require the use of the RESERVATION STATUS field, must be prepopulated with the list of reservation status data. Based on the business requirements, the current list of reservation status is as follows:

Reservation Status ID	*Reservation Status Description*
1	Confirmed.
2	Modified & reconfirmed.
3	Cancelled & Closed.
4	Fulfilled & Closed.
Etc..	Etc..

  - Currently populating the database with a reservation status record is handled manually by the database administrator. In the future we would like the application to have the necessary features for our business to be able to manage the reservation status data. This is not an immediate requirement out of the gate but should be targeted as part of a future upgrade.

Application Development & Technical Requirements

Rental Agency Application Graphical User Interface Requirements (Cont.):

- Rental Status:

  - UI screens that require the use of the RENTAL STATUS field, must be prepopulated with the list of rental status data. Based on the business requirements, the current list of rental status is as follows:

Rental Status ID	*Rental Status Description*
1	Picked up as scheduled.
2	Dropped off as scheduled.
3	Returned late
4	In progress.
5	Roadside assistance in progress.
7	Unknown

  - Currently populating the database with a rental status record is handled manually by the database administrator. In the future we would like the application to have the necessary features for our business to be able to manage the rental status data. This is not an immediate requirement out of the gate but should be targeted as part of a future upgrade.

- Rental Fuel Option:

  - UI screens that require the use of the RENTAL FUEL OPTION field, must be prepopulated with the list of rental fuel options data. Based on the business requirements, the current list of rental fuel option is as follows:

Rental Fuel Option ID	*Rental Fuel Option Description*	*Rental Fuel Option Additional Cost*
1	Return with a full tank or on return, pay for gas that is missing.	$13.97 (Important, this Decimal value of $13.97 is just an example, since the value is calculated during car return process and is based on the current price for a gallon of gas etc. therefore price will vary.)
2	Pay for full tank in advanced at time of rental, return car empty. No refund for unused gas.	$45.99 (Important, this Decimal value of $45.99 is just an example, since the value is calculated during car return process and is based on the current price for a gallon of gas etc. therefore price will vary.)

  - Currently populating the database with a rental fuel option record is handled manually by the database administrator. In the future we would like the application to have the necessary features for our business to be able to manage the rental fuel option data. This is not an immediate requirement out of the gate but should be targeted as part of a future upgrade.

Application Development & Technical Requirements

- **Rental Insurance Option:**

  - UI screens that require the use of the RENTAL INSURANCE OPTION field, must be prepopulated with the list of rental insurance options data. Based on the business requirements, the current list of rental insurance option is as follows:

Rental Insurance Option ID	*Rental Insurance Option Description*	*Rental Insurance Option Additional Cost per Day*
1	No insurance. Opt-out.	$0.00
2	Collision Damage Waiver Max - Agency will pay for damage, lost or stolen vehicle.	$49.99
3	Collision Damage Waiver 3000 - Agency will pay for first $3,000 of loss or damage, renter pays all loss & damage after $3,000.	$39.99
4	Lability Extended Protection – Agency provides renter with third party liability protection up to $1 Million per accident for bodily injury or death or property damage to others.	$89.99
5	Roadside Assistance Plus – 24/7 roadside assistance, replacement for lost keys, flat tire service, fuel delivery, etc.	$15.99

  - Currently populating the database with a rental insurance option record is handled manually by the database administrator. In the future we would like the application to have the necessary features for our business to be able to manage the rental insurance option data. This is not an immediate requirement out of the gate but should be targeted as part of a future upgrade.

Application Physical Technical Architecture

The following architecture was designed after reviewing the ***business requirements*** and ***technical requirements***:

Rental Agency Employees:

a. The system in the agencies used by the customer service representatives or front-line workers, must be able to quickly respond and execute the necessary requests such as:

   1. ***POS Customer Management (Retail Customer & Corporate Customer) features*** such as Customer Search & Print, New Customer Registration, Customer Update, Customer Deletion, & Customer Listing functionalities

   2. ***POS Vehicle Reservation, Rental & Return Management Feature*** such as vehicle reservations, Vehicle Rental & Vehicle Return functionalities.

   3. ***POS Vehicle Inventory Management Feature*** allows inventory personnel and employees to bulkmanage vehicles such as Cars, SUVs, Mini-Vans, Cargo Vans, and other vehicles to be searched, added, updated, deleted, printed, listed etc.

   4. ***POS Credit Card Management Feature*** such as Credit Card Search & Print, New Credit Card Registration, Credit Card Update, Credit Card Deletion, & Credit Card Listing functionalities.

b. Customer service agent must be provided rich user-interface experience

c. The system needs to be designed to perform well for the back-end personnel such as vehicle inventory managers and administrators, service personnel etc.

Corporate Offices:

The business operations are managed by the business employees & employees at the corporate office at rental agencies via the INTRANET Web Portal including the following features:

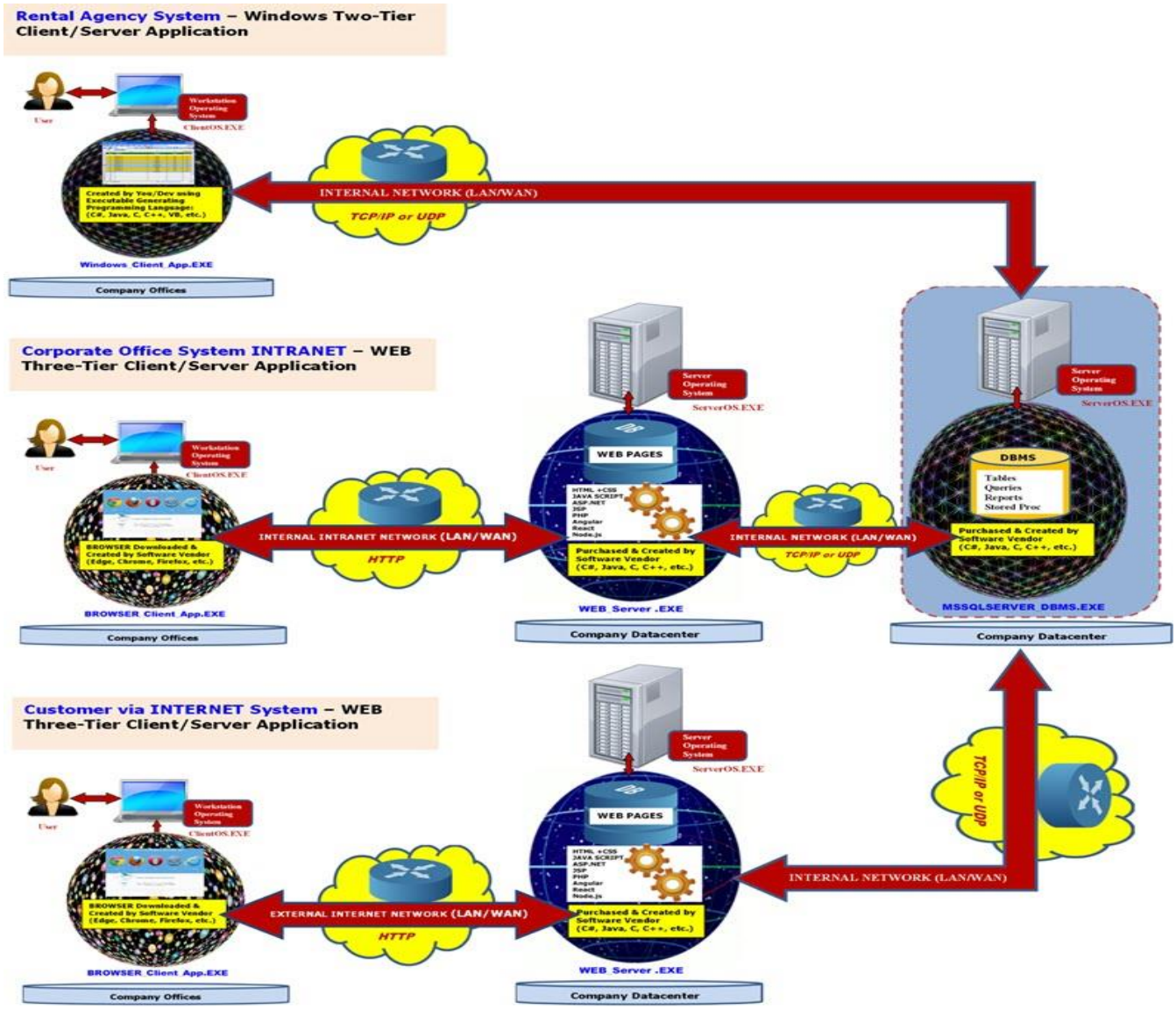
1. ***Intranet Web Enterprise Resource Planning Systems (ERP) Portal Feature*** such as providing access to Enterprise Resource Planning Systems (ERP) Applications such as: Customer Credit Card Management System, Vehicle Inventory Management System, Customer Relationship Management (CRM), Human Resource Management System, & Finance & Operations System, Marketing System, Customer & Field Service System etc.

5. ***Web EZRental Point-of-Sales Corporate Management Feature*** which allows employees to manage & execute Point-of-Sales (POS) transaction via the Intranet Web Poral such as: Search Customer Profile Information, Customer Account Management, Customer Registration, Customer Update, Customer Delete, & Customer Listing functionalities, Manage & Make Reservations of a Vehicle, Manage an existing Rental, etc.

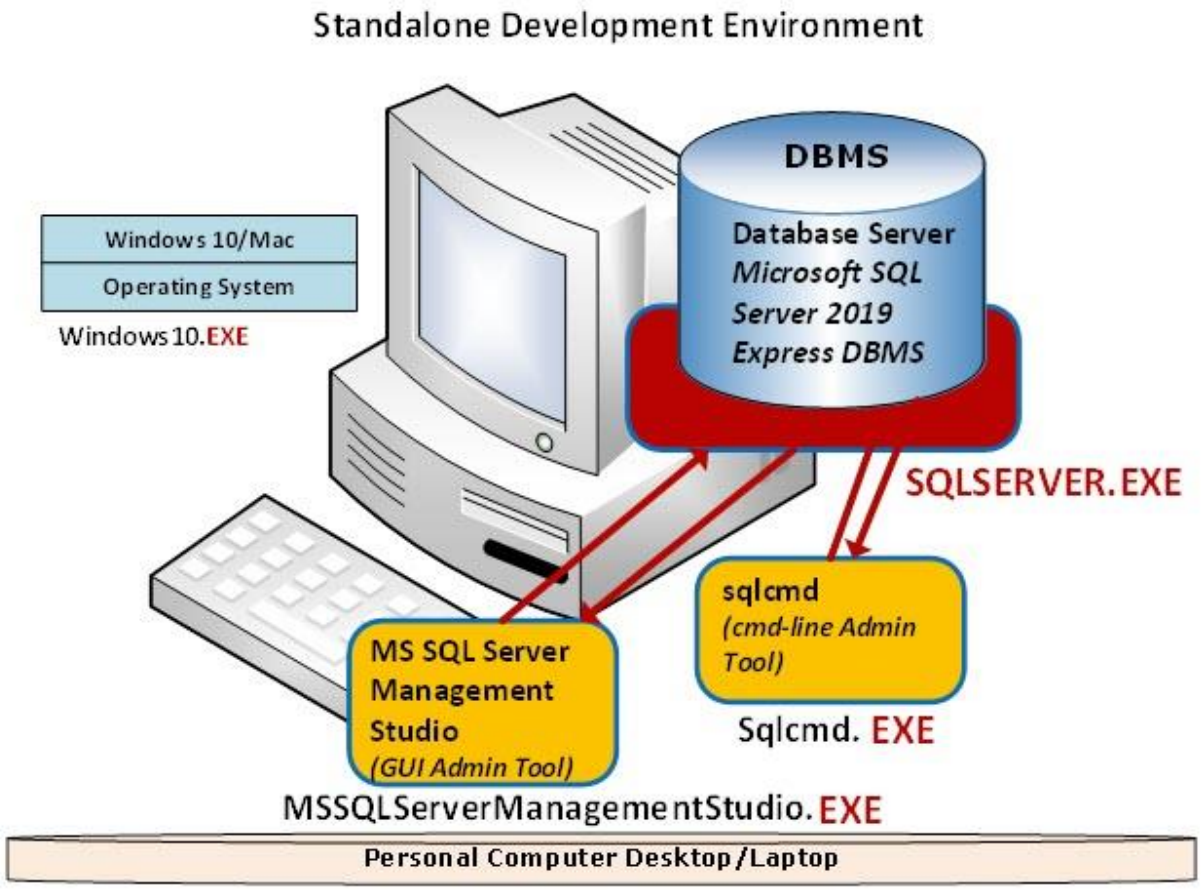
Customer self-service:

a. The customer must have ability to make reservations and manage their reservations and rentals online via the internet from anywhere in the world, considering a good user experience.

The figure below shows the demonstration of the architecture implemented:



Database Management System Development Environment & Physical Architecture



MS SQL Server 2019 Express DBMS has been used by ***EZRental Inc*** as standard database management system to store and preserve all the information. ***SQL command line admin tool*** along with ***MS SQL Server Management Studio GUI admin tool*** has been used to create the database management system as show in the figure above.

Project Roles & Responsibility

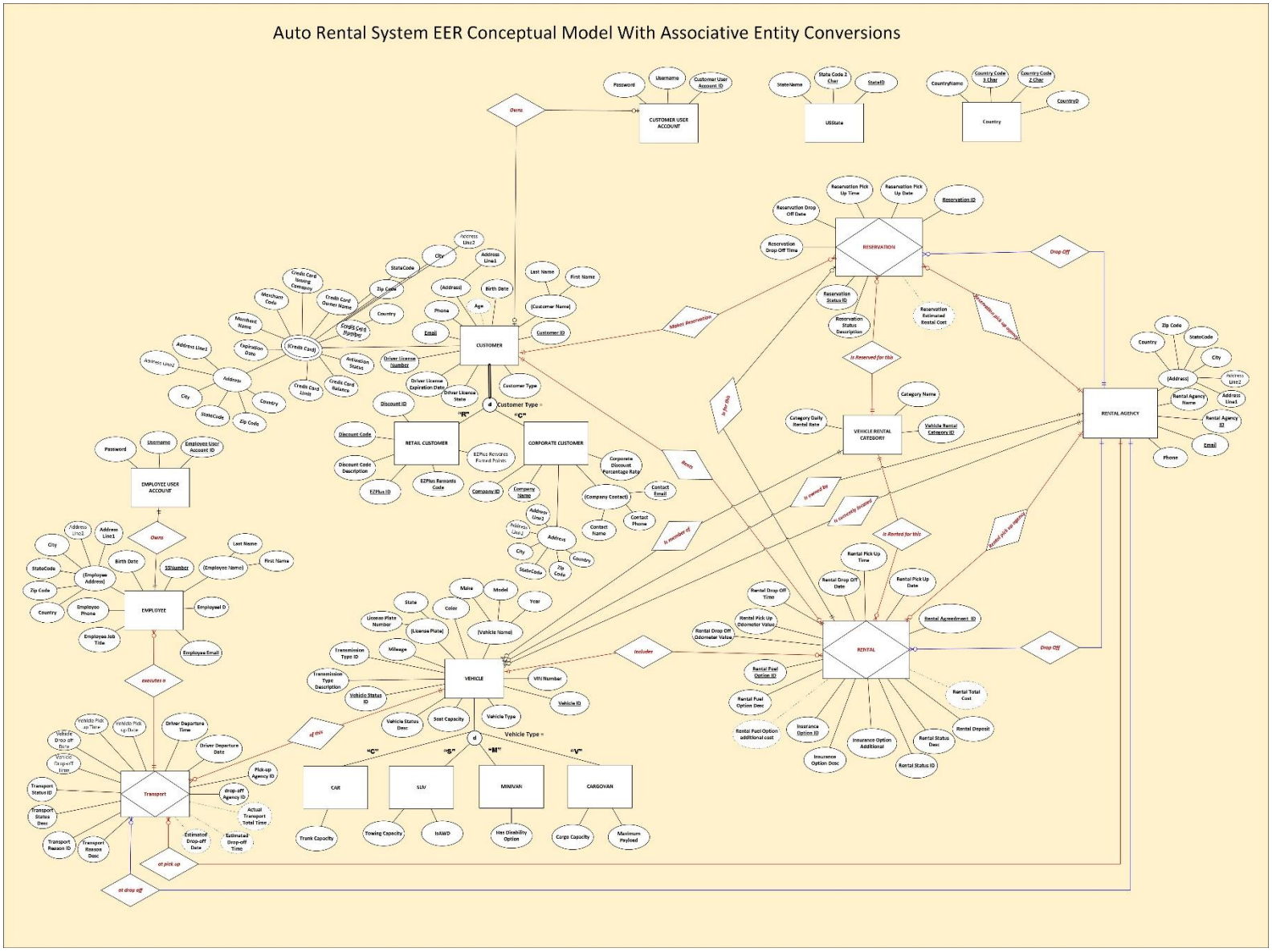
The Business/Database Analyst aligned the required database development team, and the table below describes each of the **Database Management System (DBMS) server application project** roles and the individual (s) that will execute the roles:

Person	Role	Description
Mr. Rodriguez	**Program Manager & Project Manager**	▪ Owner of the project and liaison to Manage the EZRental Inc., the customer. ▪ Activities include but not limited to. 1. Owner of project responsible for the success of the project. 2. Project Management 3. Scrum Master that ensures the project stays on time and moving in the right direction. Clear any obstacles impeding the team's progress etc.
Consultant #1 **Mr. Rodriguez**	**Business & Database Analyst**	▪ Interview with the stakeholders at **EZRental Inc** were done by the **Business/Database Analyst.** And create the ***Business Requirements*** that will be the foundation to the database design & implementation. ▪ Activities include but not limited to: 1. Engage in discovery activities & interview the stakeholders at **EZRental Inc**. 2. From the interview and discovery ***create*** **1) ER/EER Conceptual Data Model from the business requirements & 2) Normalized Logical Model**.
Consultant #2, 3, 4 & 5 **(Manjil Itani)**	**Database Developers**	▪ Use the Normalized Logical Model created by consultant #2 to create the Data Dictionary, Physical Schema Diagram, and Implement the Database Application for the Auto Rental System. ▪ Activities include but not limited to: 1. Use the ***Normalized Logical Model*** created by consultant #2 to **Create Data Dictionary tables for each logical table targeting MS SQL Server 2019 Express DBMS Data Types** & **2) Create Physical Schema Diagram**. 2. From these two deliverables, **1) implement the Database Application using MS SQL Server 2019 Express DBMS** for the Auto Rental System.
Consultant #6 **(Manjil Itani)**	**Database Administrator**	▪ As the DB Admin, install the DBMS, maintain, and operate the DBMS throughout its lifetime. ▪ Activities include but not limited to: 1. As DB Admin, you are to **1) Setup & install MS SQL Server 2019 Express DBMS. 2) MS SQL Developer Administrative tool**. 2. Also, as DB Admin, you are to **3) Operate & Maintain the DBMS**.

The Application Technical Architect/Analyst aligned the required application development team, and the table below describes each of the **Windows client & Browser client application development project** roles and the individual (s) that will execute the roles:

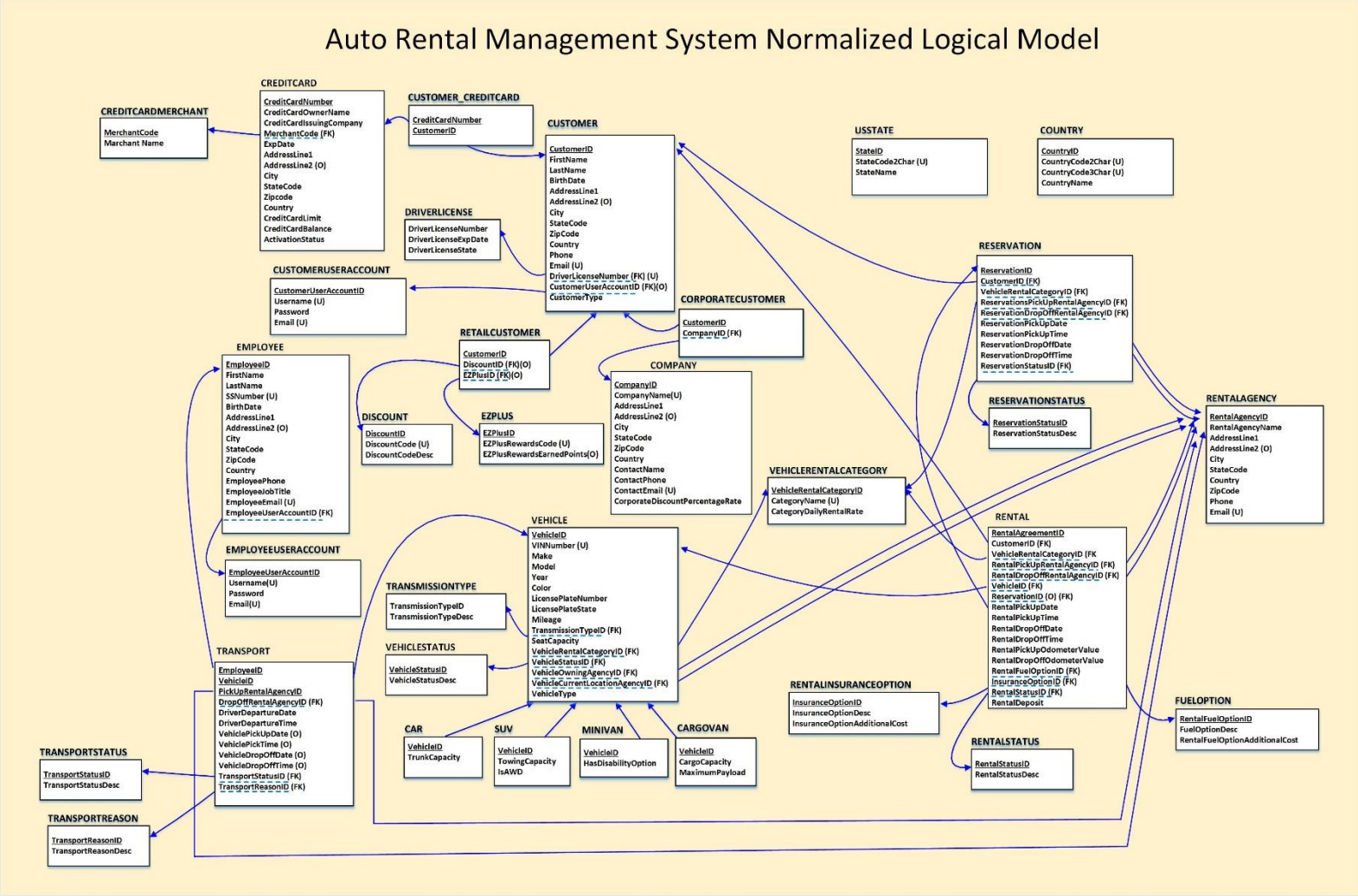
Person	Role	Description
Consultant #7 & 13 **Manjil Itani**	**Object-Oriented-Programming Architect**	▪ Interviews were held with the stakeholders at **EZRental Inc**., to derive the Application Technical Requirements and design and derive the **Class/Object Model Architecture** and plan for designing both **Windows Client Application** and the **Web Browser Application**. prospective and in addition design the Class/Object model based on interview and analysis results. ▪ Activities include but not limited to: 1. Engage in discovery activities & interview the stakeholders at **EZRental Inc**. 2. From the interview and discovery **1) Design/Architect the Object-Oriented-Programming Class/Object Model for the Windows Client Application**. 3. **Design/Architect the Object-Oriented-Programming Class/Object Model for the Web Browser Application**.
Consultants #8, 9, 10, 11, & 12 **Manjil Itani**	**Full Stack Application Developers**	▪ Object-Oriented-Programming developer to implement the **Windows Client Application** using **C# & .NET technologies** & on the database side, implement stored procedures and support the databased team as needed. ▪ Activities include but not limited to: 1. As full stack developer, **Programming & implementation of the Object-Oriented-Programming of Class/Object Model designed by consultant #7 for the Windows Client Application using C# & .NET Technologies**. 2. In addition, **Development of Database Stored Procedures, and other development requirements in the Back-end DBMS**. 3. From the technical requirements, **design a high-level Graphical User-Interface (GUID) wireframe**, & **implement the front-end UI Programming, features & functionality**
Consultant #14, 15, 16, 17 & 18 **Manjil Itani**	**Full Stack Web Developer**	▪ Object-Oriented-Programming developer to implement the **Web Browser Application** using **C# & ASP.NET technologies**. ▪ Activities include but not limited to: 1. As full stack developer, **Programming & implementation of the Object-Oriented-Programming of Class/Object Model designed by consultant #7 for the Web Browser Client Application using C# & ASP.NET Technologies**. 2. From the technical requirements, **design a high-level Graphical User-Interface (GUID) wireframe**, & **implement the Web front-end UI Programming, features & functionality in the Web Server Application**

Database Design Deliverable #1 – ER/EER Conceptual Model Diagram



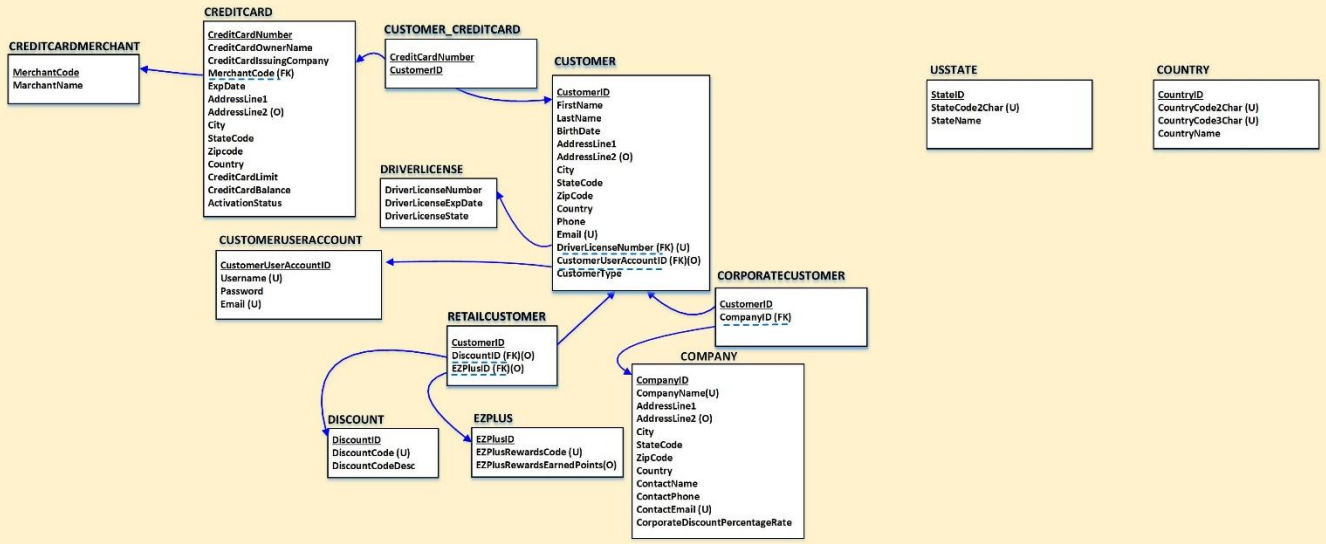
The ***ER/EER conceptual model*** (shown in the figure above) has be designed to highlight the ***relationship*** between the business ***entities*** identified in the **Application Business Requirements** as a part of **Analysis Phase** deliverables under ***Waterfall Methodology***.

Database Design Deliverable #2 – Normalized Logical Model Diagram



Normalized Logical Model is derived from the **Logical Model Diagram** which is the **ER/EER Diagram**, the ***Database Analyst/Architect*** has developed this model as part of the ***Design phase*** of the ***Waterfall methodology*** and has converted the entities in the **ER/EER Diagram** into ***database tables***.

Auto Rental Management System Normalized Logical Model Pilot Development & Deployment



Small **PILOT** (diagram above) has been implemented to test and prove the design for the database using the **ER/EER Diagram**.

Database Design Deliverable #3 – Physical Model Data Dictionary

The **13 tables** below represent the **Normalized Logical Model** developed into **Data dictionary.** These **13 tables** represent tabular listing of the metadata (**Column Name, Column Database Data Type, Column Constraints & Column Description**) for the all the columns of every table in the **Normalized Logical Model.**

#1:

CREDITCARD							
Column Num.	**Attribute/Column Name**	**Generic Data Type Name**	**MS SQL SERVER Data Type Name**	**Is it Required?**	**Length/Size /Format**	**Constraints**	**Description/ purpose**
1.	**CreditCardNumber**	String	**VARCHAR (16)**	Y	16	**PRIMARY KEY**	16-digit, customer's credit card number, **PRIMARY KEY HAS BUSINESSS MEANING**
2.	**CreditCardOwnerName**	String	**VARCHAR (100)**	Y	100	**NOT NULL**	Credit card owner's full name
3.	**CreditCardIssuingCompany**	String	**VARCHAR (50)**	Y	50	**NOT NULL**	Credit card company name
4.	**MerchantCode**	Number	**TINYINT**	Y	Default size of **TINYINT** data type	**FOREIGN KEY NOT NULL**	Code for company that processes the payment between customer and our business
5.	**ExpDate**	Date	**DATE**	Y	MM/DD/YY	**NOT NULL**	Date credit card expires
6.	**AddressLine1**	String	**VARCHAR (50)**	Y	50	**NOT NULL**	House number & street part 1
7.	**AddressLine2**	String	**VARCHAR (50)**	N	50	**NULL**	House number & street part 2 (OPTIONAL)
8.	**City**	String	**VARCHAR (30)**	Y	30	**NOT NULL**	City name
9.	**StateCode**	Character	**CHAR (2)**	Y	2	**NOT NULL**	2-character code representing US state
10.	**Zipcode**	String	**VARCHAR (10)**	Y	10	**NOT NULL**	US zip code
11.	**Country**	String	**VARCHAR (100)**	Y	100	**NOT NULL**	Country columns with international scope
12.	**CreditCardLimit**	Number/Decimal	**DECIMAL(8,2)**	Y	X= 8 Y = 2	**NOT NULL CHECK(CreditCard Limit greater than** 1000)	Discount rate, X = total number of digits before & after decimals, Y= total number of digits after decimal

13.	**CreditCardBalance**	Number/Decimal	**DECIMAL(8,2)**	Y	X= 8 Y = 2	**NOT NULL CHECK(CreditCardBalance greater than** 1000)	Discount rate, X = total number of digits before & after decimals, Y= total number of digits after decimal
14.	**ActivationStatus**	Boolean	**BIT**	Y	1	**NOT NULL**	True, if credit card is active False if the credit card is disabled

#2:

CREDITCARDMERCHANT							
Column Num.	**Attribute/Column Name**	**Generic Data Type Name**	**MS SQL SERVER Data Type Name**	**Is it Required?**	**Length/ Size /Format**	**Constraints**	**Description/ purpose**
1.	**MerchantCode**	Number	**TINYINT**	Y	Default size of **TINYINT** data type	**PRIMARY KEY CHECK(MerchantCode between** 1 and 15)	Code for company that processes the payment between customer and our business
2.	**MerchantName**	String	**VARCHAR (50)**	Y	50	**NOT NULL**	Name of the payment processing company

#3:

CUSTOMER							
Column Num.	**Attribute/Column Name**	**Generic Data Type Name**	**MS SQL SERVER Data Type Name**	**Is it Required?**	**Length/Size /Format**	**Constraints**	**Description/ purpose**
1.	**CustomerID**	Number	**INT**	Y	Default size of **INT** data type	**PRIMARY KEY IDENTITY**	Unique number to identify both **Retail & Corporate** customers
2.	**FirstName**	String	**VARCHAR (50)**	Y	50	**NOT NULL**	First name of customer
3.	**LastName**	String	**VARCHAR (50)**	Y	50	**NOT NULL**	Last name of customer
4.	**BirthDate**	Date	**DATE**	Y	MM/DD/YY	**NOT NULL**	Customer's date of birth
5.	**AddressLine1**	String	**VARCHAR (50)**	Y	50	**NOT NULL**	House number & street part 1
6.	**AddressLine2**	String	**VARCHAR (50)**	N	50	**NULL**	House number & street part2 (OPTIONAL)
7.	**City**	String	**VARCHAR (30)**	Y	30	**NOT NULL**	City name

8.	**StateCode**	Character	**CHAR (2)**	Y	2	**NOT NULL**	2-character code representing US state
9.	**ZipCode**	String	**VARCHAR (10)**	Y	10	**NOT NULL**	US zip code
10.	**Country**	String	**VARCHAR (100)**	Y	100	**NOT NULL**	Country columns with international scope
11.	**Phone**	String	**VARCHAR (20)**	Y	20	**NOT NULL**	Phone number- scope is international
12.	**Email**	String	**VARCHAR (75)**	Y	75	**NOT NULL UNIQUE**	Email address. Must be unique.
13.	**DriverLicenseNumber**	String	**VARCHAR (25)**	Y	25	**FOREIGN KEY UNIQUE NOT NULL**	**Has business meaning** (Alpha-numeric)
14.	**CustomerUserAccountID**	String	**UNIQUEIDENTIFIER**	N	Default size of **UNIQUEIDENTIFIER** data type	**FOREIGN KEY NULL**	
15.	**CustomerType**	Character	**CHAR (1)**	Y	1	**NOT NULL**	**"R"** for **Retail Customers** **"C"** for **Corporate Customers**

#4:

CUSTOMER_CREDITCARD							
Column Num.	**Attribute/Column Name**	**Generic Data Type Name**	**MS SQL SERVER Data Type Name**	**Is it Required?**	**Length/Size /Format**	**Constraints**	**Description/ purpose**
1.	**CreditCardNumber**	String	**VARCHAR (16)**	Y	16	**PRIMARY KEY**	16-digit, customer's credit card number
2.	**CustomerID**	Number	**INT**	Y	Default size of **INT** data type	**PRIMARY KEY**	Unique number to identify both **Retail & Corporate** customers

#5:

RETAILCUSTOMER							
Column Num	**Attribute/Column Name**	**Generic Data Type Name**	**MS SQL SERVER Data Type Name**	**Is it Required?**	**Length/Size /Format**	**Constraints**	**Description/ purpose**
1.	**CustomerID**	Number	**INT**	Y	Default size of **INT** data type	**PRIMARY KEY**	Unique number to identify **ONLY Retail** customers

2.	**DiscountID**	Number	**INT**	N	Default size of **INT** data type	**FOREIGN KEY NULL**	Random number generated to apply discount to Retail customer
3.	**EZPlusID**	Number	**INT**	N	Default size of **INT** data type	**FOREIGN KEY NULL**	ID for **Retail** customers enrolled in EZPlus reward program which gives points on rental

#6:

DISCOUNT							
Column Num	**Attribute/Column Name**	**Generic Data Type Name**	**MS SQL SERVER Data Type Name**	**Is it Required?**	**Length/Size /Format**	**Constraints**	**Description/ purpose**
1.	**DiscountID**	Number	**INT**	Y	Default size of **INT** data type	**PRIMARY KEY IDENTITY**	Random number generated to apply discount to Retail customer. **HAS NO BUSINESS MEANING**
2.	**DiscountCode**	String	**VARCHAR (10)**	Y	10	**UNIQUE NOT NULL**	Code used to redeem the discount coupon for Retail customers
3.	**DiscountCodeDesc**	String	**VARCHAR (150)**	Y	150	**NOT NULL**	Discount description

#7:

EZPLUS							
Column Num	**Attribute/Column Name**	**Generic Data Type Name**	**MS SQL SERVER Data Type Name**	**Is it Required?**	**Length/Size /Format**	**Constraints**	**Description/ purpose**
1.	**EZPlusID**	Number	**INT**	Y	Default size of **INT** data type	**PRIMARY KEY IDENTITY**	ID to represent the **Retail** customers enrolled in **EZPlus reward program,** this number is randomly generated
2.	**EZPlusRewardsCode**	String	**VARCHAR (13)**	Y	Length: 13 char Format : EZP+10digit	**UNIQUE NOT NULL**	Randomly generated code to be assigned to **Retail** customers by client application
3.	**EZPlusRewardsEarnedPoints**	Number	**INT**	N	Default size of **INT** data type	**NULL**	Point earned by retail customer every time they rent, this point can be redeemed in the future rentals

#8:

COPRORATECUSTOMER							
Column Num	**Attribute/Column Name**	**Generic Data Type Name**	**MS SQL SERVER Data Type Name**	**Is it Required?**	**Length/Size /Format**	**Constraints**	**Description/ purpose**
1.	**CustomerID**	Number	**INT**	Y	Default size of **INT** data type	**PRIMARY KEY**	ONLY **corporate customers**
2.	**CompanyID**	Number	**SMALLINT**	Y	Default size of **SMALLINT** data type	**FOREIGN KEY NOT NULL CHECK(CompanyID between** 1 and 20000)	ID representing company we have business with

#9:

COMPANY							
Column Num.	**Attribute/Column Name**	**Generic Data Type Name**	**MS SQL SERVER Data Type Name**	**Is it Required?**	**Length/Size /Format**	**Constraints**	**Description/ purpose**
1.	**CompanyID**	Number	**SMALLINT**	Y	Default size of **SMALLINT** data type	**PRIMARY KEY CHECK(CompanyID between** 1 and 20000)	**HAS BUSINESS MEANING** Id to represent the company we have business with
2.	**CompanyName**	String	**VARCHAR (100)**	Y	100	**UNIQUE NOT NULL**	Name of the company
3.	**AddressLine1**	String	**VARCHAR (50)**	Y	50	**NOT NULL**	House number & street part 1
4.	**AddressLine2**	String	**VARCHAR (50)**	N	50	**NULL**	House number & street part 1 (OPTIONAL)
5.	**City**	String	**VARCHAR (30)**	Y	30	**NOT NULL**	City name
6.	**StateCode**	Character	**CHAR (2)**	Y	2	**NOT NULL**	2-character code representing US state
7.	**ZipCode**	String	**VARCHAR (10)**	Y	10	**NOT NULL**	US zip code
8.	**Country**	String	**VARCHAR (100)**	Y	100	**NOT NULL**	Country columns with

							international scope
9.	**CompanyRepName**	String	**VARCHAR (100)**	Y	100	**NOT NULL**	Name of the person representing this company
10.	**ContactPhone**	String	**VARCHAR (20)**	Y	20	**NOT NULL**	Phone number-scope is international
11.	**ContactEmail**	String	**VARCHAR (75)**	Y	75	**NOT NULL UNIQUE**	Email address of the companyMust be unique.
12.	**CorporateDiscountPercentageRate**	Number/Decimal	**DECIMAL(3,2)**	Y	X=3 Y=2	**NOT NULL**	Discount rate, X = total number of digits before & after decimals, Y= total number of digits after decimal

#10:

DRIVERLICENSE							
Column Num.	**Attribute/Column Name**	**Generic Data Type Name**	**MS SQL SERVER Data Type Name**	**Is it Required?**	**Length/Size /Format**	**Constraints**	**Description/ purpose**
1.	**DriverLicenseNumber**	String	**VARCHAR (25)**	Y	25	**PRIMARY KEY**	Alpha-numeric drivers license, **HAS BUSINESS MEANING**
2.	**DriverLicenseExpDate**	Date	**DATE**	Y	MM/DD/YY	**NOT NULL**	Date the license expires
3.	**DriverLicenseState**	Character	**CHAR (2)**	Y	2	**NOT NULL**	2 char state code of the driver license

#11:

CUSTOMERUSERACCOUNT							
Column Num.	**Attribute/Column Name**	**Generic Data Type Name**	**MS SQL SERVER Data Type Name**	**Is it Required?**	**Length/Size /Format**	**Constraints**	**Description/ purpose**
1.	**UserAccountID**	Number	**UNIQUEIDENTIFIER**	Y	Default size of **UNIQUEIDENTIFIER** data type	**PRIMARY KEY DEFAULT NEWID()**	User account ID auto generated with GUID
2.	**Username**	String	**VARCHAR(50)**	Y	50	**NOT NULL UNIQUE**	User name
3.	**Password**		**VARCHAR(50)**	Y	50	**NOT NULL**	Password for the user account

4.	**Email**	String	**VARCHAR (75)**	Y	75	**NOT NULL UNIQUE**	Email address of for the account

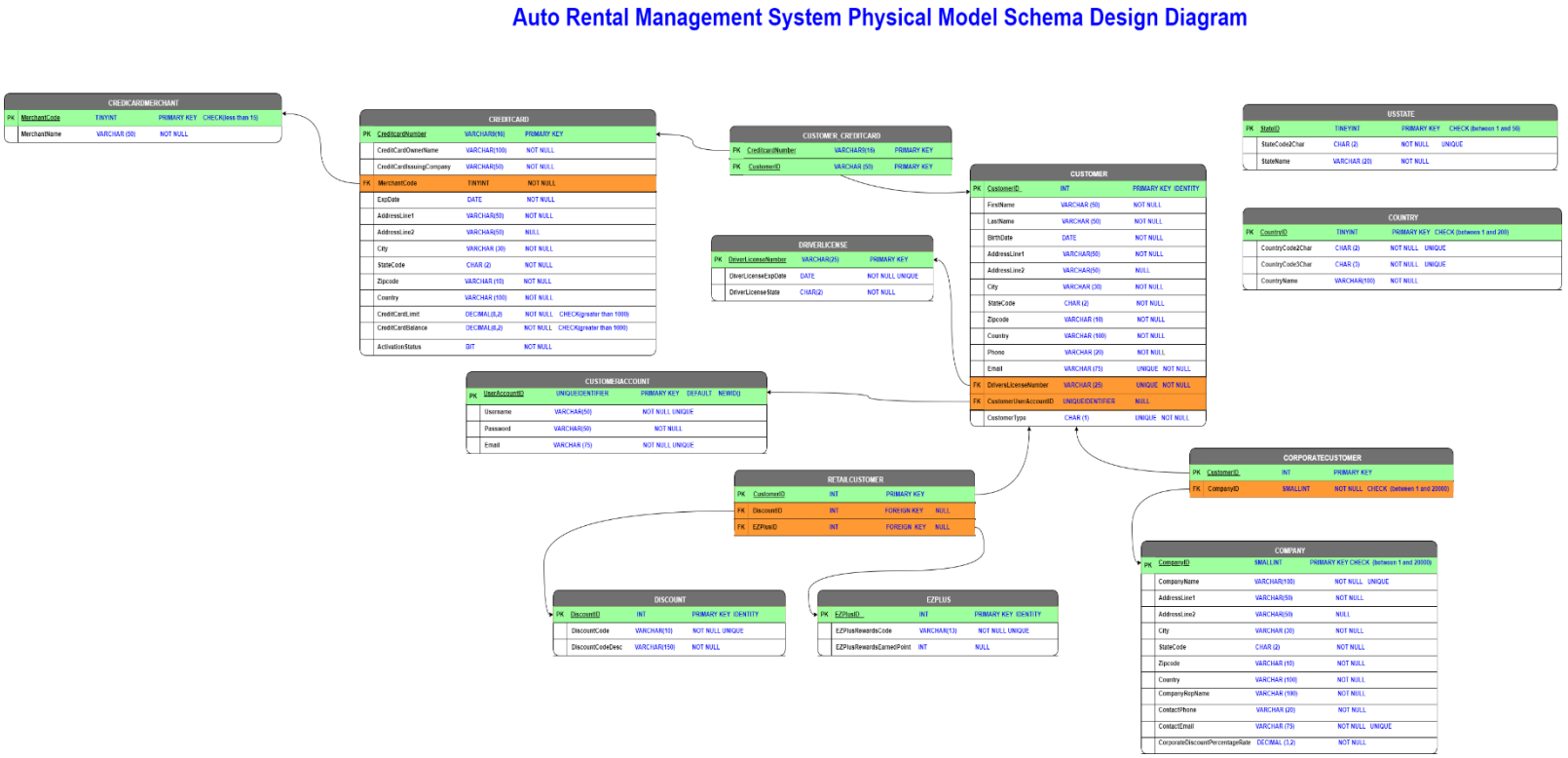
#12:

USSTATE							
Column Num.	**Attribute/Column Name**	**Generic Data Type Name**	**MS SQL SERVER Data Type Name**	**Is it Required?**	**Length/Size /Format**	**Constraints**	**Description/ purpose**
1.	**StateID**	Number	**TINYINT**	Y	Default size of **INT** data type	**PRIMARY KEY CHECK(StateID between** 1 and 56)	Numeric code representing the State
2.	**StateCode2Char**	Character	**CHAR (2)**	Y	2	**NOT NULL UNIQUE**	2-character code representing US state
3.	**StateName**	String	**VARCHAR(20)**	Y	20	**NOT NULL**	Full name of the state

#13:

COUNTRY							
Column Num.	**Attribute/Column Name**	**Generic Data Type Name**	**MS SQL SERVER Data Type Name**	**Is it Required?**	**Length/Size /Format**	**Constraints**	**Description/ purpose**
1.	**CountryID**	Number	**TINYINT**	Y	Default size of **INT** data type	**PRIMARY KEY CHECK(CountryID between** 1 and 200)	ID representing the country number
2.	**CountryCode2Char**	Character	**CHAR (2)**	Y	2	**NOT NULL UNIQUE**	2 char code representing country
3.	**CountryCode3Char**	Character	**CHAR (3)**	Y	3	**NOT NULL UNIQUE**	3 char code representing the country
4.	**CountryName**	String	**VARCHAR(100)**	Y	100	**NOT NULL**	Full name of the country

Database Design Deliverable #4– Physical Model Schema Design Diagram



The diagram above represents the **Physical Model Schema Design Diagram**, derived by combining **Normalized Logical Model Diagram** and the **Physical Model Data Dictionary**. This diagram summarizes the tables & relationships required to implement the final design component of the **DBMS Application**.

Database Implementation Deliverable #5 – Development & Implementation

The following code is implemented leveraging the **Physical Model Data Dictionary** into the **MS SQL Server** to generate **database tables** required to store information for ***EZRentals Inc*** services.

```
CREATE DATABASE EZRental;

USE EZRental;

--Create table to store Driver license information
CREATE TABLE DRIVERLICENSE
(
 DriverLicenseNumber     VARCHAR(25)     PRIMARY KEY,
 DriverLicenseExpDate    DATE            NOT NULL,
 DriverLicenseState      CHAR(2)         NOT NULL
);

--Create table to store Customer account information
CREATE TABLE CUSTOMERACCOUNT
(
 UserAccountID     UNIQUEIDENTIFIER    PRIMARY KEY DEFAULT NEWID(),
 UserName          VARCHAR(50)         UNIQUE NOT NULL,
 Password          VARCHAR(50)         NOT NULL,
 Email             VARCHAR(75)         UNIQUE NOT NULL
 );

 --Create table to store Customer's personal information
CREATE TABLE CUSTOMER
(
CustomerID                INT                 PRIMARY KEY IDENTITY,
FirstName                 VARCHAR(50)         NOT NULL,
LastName                  VARCHAR(50)         NOT NULL,
BirthDate                 DATE                NOT NULL,
AddressLine1              VARCHAR(50)         NOT NULL,
AddressLine2              VARCHAR(50)         NULL,
City                      VARCHAR(30)         NOT NULL,
StateCode                 CHAR(2)             NOT NULL,
ZipCode                   VARCHAR(10)         NOT NULL,
Country                   VARCHAR(100)        NOT NULL,
Phone                     VARCHAR(20)         NOT NULL,
Email                     VARCHAR(75)         UNIQUE NOT NULL,
DriverLicenseNumber       VARCHAR(25)         FOREIGN KEY REFERENCES DRIVERLICENSE(DriverLicenseNumber)
                                              ON DELETE CASCADE ON UPDATE CASCADE
                                              UNIQUE NOT NULL,
CustomerUserAccountID     UNIQUEIDENTIFIER    FOREIGN KEY REFERENCES CUSTOMERACCOUNT(UserAccountID)
                                              ON DELETE CASCADE ON UPDATE CASCADE
                                              NULL,
CustomerType              CHAR(1)             NOT NULL );
```

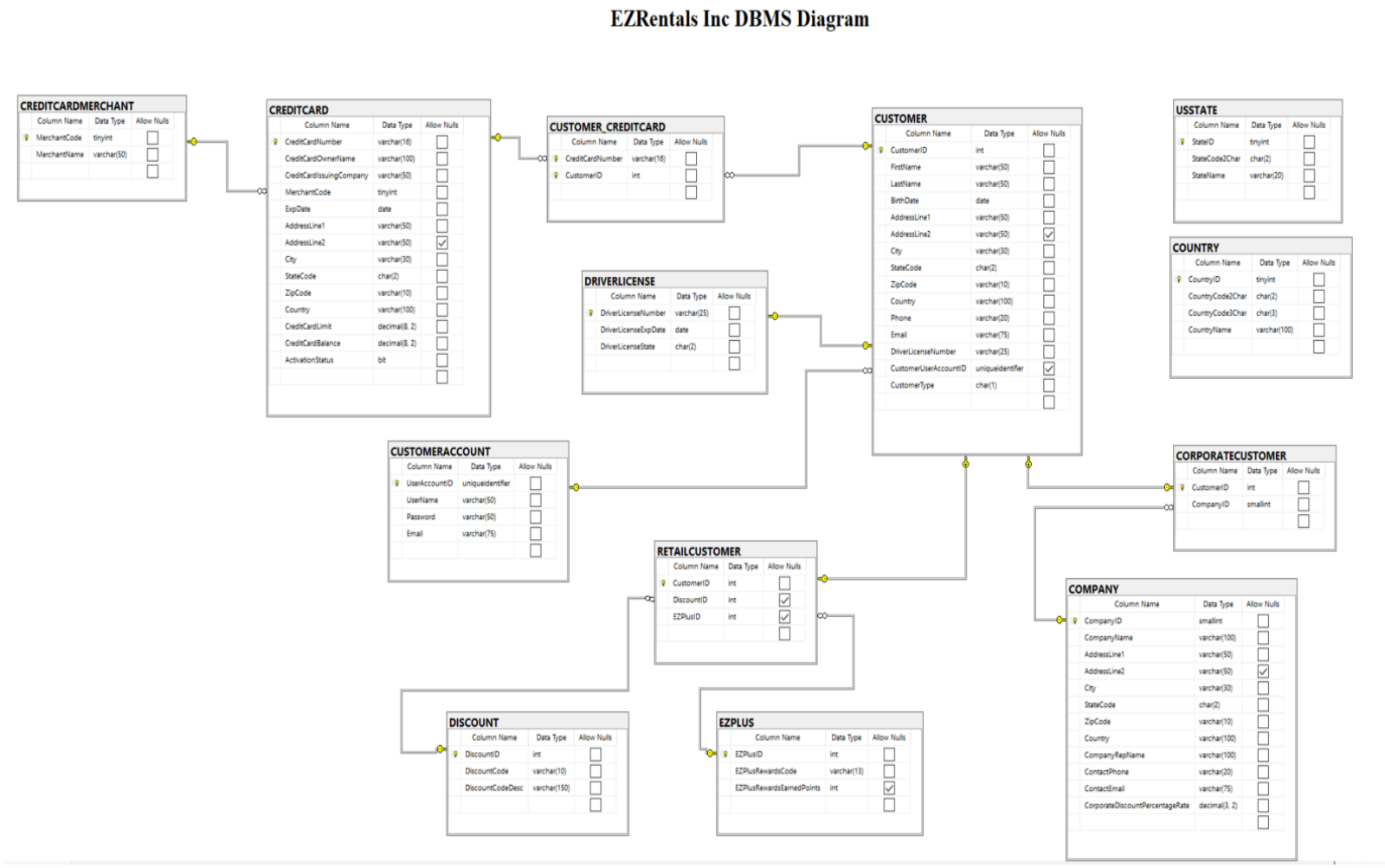
```
--Create table to store information Merchant's who deal with credit card transanction between customers and
EZRentals Inc
CREATE TABLE CREDITCARDMERCHANT
(
MerchantCode  TINYINT       PRIMARY KEY  CHECK(MerchantCode >= 1 AND MerchantCode <= 15),
MerchantName  VARCHAR(50)   NOT NULL
);

--Create table to store  credit card information
CREATE TABLE CREDITCARD
(
CreditCardNumber              VARCHAR(16)     PRIMARY KEY,
CreditCardOwnerName           VARCHAR(100)    NOT NULL,
CreditCardIssuingCompany      VARCHAR(50)     NOT NULL,
MerchantCode                  TINYINT         FOREIGN KEY
                                              REFERENCES CREDITCARDMERCHANT(MerchantCode)
                                                                    ON DELETE CASCADE ON UPDATE
CASCADE
                                                                    CHECK(MerchantCode >= 1 AND
MerchantCode <= 15)
                                                                    NOT NULL,
ExpDate                       DATE            NOT NULL,
AddressLine1                  VARCHAR(50)     NOT NULL,
AddressLine2                  VARCHAR(50)     NULL,
City                          VARCHAR(30)     NOT NULL,
StateCode                     CHAR(2)         NOT NULL,
ZipCode                       VARCHAR(10)     NOT NULL,
Country                       VARCHAR(100)    NOT NULL,
CreditCardLimit               DECIMAL(8,2)    NOT NULL CHECK( CreditCardLimit >= 1000),
CreditCardBalance             DECIMAL(8,2)    NOT NULL CHECK(CreditCardBalance >= 1000),
ActivationStatus              BIT             NOT NUL);


--Create table to store Customer's credit card information
CREATE TABLE CUSTOMER_CREDITCARD
(
CreditCardNumber  VARCHAR (16),
CustomerID        INT

CONSTRAINT pk_Customer_CreditCard
PRIMARY KEY (CreditCardNumber, CustomerID), -- Syntax #2-Create composite key

CONSTRAINT fk_Customer_CC_CardNumber -- Syntax #3-Create relationship to CreditCard table
FOREIGN KEY (CreditCardNumber)
REFERENCES CreditCard(CreditCardNumber)
ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT fk_Customer_CC_CustomerID -- Syntax #3-Create relationship to customer table
FOREIGN KEY(CustomerID)
REFERENCES Customer(CustomerID)
ON DELETE CASCADE ON UPDATE CASCADE
);
```

```
--Create table to store information about all the discount codes/coupns generated by EZRentals Inc.
CREATE TABLE DISCOUNT
(
  DiscountID        INT            PRIMARY KEY IDENTITY,
  DiscountCode      VARCHAR(10)    UNIQUE  NOT NULL,
  DiscountCodeDesc  VARCHAR(150)   NOT NULL
);

--Create table to store Customers EZPlus information
CREATE TABLE EZPLUS
(
EZPlusID                    INT             PRIMARY KEY  IDENTITY,
EZPlusRewardsCode           VARCHAR(13)     UNIQUE  NOT NULL,
EZPlusRewardsEarnedPoints   INT             NULL
);

--Create table to store Retail customers information
CREATE TABLE RETAILCUSTOMER
(
CustomerID   INT   PRIMARY KEY,
DiscountID   INT   FOREIGN KEY
             REFERENCES DISCOUNT(DiscountID)
                     ON DELETE CASCADE ON UPDATE CASCADE
                     NULL,
EZPlusID     INT   FOREIGN KEY
             REFERENCES EZPLUS(EZPlusID)
                     ON DELETE CASCADE ON UPDATE CASCADE
                     NULL,

CONSTRAINT fk_RetailCustomer_CC_CustomerID
FOREIGN KEY(CustomerID)
REFERENCES Customer(CustomerID)
ON DELETE CASCADE ON UPDATE CASCADE
);
--Create table to store information about Company associated to EZRentalInc
CREATE TABLE COMPANY
(
CompanyID                       SMALLINT            PRIMARY KEYCHECK(CompanyID >= 1 and CompanyID <= 20000),
CompanyName                        VARCHAR(100)        UNIQUE   NOT NULL,
AddressLine1                       VARCHAR(50)         NOT NULL,
AddressLine2                       VARCHAR(50)         NULL,
City                               VARCHAR(30)         NOT NULL,
StateCode                          CHAR(2)             NOT NULL,
ZipCode                            VARCHAR(10)         NOT NULL,
Country                            VARCHAR(100)        NOT NULL,
CompanyRepName                     VARCHAR(100)        NOT NULL,
ContactPhone                       VARCHAR(20)         NOT NULL,
ContactEmail                       VARCHAR(75)         NOT NULL  UNIQUE,
CorporateDiscountPercentageRate    DECIMAL(3,2)        NOT NULL
);
```

```
--Create table to store information about Corporate Customers
CREATE TABLE CORPORATECUSTOMER
(
CustomerID       INT       PRIMARY KEY,
CompanyID        SMALLINT
                 FOREIGN KEY REFERENCES COMPANY(CompanyID)
                            ON DELETE CASCADE ON UPDATE CASCADE
                            CHECK(CompanyID >= 1 and CompanyID <= 20000)
                            NOT NULL,

CONSTRAINT fk_CorporateCustomer_CC_CustomerID
FOREIGN KEY(CustomerID)
REFERENCES Customer(CustomerID)
ON DELETE CASCADE ON UPDATE CASCADE
);

--Create table to store information US states
CREATE TABLE USSTATE
(
StateID             TINYINT       PRIMARY KEY  CHECK( StateID >= 1 and StateID <= 56),
StateCode2Char      CHAR(2)       UNIQUE NOT NULL,
StateName           VARCHAR(20)   NOT NULL
);

--Create table to store information about all the countries that EZRental provide service to
CREATE TABLE COUNTRY
(
CountryID               TINYINT        PRIMARY KEY  CHECK(CountryID >= 1 AND CountryID <= 200),
CountryCode2Char        CHAR(2)        UNIQUE NOT NULL,
CountryCode3Char        CHAR (3)       UNIQUE NOT NULL,
CountryName             VARCHAR(100)   NOT NULL
);
```

Database Implementation Deliverable #6 – Implemented Physical Schema Diagram

The diagram below is generated in **MS SQL Serve** to show the implementation of table and relationship of the ***EZRental*** **Database**.



Database Implementation Deliverable #7 – Database Validation Unit Testing

The statements below show various **Data Manipulation Language (DML) SQL,** validation performed in tables in ***EZRental databse***.

1. INSERT STATEMENTS

a) Insert into CREDITCARDMERCHANT table

The following statement is executed in the SQL server to populate 11 rows of Credit Card Merchant information affiliated to EZRentals transactions into CREDIITCARDMERCHANT table in the EZRentals database.

```
--Name of database to perform operations
USE EZRental;

--insert into CREDITCARDMERCHANT table
INSERT INTO CREDITCARDMERCHANT(MerchantCode,MerchantName) VALUES(1,'Stax by Fattmerchant');
INSERT INTO CREDITCARDMERCHANT(MerchantCode,MerchantName) VALUES(2,'Helcim');
INSERT INTO CREDITCARDMERCHANT(MerchantCode,MerchantName) VALUES(3,'Dharma Merchant Services');
INSERT INTO CREDITCARDMERCHANT(MerchantCode,MerchantName) VALUES(4,'Payment Depot');
INSERT INTO CREDITCARDMERCHANT(MerchantCode,MerchantName) VALUES(5,'National Processing');
INSERT INTO CREDITCARDMERCHANT(MerchantCode,MerchantName) VALUES(6,'Block');
INSERT INTO CREDITCARDMERCHANT(MerchantCode,MerchantName) VALUES(7,'Intuit Quickbooks');
INSERT INTO CREDITCARDMERCHANT(MerchantCode,MerchantName) VALUES(8,'PayPal');
INSERT INTO CREDITCARDMERCHANT(MerchantCode,MerchantName) VALUES(9,'Stripe');
INSERT INTO CREDITCARDMERCHANT(MerchantCode,MerchantName) VALUES(10,'Flagship Merchant Services');
INSERT INTO CREDITCARDMERCHANT(MerchantCode,MerchantName) VALUES(11,'Clover');

--show data in CREDITCARDMERCHANT table;
SELECT * FROM CREDITCARDMERCHANT;
```

MerchantCode	MerchantName
1	Stax by Fattmerchant
2	Helcim
3	Dharma Merchant Services
4	Payment Depot
5	National Processing
6	Block
7	Intuit Quickbooks
8	PayPal
9	Stripe
10	Flagship Merchant Services
11	Clover

b) Insert in CREDITCARD table

The following statement is executed in the SQL server to populate 5 rows of customer Credit Card information into CREDIITCARD table in the EZRentals database.

```
--Name of database to perform operations
USE EZRental;

--insert into CREDITCARD table
INSERT INTO CREDITCARD(CreditCardNumber, CreditCardOwnerName, CreditCardIssuingCompany, MerchantCode, ExpDate,
AddressLine1,
AddressLine2, City, StateCode, ZipCode, Country, CreditCardLimit, CreditCardBalance,ActivationStatus)
VALUES('375678962002379','Max Vin','Amex',4,'2030-9-1','222 Broadway St','Apt 277','West
Village','36','50537','US',10000.00,5000.00,1);

INSERT INTO CREDITCARD(CreditCardNumber, CreditCardOwnerName, CreditCardIssuingCompany, MerchantCode, ExpDate,
AddressLine1,
AddressLine2, City, StateCode, ZipCode, Country, CreditCardLimit, CreditCardBalance,ActivationStatus)
VALUES('4845691230024699','Randy Rox','Visa',7,'2030-7-13','123 Brian Ave','Apt 9','West
Village','36','50537','US',20000.00,6000.00,1);

INSERT INTO CREDITCARD(CreditCardNumber, CreditCardOwnerName, CreditCardIssuingCompany, MerchantCode, ExpDate,
AddressLine1,
AddressLine2, City, StateCode, ZipCode, Country, CreditCardLimit, CreditCardBalance,ActivationStatus)
VALUES('375756910023649','John Doe','Amex',9,'2025-2-11','123 Mery
Blvd','2fl','Cooker','36','29782','US',30000.00,7000.00,1);

INSERT INTO CREDITCARD(CreditCardNumber, CreditCardOwnerName, CreditCardIssuingCompany, MerchantCode, ExpDate,
AddressLine1,
AddressLine2, City, StateCode, ZipCode, Country, CreditCardLimit, CreditCardBalance,ActivationStatus)
VALUES('6450004987603246','Kane Page','Discover',4,'2023-7-14','150 Yates Blvd','Apt
69','Jamaica','36','11456','US',40000.00,8000.00,1);

INSERT INTO CREDITCARD(CreditCardNumber, CreditCardOwnerName, CreditCardIssuingCompany, MerchantCode, ExpDate,
AddressLine1,
AddressLine2, City, StateCode, ZipCode, Country, CreditCardLimit, CreditCardBalance,ActivationStatus)
VALUES('374569003158945','Bill Cape','Amex',4,'2022-12-25','333 Washington Ave','15fl','East
Village','36','51489','US',50000.00,9000.00,1);


--show data in CREDITCARD table;
SELECT * FROM CREDITCARD;
```

CreditCardNumber	CreditCardOwnerName	CreditCardIssuingCompany	MerchantCode	ExpDate	AddressLine1	AddressLine2	City	StateCode	ZipCode	Country	CreditCardLimit	CreditCardBalance	ActivationStatus
1	374569003158945	Bill Cape	Amex	4	2022-12-25	333 Washington Ave	15fl	East Village	36	51489	US	50000.00	9000.00	1
2	375678962002379	Max Vin	Amex	4	2030-09-01	222 Broadway St	Apt 277	West Village	36	50537	US	10000.00	5000.00	1
3	375756910023649	John Doe	Amex	9	2025-02-11	123 Mery Blvd	2fl	Cooker	36	29782	US	30000.00	7000.00	1
4	4845691230024699	Randy Rox	Visa	7	2030-07-13	123 Brian Ave	Apt 9	West Village	36	50537	US	20000.00	6000.00	1
5	6450004987603246	Kane Page	Discover	4	2023-07-14	150 Yates Blvd	Apt 69	Jamaica	36	11456	US	40000.00	8000.00	1

2. SELECT STAMENT

c) Select a data from CREDITCARD table

Displays all columns from CREDITCARD table with the CreditCardNumber matching to

```
--Name of database to perform operations
USE EZRental;


--Selects data with matching parameter from CREDITCARD table
SELECT * FROM CREDITCARD WHERE CreditCardNumber='375678962002379';
```

CreditCardNumber	CreditCardOwnerName	CreditCardIssuingCompany	MerchantCode	ExpDate	AddressLine1	AddressLine2	City	StateCode	ZipCode	Country	CreditCardLimit	CreditCardBalance	ActivationStatus
375678962002379	Max Vin	Amex	4	2030-09-01	222 Broadway St	Apt 277	West Village	36	50537	US	10000.00	5000.00	1

3. UPDATE STATEMENT

a) Update a record in CREDITCARD table

The following statement updates all the columns in CREDITCARD Table with the CreditCardNumber = '375678962002379' only.

```
--Name of database to perform operations
USE EZRental;

--Update all the columns in CREDITCARD table except CreditCardNumber
UPDATE CREDITCARD SET CreditCardOwnerName = 'Manjil Itani', CreditCardIssuingCompany = 'Discover',
MerchantCode = 9,
ExpDate = '2027-02-01', AddressLine1 = '5789 S BlackStoen Ave', AddressLine2 = 'Apt 107', City='Chicago',
StateCode='17',
ZipCode = '60637', Country='US', CreditCardLimit=20000.00, CreditCardBalance=17000.00,
ActivationStatus = 0
WHERE CreditCardNumber='375678962002379';

--Selects all data from CREDITCARD table
SELECT * FROM CREDITCARD;
```

	CreditCardNumber	CreditCardOwnerName	CreditCardIssuingCompany	MerchantCode	ExpDate	AddressLine1	AddressLine2	City	StateCode	ZipCode	Country	CreditCardLimit	CreditCardBalance	ActivationStatus
1	374569003158945	Bill Cape	Amex	4	2022-12-25	333 Washington Ave	15fl	East Village	36	51489	US	50000.00	9000.00	1
2	375678962002379	Max Vin	Amex	4	2030-09-01	222 Broadway St	Apt 277	West Village	36	50537	US	10000.00	5000.00	1
3	375756910023649	John Doe	Amex	9	2025-02-11	123 Mery Blvd	2fl	Cooker	36	29782	US	30000.00	7000.00	1
4	4845691230024699	Randy Rox	Visa	7	2030-07-13	123 Brian Ave	Apt 9	West Village	36	50537	US	20000.00	6000.00	1
5	6450004987603246	Kane Page	Discover	4	2023-07-14	150 Yates Blvd	Apt 69	Jamaica	36	11456	US	40000.00	8000.00	1

CreditCardNumber	CreditCardOwnerName	CreditCardIssuingCompany	MerchantCode	ExpDate	AddressLine1	AddressLine2	City	StateCode	ZipCode	Country	CreditCardLimit	CreditCardBalance	ActivationStatus
374569003158945	Bill Cape	Amex	4	2022-12-25	333 Washington Ave	15fl	East Village	36	51489	US	50000.00	9000.00	1
375678962002379	Manjil Itani	Discover	9	2027-02-01	5789 S BlackStoen Ave	Apt 107	Chicago	17	60637	US	20000.00	17000.00	0
375756910023649	John Doe	Amex	9	2025-02-11	123 Mery Blvd	2fl	Cooker	36	29782	US	30000.00	7000.00	1
4845691230024699	Randy Rox	Visa	7	2030-07-13	123 Brian Ave	Apt 9	West Village	36	50537	US	20000.00	6000.00	1
6450004987603246	Kane Page	Discover	4	2023-07-14	150 Yates Blvd	Apt 69	Jamaica	36	11456	US	40000.00	8000.00	1

4. DELETE STATEMENT

b) Delete record from CREDITCARD table

The following statement deletes record from CREDITCARD table with CreditCardNumber='375678962002379'

```
--Name of database to perform operations
USE EZRental;

--Delete record with selected criteria
DELETE FROM CREDITCARD WHERE CreditCardNumber='375678962002379'

--Selects all data from CREDITCARD table
SELECT * FROM CREDITCARD;
```

CreditCardNumber	CreditCardOwnerName	CreditCardIssuingCompany	MerchantCode	ExpDate	AddressLine1	AddressLine2	City	StateCode	ZipCode	Country	CreditCardLimit	CreditCardBalance	ActivationStatus
374569003158945	Bill Cape	Amex	4	2022-12-25	333 Washington Ave	15fl	East Village	36	51489	US	50000.00	9000.00	1
375678962002379	Manjil Itani	Discover	9	2027-02-01	5789 S BlackStoen Ave	Apt 107	Chicago	17	60637	US	20000.00	17000.00	0
375756910023649	John Doe	Amex	9	2025-02-11	123 Mery Blvd	2fl	Cooker	36	29782	US	30000.00	7000.00	1
4845691230024699	Randy Rox	Visa	7	2030-07-13	123 Brian Ave	Apt 9	West Village	36	50537	US	20000.00	6000.00	1
6450004987603246	Kane Page	Discover	4	2023-07-14	150 Yates Blvd	Apt 69	Jamaica	36	11456	US	40000.00	8000.00	1

CreditCardNumber	CreditCardOwnerName	CreditCardIssuingCompany	MerchantCode	ExpDate	AddressLine1	AddressLine2	City	StateCode	ZipCode	Country	CreditCardLimit	CreditCardBalance	ActivationStatus
374569003158945	Bill Cape	Amex	4	2022-12-25	333 Washington Ave	15fl	East Village	36	51489	US	50000.00	9000.00	1
375756910023649	John Doe	Amex	9	2025-02-11	123 Mery Blvd	2fl	Cooker	36	29782	US	30000.00	7000.00	1
4845691230024699	Randy Rox	Visa	7	2030-07-13	123 Brian Ave	Apt 9	West Village	36	50537	US	20000.00	6000.00	1
6450004987603246	Kane Page	Discover	4	2023-07-14	150 Yates Blvd	Apt 69	Jamaica	36	11456	US	40000.00	8000.00	1

Conclusion

a) Purpose

The purpose of this project was to understand various elements and requirements in-order to create a meaningful database to store all the information surrounded by the **EZRental** business and make a smooth operation for the business to run.

b) What was done?

The techniques of Agile and WaterFall was implemented in-order to breakdown and divide various aspect of the project. With successful **Planning, Analysis, Design, Development, Validation** a central database management system has been created to store and preserve all the data/information required to run the **EZRental** business smoothly.

c) Key points of the project

The project was divided into various sprints:

SPRINT#1: Documentation skeleton to present the workflow to customers was created.

SPRINT#2: **EZRental** System Management Physical Model Schema was Developed.

SPRINT#3: Physical Model for **EZRental** DBMS was transformed to actual database using Microsoft SQL server.

SPRINT#4: Various DML SQL validations were performent to **EZRental** DBMS.

d) Outcome

A well formed database system is ready to store, preserve and present all the necessary information's abiding the business requirements for running **EZRental** has been created.