KUZEY KIBRIS KAMPÜSÜ
ODTÜ METU
NORTHERN CYPRUS CAMPUS

CNG 352

Term Project – Step 2 Report

Name: Shayan

Nadeem

Bhutta + Ali Murat

Chatkha

Student ID: 2542413

+ 2486843

Project Description:

This term project aims to develop a web application that will be used to handle the inventory and sale management of a car dealership called “Bhutta Carz”. This system is called BCMS (Bhutta Carz Management System). As of today, the company Bhutta Carz uses a very subtle and manual approach of logging every single invoice, hand out manual quotations to each potential buyer and having basically no record of any car present in the inventory and just using basic memory as a tool which is very inefficient and not feasible in terms of management. Through this application, firstly we will make a panel that will allow admins to manage the system and new users to sign in/up to the system. The users, after signing up, can also request a quotation for a specific car. The admins will login and add, update, or delete cars in the inventory. Moreover, they will be able to see all the available requested quotations for cars for each individual user and reply and if the quotation is accepted or suitable for the user can schedule an appointment through the system. This will also make use of the user details while making a digital invoice for the user after they have made a purchase through the system. This will better help record all the required details. Another thing registered users can do is sign up a form for their car to be parked at the dealership (not sold) and of course after inspection and scheduled appointment through the system, all these details can be logged into the system. This way, the car will also be added to the system but as a user’s car.

Data Requirements:

User

Each user after registration will be added to the BCMS. The data describing a certain user will include their unique cnic (computerized national identity card) number, their first and last names, date of birth, email address and their contact phone number. A user can be two types disjointly, Admin or Customer.

Admin

An admin will have two extra attributes to be stored, their username and password details.

Customer

A customer will be empty on its own having all the attributes of User entity. A customer can be two types overlappingly, Seller and Buyer.

Buyer

A buyer on its own will be empty and will store everything stored by the Customer entity.

Seller

A seller on its own will be empty and will store everything stored by the Customer entity. Furthermore, a seller can be of two types overlappingly, a Space Renter or a Car Dealer.

Space Renter

A space renter is a seller who is trying to rent a space for their car for a certain duration in the dealership and will pay rent depending on the duration their car stays for in the dealership. It is described by the agreed rent rate and commission decided between the space renter and the dealership.

Car Dealer

A car dealer is an actual seller and not a renter who wants to sell a car to us. It is an empty entity that stores everything in the parent entity i.e. Seller.

Car

A Car is the main component of our system. The data describing a specific car will have a unique vehicle id. It will have a make (example Honda), a model (example Civic), year of assembly, year of registration and year of import.
Moreover, mileage (in kilometers), condition of the car on a scale of 1 to 10 with 1 being extremely poor, full of scratches and dents to 10 being excellent with no scratches or dents at all will be stored. An attribute named accidental will be stored as a Boolean value with true meaning it is accidental and false meaning it is not accidental will be stored. The number of previous owners of the car will be stored. License plate-no (if the car is already registered) will be stored. The number of keys belonging to the car will be stored. The color of the car will be stored. A Boolean attribute named “reserved?” will be stored with true meaning that the car has already been reserved by another customer to be looked at or payment being in process or false being that the car is free to be looked at or bought by another user. Also, a Boolean attributed named “Sold?” will be stored, which when true means that the car has been sold and will no longer be shown in the inventory and false meaning it has yet to be sold and is still shown in the inventory. A car can be of two types disjointly, either a system car or a user car.

System Car

A System Car refers to a car completely owned by Bhutta Carz itself. The extra data stored will be related to the documentation of the car available. Separate ids of

documents namely transfer letter, sales invoice, sales certificate, police clearance and auction sheet will be stored in the system.

Customer Car

A Customer Car refers to a car that has not been bought by Bhutta Carz but is just being rented out on a space in the car dealership. For the specific type, everything that is stored in the parent car entity is stored here plus the duration the car has been parked for in the dealership as well except all the document ids which are requested from the space renter seller if the car wants to be potentially bought.

Quotation

A Quotation in BCMS generally describes a form that is filled in by a user who wishes to sell their car to us. This form is filled out and then in return, a quotation/price is given to the customer for the car they are trying to sell to us. For a quotation (form), yet again, the make, model, year of assembly, year of registration and year of import is stored. Accidental Boolean attribute is stored (look in Car section for details), number of keys, number of previous owners and condition of the car is stored. Additionally, a new Boolean attribute called “all_docs_avail?” is stored which if true means that for a specific quotation requested, all documents for that car are available and false meaning not all documents are available.
The demand price of the car given by the customer is stored while the offer price given by one of our staff members is also recorded. Finally, another Boolean attribute called “deal_made?” is stored which when true means that the customer agreed on the quotation provided and that the deal was successfully sealed or false meaning that the customer declined the quotation provided.

Transaction

A Transaction in BCMS describes any form of transaction payment to the customer or from the customer recorded in the system. Every transaction in the system will be described by a unique transaction id. Furthermore, the date and time of the transaction will be recorded. Finally, a Boolean attribute called “cash?” will be stored which when true means that cash was used in the transaction and false meaning that bank transfer was used in the transaction as our system does not allow card payments.

Appointment

A customer, after looking through our inventory and signing up, can schedule an appointment with us. After filling out the required appointment form for a certain appointment, a uniquely identified appointment id is stored together with the date and time the appointment is requested for is stored.

Potential Car

A potential car in BCMS is a weak entity that cannot exist without a car dealer (seller). It is described by all entities stored in the System Car entity. Once a deal has been made, a potential car becomes a system car.

Transaction Requirements:

Data Entry

Enter the details of a new admin (such as Ahmed Othman with CNIC as 35202-56569869)
Enter the details of a new buyer (such as Shayan Nadeem Bhutta with date of birth as 11/06/2001)
Enter the details of a new seller (such as Ali Murat Chatkha with phone number as +90 533 877 4143.
Enter the details of a new space renter (such as Muhammad Tabraiz Bilal with rent duration of 1 month)
Enter the details of a new system car (such as Suzuki Swift 2018 with transfer letter Id as 22451689)
Enter the details of a new customer car (such as Suzuki Mehran 2005 with color as silver)
Enter the details of a new quotation (such as Toyota Corolla 2023 with one previous owner and demand of 5000000 Rs)
Enter the details of a new transaction (such as transaction id 2526 with time as 20:40)
Enter the details of a new appointment (such as appointment id 4444 with date as 11/06/2024)

Data Update/Deletion

Update/delete the details of a system car.
Update/delete the details of a customer car.
Update/delete the details of an admin.
Update/delete the details of a space renter.
Update/delete the details of an actual seller.
Update/delete the details of a buyer.
Update/delete the details of a seller.

Data Queries

List all system cars owned by the dealership.
List all customer cars parked in the dealership.
List all system cars with milage less than 50000 (kms).
List all system cars with make of “Honda”.

List all system cars with model as “Civic”.
List all customer cars with the color “Black”.
List all customer cars with year of assembly as 2022.
List all customer cars with condition of 9 (out of 10).
List all customer cars with no accidents.
List all customer cars with the number of previous owners less than 2.
List all cars with no reservations in the system.
List all sellers in the system.
List all actual sellers in the system.
List all space renters in the system.
List all buyers in the system.
List phone numbers of all space renters with rent duration of less than a week.
List cnic of the buyer named “Shayan Nadeem Bhutta”.
List all quotations made to the seller named “Ali Murat Chatkha”.

EER Diagram:

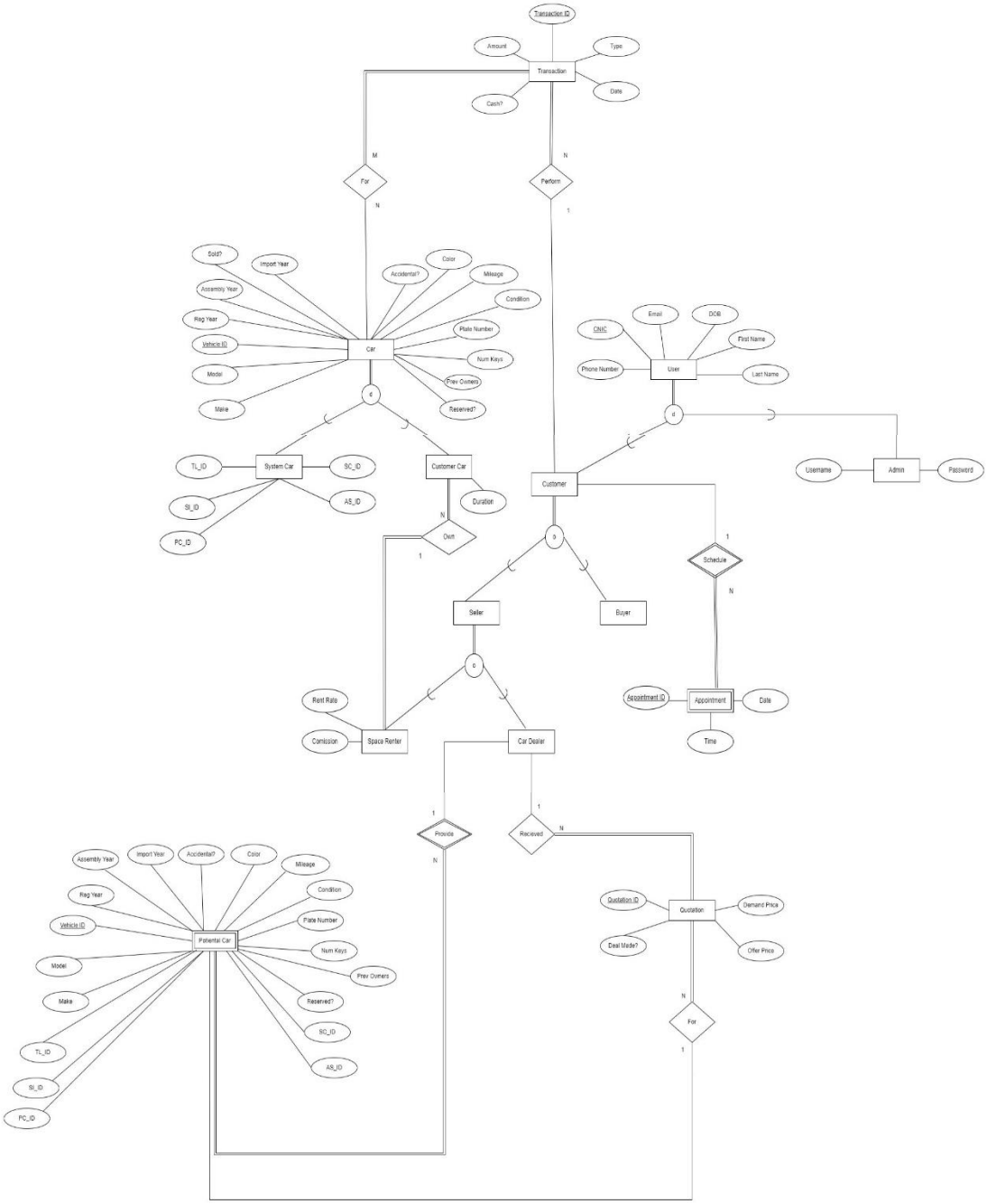

Figure 1: BCMS EER Diagram

Assumptions

1) A single car can be bought from and sold to our dealership so a particular car can have many transactions in our system.
2) We do not store any document ids for a customer car because they are required on the spot if there is a potential buyer for a customer car.
3) Appointment is a weak entity because it cannot exist without a customer to

schedule it.

4) Potential Car is a weak entity because we cannot have it without a car dealer (seller) to provide it.

Design Choices

All named attributes and entity selections have been selected after carefully analyzing the data requirements listed above in that section. All data types are either self-explanatory or explained in the data requirements section.

Constraints

Relationships

Car <-> Transaction – For – Many to Many : A transaction can be for many cars and a particular car can have many transactions in case it is both bought from and sold to the dealership. All transactions must be for a car and nothing else but not all cars have to have a transaction.

Customer <-> Transaction – Perform – One to Many : A customer can perform many transactions, but a particular transaction can only be performed by a single customer. All transactions need to be performed by a customer but not all customers have to perform transactions.

Customer <-> Appointment – Schedule – Many to One: A customer can schedule many appointments, but a particular appointment is scheduled by a single customer. All appointments need to be scheduled by customers but not all customers have to schedule an appointment. Schedule is a weak relationship here belonging to the weak entity “Appointment” as an appointment cannot exist without a customer to schedule it.

Space Renter <-> Customer Car – Own – Many to One : A space renter can own multiple customer cars in the dealership, but a particular car can only be owned by one space renter. All space renters need to own a car in the dealership and all customer cars need to be owned by space renters in the dealership.

Car Dealer <-> Quotation – Receive – Many to One: A car dealer can receive many quotations from the dealership, but a particular quotation is for one car dealer. All quotations are for a car dealer (seller) but not all car dealers need to receive a quotation and can sell a car directly without a quotation.

Quotation <-> Potential Car – For – One to Many : A quotation is for one potential car, but a potential car can have many quotations. All quotations need to be for a potential car, but a potential car does not need to have a quotation.

Car Dealer <-> Potential Cars – Provide – Many to One: A car dealer can provide

many potential cars to the dealership, but a particular potential car can only be provided by a single car dealer. Not all car dealers have to necessarily provide a potential car to the dealership, but all potential cars need to be provided by a car dealer to the dealership.

Specialization

Car <-> System Car OR Customer Car – Disjoint : Disjoint as it should either be a system car or a customer car but not at the same time.
User <-> Customer OR Admin – Disjoint : Disjoint as it should either be a user or an admin but not at the same time.
Customer <-> Seller AND Buyer – Overlapping: Overlapping as a customer can both be a buyer and seller at the same time.
Seller <-> Space Renter AND Car Dealer – Overlapping: Overlapping as a seller can both be a space renter and a car dealer at the same time.