**Project 2 Part 1: Car Rental Database**

**Aman Hogan-Bailey, George Gavin, Luke Sweeney**

The University of Texas at Arlington

2228-CSE-3330-004

11/04/2022

**Table of Contents**

[**1.** **Introduction** 3](#_Toc118406067)

[**2.** **Requirements and Assumptions** 3](#_Toc118406068)

[**3.** **ER Diagram** 4](#_Toc118406069)

[**3.1** **ER Diagram Entities Explanation** 5](#_Toc118406070)

[**3.2** **ER Diagram Relations Explanation** 7](#_Toc118406071)

[**4.** **Schema Diagram** 8](#_Toc118406072)

[**4.1** **Schema Diagram Design Choices** 9](#_Toc118406073)

[**HONOR CODE** 10](#_Toc118406074)

# **1. Introduction**

The goal of this project is to design and implement a database for keeping track of information about a car rental company. There will be 3 phases: 1st is to create an ER schema diagram, phase 2 implement it in SQL and finally in phase 3 to create queries and a GUI application of the car rental company. This document implements the first phase by creating a:

* *ER Diagram*
* *ER Schema Diagram*

# **2. Requirements and Assumptions**

1. The types of cars available for rental (compact, medium, large, suv, truck, and van) is not shown in the diagram and should be specified as such when entered into the database into the field Car\_type.
2. All cars of the same type have the same rental rate.
3. There is only one rental location.

# **3. ER Diagram**

Diagram

Description automatically generated

# **3.1 ER Diagram Entities Explanation**

1. CAR
   1. Each CAR has a Daily\_rate which is the cost per day of the specific car. It is stored as an INT.
   2. Each CAR has a Weekly\_rate which is the cost per week for a specific car and it is stored as an INT.
   3. Each CAR has a VehicleID which is a unique INT for each CAR for identification.
   4. Each CAR has a Model which is the model for a particular car and is stored as a variable CHAR.
   5. Each CAR has a Car\_type which is the type of CAR from the list: compact, medium, large, suv, truck, and van. It is stored as a variable CHAR.
   6. Each CAR has a Year which is the year the car was made/designed. It is stored as an INT.
2. CUSTOMER
   1. Each CUSTOMER has a unique IdNo that consists of an INT such as 1,2,3….
   2. Each CUSTOMER has a Name, that consists of a single initial and a last name.
   3. Each CUSTOMER has a Phone number consisting of a 12-CHAR string “817-272-5333.”
3. RENTAL
   1. Each RENTAL has a boolean value active\_or\_scheduled. This denotes whether the rental is active by a customer or is scheduled for a customer. (1 for active, 0 for scheduled).
   2. Each RENTAL has a unique Car\_id which comes from the VehicleID of CAR. It is a unique INT type.
   3. Each RENTAL has a Cust\_id which is a unique id for each customer that rents a car. It is derived from the IdNo in CUSTOMER.
   4. Each RENTAL can calculate the Amount DUE based on the other attributes. The Amount does not exist in the database and is a derived attribute.
4. DAILY
   1. Each DAILY has a Cust\_id which is a unique id for each customer that rents a car. It is derived from the IdNo in RENTAL.
   2. Each DAILY has a unique Car\_id which comes from the VehicleID of RENTAL. It is a unique INT type.
   3. Each DAILY has NoOfDays which is the number of days the rental will be rented as an INT.
   4. Each DAILY has a StartDate which is a character string that denotes the date the car was rented.
   5. Each DAILY calculates the ReturnDate which is a derived attribute. It is shown as a CHAR string.
5. WEEKLY
   1. Each WEEKLY has a Cust\_id which is a unique id for each customer that rents a car. It is derived from the IdNo in RENTAL.
   2. Each WEEKLY has a unique Car\_id which comes from the VehicleID of RENTAL. It is a unique INT type.
   3. Each WEEKLY has NoOfWeeks which is the number of days the rental will be rented as an INT.
   4. Each WEEKLY has a StartDate which is a character string that denotes the date the car was rented.
   5. Each WEEKLY calculates the ReturnDate which is a derived attribute. It is shown as a CHAR string.

# **3.2 ER Diagram Relations Explanation**

1. CUSTOMER -- MAKES A – RENTAL
   1. Each customer makes a rental for a car. This rental is issued by the company and can be scheduled to take place in the future. This means many customers can make rentals. And each rental can have many customers. Additionally, each customer must have a rental and each rental must have a customer.
2. CAR -- HAS A – RENTAL
   1. Each car has a rental. This means that the car now is planned or actively being rented by a customer. Not all cars at the company will have a rental at every moment in time. But every rental made will have a car associated with it.
3. RENTAL – IS – DAILY
   1. Rentals can be daily. This means that the customer is using the car for a period of less than seven days. Some rentals will be daily, and some will not be.
4. RENTAL – IS – WEEKLY
   1. Rentals can be weekly. This means that the customer is using the car for a period greater than or equal to seven days. Some rentals will be weekly, and some will not be.

**4. Schema DiagramDiagram

Description automatically generated with medium confidence**

# **4.1 Schema Diagram Design Choices**

1. CAR
   1. PRIMARY KEY
      1. Using a primary key for the VehicleId allows the customer and rental to be linked to each other easily for every rental made.
2. CUSTOMER
   1. PRIMARY KEY
      1. Using a primary key for the Cust\_id allows the customer to be uniquely identified and linked to their rental and the car they rented.
3. RENTAL
   1. FOREIGN KEYS
      1. Allowing the RENTAL to reference the VehicleID makes it so the RENTAL entity is able to derive information about the car and link the rental to a customer.
      2. Allowing the RENTAL to reference the Cust\_id makes it so the RENTAL entity is able to derive and link information about the customer.
4. DAILY
   1. FOREIGN KEYS
      1. Allowing the DAILY to reference the VehicleID makes it so the DAILY entity is able to derive information about the specific rental made.
      2. Allowing the DAILY to reference the Cust\_id makes it so the DAILY entity is able to derive information about the specific rental made.
5. WEEKLY
   1. FOREIGN KEYS
      1. Allowing the WEEKLY to reference the VehicleID makes it so the WEEKLY entity is able to derive information about the specific rental made.
      2. Allowing the WEEKLY to reference the Cust\_id makes it so the WEEKLY entity is able to derive information about the specific rental made.

# **HONOR CODE**

**I pledge, on my honor, to uphold UT Arlington's tradition of academic integrity, a tradition that values hard work and honest effort in the pursuit of academic excellence.**

**I promise that I will submit only work that I personally create or that I contribute to group collaborations, and I will appropriately reference any work from other sources. I will follow the highest standards of integrity and uphold the spirit of the Honor Code.**

**References**

None