CPS 510 A1 Technical Report:

By Eric Mergelas, Hetu Patel and Hunter Huchenski

## **Introduction:**

This report showcases the functionality and requirements of a car rental DBMS.

The field of car rentals deals with many types of data including but not limited to the people buying the cars and the cars themselves. Information about how long a car has been rented for and how much the rental costs are some of the subcategories of this data.

Needless to say, a DBMS system for a car rental service would greatly improve the efficiency of record keeping, especially considering the number of changes to data that would be in a business of that sort.

## **Requirements:**

Functional:

* The system must be able to deal with customer data such as name, driver's license and contact details.
* These systems must be able to deal with vehicle inventory data such as vehicle details, status and maintenance records.
* The system must also be able to manage payments.

Non-functional:

* Scalability: The system must remain relatively stable when entries increase.
* Security: The system must have strong data integrity, especially when dealing with payments and important user data.

## 

## 

## 

## Functional Tables Prototype:

### **1. Customers Table:**

* Customer\_ID (Primary Key)
* First\_Name
* Last\_Name
* Email
* Phone\_Number
* License\_Number
* Address

**Purpose:** This table stores customer details.

### **2. Cars Table**

* Car\_ID (Primary Key)
* Make
* Model
* Year
* License\_Plate
* VIN
* Daily\_Rental\_Price
* Availability\_Status (Available, Rented, Under\_Maintenance)

**Purpose:** Holds information on all cars available for rent.

### **3. Rental\_Transactions Table**

* Rental\_ID (Primary Key)
* Customer\_ID (Foreign Key from Customers Table)
* Car\_ID (Foreign Key from Cars Table)
* Rental\_Start\_Date
* Rental\_End\_Date
* Total\_Cost
* Status (Active, Completed, Canceled)

**Purpose:** Tracks each rental transaction, linking customers with cars.

### 

### 

### 

### **4. Payments Table**

* Payment\_ID (Primary Key)
* Rental\_ID (Foreign Key from Rental\_Transactions Table)
* Payment\_Amount
* Payment\_Date
* Payment\_Method (Credit Card, Cash, etc.)

**Purpose:** Stores payment details associated with rental transactions.

### **5. Car\_Maintenance Table**

* Maintenance\_ID (Primary Key)
* Car\_ID (Foreign Key from Cars Table)
* Maintenance\_Date
* Description
* Maintenance\_Cost

**Purpose:** Tracks maintenance activities for the cars.

### **Relationships:**

* **Customers to Rental\_Transactions:** One customer can have many rental transactions, but each rental is tied to one customer (1-to-Many).
* **Cars to Rental\_Transactions:** Each car can be rented multiple times, but each rental transaction involves one car (1-to-Many).
* **Rental\_Transactions to Payments:** One rental transaction can have multiple payments, but each payment is linked to one transaction (1-to-Many).
* **Cars to Car\_Maintenance:** Each car can have multiple maintenance records (1-to-Many).

### **Summary of Tables:**

* **Customers** and **Cars** are the core entities.
* **Rental\_Transactions** are the junction between customers and cars, recording when rentals happen.
* **Payments** and **Car\_Maintenance** handle financial and operational tracking respectively.