**Parking Space Management System**

The parking management system has been created to make it easier to manage and run a parking facility. Its main goal is to handle parking space allocation, vehicle tracking, ticket issuance, and payment management in an efficient manner. The system aims to streamline the process of assigning parking spots, keep track of vehicles, provide parking tickets, and handle payment transactions effectively.

1. **Key Features**

**Parking Space Management -** Ability to add and delete parking space.

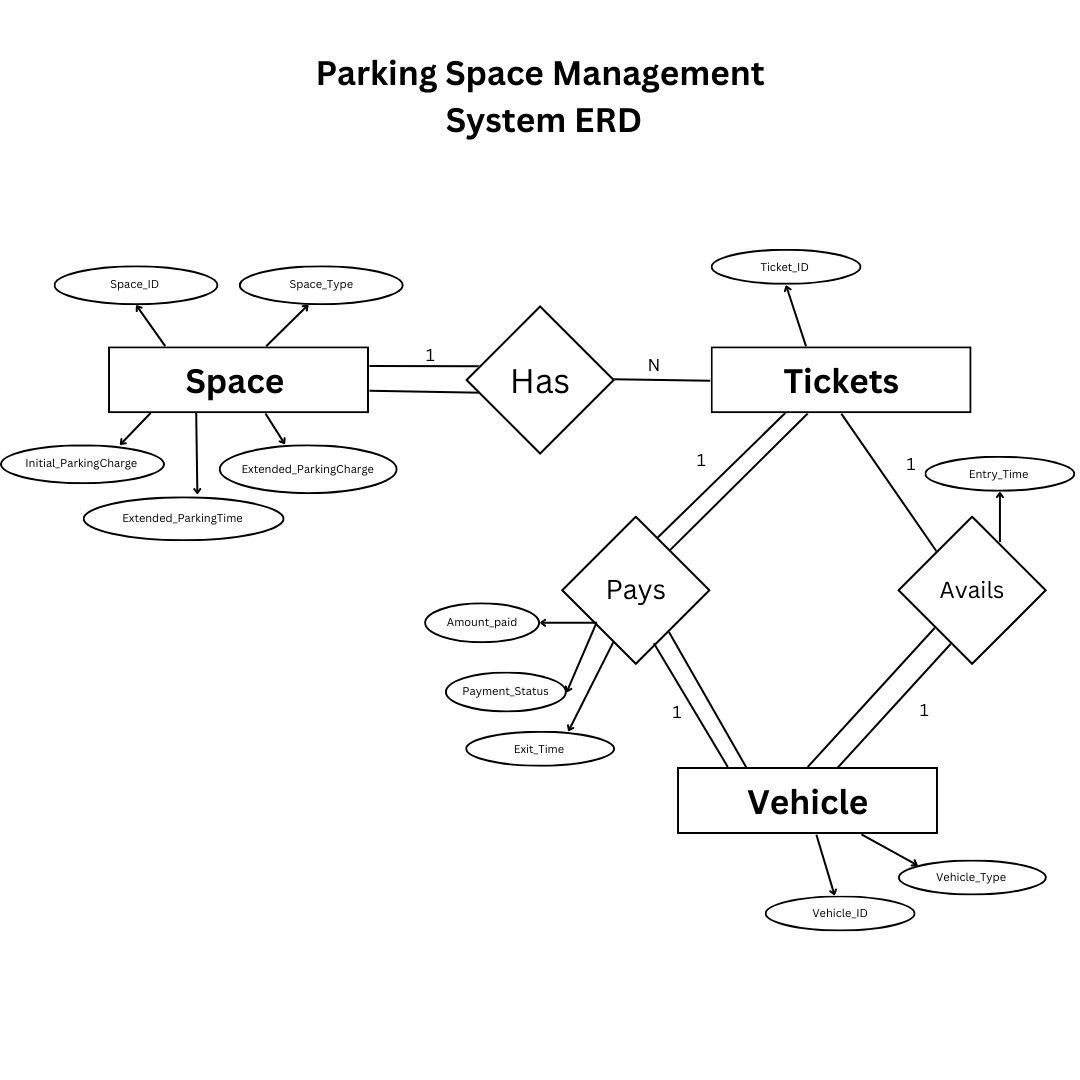
**Ticket Management** - Generating ticket for vehicles upon entry to the parking space, can record entry and exit time for each vehicle. Tracking the payment status of tickets (paid, unpaid, pending).

**Payment Management**- Process payment for parking tickets, can record payment details such as payment amount.

**Vehicle management** - Adding and managing vehicle records in the system like storing vehicle information such as Vehicle type and vehicle’s plate number which is the vehicle ID.

**Extended parking** - Calculating and applying additional charges for extended parking (updating the exit time and total charges accordingly.

1. **Entity Relationship Diagram**



**Entities**

**Parking Space**

* Represents specific parking spot in the facility

**Attributes**

* Space\_ID - identifier for each parking space
* Space\_Type - Type of the Parking spaces
* Initial\_ParkingCharge - The initial charge for parking in the space
* Extended\_ParkingCharge - Additional charge for parking beyond a certain time limit.
* Extended\_ParkingTime- The

**Tickets**

* Represents parking tickets issued to a vehicle.

**Attributes**

* Ticket\_ID - unique identifier for each parking tickets

**Vehicle**

* Represents the vehicle that can be parked in the parking spaces.

**Attributes**

* Vehicle\_ID - identifier for each vehicle
* Vehicle\_Type - Specify the type of vehicle such as 2 or 4 wheelers.

**Pays**

* Relationship between two entities Vehicle and Tickets

**Attributes**

* Amount\_Paid - Indicates the amount paid for the parking fee
* Payment\_Status - represents the status of the payment for a specific ticket and vehicle association.
* Exit\_Time - Time of exit from the parking space

**Avails**

* Relationship between two entities Vehicle and Tickets

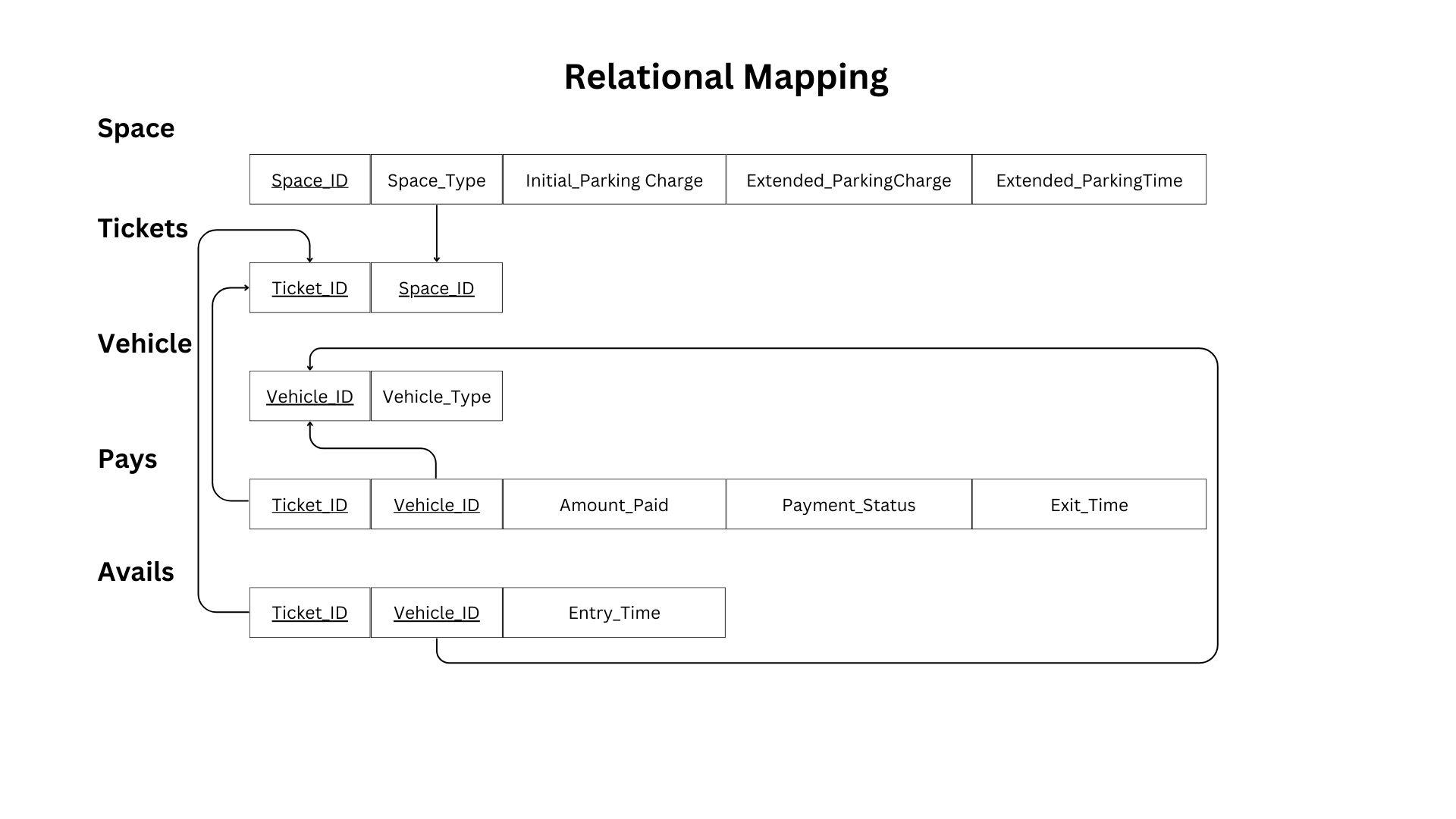
**Attributes**

* Entry\_Time - Time of entry into the parking space

**Connections between the entities**

* **Space and Tickets**
  + This connection allows a parking space to be associated with multiple tickets, indicating that multiple tickets can be linked to a single parking space. (1:N relationship).
* **Vehicle\_Pays\_Ticket** 
  + It's a relationship between tickets and a vehicle, including payment details, and Exit\_Time attribute to indicate the time which vehicle associated with the ticket exited the parking space and how much the vehicle needs to pay. (1:1 relationship)
* **Vehicle\_Avails\_Ticket** 
  + These connections indicate that a ticket can be linked to a single vehicle through both payment and availability aspects. (1:1 relationship)

1. **Relational Mapping**



1. **Data Dictionary**

| **Space** | | | | |
| --- | --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Data Format** | **Description** | **Example** |
| Space\_ID | int | NNNNNNNNNN | Identifier for each parking space | 1234567890 |
| Space\_Type | String | NNNNNN | Type of the Parking spaces | 2 Wheelers, 4 wheelers |
| Initial\_ParkingCharge | REAL | NNNNN | The initial charge for parking in the space | ₱ 100 |
| Extended\_ParkingCharge | REAL | NNNN | Additional charge for parking beyond a certain time limit. | ₱ 50 |
| Extended\_ParkingTime | Int | NN |  | 4 |

| **Ticket** | | | | |
| --- | --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Data Format** | **Description** | **Example** |
| Ticket\_ID | int | NNNN | Unique identifier for each parking tickets | 10 |

| **Vehicle** | | | | |
| --- | --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Data Format** | **Description** | **Example** |
| Vehicle\_ID | string | NNNNN | Identifier for each vehicle | HUS-275 |
| Vehicle\_Type | string | NNNNNNNN | Specify the type of vehicle | 2 wheelers, 4 wheelers |

| **Pays** | | | | |
| --- | --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Data Format** | **Description** | **Example** |
| Amount\_Paid | REAL | NNNNNNNN | Indicates the amount paid for the parking fee | ₱200.0 |
| Payment\_Status | string | NNNN | Represents the status of the payment for a specific ticket and vehicle association. | Paid, Unpaid |
| Exit\_Time | string | YYYY-MM-DD  HH:MM:SS | Date and Time of exit from the parking space | 2023-09-07  02: 00: 00 |

| **Avails** | | | | |
| --- | --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Data Format** | **Description** | **Example** |
| Entry\_Time | string | YYYY-MM-DD  HH:MM:SS | Date and Time of entry into the parking space | 2023-09-07  02: 00: 00 |

1. **SQL DDL**

CREATE TABLE Space (

Space\_ID INTEGER PRIMARY KEY,

Space\_type TEXT,

Initial\_ParkingCharge REAL,

Extended\_ParkingCharge REAL,

Extended\_ParkingTime INTEGER

);

CREATE TABLE Ticket (

Ticket\_ID INTEGER PRIMARY KEY AUTOINCREMENT,

Space\_ID INTEGER,

FOREIGN KEY (Space\_ID) REFERENCES Space(Space\_ID)

);

CREATE TABLE Vehicle (

Vehicle\_Type TEXT,

Vehicle\_ID TEXT PRIMARY KEY

);

CREATE TABLE Pays (

Ticket\_ID INTEGER,

Vehicle\_ID TEXT,

Amount\_Paid REAL,

Payment\_Status TEXT,

Exit\_Time TEXT,

PRIMARY KEY (Ticket\_ID, Vehicle\_ID),

FOREIGN KEY (TicketID) REFERENCES Ticket(TicketID),

FOREIGN KEY (Vehicle\_ID) REFERENCES Vehicle(Vehicle\_ID)

);

CREATE TABLE Avails (

Ticket\_ID TEXT,

Vehicle\_ID TEXT,

Entry\_Time TEXT,

PRIMARY KEY (Ticket\_ID, Vehicle\_ID),

FOREIGN KEY (Ticket\_ID) REFERENCES Ticket(Ticket\_ID),

FOREIGN KEY (Vehicle\_ID) REFERENCES Vehicle(Vehicle\_ID)

);