KLE Society's

KLE Technological University

# DBA Course Project Report

On

## PARKING MANAGEMENT SYSTEM

Database Applications Lab (15ECSP204)

Database Management System (15ECSC208)

Submitted By


| NAME | ROLLNO | USN |
|---|---|---|
| AKASH KURUTAGI | 202 | 01FE19BCS075 |
| SUSHEN ITAGI | 216 | 01FE19BCS089 |
| VISHAL GIRADDI | 214 | 01FE19BCS087 |
| PRAJWAL BASTI | 206 | 01FE19BCS079 |

**TEAM NUMBER**: 02(B1)

**Faculty In-charge**:

Sunita

**SCHOOL OF COMPUTER SCIENCE & ENGINEERING**

HUBLI – 580 031 (India)


Academic year 2020-21

# PARKING MANAGEMENT SYSTEM

## CONTENTS

## 1.INTRODUCTION:

Structure of parking lots:

A typical parking floor consists of one or more parking lots that are further subdivided into spots. A floor usually has a height limit that restricts certain vehicles from entering the lot so we partition the floors based on Vehicle type . Additionally, a floor contains several uniquely numbered parking slots.And on departure of the vehicle based on the entry time and the exit time the parking ticket is generated and the corresponding spot is made free.

Regular customers are usually given pass/stickers to place  on their dashboard or windshield so the parking lot management can easily determine that the customers are not in violation of any parking rules. Walk in customers have to pay for slots so they can park their vehicles in the same designated slots. Walk-in customers are given a parking slot and then assigned to the customer as the slip is generated, based on preferences they have specified

## PROBLEM STATEMENT:

• The traditional parking system is time insufficient.

• The payment process slows down the entry and exit of the other vehicles.

• Only cash payment can be used in practice.

• Parking slots can be filled prior to visiting the slot.

The parking management system is used to help managing the vehicles allocated position, avoid congestion and collect the required amount in defined manner. The system can be widely used where there are many meeting points like many shopping complexes or like megamalls

## 2.OBJECTIVES

Design a efficient semi automated parking system that:

1. Displays available parking slots in a systematic manner according to their type i.e  two wheeler slot, 4 wheeler slot

2. collects vehicle information such as vehicle type, vehicle registration number, arrival time etc

3. Ask the user about their preferred method of payment i.e cash or card

4. Calculate the amount to be paid using arrival time and departure time

5. Processes card payments

6. Reserve special slots with requisite charging system for EV’s

7.  Reserve a certain number of parking slots for shop owners to park their vehicle

8. Store user information to quickly process entry and exit of regular customer

### 2.1 DATA COLLECTION

Types of data required for parking lots to operate, we must know more about the types of people who visit parking lots.

Customers who enter parking lots belong to one of the following groups:

•A regular customer who has purchased a biweekly, monthly, or yearly pass.

•A walk-in customer who neither has a pass nor booked a slot remotely. A slot will be assigned to such a customer based on availability.

## 2.2 DATA DESIGN:

To establish efficient parking system we have to store and process the data effectively

that wolud be working with following main entities:

•Parking lot

•Customer

•Parking reservation

### Entities and the attributes

1. **FLOOR** : F_ID,NUMBER_OF_SLOTS,IS_FLOOR_FULL,IS_RESERVED

**2.PARKING_LOT** : P_ID, F_ID, IS_SLOT_AVAILABLE, IS_REENTRY_ALLOWED, IS_VALET_PARKING_AVAILABLE

3**. PARKING_SLOT** : P_ID,PS_ID,SLOT_NUMBER, IS_SLOT_FULL

4.**ELECTRICAL_PARKING_SLOT** : F_ID,CUST_ID,EV_ID,CHARGING_PREFERENCE

5**.CUSTOMER** : CUST_ID,VEHICLE_NUMBER,CONTACT_NUMBER,REGISTRATION_DATE,IS_ELECTRIC,IS_REGULAR_CUSTOMER

6**.REGULAR_CUSTOMER** : CUST_ID,REG_ID,PURCHASE_DATE,EXPIRY_DATE,COST

7. **RESERVATION** : CUST_ID, R_ID,PS_ID, BOOKIN_DATE, DURATION_IN_MINUTES

8. **PARKING_SLIP** : R_ID, S_ID, ACTUAL_ENTRY_TIME, ACTUAL_EXIT_TIME, PENALTY_TYPE, PENALTY_COST, BASIC_COST

9**. PAYMENT** : PY_ID,S_ID, CUST_ID, PER_HOUR_CHARGES, E_CHARGING_PER_HOUR, MODE, COST

### Relationship

1. FLOOR has PARKING LOT

2.PARKING_LOT has ELECTRIC_PARKING_LOT, PARKING_SLOT

3. PARKING_SLOT has RESERVATION

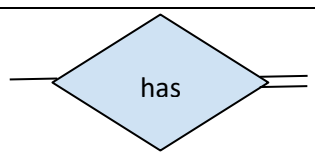
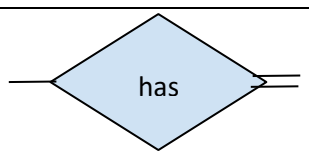
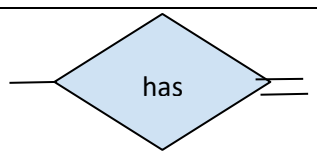
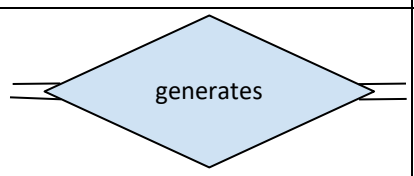
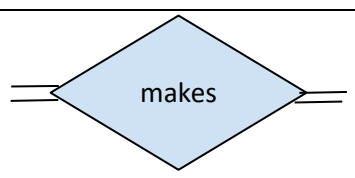
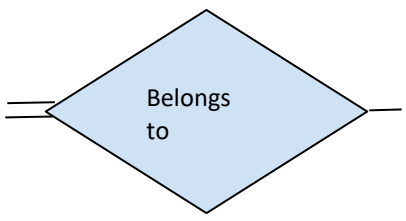
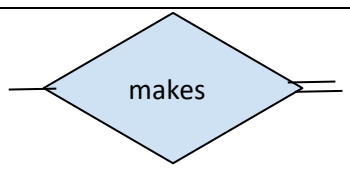
4. RESERVATION generates PARKING_SLIP

5. CUSTOMER makes RESERVATION

6.REGULAR_CUSTOMER belongs to CUSTOMER

7.CUSTOMER makes PAYMENT

## Cardinality Ratio and Participation

| SL.no. | Entity 1 | Relationship | Entity 2 | Cardinality Ratio |
|---|---|---|---|---|
| 1 | Floor<br>(0, N) |  | Parking lot<br>(1,1) | 1: N |
| 2 | Parking lot<br>(0, N) |  | EV parking slot,<br>Parking slot<br>(1,1) | 1: N<br>1: N |
| 3 | Parking slot<br>(0,1) |  | Reservation<br>(1,1) | 1:1 |
| 4 | Reservation<br>(1,1) |  | Parking slip<br>(1,1) | 1:1 |
| 5 | Customer<br>(1,1) |  | Reservation<br>(1,1) | 1:1 |
| 6 | Regular customer<br>(1,1) |  | Customer<br>(0,1) | 1:1 |
| 7 | Customer<br>(0,1) |  | Payment<br>(1,1) | 1:1 |

### 1. Floor has Parking lot

A floor can have maximum of ‘N’ parking lot and also it may not have a parking lot. So the minimum number entities from that can participate in the relation is zero and maximum number of entities can participate is ‘N’. Therefore, we get the (min, max) notation as (0, N). As 0 entities from floor can participate in this relation it is participating partially in the relationship, hence single line connecting Floor entity and the relationship.

A Parking lot is in a particular floor. Its simple to understand there will be no parking lot without a floor. The minimum and maximum entity that participate in the relationship is 1. Therefore, we get the (min, max) notation as (1, 1). As parking lot will be in a floor the participation of Parking lot is total participation, hence a double line connecting parking lot and relationship.

A Floor can have ‘N’ Parking plot. Therefore, cardinality ratio is 1: N.

### 2. Parking lot has Parking slots, EV parking slot

A Parking lot can have maximum of ‘N’ parking slot or EV parking slot and also it may not have a parking slot or EV parking slot. So, the minimum number entities from that can participate in the relation is zero and maximum number of entities can participate is ‘N’. Therefore, we get the (min, max) notation as (0, N). As 0 entities from Parking lot can participate in this relation it is participating partially in the relationship, hence single line connecting Parking lot entity and the relationship.

A Parking slot or EV Parking slot should belong to a particular floor. Its simple to understand there will be no parking slot or EV Parking slot without a Parking lot. The minimum and maximum entity that participate in the relationship is 1. Therefore, we get the (min, max) notation as (1, 1). As parking slot and Ev parking slot will be in a Parking lot the participation of both the entities Parking slot and EV parking slot is total participation, hence a double line connecting both the entities to the relationship.

A Parking lot has ‘N’ Parking slot or EV Parking slot. Therefore, cardinality ration is 1: N.

### 3. Parking Slot has reservation.

A Parking slot can have maximum of ‘1’ Reservation and also it may not be reserved. So, the minimum number entities from Parking slot that can participate in the relation is zero and maximum number of entities can participate is ‘1’. Therefore, we get the (min, max) notation as (0, 1). As 0 entities from Parking slot can participate in this relation it is participating partially in the relationship, hence single line connecting Parking slot entity and the relationship.
A Reservation will be done for particular Parking slot. Its simple to understand there will be no reservation for no Parking slot. The minimum and maximum entity that participate in the relationship is 1. Therefore, we get the (min, max) notation as (1, 1). Reservation is in total participation, hence a double line connecting both the entities to the relationship.
A parking slot will be having a only single Reservation. Therefore cardinality ratio is 1: 1

### 4. Reservation generates Parking slip

A Reservation will generate maximum of ‘1’ Parking slip. So, the minimum and maximum number entities from Reservation that can participate in the relation is 1. Therefore, we get the (min, max) notation as (1, 1). As 1 entities from Reservation will participate in this relation it is participating total in the relationship, hence double line connecting Reservation entity and the relationship.
A Parking slip will be generated for particular Reservation. Its simple to understand there will be no Parking slip generated without Reservation. The minimum and maximum entity that participate in the relationship is 1. Therefore, we get the (min, max) notation as (1, 1). Parking slip is in total participation, hence a double line connecting both the entities to the relationship.
A Reservation will generate single Parking slip. Therefore, cardinality ratio is 1: 1

### 5. Customer makes Reservation

A Customer will make a maximum of ‘1’ Reservation. So, the minimum and maximum number entities from Customer that can participate in the relation is 1. Therefore, we get the (min, max) notation as (1, 1). As 1 entities from Customer will participate in this relation it is participating total in the relationship, hence double line connecting Customer entity and the relationship.
A Reservation is done by a Customer. Its simple to understand there will be no Reservation without Customer. The minimum and maximum entity from Reservation that participate in the relationship is 1. Therefore, we get the (min, max) notation as (1, 1). Reservation is in total participation, hence a double line connecting both the entities to the relationship.
A Customer makes a single Reservation. Therefore, cardinality ratio is 1: 1.

### 6. Regular Customer belongs to Customer

A Regular Customer has to be Customer first to become regular customer. So, the minimum and maximum number entities from Regular Customer that can participate in the relation is 1. Therefore, we get the (min, max) notation as (1, 1). As 1 entity from Regular Customer will participate in this relation it is participating total in the relationship, hence double line connecting Regular Customer entity and the relationship.
A Customer may or may not be Regular Customer. The minimum entity from Customer that participate in the relationship is 0. The maximum number of entity that can participate in the relationship is 1. Therefore, we get the (min, max) notation as (0, 1). Customer is in partial participation, hence a single line connecting both the entities to the relationship.
A Regular Customer belongs to Customer. Therefore, cardinality ratio is 1: 1.

### 7. Customer makes Payment

A Customer might have parked his vehicle but not yet exited the Parking Plot. Therefore he/she might not have made the payment yet. So the minimum number of entity from Customer that can participate in the relationship is 0 and maximum number of payment can be done is 1 by a customer. Therefore, we get the (min, max) notation as (0, 1). Customer is in partial participation, hence a single line connecting both the entities to the relationship.
A payment will be done by a Customer. The minimum and maximum number of entities that can participate in the relationship is 1. Therefore, we get the (min, max) notation as (1, 1). Payment is in total participation, hence a double line connecting both the entities to the relationship.
A Customer make only one payment. Therefore, cardinality ratio is 1: 1.

## Implementation

As a part of our implementation, we are proceeding in two ways.

1. Website for the parking attendant.
2. Android app for the customers.

1. To store the required data and to access the information we are using Mysql, this is about the backend part and for front end part we are using HTML for the body and structure, CSS for styling and design and lastly Javascript for the compling the information and XAMPP server which acts as our local server. (Keywords: MYSQL, HTML, CSS, Javascript, XAMPP local server)

2. Here in our app we are using SQL lite as a part of our backend, which is used as database management system. And as a part of our front end we are using the android studio as the app development

Keyword: Java, SQL lite

## 3.REQUIREMENTS

For our Parking management system our main requirement is to deal about how to handle the payment criteria based on the type of the vehicle entering , Should be alloted to the specific floor and spot.

### 3.1 DATA REQUIREMENTS:

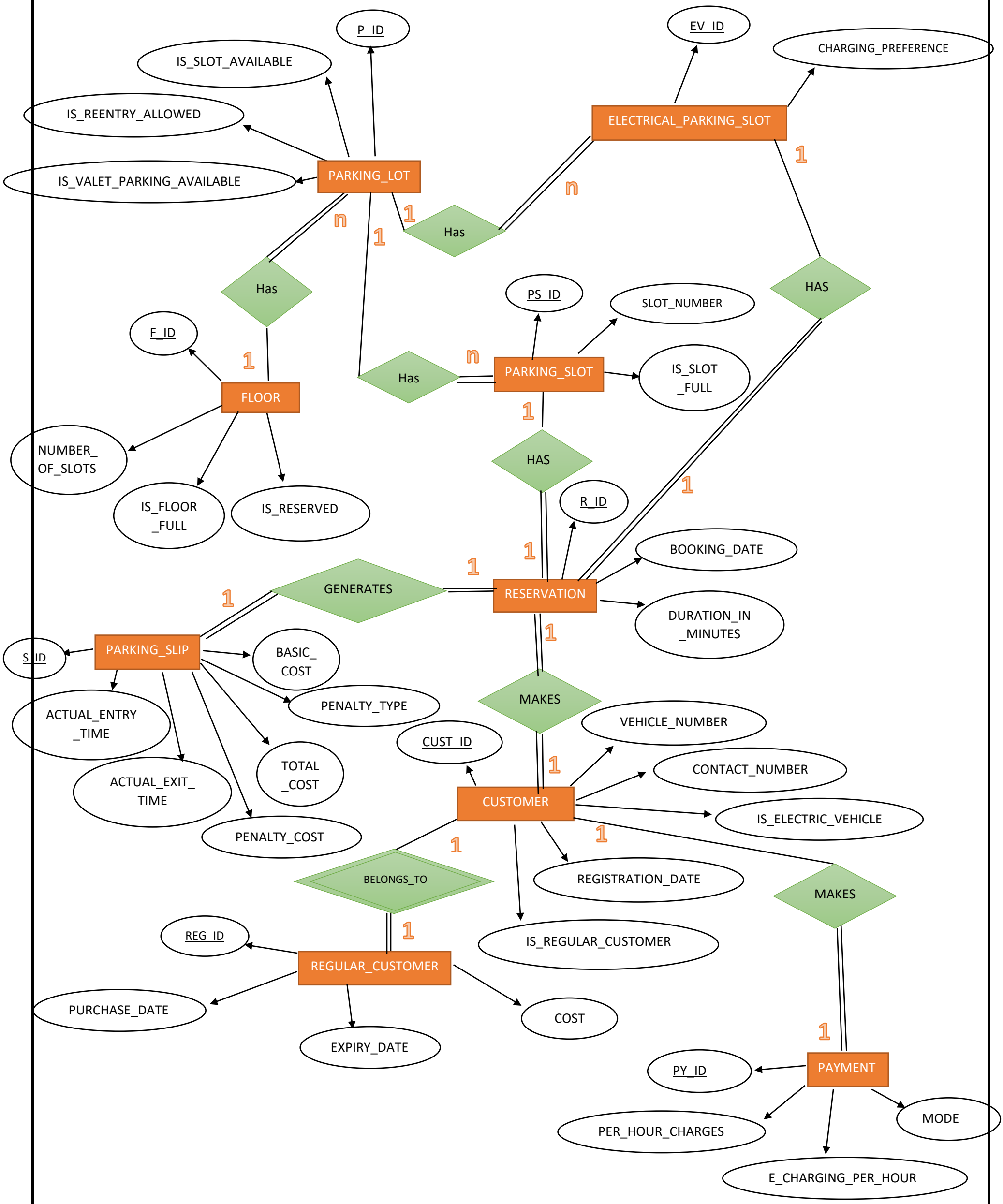
* Is the vehicle a reserved vehicle or the walk-in vehicle.
* Is the vehicle an electric vehicle, if yes, is charging point preferred or not.
* For regular customers the parking pass start date and the expiry date is checked.
* And for normal customers the vehicle number, phone number is noted.
* And on successful reservation the date is noted and the entry time and the exit time is noted.

### 3.2 FUNCTIONAL REQUIREMENTS:

* When the vehicle enters the parking lot the reservation is checked, if reserved the already allotted spot is allotted,
if not, the availability is checked based on type of vehicle.
* If available the corresponding slot in the particular floor is allotted.
* And on exit of the vehicle the parking slip is generated stating the entry time and the exit time and other details.
* And based on entry time and exit time and charges are calculated
* And to this calculated amount the penalty, electric charging rate is added if applies.
* And on successful payment by their preferred payment mode, the spot is made available for next Parking

# 4.DESIGN PHASE

## 4.1 ER-MODEL



## 4.2 RELATIONAL SCHEMA

FLOOR

| F_ID | NUMBER_OF_SLOTS | IS_FLOOR_FULL | IS_RESERVED |
|---|---|---|---|

PARKING_LOT

| P_ID | F_ID | IS_SLOT_AVAILABLE | IS_REENTRY_ALLOWED | IS_VALET_PARKING_AVAILABLE |
|---|---|---|---|---|

PARKING_SLOT

| P_ID | PS_ID | SLOT_NUMBER | IS_SLOT_FULL |
|---|---|---|---|

CUSTOMER

| CUST_ID | PS_ID | VEHICLE_NUMBER | CONTACT_NUMBER | REGISTRATION_DATE | IS_VEHICLE_ELECTRIC | IS_REGULAR_CUSTOMER |
|---|---|---|---|---|---|---|

ELETRICAL_PARKING_SLOT

| EV_ID | CUST_ID | P_ID | CHARGING_PREFERENCE |
|---|---|---|---|

REGULAR_CUSTOMER

| CUST_ID | REG_ID | PURCHASE_DATE | EXPIRY_DATE | COST |
|---|---|---|---|---|

RESERVATION

| CUST_ID | R_ID | PS_ID | BOOKIN_DATE | DURATION_IN_MINUTES |
|---|---|---|---|---|

PARKING_SLIP

| R_ID | S_ID | ACTUAL_ENTRY_TIME | ACTUAL_EXIT_TIME | PENALTY_TYPE | PENALTY_COST | BASIC_COST |
|---|---|---|---|---|---|---|

PAYMENT

| PY_ID | S_ID | PER_HOUR_CHARGES | E_CHARGING_PER_HOUR | MODE | COST | CUST_ID |
|---|---|---|---|---|---|---|

## 5.NORMALIZATION

1. PARKING_LOT, FLOOR, PARKING_SLOT, CUSTOMER, ELECTRICAL_PARKING_SLOT, REGULAR_CUSTOMER, RESERVATION, PARKING_SLIP, PAYMENT.

Since above mentioned tables doesn't contain any multi valued attributes it's in 1NF.

2. PARKING_SLOT

| P_ID | PS_ID | SLOT_NUMBER | IS_SLOT_FULL |
|---|---|---|---|

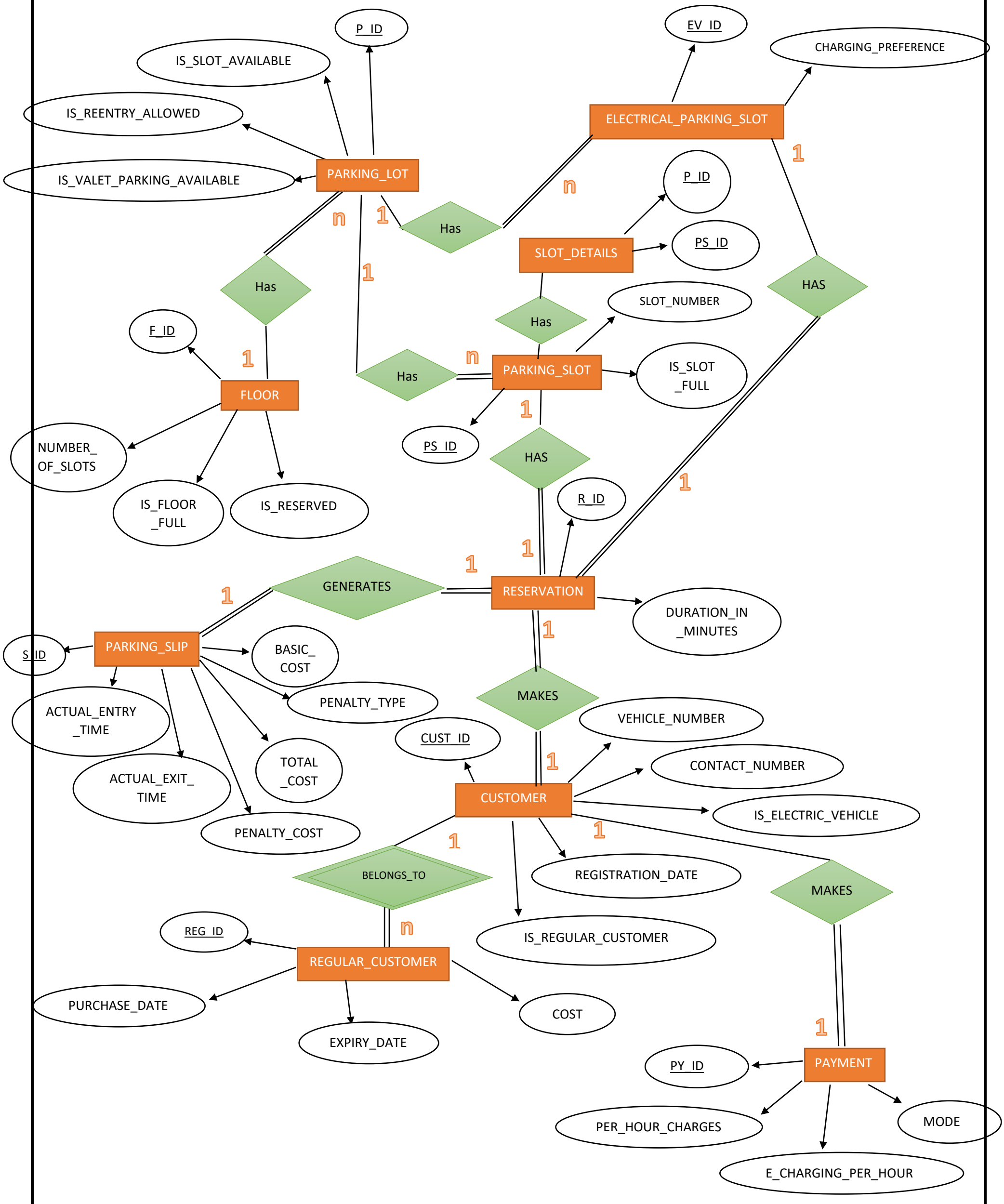
The table PARKING_SLOT we have multi-key attributes where non-prime key attribute depends on those key attributes.

This entity violates 2NF. Therefore, we have to consider two different tables.

1. PARKING_SLOT {PS_ID, SLOT_NUMBER, IS_SLOT _FULL}
2. SLOT_DETAILS {P_ID, PS_ID}

3. Schema is already in 3NF because, only key is determining the other attributes. And there is no transitive relationship between the attributes. Hence the schema remains same after normalization.

## 5.1 ER-MODEL AFTER NORMLISATION



## 5.2 RELATIONAL SCHEMA AFTER NORMALIZATION

FLOOR

| F_ID | NUMBER_OF_SLOTS | IS_FLOOR_FULL | IS_RESERVED |
|---|---|---|---|

PARKING_LOT

| P_ID | F_ID | IS_SLOT_AVAILABLE | IS_REENTRY_ALLOWED | IS_VALET_PARKING_AVAILABLE |
|---|---|---|---|---|

PARKING_SLOT

| P_ID | PS_ID | SLOT_NUMBER | IS_SLOT_FULL |
|---|---|---|---|

SLOT_DETAILS

| P_ID | PS_ID |
|---|---|

CUSTOMER

| CUST_ID | PS_ID | VEHICLE_NUMBER | CONTACT_NUMBER | REGISTRATION_DATE | IS_VEHICLE_ELECTRIC | IS_REGULAR_CUSTOMER |
|---|---|---|---|---|---|---|

ELETRICAL_PARKING_SLOT

| EV_ID | CUST_ID | P_ID | CHARGING_PREFERENCE |
|---|---|---|---|

REGULAR_CUSTOMER

| CUST_ID | REG_ID | PURCHASE_DATE | EXPIRY_DATE | COST |
|---|---|---|---|---|

RESERVATION

| CUST_ID | R_ID | PS_ID | BOOKIN_DATE | DURATION_IN_MINUTES |
|---|---|---|---|---|

PARKING_SLIP

| R_ID | S_ID | ACTUAL_ENTRY_TIME | ACTUAL_EXIT_TIME | PENALTY_TYPE | PENALTY_COST | BASIC_COST |
|---|---|---|---|---|---|---|

PAYMENT

| PY_ID | S_ID | PER_HOUR_CHARGES | E_CHARGING_PER_HOUR | MODE | COST | CUST_ID |
|---|---|---|---|---|---|---|

# THANK YOU