



Smart City Car Parking System

BD2 Project - Stage 0

20L

10.04.2020

Michał Smutkiewicz (283538)

Khanh Do Van (308946)

Przemek Kacprowicz (267976)

Artur Nasiadko (308805)

0. Contents

1. Project Overview
2. Conceptual Model - ER Diagram
3. Conceptual Model - Description

1. Project Overview

Main aim

The aim of the project is to make a system which will **make it easier to park in the city** by eliminating the need to manually get a ticket from a parking meter. What is more, our system will take advantage of the database containing up-to-date information about current state of local parking lots to save drivers' time wasted every day on seeking parking spots.

Detecting free spots

To achieve this, we'll need to provide mechanisms to detect free spots on our parking lots. Every parking lot will have **smart cameras**. When a car will park at a parking spot within the observation range of one of them, this parking spot will be reported as occupied and other drivers looking for a free parking spot will see an updated amount of free spots in their mobile applications.

Smart Parking Meters and ETickets

Every parking lot will have several **smart parking meters** with NFC tags. Scanning a tag with a mobile application and then registering for parking will replace normal ticket printing in parking meters, making the process quicker and more friendly for the environment. When registering on mobile, users won't need a printed ticket, as he will have it online as our system's **EParkingTicket** - online representation of standard tickets.

When a car leaves the area the time it spent on the parking lot is counted and the owner of the car is billed for an appropriate amount. Parking lots are assigned to zones which define price information. The system won't track which individual spots are taken, **only how many spots are occupied per parking lot**. Also, users won't have ability to book particular spots on parking lot, as this won't be system's responsibility to take care of reservations.

Users from outside the system

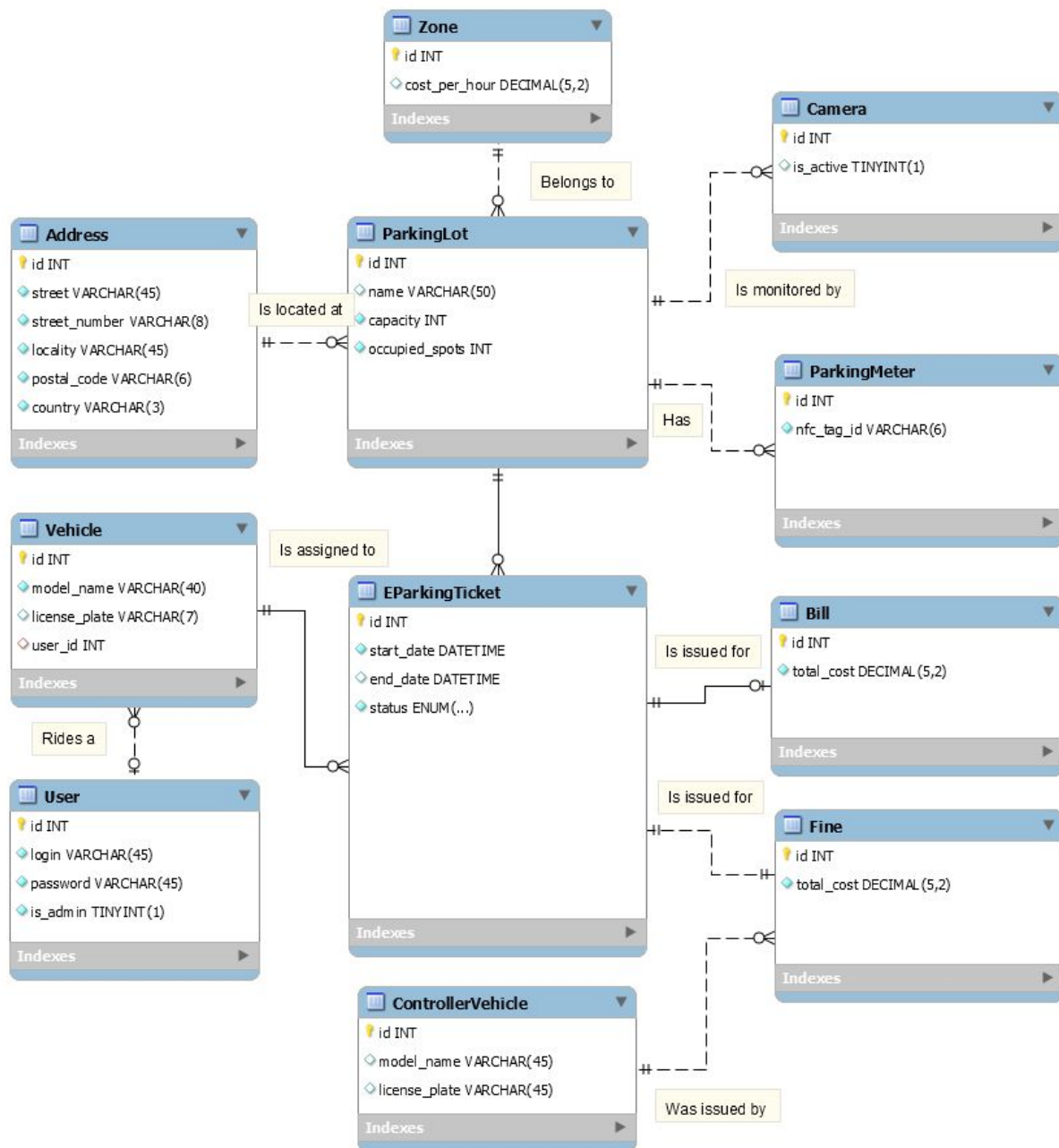
In case an user isn't a client of our system or doesn't use the app, **parking meters will provide the possibility to prepay the tickets - pay directly in parking meter**. In that case, the user registers vehicle manually in the parking meter, pre pays for fixed

amount of time and gets a printed ticket. The only difference between registering for parking online and manually in a parking meter is that the user won't be able to check his EParkingTicket and pay by our app. However, the system should provide him with the possibility to check his history of payments for his vehicle once he signs up to the system in the future.

Dealing with illegally parked cars

Special cameras placed on controlling cars will scan license plates of parked cars and then automatically create a ticket fine - regular ETicket and special Fine objects in system database for a fixed amount of money. Procedure of dealing with unpaid fines later will be a task of local Municipal Police (information via post for example). All further procedures after issuing the bill will be sorted out in classic, "non-smart" way outside of our system.


2. Conceptual Model - ER Diagram



3. Conceptual Model - Description


ParkingLot

Represents a large area where drivers can park their vehicles for a period of time.

Attribute name	Domain	Required	Description
 id	Int	Yes	Identification of the parking lot.
name	Varchar(50)	No	Additional field, representing parking lot name if there is one.
capacity	Int	Yes	Maximum number of spots can serves.
occupied_spots	Int	Yes	Number of spots are taken currently.


EParkingTicket

Represents virtual reservation - a kind of ticket from classic parking meter. It is always created whenever a driver wants to start counting the period of parking. This is also the entity on which future bills are based on.

Attribute name	Domain	Required	Description
 id	Int	Yes	Identification of the EParkingTicket.
start_date	datetime	Yes	The time when a car starts being parked.
end_date	datetime	No	The time when a car leaves.
status	Status { 'STARTED', 'ENDED', 'PAID', 'CANCELLED' }	Yes	Indicates the status of EParkingTicket (started, ended, paid, cancelled).


Camera

Every parking lot can have several smart cameras that can detect occupation of parking spots and signal it to the main system.

Attribute name	Domain	Required	Description
 id	Int	Yes	Identification of the camera.
is_active	Tinyint(1)	Yes	The status of the camera (working, not working).


Zone

Zone represents the pay rate in the particular area. There may be several parking lots in one Zone.

Attribute name	Domain	Required	Description
 id	Int	Yes	Identification of the zone.
cost_per_hour	Decimal(5,2)	Yes	The amount of money paid per hour.


ParkingMeter

Represents a small post or plate with NFC tag which can be scanned to occupy a spot on the parking lot.

Attribute name	Domain	Required	Description
 id	Int	Yes	Identification of the parking meter
nfc_tag_ID	Varchar(6)	Yes	A tag of the parking meter

Bill


Represents payment for only one specific EParkingTicket.

Attribute name	Domain	Required	Description
 id	Int	Yes	Identification of the bill.

total_cost	Decimal(5,2)	Yes	Total amount of the bill.
------------	--------------	-----	---------------------------


User

Represents a user who might own car(s) on the system. He/she is supposed to make use of the parking service of the system.

Attribute name	Domain	Required	Description
 id	Int	Yes	Identification of the user.
login	Varchar(45)	Yes	The login name of the user for logging into the system.
password	Varchar(45)	Yes	The hashed string of the password of the user account.
is_admin	Tinyint(1)	Yes	Whether or not the user is a system administrator.


Vehicle

Represents a vehicle on the system. Users can register zero or many vehicles.

Attribute name	Domain	Required	Description
 id	Int	Yes	Identification of the vehicle.
model_name	Varchar(40)	Yes	The model name of the vehicle.
license_plate	Varchar(7)	Yes	The license plate numbers of the vehicle.

Address


Representation of addresses where all parking lots are located in. These addresses recorded in this table are kind of conceptual addresses for parking lots.

Attribute name	Domain	Required	Description
 id	Int	Yes	Identification of the Address.
street	Varchar(45)	Yes	The name of street.
street_number	Varchar(8)	Yes	The street number.
locality	Varchar(45)	Yes	The locality where the address belongs to.

postal_code	Varchar(6)	Yes	The postal code of this address.
country	Varchar(3)	Yes	ISO country code of this address.


Fine

Represents an issued fine for unauthorized use of the parking spot. For each EParkingTicket can be created, at most, one Fine.

Attribute name	Domain	Required	Description
 id	Int	Yes	Identification of the vehicle.
total_cost	Decimal(5,2)	Yes	The amount fines the driver

ControllerVehicle

Represents a controller vehicle of the system. We use the controller vehicles for monitoring unauthorized use of parking slots, issuing a fine for the appropriate driver.

Attribute name	Domain	Required	Description
 id	Int	Yes	Identification of the controller vehicle.
model_name	Varchar(40)	Yes	The model name of the controller vehicle.
license_plate	Varchar(7)	Yes	The license plate numbers of the controller vehicle.