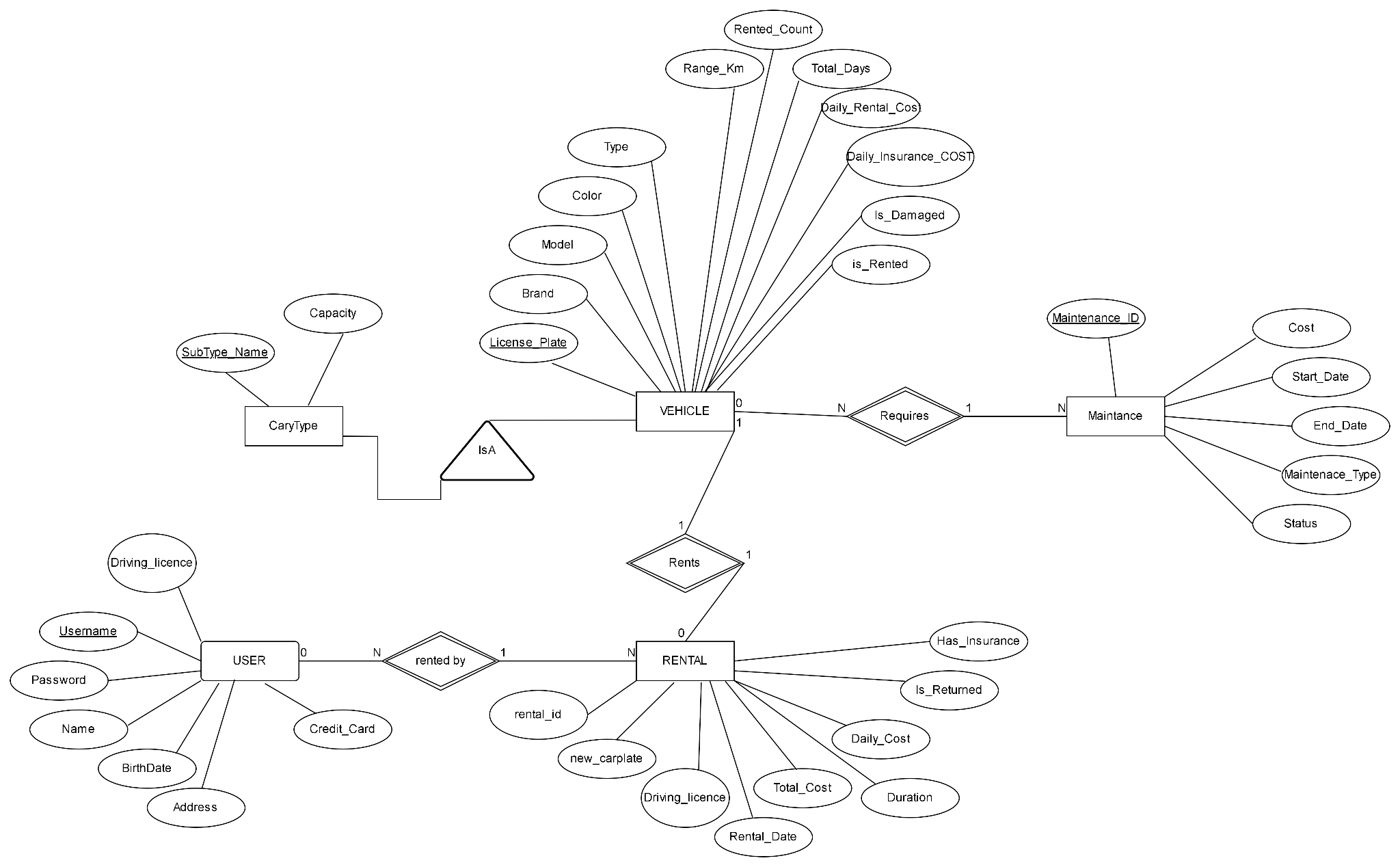
**Αναφορα 360 ομαδα 18**

**csd5010,csd4871,csd4732**

Technologies**:**

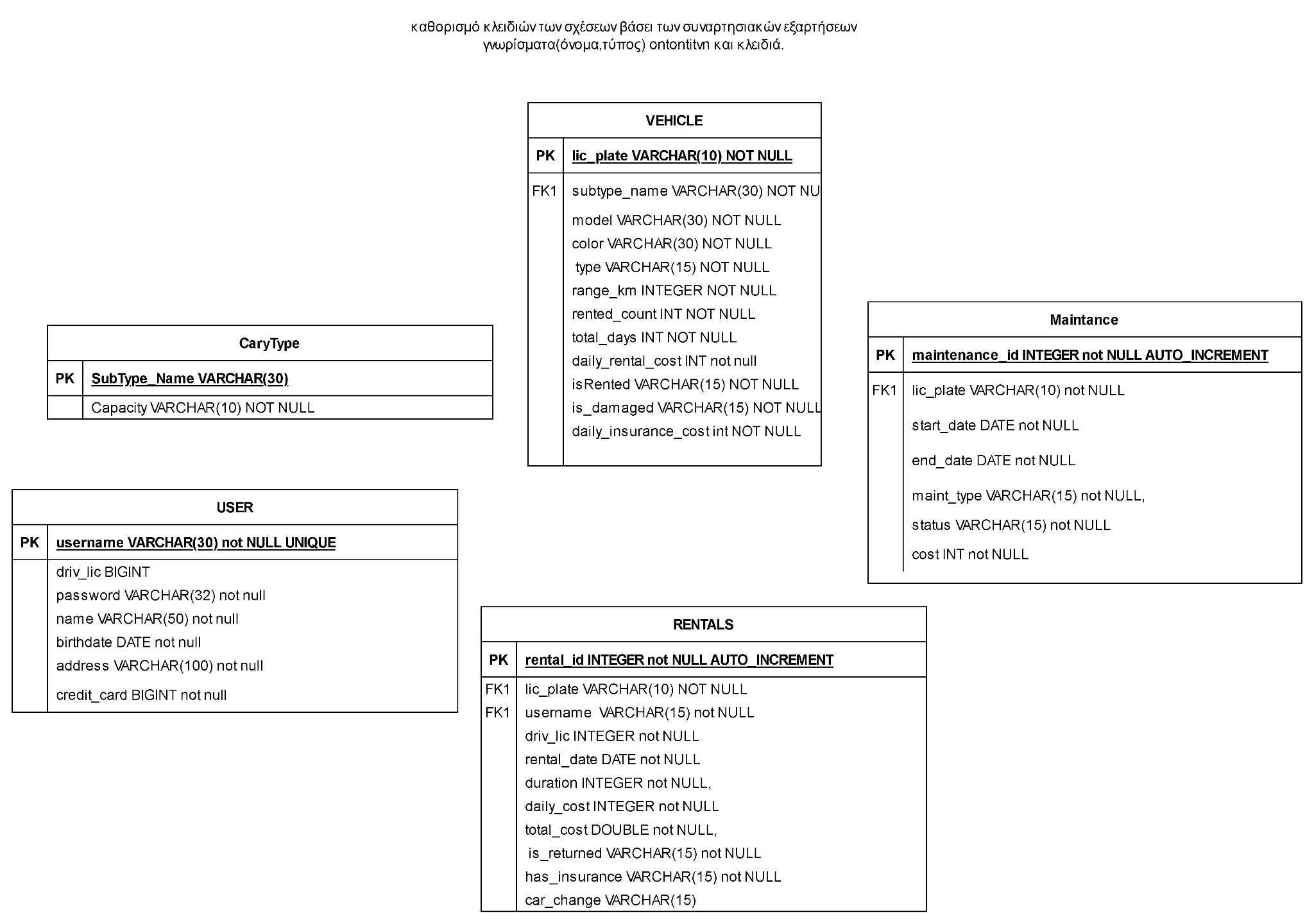
Jdk 11.0.21 Tomcat 9.0.83 IDE:netbeans 20

**1. Διάγραμμα Οντοτήτων-Σχέσεων (Entity-Relationship Diagram - ERD**

****

**2.Γνωρίσματα Οντοτήτων και Σχέσεων/Μετατροπή σε Σχεσιακό Μοντέλο**

σημείωση:PK(primary key),FK(foreign key)

****

**3. Κανονικοποίηση 3nf**

VEHICLE(lic\_plate, subtype\_name, model, color, type, range\_km, rented\_count, total\_days, daily\_rental\_cost, isRented, is\_damaged, daily\_insurance\_cost)

MAINTANCE(maintance\_id, lic\_plate, start\_date, end\_date, maint\_type, status, cost)

RENTALS(rental\_id, lic\_plate, username, driv\_lic, rental\_date, duration, daily\_cost, total\_cost, is\_returned, has\_insurance, car\_change)

USER(username, lic\_plate, driv\_lic, password, name, birthdate, address, credit\_card)

CARTYPE(SubType\_Name, capacity)

Each of the above relations have just one candidate key those are:

* lic\_plate for VEHICLE
* Maintenance\_id for MAINTANCE
* rental\_id for RENTALS
* username for USER
* SubType\_Name for CARTYPE

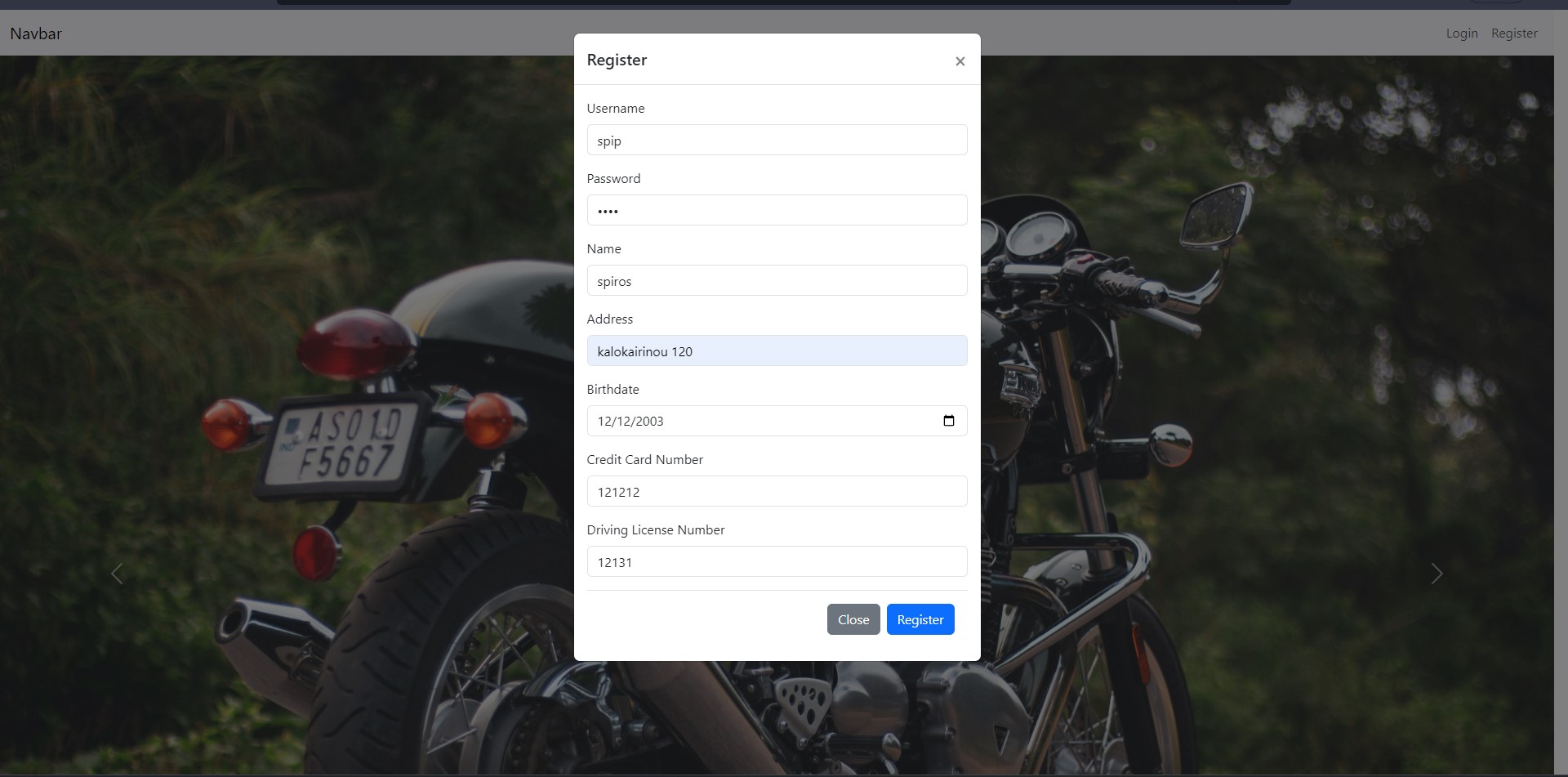
There are no relations with one or more candidate keys, so there is no need to remove any attributes.

There are no transitive dependencies (because of the implementation) in relations, so yet again there is no need to remove any attributes.

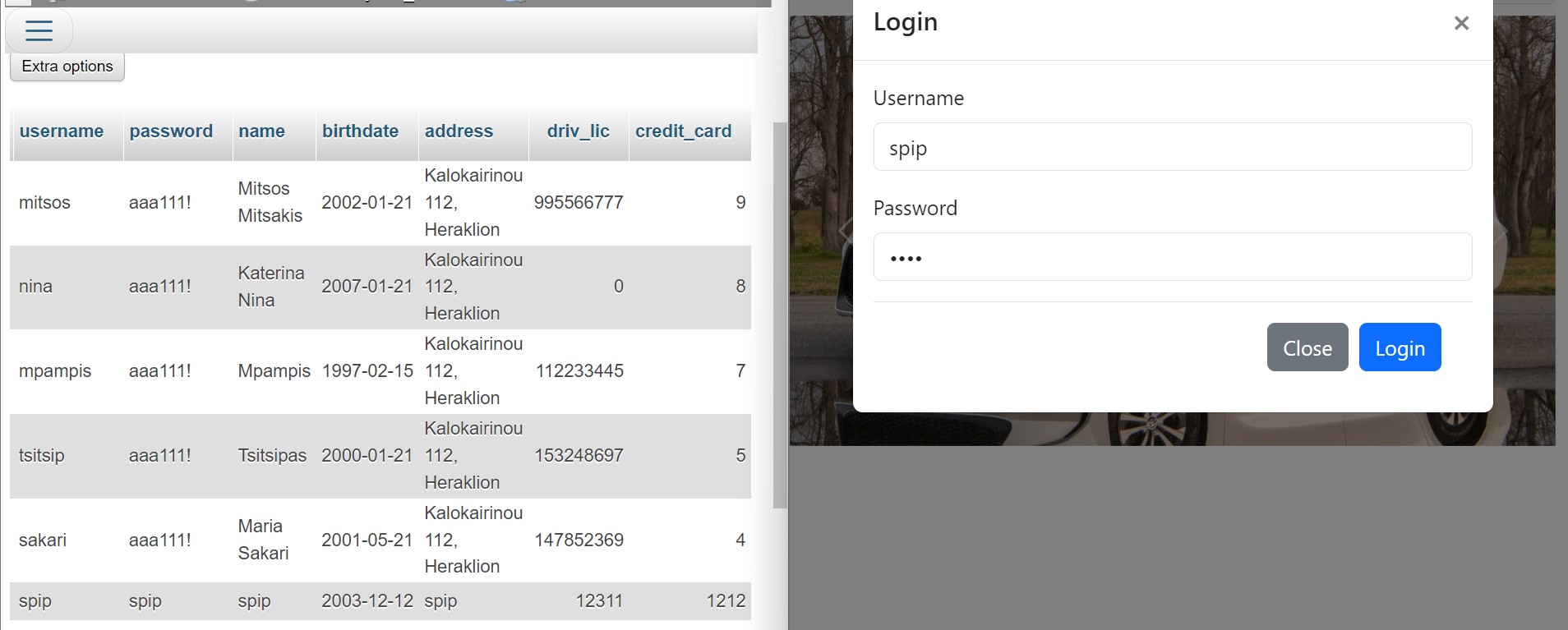
Therefore the schema is already in 3NF

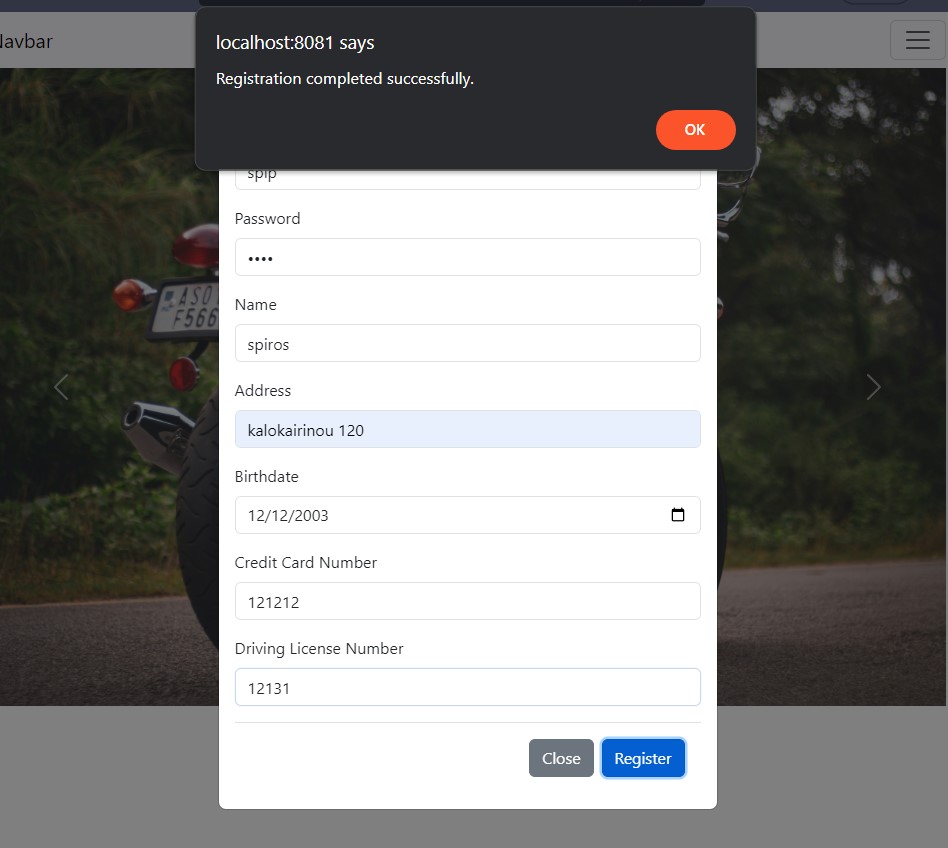
**4. Συμπεριφορα screenshots**

Register

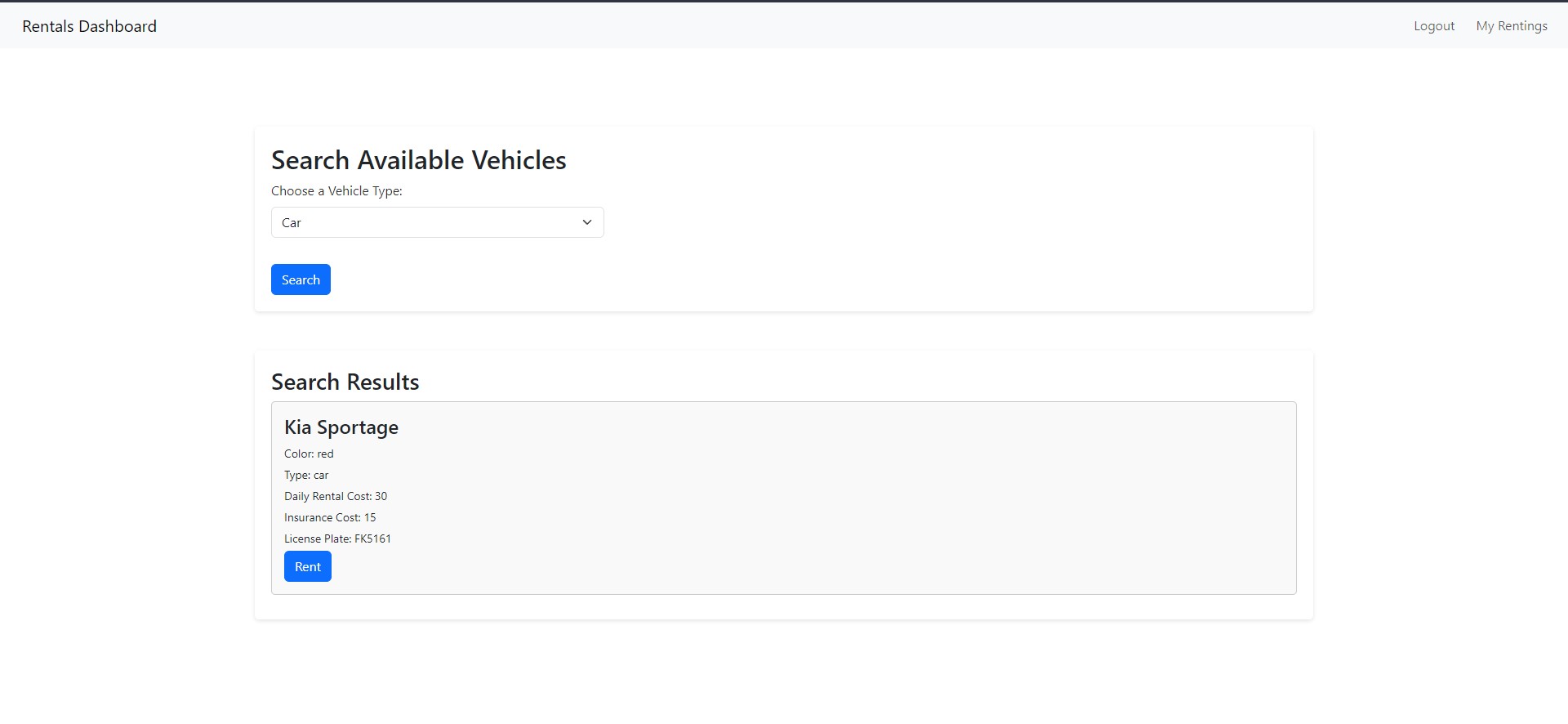
****

Register complete message + database

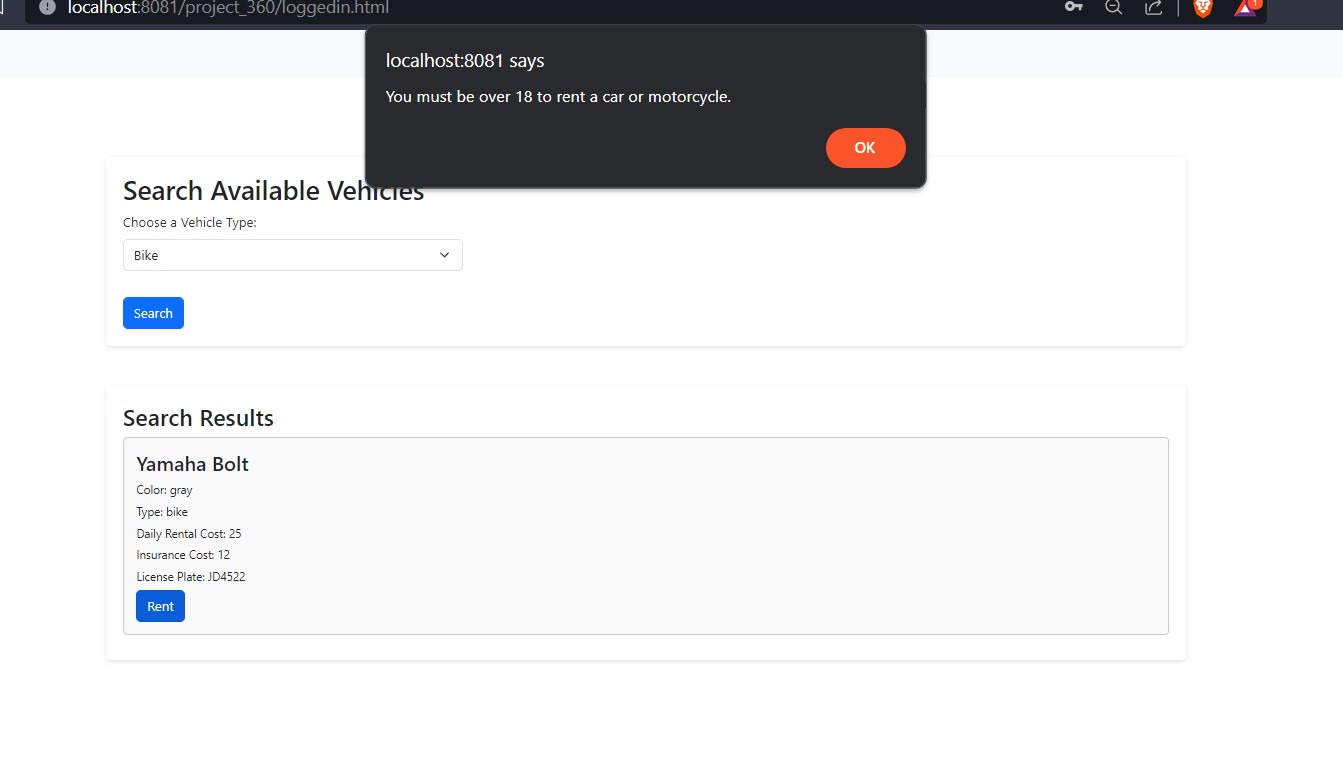
****

****

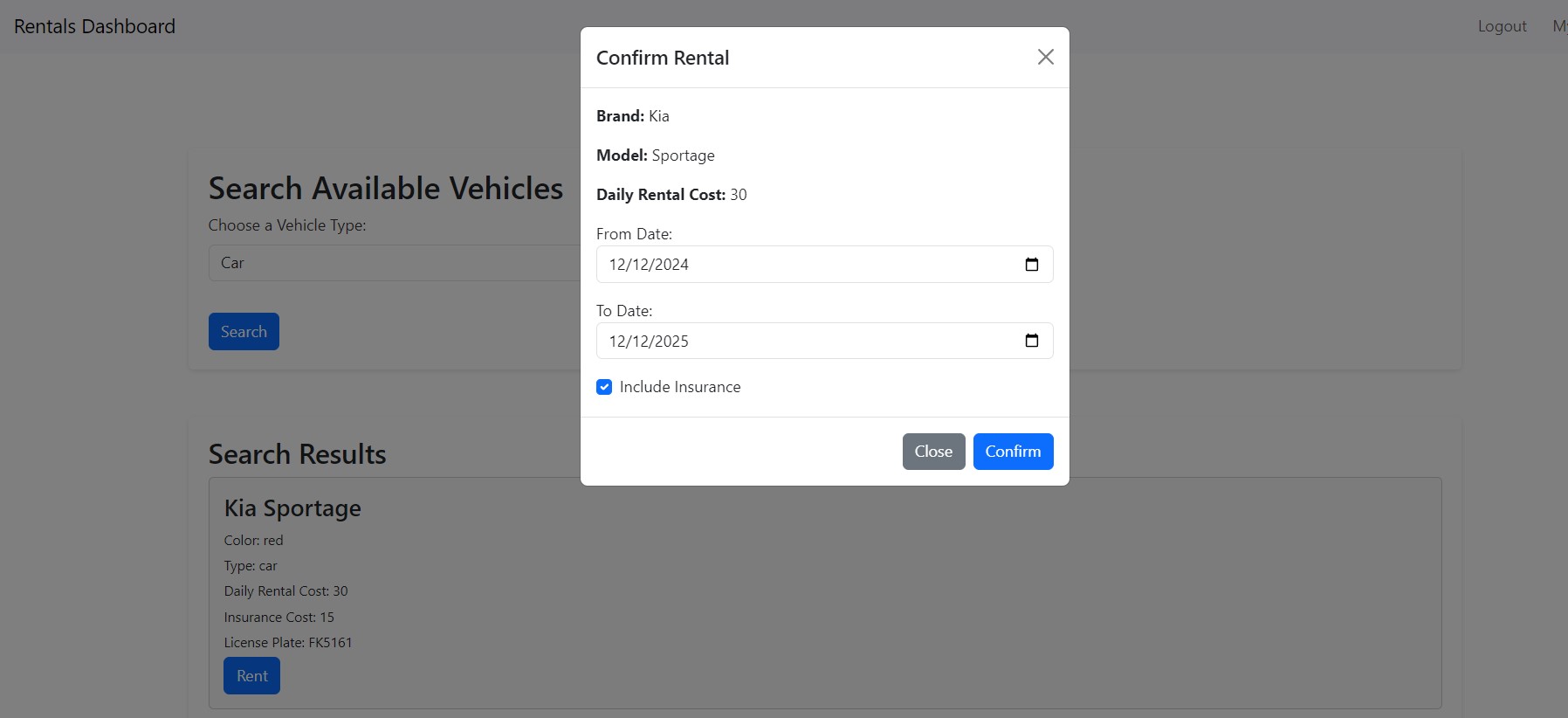
Loggedin User interface

****

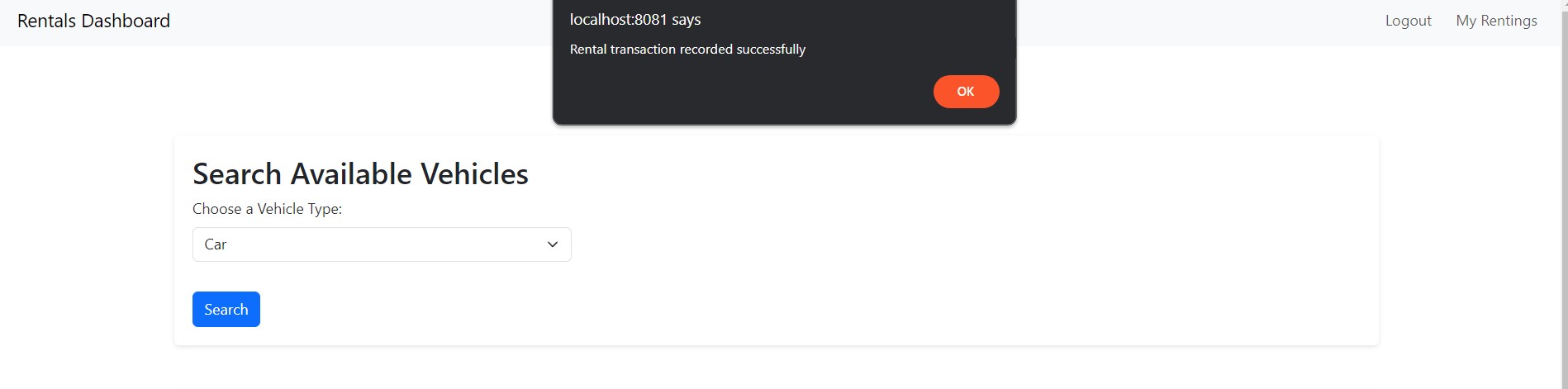
Underage Renting(Other user):



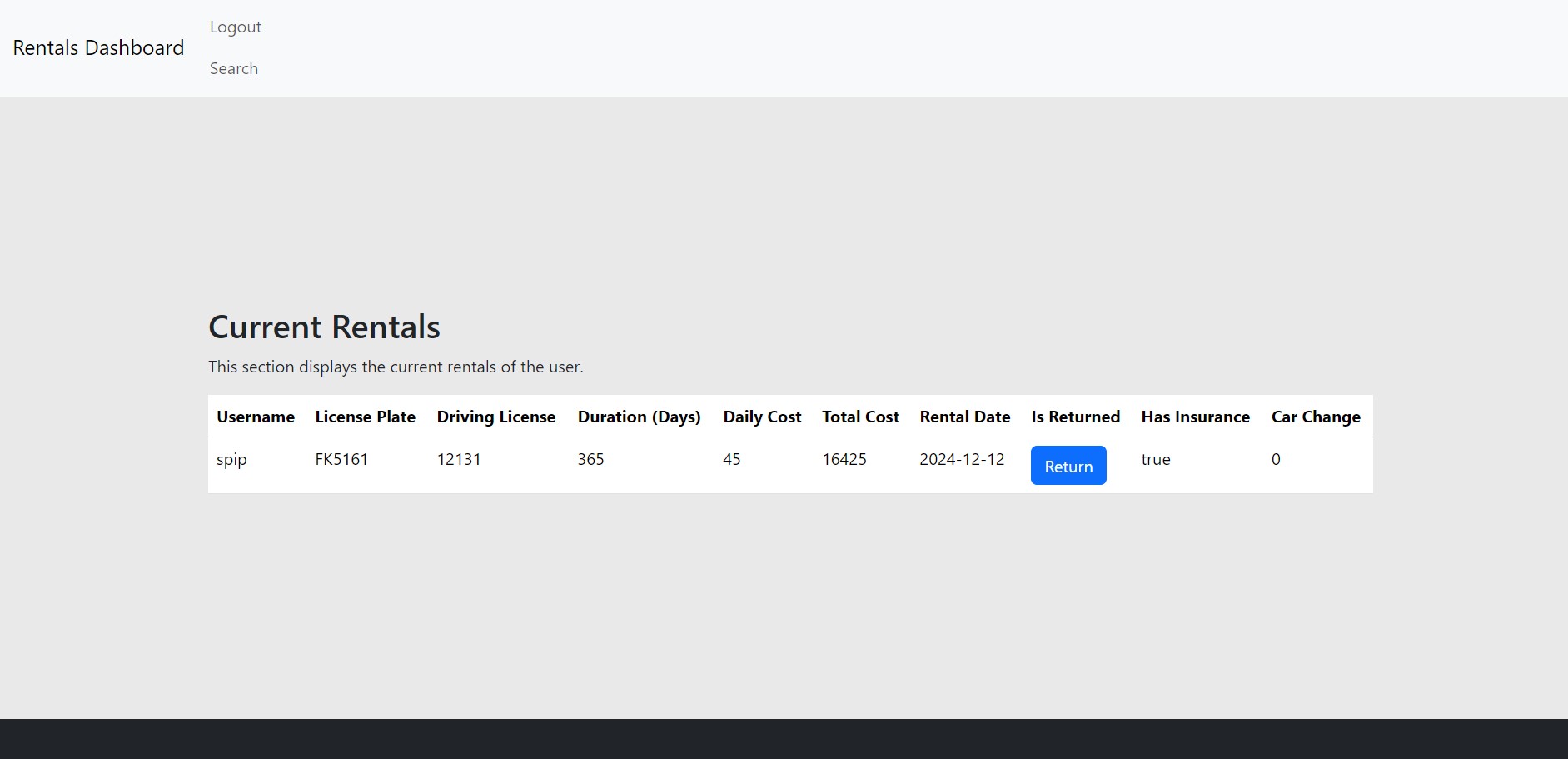
Make Renting (with over 18 account and has licence):

****

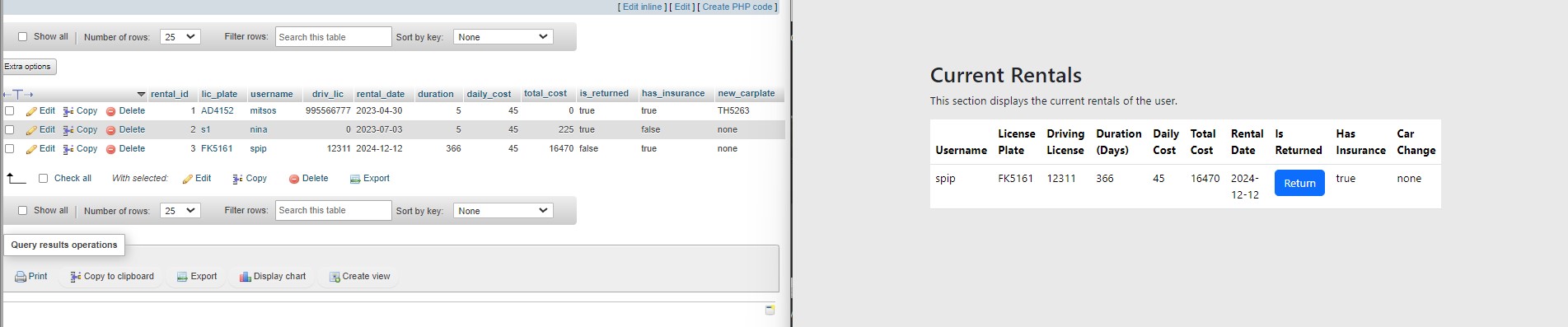
Made Renting:



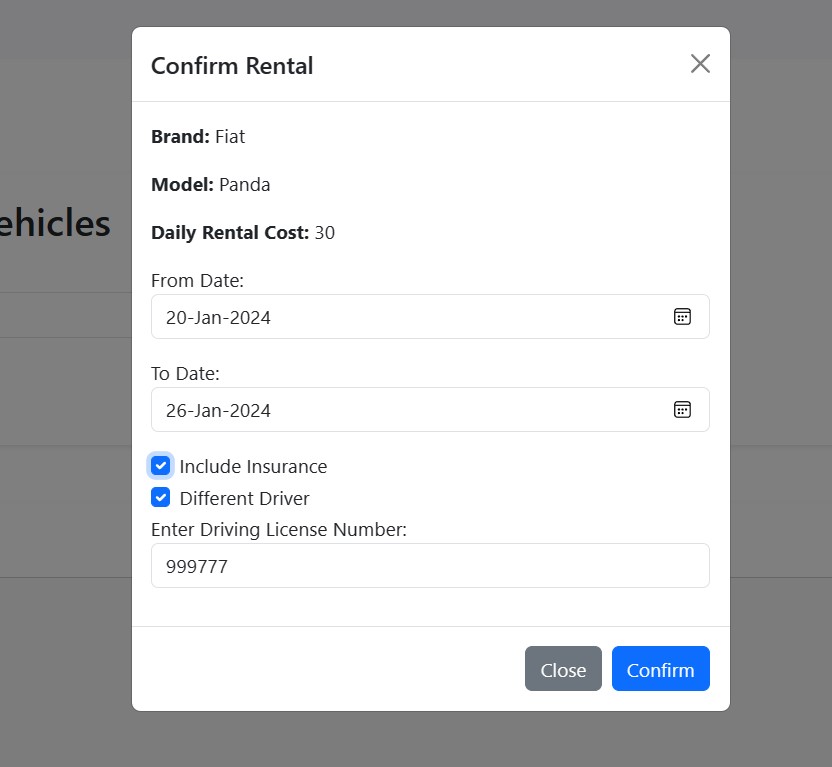
My Renting(top right in navbar)



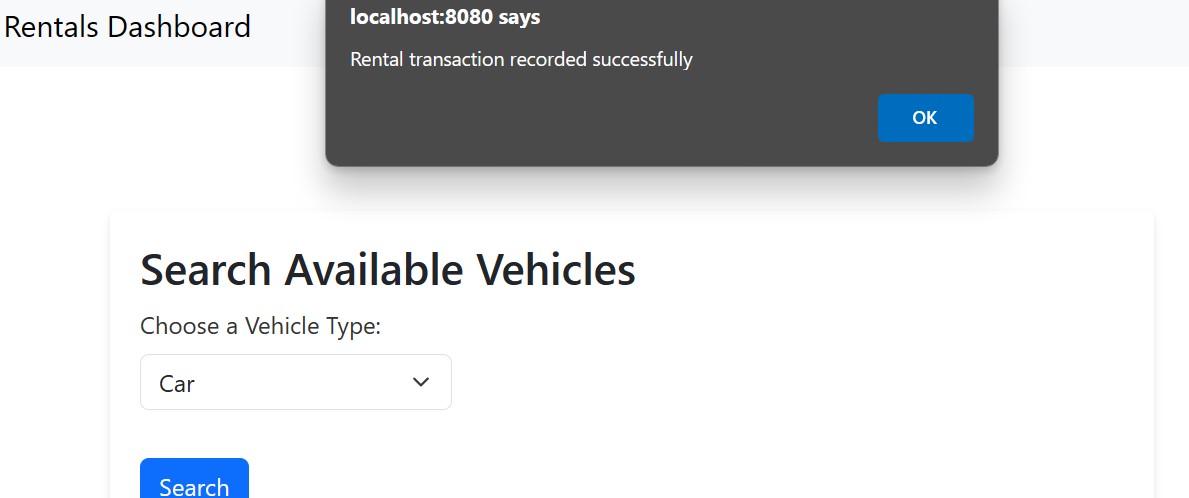
Rentals Table update:

****

Make Renting with different Driver:



Made the renting:



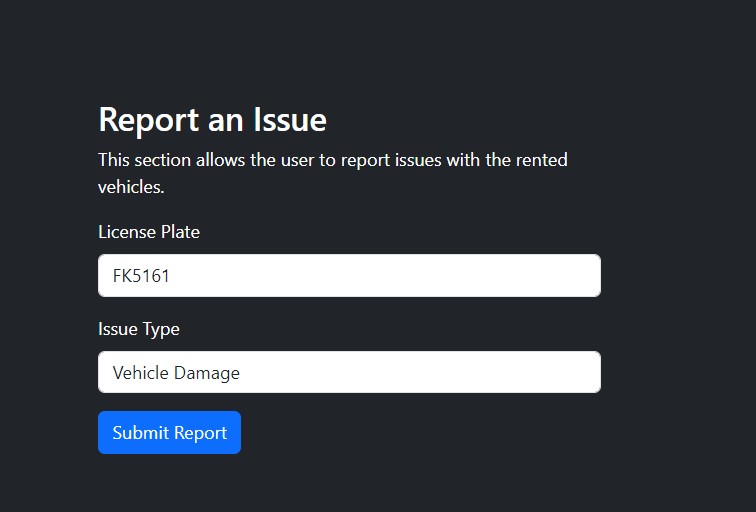
Rental Table before:

****

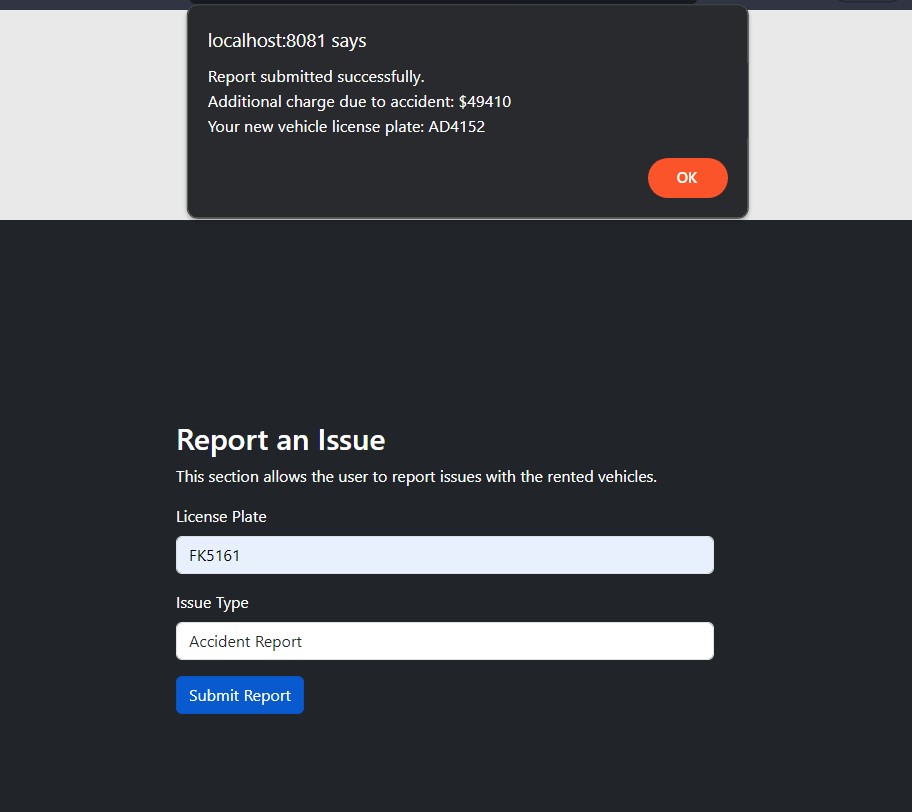
Rental Table after:



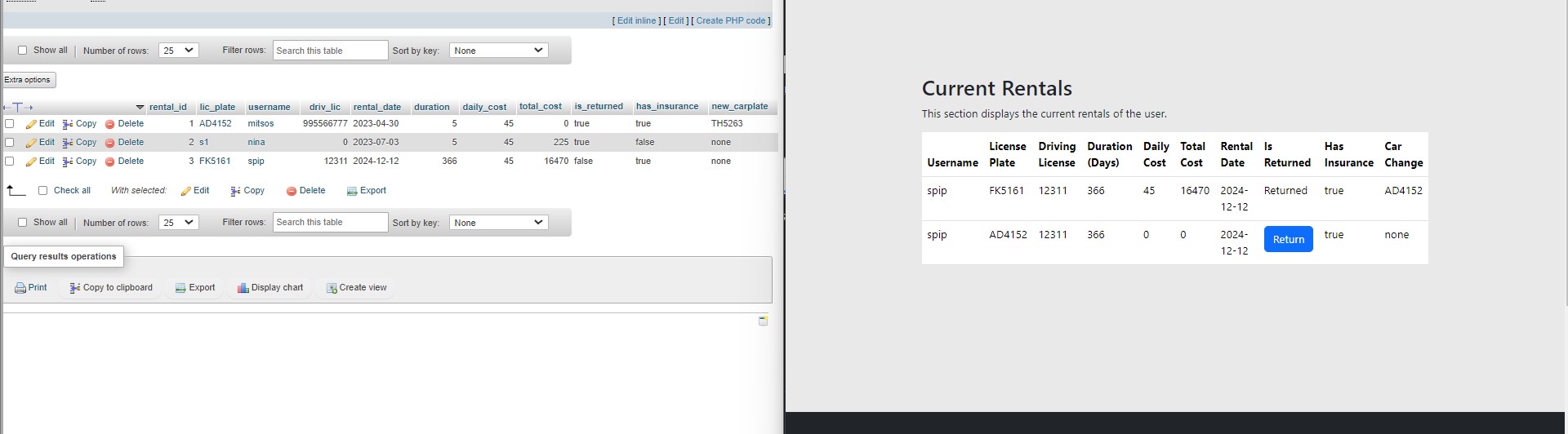
Report Issue UI:

****

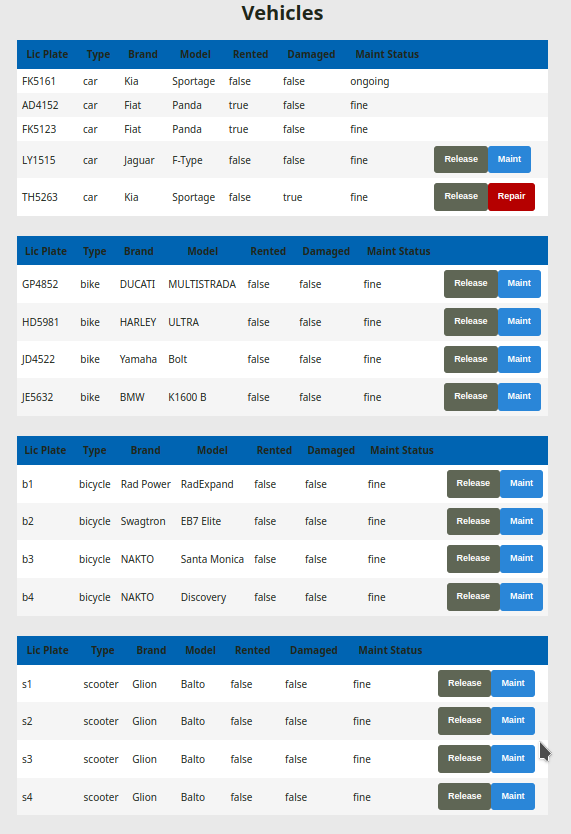
Report problem without insurance:

****

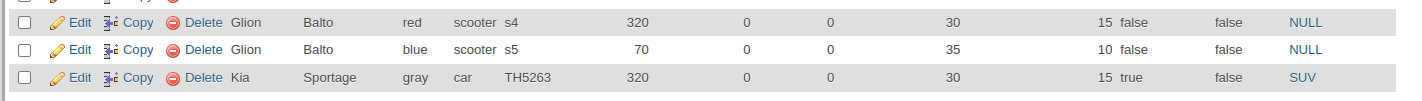
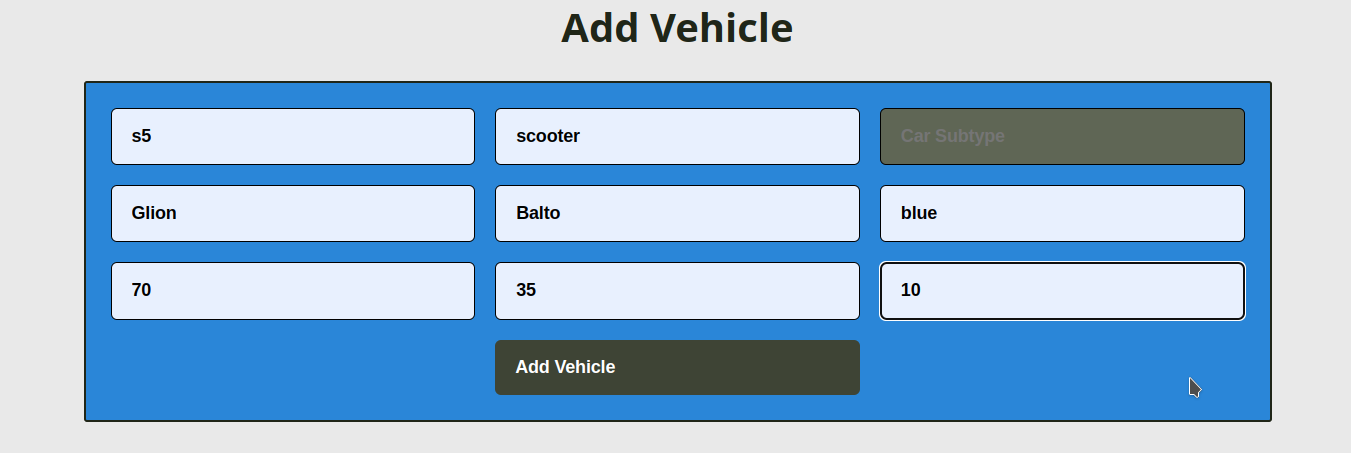
Rentals table after(explaining: in our logic we put 3\*times pay in the first original car and we do not count in in the second one so we do not duplicate pay and change actual statistics)



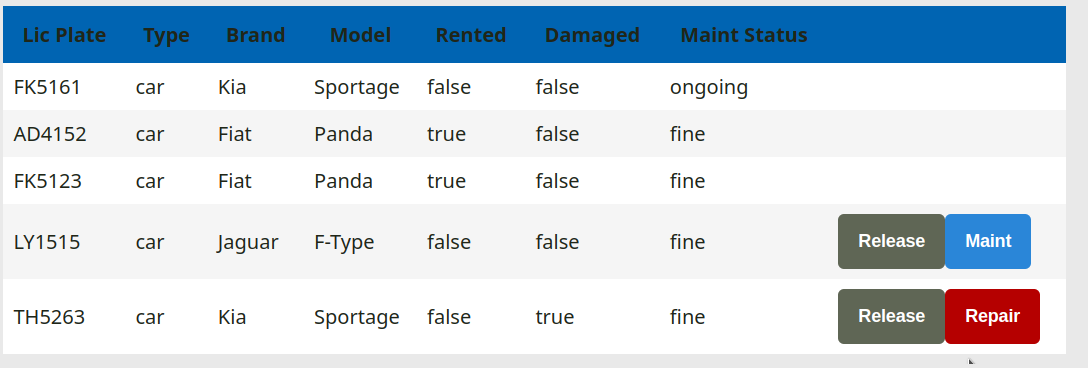
Else if insurance true no extra pay return new car to user

****

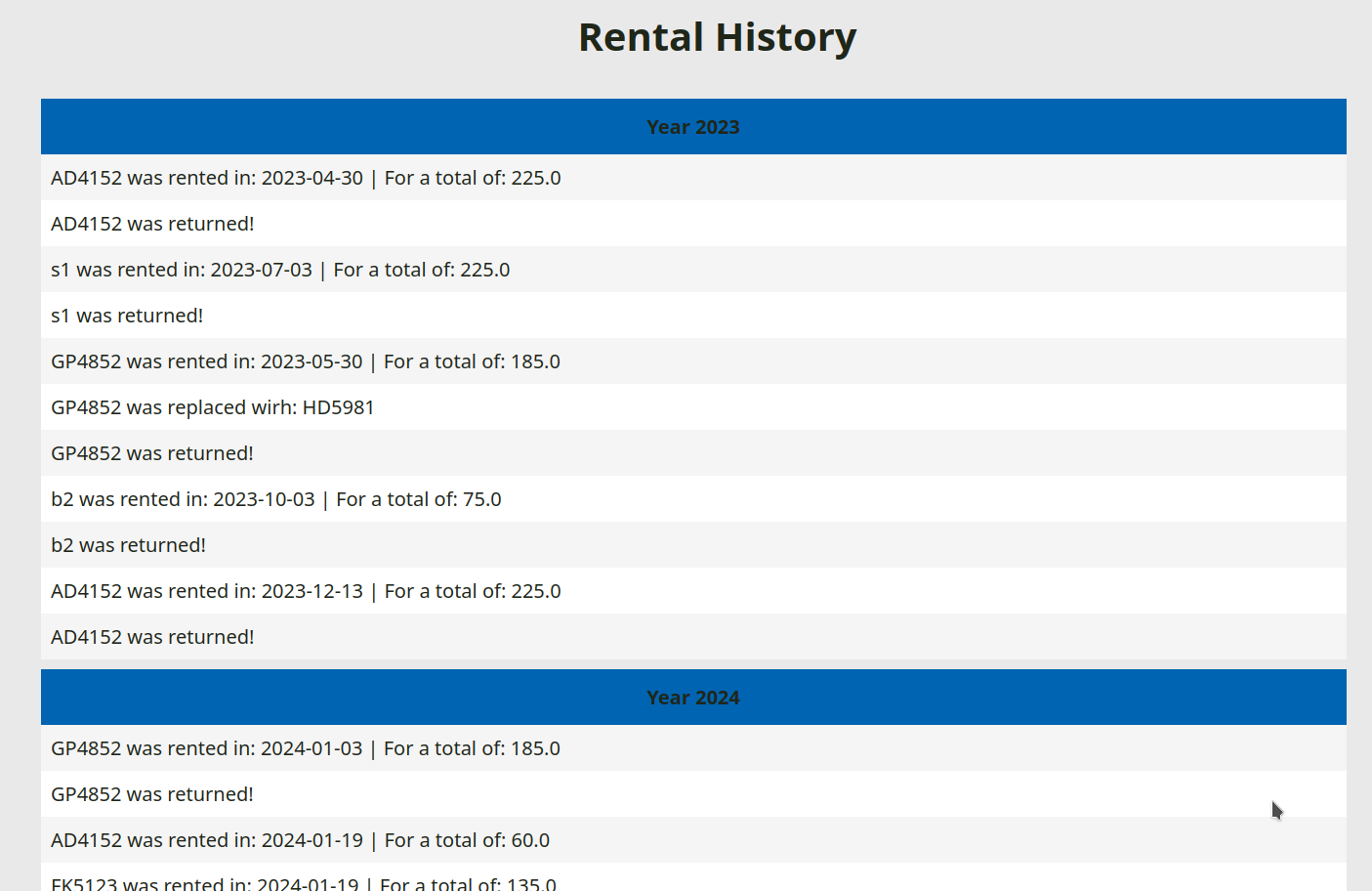
Show all available vehicles by type

****

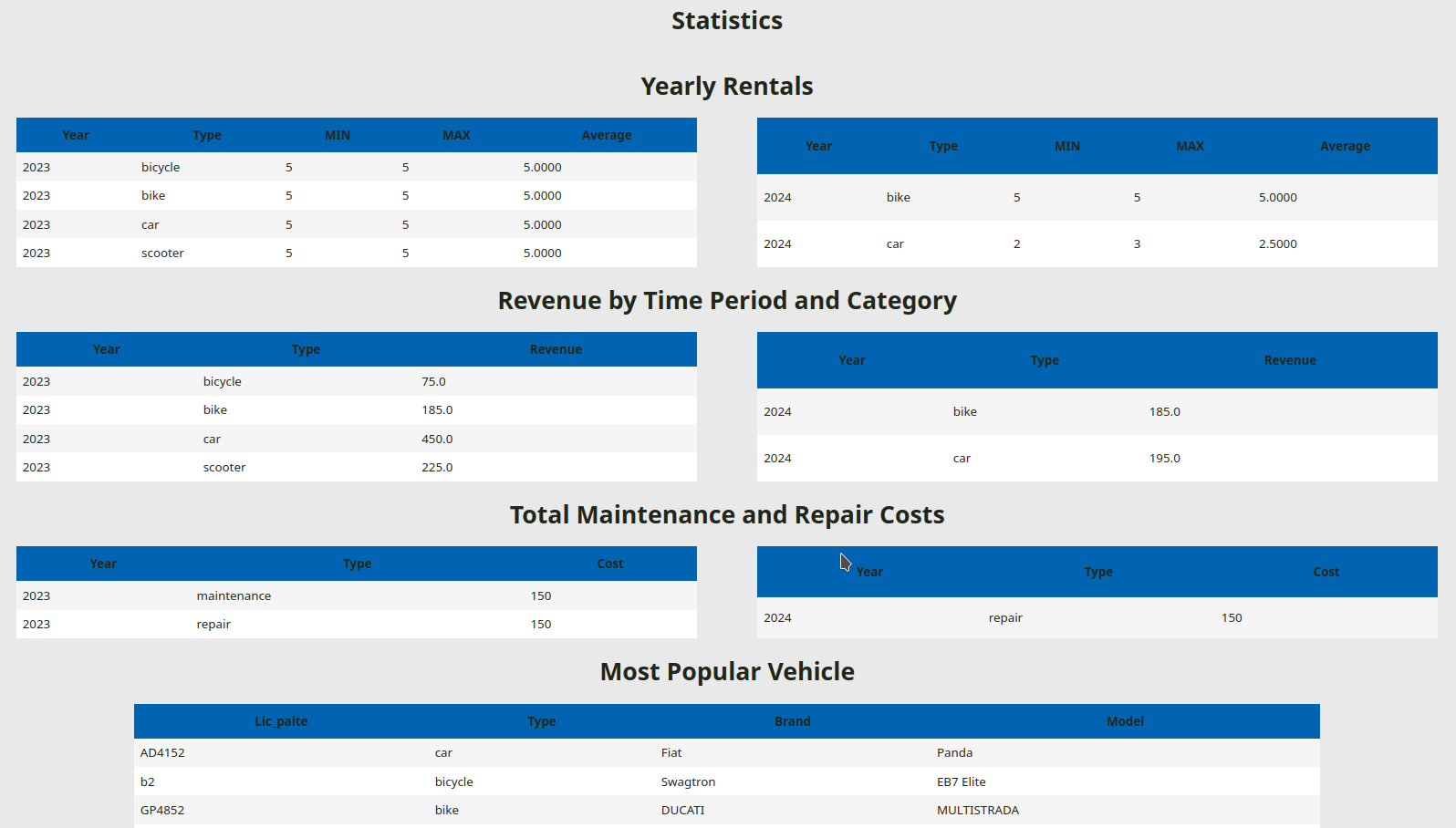
Add new vehicle

****

Availability of vehicles and to repair/maintenance them

****

Rental History: the status of rental in yearly period

****

Statistics in yearly period and Most Popular vehicle by type

**5. Συμπεράσματα**

Οτι μας ζητήθηκε από την εκφώνηση το υλοποιήσαμε σε ικανοποιητικό βαθμό. Tα μειονεκτήματα της υλοποίησης μας βρίσκονται κυρίως στην ασφάλεια καθώς δεν κάνουμε όλους τους επαρκής ελέγχους και δεν κρύβουμε στοιχεία των χρήστη. Λόγω sql injections, σίγουρα θα υπάρχουν προβλήματα στην ασφάλεια. Επίσης σε μερικά σημεία δεν βασιζόμαστε στη βάση να μας δώσει συγκεκριμένα δεδομένα τα βρίσκουμε μέσο JAVA (επαρκές όμως πάντα χρήση queries κτλ για πρόσβαση στα κατάλληλα δεδομένα).