



Ride or Cry **v.1.0**

ΑΠΟΘΕΤΗΡΙΟ ΕΦΑΡΜΟΓΗΣ:

<https://github.com/ihopewecanchangeitlater/Cloud-Services-Development>

ΒΙΝΤΕΟ ΕΦΑΡΜΟΓΗΣ:

<https://youtu.be/-YudLKLcTu0>

ΟΜΑΔΑ ΑΝΑΠΤΥΞΗΣ:

ihopewecanchangeitlater

ΠΡΟΓΡΑΜΜΑΤΙΣΤΕΣ:

Κωνσταντίνος Θωμασιάδης

Ευστάθιος Ιωσηφίδης

Αικατερίνη Κρότκα

ΘΕΣΣΑΛΟΝΙΚΗ Ιανουάριος 2025

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

Εισαγωγή	3
Περιγραφή Συστήματος	3
Κοινό Στόχος	3
Τεχνολογίες	3
Εγκατάσταση και εκκίνηση	4
Προβολή	4
Τερματισμός εκτέλεσης	4
ΕΙΣΟΔΟΣ / ΕΓΓΡΑΦΗ	5
Σύνδεση	5
Εγγραφή	5
Εγγραφή ως πολίτης	6
Εγγραφή ως Αντιπρόσωπος	7
ΠΟΛΙΤΗΣ	7
Λειτουργικότητα	8
Επιλογή Αυτοκινήτου	8
Κράτηση Αυτοκινήτου	9
Αγορά Αυτοκινήτου	10
Έξοδος	10
ΑΝΤΙΠΡΟΣΩΠΟΣ	11
Λειτουργικότητα	11
Προσθήκη Αυτοκινήτου στον στόλο	11
Επεξεργασία Ποσότητας Αυτοκινήτου	12
Επιστροφή αυτοκινήτου από Test Drive	15
Έξοδος	16
ΤΕΚΜΗΡΙΩΣΗ ΣΥΣΤΗΜΑΤΟΣ	17
Γενικές Πληροφορίες	17
Περιβάλλον Ανάπτυξης	17
Τεχνολογίες	17
Περιγραφή API	18
Endpoints	18
Για Κάθε Endpoint	19
Περιγραφή οντοτήτων	36
1. Car (Αυτοκίνητο)	36
2. User (Χρήστης)	36
3. Reservation (Κράτηση)	37

Εισαγωγή

Περιγραφή Συστήματος

Το σύστημα **RideOrCry** παρέχει μια ολοκληρωμένη και ενοποιημένη λύση για τη διασύνδεση των αντιπροσωπειών αυτοκινήτων με δυνητικούς πελάτες, προσφέροντας μια κοινή πλατφόρμα για την προσφορά, προβολή και επιλογή κάποιου αυτοκινήτου. Αναλυτικότερα:

- Χρήστης αντιπροσωπείας
 - Προσθήκη αυτοκινήτου στην κοινή βάση
 - Ανανέωση αριθμού διαθέσιμων αυτοκινήτων
 - Επιστροφή αυτοκινήτου δοκιμαστικής διαδρομής στον στόλο
 - Προβολή αυτοκινήτων που διατίθενται από την αντιπροσωπεία
 - Ημερήσιο πρόγραμμα με τις συνολικές κρατήσεις των αυτοκινήτων για δοκιμή
- Χρήστης πελάτης
 - Αγορά αυτοκινήτου
 - Καταχώρηση ραντεβού δοκιμαστικής διαδρομής
 - Προβολή όλων των διαθέσιμων αυτοκινήτων
 - Ημερήσιο πρόγραμμα με τις καταχωρημένες κρατήσεις για δοκιμή
- Όλοι οι χρήστες
 - Δυνατότητα εγγραφής στην πλατφόρμα
 - Δυνατότητα αυθεντικοποίησης για την είσοδο στην πλατφόρμα
 - Δυνατότητα προβολής με πολλαπλά κριτήρια (π.χ. μάρκα, μοντέλο, τιμή, κυβισμός, καύσιμο)

Ο στόχος του συστήματος είναι να παρέχει ένα εύχρηστο περιβάλλον για τους χρήστες του, συμπεριλαμβάνοντας όλες τις ενέργειες και διαδικασίες που βελτιστοποιούν την εμπειρία χρήσης του.

Κοινό Στόχος

- **Πολίτες:** Μπορούν να αναζητούν αυτοκίνητα, να κάνουν κρατήσεις δοκιμαστικής διαδρομής, και να αγοράζουν αυτοκίνητα εύκολα.
- **Αντιπρόσωποι:** Έχουν πρόσβαση στη διαχείριση του στόλου τους, προσθέτουν νέα αυτοκίνητα, και παρακολουθούν τις κρατήσεις.

Τεχνολογίες

Το σύστημα αναπτύχθηκε χρησιμοποιώντας τις εξής τεχνολογίες:

- **React:** Για την υλοποίηση του περιβάλλοντος χρήστη στο frontend.
- **Spring Boot:** Για την ανάπτυξη RESTful API στο backend.
- **JWT (JSON Web Tokens):** Για αυθεντικοποίηση και εξουσιοδότηση.
- **Hibernate/JPA:** Για την ενοποίηση της βάσης δεδομένων με το σύστημα.
- **PostgreSQL:** Ως σύστημα διαχείρισης σχεσιακής βάσης δεδομένων.
- **Swagger/OpenAPI:** Για τη δημιουργία και τεκμηρίωση του API.

ΕΚΤΕΛΕΣΗ ΕΦΑΡΜΟΓΗΣ

Εγκατάσταση και εκκίνηση

Για την εκτέλεση της εφαρμογής θα χρειαστούν τα ακόλουθα βήματα:

1. Κατεβάζουμε το αποθετήριο από το GitHub
git clone https://github.com/ihopewecanchangeitlater/Cloud-Services-Development
2. Εκτελούμε τις παρακάτω εντολές, ώστε να ανανεώσουμε τους τοπικούς καταλόγους
cd Cloud-Services-Development
git submodule init
git submodule update
3. Εκτελούμε τη ρουτίνα δημιουργίας των container με τη χρήση του docker compose
docker-compose up --build -d
** η παράμετρος *-d* απομονώνει την έξοδο της εντολής

Προβολή

Όταν ολοκληρωθεί η παραπάνω διαδικασία, μπορείτε να ανοίξετε τον browser στην ιστοσελίδα

<http://localhost:3000>

, στην οποία διατίθεται το περιβάλλον χρήστη.

Τερματισμός εκτέλεσης

Για να τερματιστεί την εκτέλεση των container, εκτελείτε την εντολή


docker-compose down

ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΤΗ

ΕΙΣΟΔΟΣ / ΕΓΓΡΑΦΗ

Σύνδεση

Η πρώτη οθόνη που θα εμφανιστεί είναι η οθόνη σύνδεσης, όπου θα σας ζητηθεί να εισάγετε τα διαπιστευτήριά σας (email και κωδικό πρόσβασης). Εάν διαθέτετε ήδη λογαριασμό, μπορείτε να εισαγάγετε τα στοιχεία σας και να συνδεθείτε στο σύστημα.



Login

Email

Password

Don't have an account? [Register](#)

Εγγραφή

Σε περίπτωση που δεν έχετε δημιουργήσει λογαριασμό, μπορείτε να επιλέξετε την επιλογή **Register**, η οποία θα σας μεταφέρει στην οθόνη εγγραφής.



Register

Στην οθόνη εγγραφής, θα πρέπει να επιλέξετε τον τύπο λογαριασμού που επιθυμείτε να δημιουργήσετε: **Πολίτης** ή **Αντιπρόσωπος** (επιλέγοντας το αντίστοιχο κουμπί). Στη συνέχεια, συμπληρώστε τα απαραίτητα πεδία με τα στοιχεία σας και πατήστε **Register** για να ολοκληρώσετε τη διαδικασία.

Εγγραφή ως πολίτης



Register

AFM:

Name:

Surname:


Email:

Password:

Back

Register

Εγγραφή ως Αντιπρόσωπος



Register

CITIZEN AGENCY

AFM:

Name:

Owner Name:

Email:

Password:

Back Register

Μετά την επιτυχή εγγραφή σας, θα επιστρέψετε στην αρχική οθόνη σύνδεσης, όπου μπορείτε να εισάγετε το email και τον κωδικό πρόσβασης που ορίσατε, προκειμένου να συνδεθείτε.

ΣΗΜΑΝΤΙΚΟ: Φροντίστε να θυμάστε τα στοιχεία σύνδεσης που εισάγετε (email και κωδικό πρόσβασης), καθώς θα απαιτούνται κάθε φορά που θέλετε να αποκτήσετε πρόσβαση στο σύστημα.

ΠΟΛΙΤΗΣ


Στην οθόνη σας εμφανίζονται τα εξής στοιχεία:

- Μπάρα στοιχείων:** Η επάνω μπάρα που περιέχει πληροφορίες σχετικά με τον λογαριασμό σας, όπως το όνομα χρήστη και άλλες σχετικές λεπτομέρειες.
- Μπάρα αναζήτησης:** Πεδία όπου μπορείτε να εισάγετε χαρακτηριστικά για την αναζήτηση αυτοκινήτων, όπως Μάρκα, Μοντέλο, Τύπο Καυσίμου, Κινητήρα και Αριθμό Θέσεων.
- Λίστα αυτοκινήτων:** Παρουσίαση όλων των διαθέσιμων αυτοκινήτων που πληρούν τα κριτήρια αναζήτησής σας ή όλων των αυτοκινήτων εάν δεν έχετε εισάγει φίλτρα.
- Ημερολόγιο:** Εμφάνιση των ημερομηνιών που έχουν ήδη γίνει κρατήσεις για test drive.

Λειτουργικότητα

Επιλογή Αυτοκινήτου

Στην οθόνη εμφανίζεται η λίστα των διαθέσιμων αυτοκινήτων. Ο χρήστης μπορεί να περιηγηθεί στη λίστα χρησιμοποιώντας τη μπάρα κύλισης για να εντοπίσει το αυτοκίνητο που τον ενδιαφέρει.



CITIZEN CITIZEN - (CITIZEN1@UOM.GR)[SIGN OUT](#)

Brand	Brand	Model	Price	Engine	Seats	Fuel
	Peugeot	208	18000	1500	5	Diesel
	Peugeot	308	23000	1200	5	Petrol
	Peugeot	5008	33000	1200	7	Petrol

January 2025<>

S	M	T	W	T	F	S
				1	2	3
				4		
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Price

033000


Engine

8001500

Seats

FIND

Εάν ο χρήστης αναζητά ένα συγκεκριμένο αυτοκίνητο και δεν το βρίσκει εύκολα στη λίστα, μπορεί να χρησιμοποιήσει τη μπάρα αναζήτησης, η οποία βρίσκεται στην αριστερή πλευρά της οθόνης. Εκεί μπορεί να εισάγει συγκεκριμένα κριτήρια αναζήτησης, όπως Μάρκα, Μοντέλο, Τύπο Καυσίμου, Κινητήρα ή Αριθμό Θέσεων.



CITIZEN CITIZEN - (CITIZEN1@UOM.GR)[SIGN OUT](#)

Brand	Brand	Model	Price	Engine	Seats	Fuel
	Peugeot	208	18000	1500	5	Diesel

January 2025<>

S	M	T	W	T	F	S
				1	2	3
				4		
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Price

019000

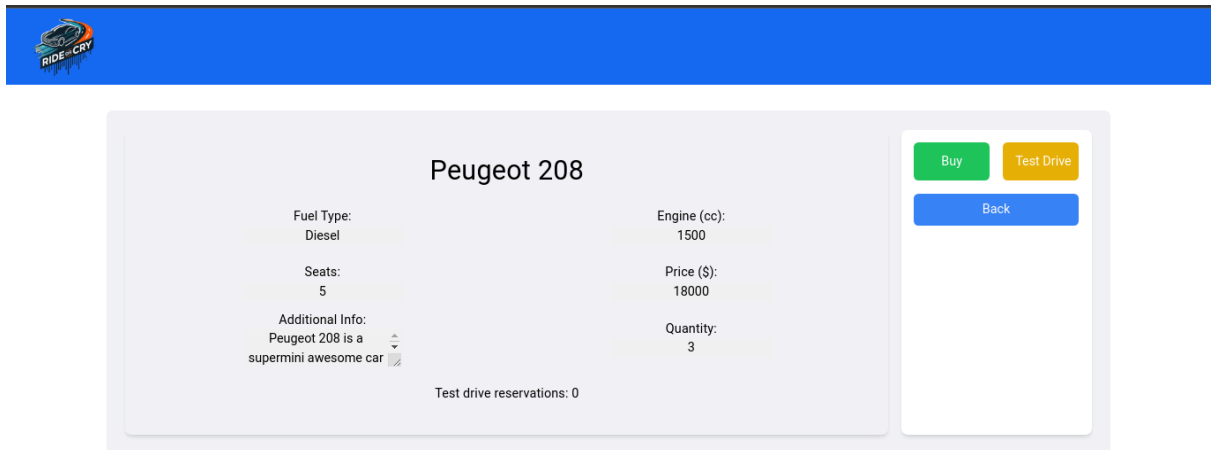
Engine

8001500

Seats

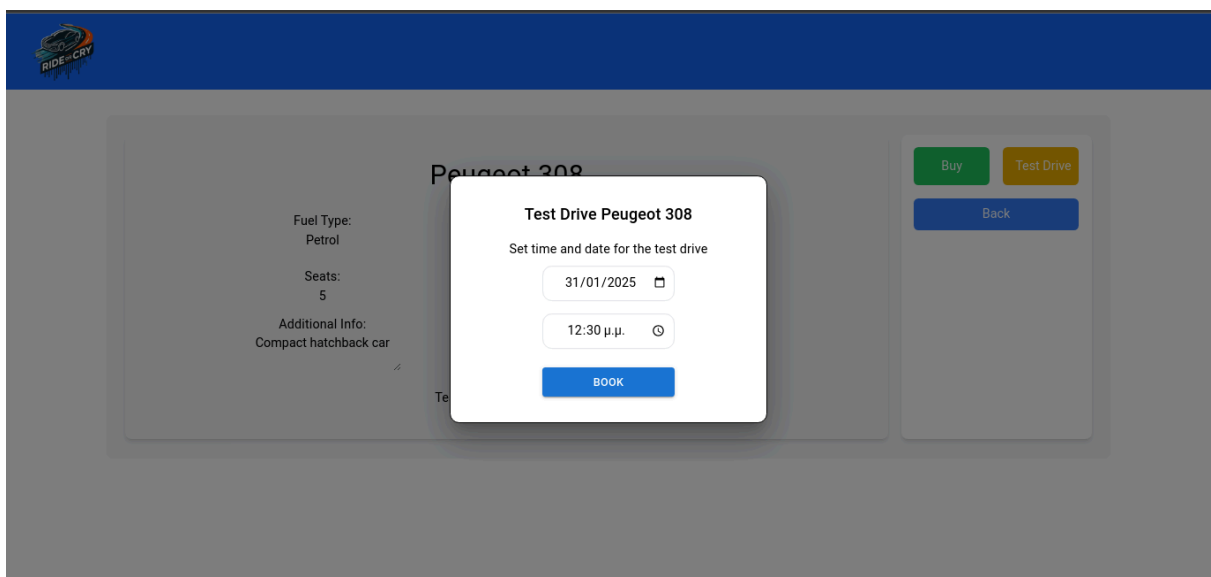
FIND

Μόλις εντοπίσει το επιθυμητό αυτοκίνητο, μπορεί να κάνει κλικ επάνω του για να μεταβεί στην οθόνη με τις λεπτομέρειες του οχήματος. Στην οθόνη αυτή εμφανίζονται αναλυτικές πληροφορίες για το αυτοκίνητο, όπως χαρακτηριστικά και διαθεσιμότητα. Εάν ο χρήστης αποφασίσει ότι δεν τον ενδιαφέρει, μπορεί να πατήσει το κουμπί **Back** για να επιστρέψει στην κεντρική οθόνη.




Κράτηση Αυτοκινήτου

Στην ίδια οθόνη, εάν ο χρήστης επιθυμεί να δοκιμάσει το αυτοκίνητο, μπορεί να πατήσει το κουμπί **Test Drive**. Στο παράθυρο που θα εμφανιστεί, μπορεί να εισάγει την επιθυμητή ημερομηνία και ώρα για το test drive και στη συνέχεια να πατήσει το κουμπί **Book**.



Η κράτηση θα καταχωρηθεί στο ημερολόγιό του, το οποίο είναι ορατό στην κεντρική οθόνη.



CITIZEN CITIZEN - (CITIZEN1@UOM.GR) [SIGN OUT](#)

Brand

Model

Fuel

Price

Engine

Seats

Peugeot

208

Diesel

18000

1500

5

Peugeot

308

Petrol

23000

1200

5

Peugeot

5008

Petrol

33000

1200

7

January 2025

<

>


S	M	T	W	T	F	S
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

FIND

Εάν ο χρήστης επιθυμεί να αγοράσει το αυτοκίνητο μετά το test drive, μπορεί να πατήσει το κουμπί **Buy** για να ολοκληρώσει την αγορά.

Αγορά Αυτοκινήτου

Στην οθόνη με τις πληροφορίες του αυτοκινήτου, εάν ο χρήστης επιθυμεί να αγοράσει το συγκεκριμένο αυτοκίνητο, μπορεί να πατήσει το κουμπί **Buy**. Μετά την επιλογή, εμφανίζεται μήνυμα που επιβεβαιώνει την επιτυχή ολοκλήρωση της αγοράς (αφαιρείται από τον στόλο).



Success

×

Thank you for your purchase

Peugeot 208

Fuel Type:
Diesel

Seats:
5

Additional Info:
Peugeot 208 is a
supermini awesome car

Engine (cc):
1500

Price (\$):
18000

Quantity:
2

Test drive reservations: 0

Buy

Test Drive

Back

Μόλις ολοκληρωθεί η διαδικασία, ο χρήστης μπορεί να πατήσει το κουμπί **Back** για να επιστρέψει στην κεντρική οθόνη.

Έξοδος

Όταν ο χρήστης ολοκληρώσει τις ενέργειες, μπορεί να πατήσει το κουμπί **Signout** και να εξέλθει του προγράμματος.

ΑΝΤΙΠΡΟΣΩΠΟΣ

Στην οθόνη σας εμφανίζονται τα εξής στοιχεία:

1. **Μπάρα στοιχείων:** Η επάνω μπάρα που περιέχει πληροφορίες σχετικά με τον λογαριασμό σας, όπως το όνομα χρήστη και άλλες σχετικές λεπτομέρειες.
2. **Μπάρα αναζήτησης:** Πεδία όπου μπορείτε να εισάγετε χαρακτηριστικά για την αναζήτηση αυτοκινήτων, όπως Μάρκα, Μοντέλο, Τύπο Καυσίμου, Κινητήρα και Αριθμό Θέσεων.
3. **Λίστα αυτοκινήτων:** Παρουσίαση όλων των διαθέσιμων αυτοκινήτων που πληρούν τα κριτήρια αναζήτησής σας ή όλων των αυτοκινήτων του στόλου της αντιπροσωπείας, εάν δεν έχετε εισάγει φίλτρα.
4. **Προσθήκη αυτοκινήτου:** Κουμπί + που ένας αντιπρόσωπος μπορεί να εισάγει ένα αυτοκίνητο στην εταιρία του.
5. **Ημερολόγιο:** Εμφάνιση των ημερομηνιών που έχουν ήδη γίνει κρατήσεις για test drive.

Λειτουργικότητα

Προσθήκη Αυτοκινήτου στον στόλο

Στην οθόνη εμφανίζεται η λίστα με τα διαθέσιμα αυτοκίνητα του αντιπροσώπου.

The screenshot displays the RIDE=CRY application interface. At the top, there is a blue header bar with the RIDE=CRY logo on the left, the user information 'AGENT AGENT - (AGENT1@UOM.GR)' in the center, and a 'SIGN OUT' button on the right. Below the header, the main content area is divided into two sections. On the left, there is a search filter panel with fields for 'Brand', 'Model', 'Fuel', 'Price' (with a slider from 0 to 0), 'Engine' (with a slider from 800 to 0), and 'Seats'. A 'FIND' button is located at the bottom of this panel. On the right, there is a table with columns 'Brand', 'Model', 'Price', 'Engine', 'Seats', and 'Fuel'. The table currently shows 'No cars found'. To the right of the table is a calendar for 'January 2025' with a grid of dates. A large blue circular button with a white '+' sign is positioned below the calendar.

Ο αντιπρόσωπος μπορεί να προσθέσει ένα νέο αυτοκίνητο στον στόλο του κάνοντας κλικ στο κουμπί + (**Add Car**). Στη συνέχεια, θα εμφανιστεί φόρμα όπου ο αντιπρόσωπος εισάγει τα απαραίτητα στοιχεία του αυτοκινήτου, όπως Μάρκα, Μοντέλο, Τύπο Καυσίμου, Κινητήρα κ.λπ.

ADD A NEW CAR

Complete the following form to create a new car entry

Brand Model

Fuel Engine

Seats Quantity

Price

Additional Information

CREATE

Μετά την εισαγωγή των στοιχείων, πατάει το κουμπί **Create** για να ολοκληρώσει τη διαδικασία. Το νέο αυτοκίνητο προστίθεται αυτόματα στη λίστα των διαθέσιμων αυτοκινήτων.

AGENT AGENT - (AGENT1@UOM.GR) **SING OUT**

Brand	Model	Price	Engine	Seats	Fuel
Peugeot	208	18000	1500	5	Diesel

January 2025

S M T W T F S

1 2 3 4

5 6 7 8 9 10 11

12 13 14 15 16 17 18


19 20 21 22 23 24 25

26 27 28 29 30 31

+

Επεξεργασία Ποσότητας Αυτοκινήτου

Στην οθόνη όπου εμφανίζεται η λίστα με τα διαθέσιμα αυτοκίνητα, ο χρήστης μπορεί να περιηγηθεί χρησιμοποιώντας την μπάρα κύλισης για να εντοπίσει το αυτοκίνητο που θέλει να επεξεργαστεί.



AGENT AGENT - (AGENT1@UOM.GR) SING OUT

Brand
Model
Fuel
Price
Engine
Seats

Brand	Model	Price	Engine	Seats	Fuel
Peugeot	208	18000	1500	5	Diesel
Peugeot	308	23000	1200	5	Petrol
Peugeot	5008	33000	1200	7	Petrol

January 2025
<
>

S M T W T F S
1 2 3 4
5 6 7 8 9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31


+

0 33000
800 1500

Seats

FIND

Εάν ο χρήστης δυσκολεύεται να βρει το επιθυμητό αυτοκίνητο, μπορεί να χρησιμοποιήσει τη μπάρα αναζήτησης, που βρίσκεται στην αριστερή πλευρά της οθόνης. Εκεί, μπορεί να εισάγει συγκεκριμένα κριτήρια, όπως Μάρκα, Μοντέλο, Τύπο Καυσίμου, Κινητήρα ή Αριθμό Θέσεων, για να περιορίσει τα αποτελέσματα.



AGENT AGENT - (AGENT1@UOM.GR) SING OUT

Brand
Model
Fuel
Price
Engine
Seats

Brand	Model	Price	Engine	Seats	Fuel
Peugeot	308	23000	1200	5	Petrol

January 2025
<
>

S M T W T F S
1 2 3 4
5 6 7 8 9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31

+

0 33000
800 1200

5

FIND

Μόλις εντοπιστεί το αυτοκίνητο, ο χρήστης μπορεί να κάνει κλικ επάνω του για να μεταβεί στην οθόνη με τις λεπτομέρειες του οχήματος. Στην οθόνη αυτή εμφανίζονται αναλυτικές πληροφορίες για το αυτοκίνητο, όπως τα χαρακτηριστικά του και η διαθέσιμη ποσότητα.



Peugeot 308

Fuel Type:	Petrol	Engine (cc):	1200
Seats:	5	Price (\$):	23000
Additional Info:	Compact hatchback car	Quantity:	2

Test drive reservations: 0

Update Quantity

0

OK

Test Drive Return

Back

Για να επεξεργαστεί την ποσότητα, ο χρήστης ενημερώνει το πεδίο **Update Quantity** και πατάει **OK** για να ενημερωθεί τη διαθεσιμότητα του αυτοκινήτου. Με την ολοκλήρωση της διαδικασίας, η λίστα ενημερώνεται αυτόματα με τη νέα ποσότητα.



Peugeot 308

Fuel Type:	Petrol	Engine (cc):	1200
Seats:	5	Price (\$):	23000
Additional Info:	Compact hatchback car	Quantity:	3

Test drive reservations: 0

Update Quantity

0

OK

Test Drive Return


Back

Αλλαγή ποσότητας (Quantity) από 2 σε 3

Στη συνέχεια ο χρήστης πατάει το κουμπί Back για να επιστρέψει στην κεντρική οθόνη.

Επιστροφή αυτοκινήτου από Test Drive

Στην οθόνη όπου εμφανίζεται η λίστα με τα διαθέσιμα αυτοκίνητα, ο χρήστης μπορεί να δει στο ημερολόγιο ότι υπάρχει κράτηση ενός αυτοκινήτου για Test Drive.



AGENT AGENT - (AGENT1@UOM.GR)SING OUT

Brand

Model

Fuel

Price

Engine

Seats

FIND

Brand	Model	Price	Engine	Seats	Fuel
Peugeot	208	18000	1500	5	Diesel
Peugeot	308	23000	1200	5	Petrol
Peugeot	5008	33000	1200	7	Petrol

033000

8001500

January 2025

S

M

T

W

T

F

S

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29


30

31

Test Drive

+

Αφού περιηγηθεί χρησιμοποιώντας τη μπάρα κύλισης για να εντοπίσει το αυτοκίνητο που θέλει να επεξεργαστεί. Πατάει επάνω του και εμφανίζεται η οθόνη με τις πληροφορίες για το αυτοκίνητο.



Peugeot 308

Fuel Type:
Petrol

Seats:
5

Additional Info:
Compact hatchback car

Engine (cc):
1200

Price (\$):
23000

Quantity:
2

Test drive reservations: 1

Update Quantity

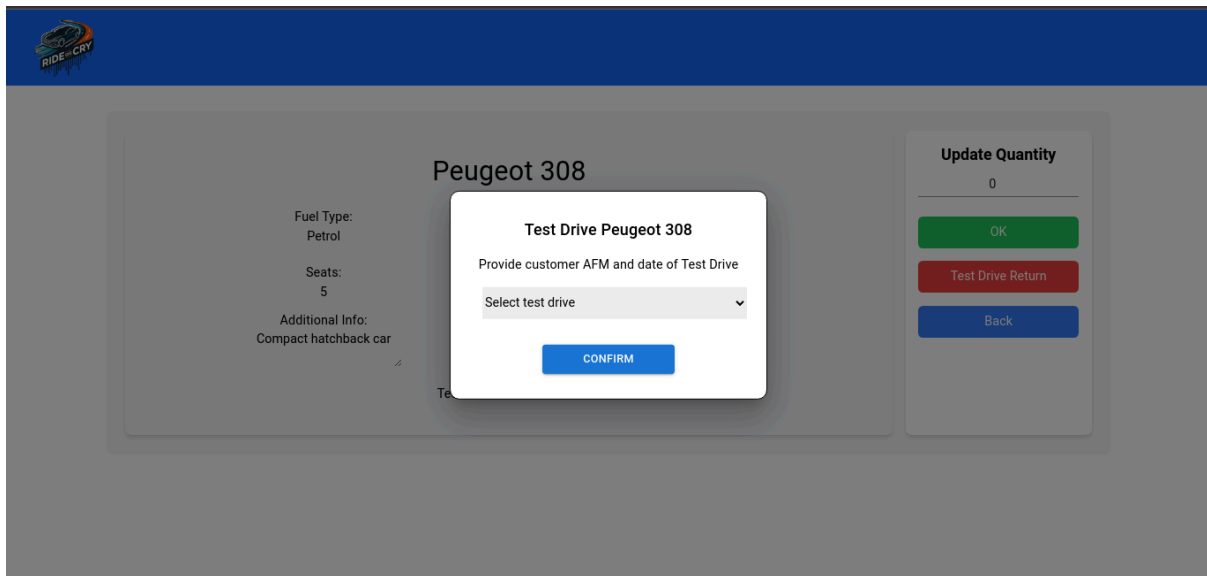
0

OK

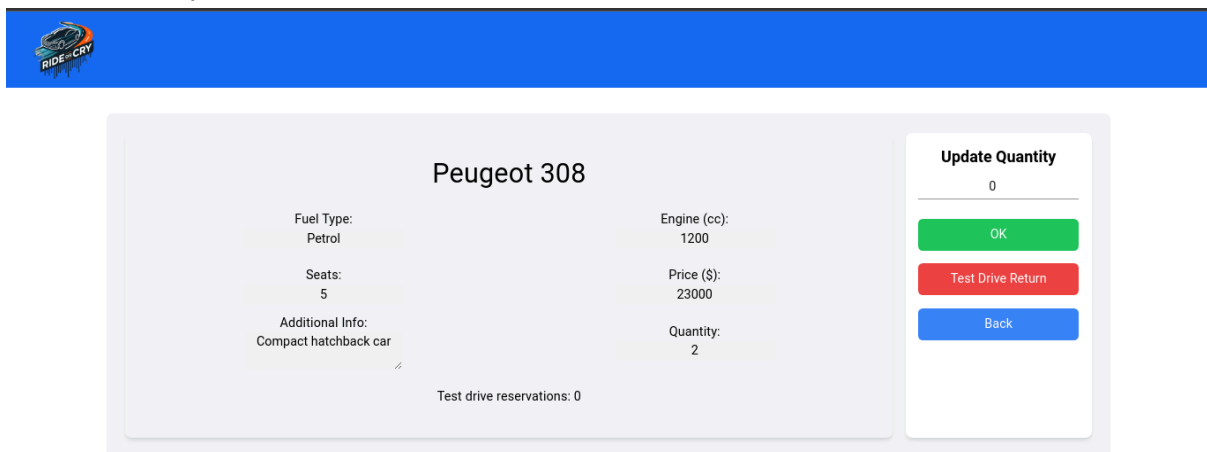
Test Drive Return

Back

Το αυτοκίνητο του παραδείγματος, έχει ένα αυτοκίνητο σε Test Drive μορφή. Ο χρήστης μπορεί να πατήσει το κουμπί **Test Drive Return** για να επεξεργαστεί την διαθεσιμότητά του.



Αφού επιλέξει τον υποψήφιο αγοραστή που είχε κρατημένο το αυτοκίνητο, πατάει το Confirm και το αυτοκίνητο είναι διαθέσιμο στον στόλο.



Έξοδος

Όταν ο χρήστης ολοκληρώσει τις ενέργειες που επιθυμεί, μπορεί να αποσυνδεθεί από το σύστημα πατώντας το κουμπί Signout. Με την αποσύνδεση, ο χρήστης μεταφέρεται στην αρχική οθόνη εισόδου. Η αποσύνδεση διασφαλίζει την προστασία του λογαριασμού και των προσωπικών του δεδομένων.

ΤΕΚΜΗΡΙΩΣΗ ΣΥΣΤΗΜΑΤΟΣ

Γενικές Πληροφορίες

Περιβάλλον Ανάπτυξης

Το σύστημα αναπτύχθηκε χρησιμοποιώντας το **Spring Boot** για το backend και το **React** για το frontend. Ακολουθούν οι βασικές πληροφορίες για το περιβάλλον εκτέλεσης:

- **Base URL Backend:** <http://localhost:8080>
- **Base URL API:** <http://localhost:8080/api-docs>
- **Swagger URL:** <http://localhost:8080/api-ui>
- **Frontend URL:** <http://localhost:3000>
- **Database URL:** <http://localhost:5432>

Τεχνολογίες

Για να τρέξει το σύστημα, απαιτούνται τα παρακάτω εργαλεία και βιβλιοθήκες:

1. **Backend:**
 - **Java 17** ή νεότερη έκδοση.
 - **Maven** (για τη διαχείριση εξαρτήσεων).
 - **Spring Boot:** Πακέτα για REST API, ασφάλεια, και JPA:
 - spring-boot-starter-web
 - spring-boot-starter-data-jpa
 - spring-boot-starter-security
 - springdoc-openapi-ui (Swagger)
 - **PostgreSQL Driver:** Για σύνδεση με τη βάση δεδομένων
 - **JWT Library:** Χρησιμοποιείται για τη διαχείριση tokens
2. **Frontend:**
 - **Node.js** (>=16) και **npm** ή **yarn** για τη διαχείριση εξαρτήσεων.
 - **React** με τις ακόλουθες βιβλιοθήκες:
 - axios για HTTP REST αιτήματα
 - react-router-dom για πλοήγηση
 - MUI για λειτουργικά στοιχεία
 - TailwindCSS για εφαρμογή θεμάτων
3. **Βάση Δεδομένων:**
 - **PostgreSQL 14.x** ή νεότερη έκδοση
4. **Άλλα εργαλεία:**
 - **Docker** (προαιρετικά): container για κατασκευή και ανάπτυξη

Περιγραφή API

Endpoints

Παρακάτω παρουσιάζεται η λίστα των διαθέσιμων endpoints με τις μεθόδους HTTP και τις περιγραφές λειτουργίας τους:

HTTP Μέθοδος	Endpoint	Περιγραφή
POST	/auth/login	Είσοδος χρήστη και λήψη JWT token.
POST	/auth/logout	Αποσύνδεση χρήστη.
POST	/auth/register/citizen	Εγγραφή νέου πολίτη.
POST	/auth/register/agency	Εγγραφή νέου αντιπροσώπου.
GET	/cars/carlist	Επιστρέφει όλα τα διαθέσιμα αυτοκίνητα.
GET	/cars/byagency/{agencyName}	Επιστρέφει όλα τα αυτοκίνητα ενός αντιπροσώπου.
GET	/cars/{carId}	Επιστρέφει πληροφορίες για συγκεκριμένο αυτοκίνητο.
GET	/cars/search	Αναζητά αυτοκίνητα με βάση φίλτρα.
POST	/cars/addcar/{agencyAfm}	Προσθέτει νέο αυτοκίνητο σε έναν αντιπρόσωπο.
PATCH	/cars/updateQuantity/{carId}	Ενημερώνει την ποσότητα ενός αυτοκινήτου.
PATCH	/cars/buy/{id}	Μειώνει την ποσότητα ενός αυτοκινήτου κατά 1 (αγορά).
GET	/reservations	Επιστρέφει όλες τις κρατήσεις.
GET	/reservations/user/{userId}	Επιστρέφει τις κρατήσεις ενός χρήστη.
GET	/reservations/car/{carId}	Επιστρέφει τις κρατήσεις ενός αυτοκινήτου.
POST	/reservations	Δημιουργεί νέα κράτηση.

DELETE	/reservations/{reservationId}	Διαγράφει μια κράτηση.
--------	-------------------------------	------------------------

Για Κάθε Endpoint

1. POST /auth/login

- **Περιγραφή:** Είσοδος χρήστη και λήψη JWT token.

Σώμα Αιτήματος:

```
{
  "email": "string",
  "password": "string"
}
```

Παραδείγματα Απόκρισης:

Επιτυχία (200):

```

1  {
2    "user": {
3      "afm": "123456789",
4      "email": "user@example.com",
5      "name": "John Doe"
6    },
7    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
8    "error": null
9  }
10
```

2. POST /auth/logout

- **Περιγραφή:** Αποσύνδεση χρήστη, καθαρίζοντας το session ή το token.
- **Απόκριση:**

"Logged out successfully"

3. POST /auth/register/citizen

- **Περιγραφή:** Εγγραφή νέου πολίτη.

Σώμα Αιτήματος:

```
{  
  "afm": "string",  
  "email": "string",  
  "name": "string",  
  "password": "string",  
  "surname": "string"  
}
```

Παραδείγματα Απόκρισης:

Επιτυχία (200):

"User registered successfully"

4. POST /auth/register/agency

- **Περιγραφή:** Εγγραφή νέου αντιπροσώπου.

Σώμα Αιτήματος:

```
{  
  "afm": "string",  
  "email": "string",  
  "name": "string",  
  "password": "string",  
  "owner": "string"  
}
```

Απόκριση:

"User registered successfully"

5. GET /cars/carlist

- **Περιγραφή:** Επιστρέφει όλα τα διαθέσιμα αυτοκίνητα.
- **Παραδείγματα Απόκρισης:**

Επιτυχία (200):

```
1  [  
2    {  
3      "id": 0,  
4      "agency": {  
5        "afm": "string",  
6        "email": "string",  
7        "name": "string",  
8        "password": "string",  
9        "owner": "string"  
10     },  
11     "brand": "string",  
12     "model": "string",  
13     "fuel": "string",  
14     "engine": 0,  
15     "seats": 0,  
16     "price": 0,  
17     "quantity": 0,  
18     "additionalInfo": "string"  
19   }  
20 ]  
21
```

6. GET /cars/byagency/{agencyName}

- **Περιγραφή:** Επιστρέφει όλα τα αυτοκίνητα ενός αντιπροσώπου.
- **Παράμετροι:**
 - agencyName (String): Το όνομα του αντιπροσώπου.

Παράδειγμα Απόκρισης:

```
1  [  
2    {  
3      "id": 0,  
4      "agency": {  
5        "afm": "string",  
6        "email": "string",  
7        "name": "string",  
8        "password": "string",  
9        "owner": "string"  
10     },  
11     "brand": "string",  
12     "model": "string",  
13     "fuel": "string",  
14     "engine": 0,  
15     "seats": 0,  
16     "price": 0,  
17     "quantity": 0,  
18     "additionalInfo": "string"  
19   }  
20 ]  
21
```

7. GET /cars/{carId}

- **Περιγραφή:** Επιστρέφει πληροφορίες για συγκεκριμένο αυτοκίνητο.
- **Παράμετροι:**
 - carId (Long): Το μοναδικό ID του αυτοκινήτου.

Παράδειγμα Απόκρισης:

```
1  {
2    "id": 0,
3    "agency": {
4      "afm": "string",
5      "email": "string",
6      "name": "string",
7      "password": "string",
8      "owner": "string"
9    },
10   "brand": "string",
11   "model": "string",
12   "fuel": "string",
13   "engine": 0,
14   "seats": 0,
15   "price": 0,
16   "quantity": 0,
17   "additionalInfo": "string"
18 }
19
```

8. GET /cars/search

- **Περιγραφή:** Αναζητά αυτοκίνητα με βάση φίλτρα.
- **Παράμετροι:**
 - **filters** (Query Parameters): Περιλαμβάνει τα κριτήρια αναζήτησης (π.χ., brand=Toyota, price_lt=30000).

Παράδειγμα Απόκρισης:

```
1  [
2    {
3      "id": 0,
4      "agency": {
5        "afm": "string",
6        "email": "string",
7        "name": "string",
8        "password": "string",
9        "owner": "string"
10     },
11     "brand": "string",
12     "model": "string",
13     "fuel": "string",
14     "engine": 0,
15     "seats": 0,
16     "price": 0,
17     "quantity": 0,
18     "additionalInfo": "string"
19   }
20 ]
21
```


9. POST /cars/addcar/{agencyAfm}

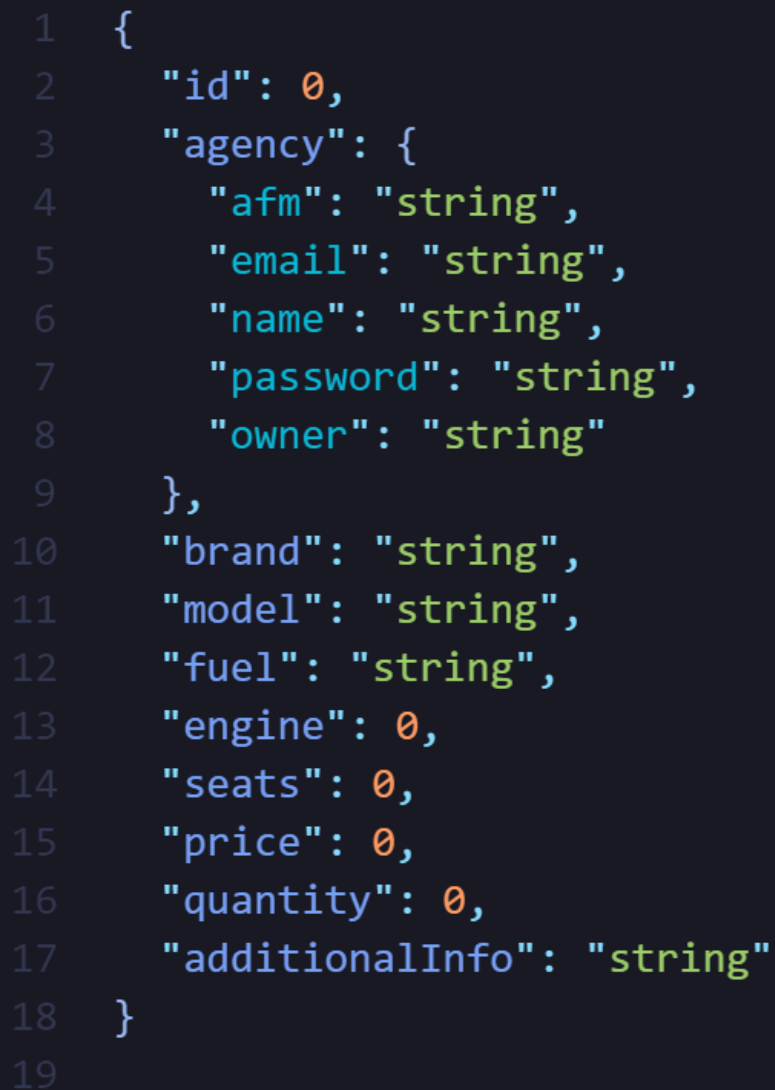
- **Περιγραφή:** Προσθέτει νέο αυτοκίνητο σε έναν αντιπρόσωπο.

Σώμα Αιτήματος:



```
1  {  
2    "brand": "Toyota",  
3    "model": "Corolla",  
4    "fuel": "Petrol",  
5    "engine": 1800,  
6    "seats": 5,  
7    "price": 20000.0,  
8    "quantity": 10  
9  }  
10
```

Απόκριση:



```
1  {
2    "id": 0,
3    "agency": {
4      "afm": "string",
5      "email": "string",
6      "name": "string",
7      "password": "string",
8      "owner": "string"
9    },
10   "brand": "string",
11   "model": "string",
12   "fuel": "string",
13   "engine": 0,
14   "seats": 0,
15   "price": 0,
16   "quantity": 0,
17   "additionalInfo": "string"
18 }
19
```

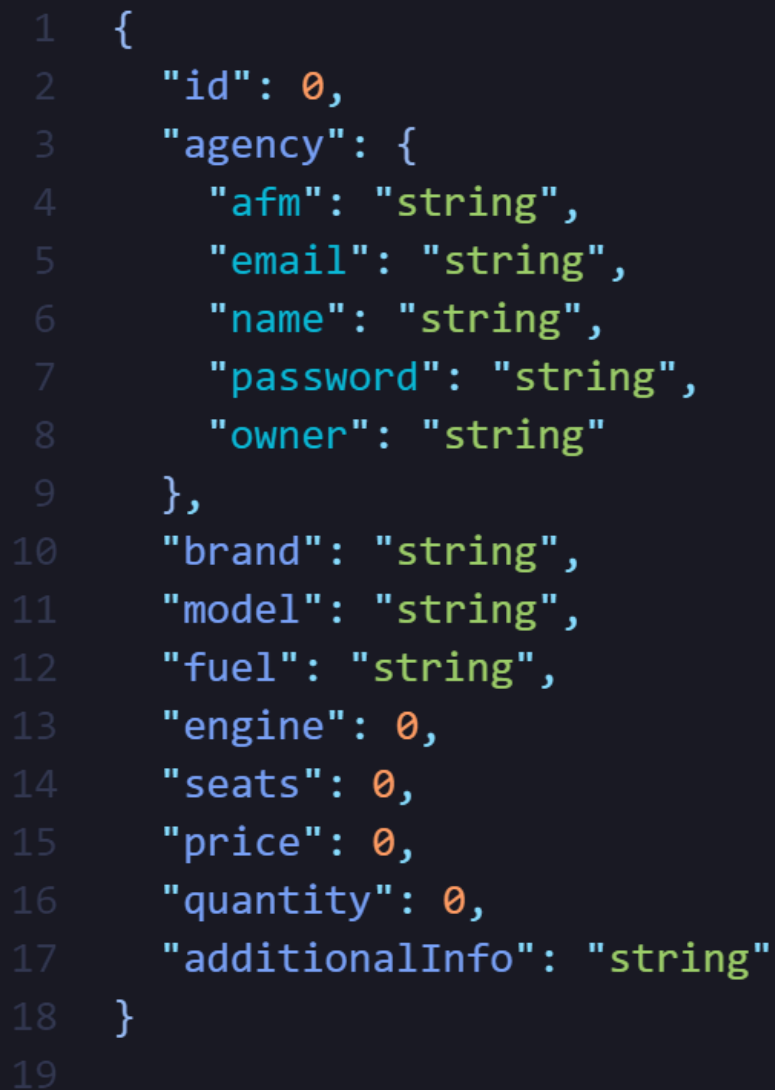
10. PATCH /cars/updateQuantity/{carId}

- **Περιγραφή:** Ενημερώνει την ποσότητα ενός αυτοκινήτου.

Σώμα Αιτήματος:



Απόκριση:



```
1  {
2    "id": 0,
3    "agency": {
4      "afm": "string",
5      "email": "string",
6      "name": "string",
7      "password": "string",
8      "owner": "string"
9    },
10   "brand": "string",
11   "model": "string",
12   "fuel": "string",
13   "engine": 0,
14   "seats": 0,
15   "price": 0,
16   "quantity": 0,
17   "additionalInfo": "string"
18 }
19
```

11. PATCH /cars/buy/{id}

- **Περιγραφή:** Μειώνει την ποσότητα ενός αυτοκινήτου κατά 1.

Απόκριση:

```
1  {
2    "id": 0,
3    "agency": {
4      "afm": "string",
5      "email": "string",
6      "name": "string",
7      "password": "string",
8      "owner": "string"
9    },
10   "brand": "string",
11   "model": "string",
12   "fuel": "string",
13   "engine": 0,
14   "seats": 0,
15   "price": 0,
16   "quantity": 0,
17   "additionalInfo": "string"
18 }
19
```

12. GET /reservations

- **Περιγραφή:** Επιστρέφει όλες τις κρατήσεις.

Απόκριση:

```
1  [
2    {
3      "id": 0,
4      "citizen": {
5        "afm": "string",
6        "email": "string",
7        "name": "string",
8        "password": "string",
9        "surname": "string"
10     },
11     "car": {
12       "id": 0,
13       "agency": {
14         "afm": "string",
15         "email": "string",
16         "name": "string",
17         "password": "string",
18         "owner": "string"
19       },
20       "brand": "string",
21       "model": "string",
22       "fuel": "string",
23       "engine": 0,
24       "seats": 0,
25       "price": 0,
26       "quantity": 0,
27       "additionalInfo": "string"
28     },
29     "date": "2025-01-31T14:26:02.303Z",
30     "time": "2025-01-31T14:26:02.303Z"
31   }
32 ]
33
```

13. GET /reservations/user/{userId}

- **Περιγραφή:** Επιστρέφει τις κρατήσεις ενός χρήστη.

Απόκριση:

```
1  [  
2    {  
3      "id": 0,  
4      "citizen": {  
5        "afm": "string",  
6        "email": "string",  
7        "name": "string",  
8        "password": "string",  
9        "surname": "string"  
10     },  
11     "car": {  
12       "id": 0,  
13       "agency": {  
14         "afm": "string",  
15         "email": "string",  
16         "name": "string",  
17         "password": "string",  
18         "owner": "string"  
19       },  
20       "brand": "string",  
21       "model": "string",  
22       "fuel": "string",  
23       "engine": 0,  
24       "seats": 0,  
25       "price": 0,  
26       "quantity": 0,  
27       "additionalInfo": "string"  
28     },  
29     "date": "2025-01-31T14:26:02.303Z",  
30     "time": "2025-01-31T14:26:02.303Z"  
31   }  
32 ]  
33
```

14. GET /reservations/car/{carId}

- **Περιγραφή:** Επιστρέφει τις κρατήσεις ενός αυτοκινήτου.

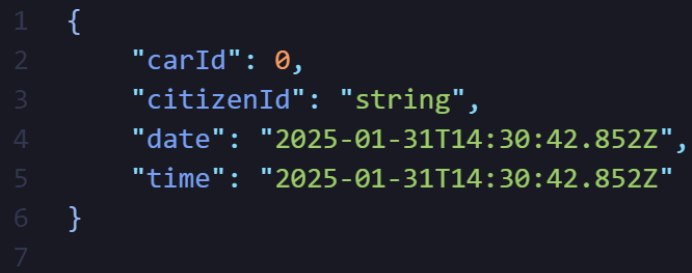
Απόκριση:

```
1  [  
2    {  
3      "id": 0,  
4      "citizen": {  
5        "afm": "string",  
6        "email": "string",  
7        "name": "string",  
8        "password": "string",  
9        "surname": "string"  
10     },  
11     "car": {  
12       "id": 0,  
13       "agency": {  
14         "afm": "string",  
15         "email": "string",  
16         "name": "string",  
17         "password": "string",  
18         "owner": "string"  
19       },  
20       "brand": "string",  
21       "model": "string",  
22       "fuel": "string",  
23       "engine": 0,  
24       "seats": 0,  
25       "price": 0,  
26       "quantity": 0,  
27       "additionalInfo": "string"  
28     },  
29     "date": "2025-01-31T14:26:02.303Z",  
30     "time": "2025-01-31T14:26:02.303Z"  
31   }  
32 ]  
33
```


15. POST /reservations

- **Περιγραφή:** Δημιουργεί νέα κράτηση.

Σώμα Αιτήματος:



```
1 {  
2   "carId": 0,  
3   "citizenId": "string",  
4   "date": "2025-01-31T14:30:42.852Z",  
5   "time": "2025-01-31T14:30:42.852Z"  
6 }  
7
```

Απόκριση:

```
1  {
2    "id": 0,
3    "citizen": {
4      "afm": "string",
5      "email": "string",
6      "name": "string",
7      "password": "string",
8      "surname": "string"
9    },
10   "car": {
11     "id": 0,
12     "agency": {
13       "afm": "string",
14       "email": "string",
15       "name": "string",
16       "password": "string",
17       "owner": "string"
18     },
19     "brand": "string",
20     "model": "string",
21     "fuel": "string",
22     "engine": 0,
23     "seats": 0,
24     "price": 0,
25     "quantity": 0,
26     "additionalInfo": "string"
27   },
28   "date": "2025-01-31T14:26:02.303Z",
29   "time": "2025-01-31T14:26:02.303Z"
30 }
```

16. DELETE /api/reservations/{reservationId}

- **Περιγραφή:** Διαγράφει μια κράτηση.

Απόκριση:

```
1  {
2    "id": 0,
3    "citizen": {
4      "afm": "string",
5      "email": "string",
6      "name": "string",
7      "password": "string",
8      "surname": "string"
9    },
10   "car": {
11     "id": 0,
12     "agency": {
13       "afm": "string",
14       "email": "string",
15       "name": "string",
16       "password": "string",
17       "owner": "string"
18     },
19     "brand": "string",
20     "model": "string",
21     "fuel": "string",
22     "engine": 0,
23     "seats": 0,
24     "price": 0,
25     "quantity": 0,
26     "additionalInfo": "string"
27   },
28   "date": "2025-01-31T14:26:02.303Z",
29   "time": "2025-01-31T14:26:02.303Z"
30 }
```

Περιγραφή οντοτήτων

1. Car (Αυτοκίνητο)

- Περιγραφή:** Αντιπροσωπεύει τα αυτοκίνητα που διαχειρίζονται οι αντιπρόσωποι.

Πεδία:

Πεδίο	Τύπος Δεδομένων	Περιγραφή
id	Long	Μοναδικό αναγνωριστικό αυτοκινήτου.
brand	String	Μάρκα αυτοκινήτου (π.χ., Toyota).
model	String	Μοντέλο αυτοκινήτου (π.χ., Corolla).
fuel	String	Τύπος καυσίμου (π.χ., Petrol).
engine	Integer	Κυβισμός κινητήρα σε κυβικά εκατοστά.
seats	Integer	Αριθμός θέσεων.
price	Double	Τιμή πώλησης του αυτοκινήτου.
quantity	Integer	Διαθέσιμο απόθεμα.
additionalInfo	String	Πρόσθετες πληροφορίες.
agency	User	Συσχετισμένος αντιπρόσωπος (Many-to-One).

- Σχέσεις:**
 - Many-to-One** με την οντότητα User (Agency).

2. User (Χρήστης)

- Περιγραφή:** Αντιπροσωπεύει έναν χρήστη του συστήματος, είτε ως πολίτη είτε ως αντιπρόσωπο.

Πεδία:

Πεδίο	Τύπος Δεδομένων	Περιγραφή
afm	String	ΑΦΜ του χρήστη (μοναδικό).
email	String	Email χρήστη.
password	String	Κωδικός πρόσβασης χρήστη.
name	String	Όνομα χρήστη.
customField	String	Όνομα αντιπροσώπου ή επώνυμο πολίτη.
roles	Set<UserRole>	Ρόλοι χρήστη (CITIZEN, AGENCY).

- **Σχέσεις:**
 - **One-to-Many** με την οντότητα Car (Agency → Αυτοκίνητα).
 - **One-to-Many** με την οντότητα Reservation (Citizen → Κρατήσεις).

3. Reservation (Κράτηση)

- **Περιγραφή:** Αντιπροσωπεύει μια κράτηση test drive.

Πεδία:

Πεδίο	Τύπος Δεδομένων	Περιγραφή
id	Long	Μοναδικό αναγνωριστικό κράτησης.
car	Car	Το αυτοκίνητο που κρατήθηκε.
citizen	User	Ο πολίτης που έκανε την κράτηση.
date	Date	Ημερομηνία κράτησης.
time	Time	Ωρα κράτησης.

- **Σχέσεις:**
 - **Many-to-One** με την οντότητα User (Citizen).
 - **Many-to-One** με την οντότητα Car.