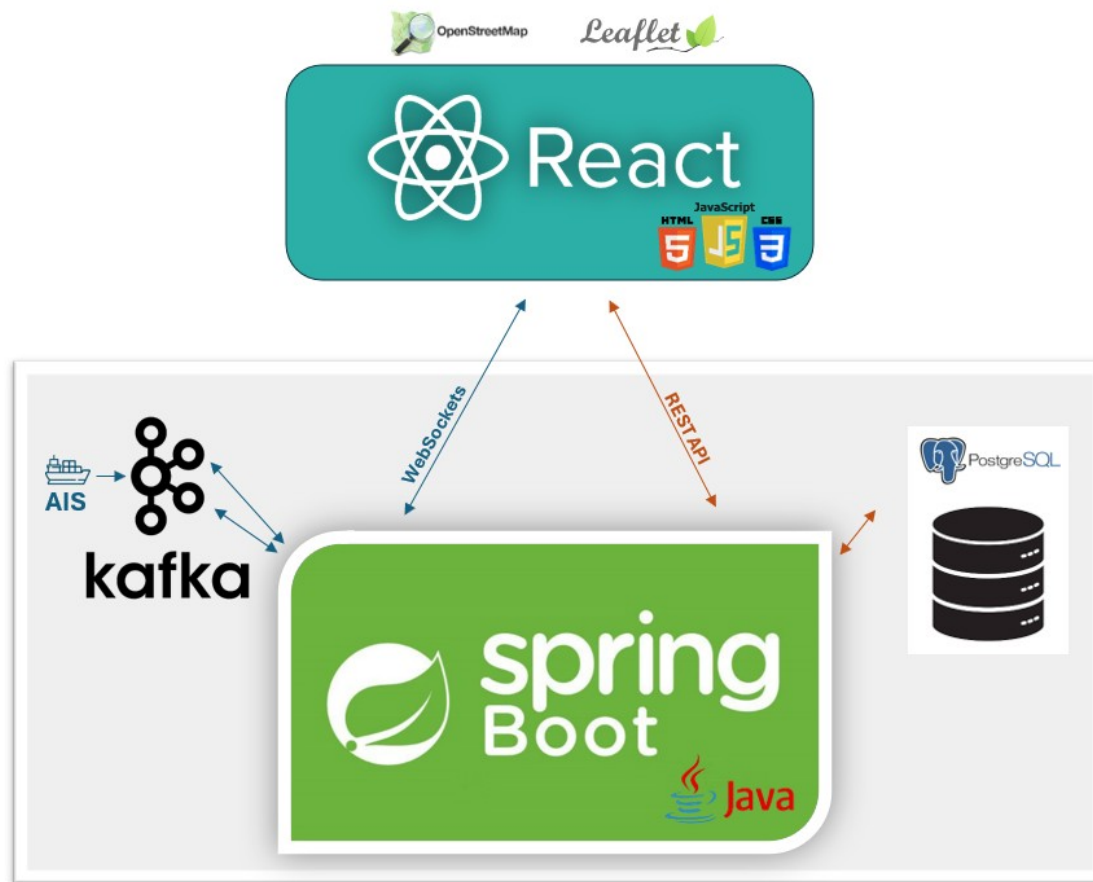


# System Modeling – Software Design

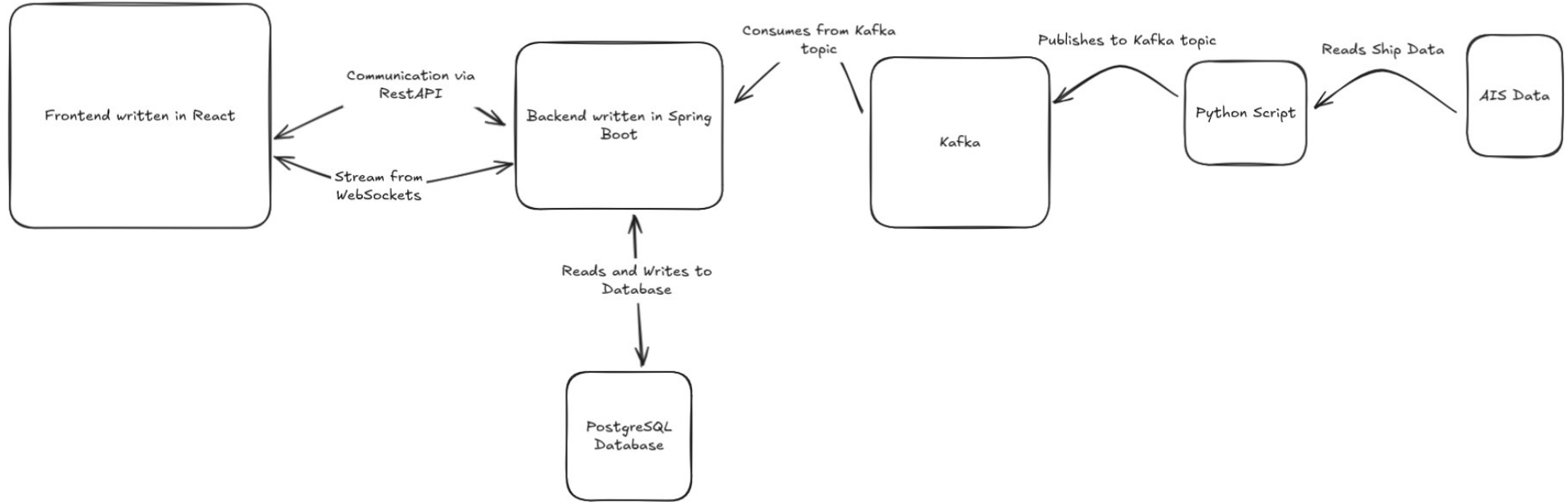
## MarineTracker

Μοντελοποίηση του "ΠΩΣ" θα υλοποιηθούν οι απαιτήσεις που έχουν προδιαγραφεί στο software-requirements-specification (srs.md).

# Σχεδιάγραμμα Βασικών Τεχνολογιών



# Σχεδιάγραμμα Βασικής Αρχιτεκτονικής



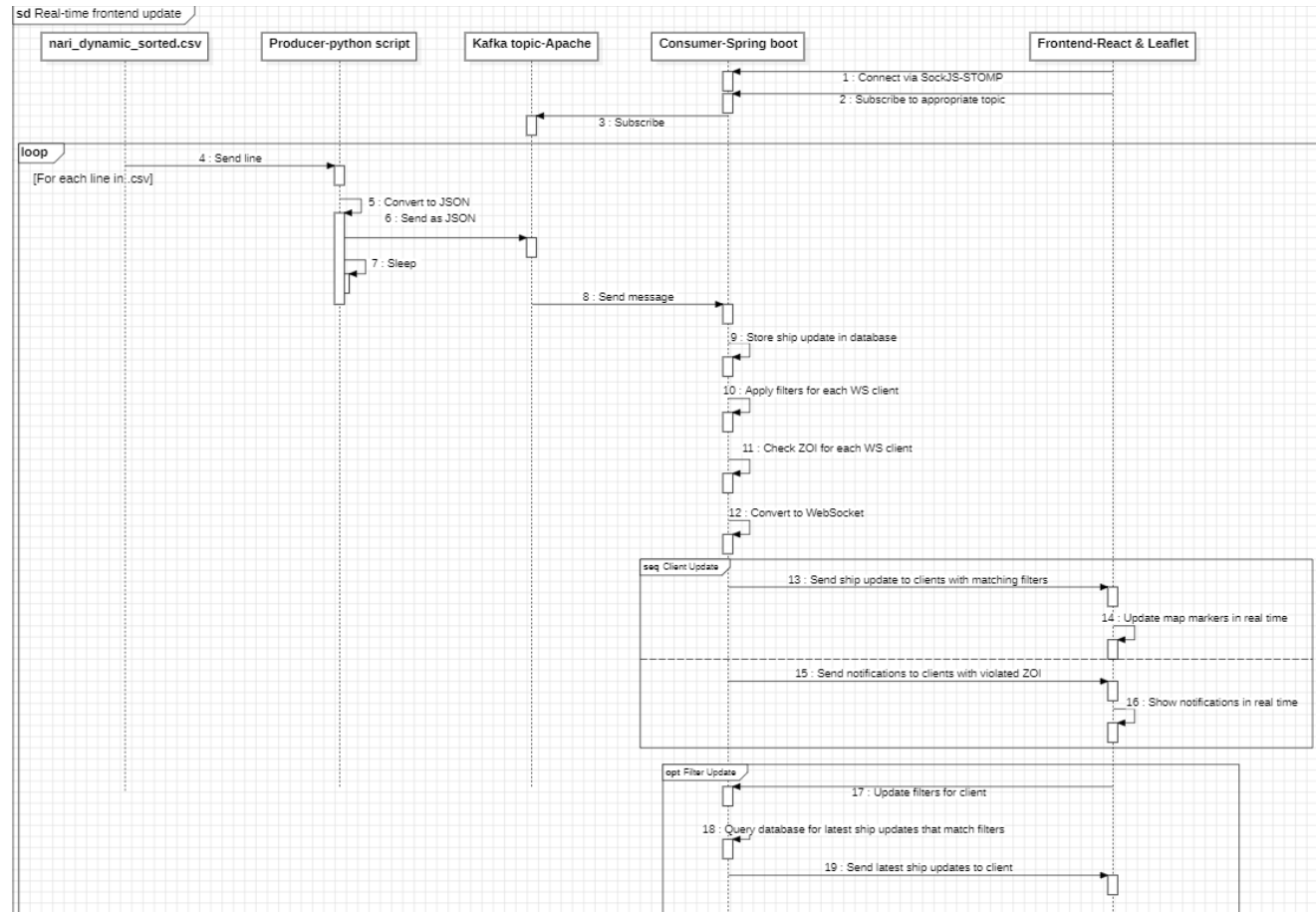
# Αυθεντικοποίηση & Ρόλοι

## Json Web Token

- Μοναδικό για κάθε χρήστη
- Δημιουργείται κατά το login
- Αποθηκεύεται στο localStorage του browser
- Αποστέλλεται με κάθε αίτηση

# Ζωντανή Προβολή Πλοίων

- 1.Producer  
(Python script)
- 2.Kafka Topic
- 3.Consumer  
(Spring Boot)
- 4.WebSocket
- 5.Frontend  
(React + Leaflet)

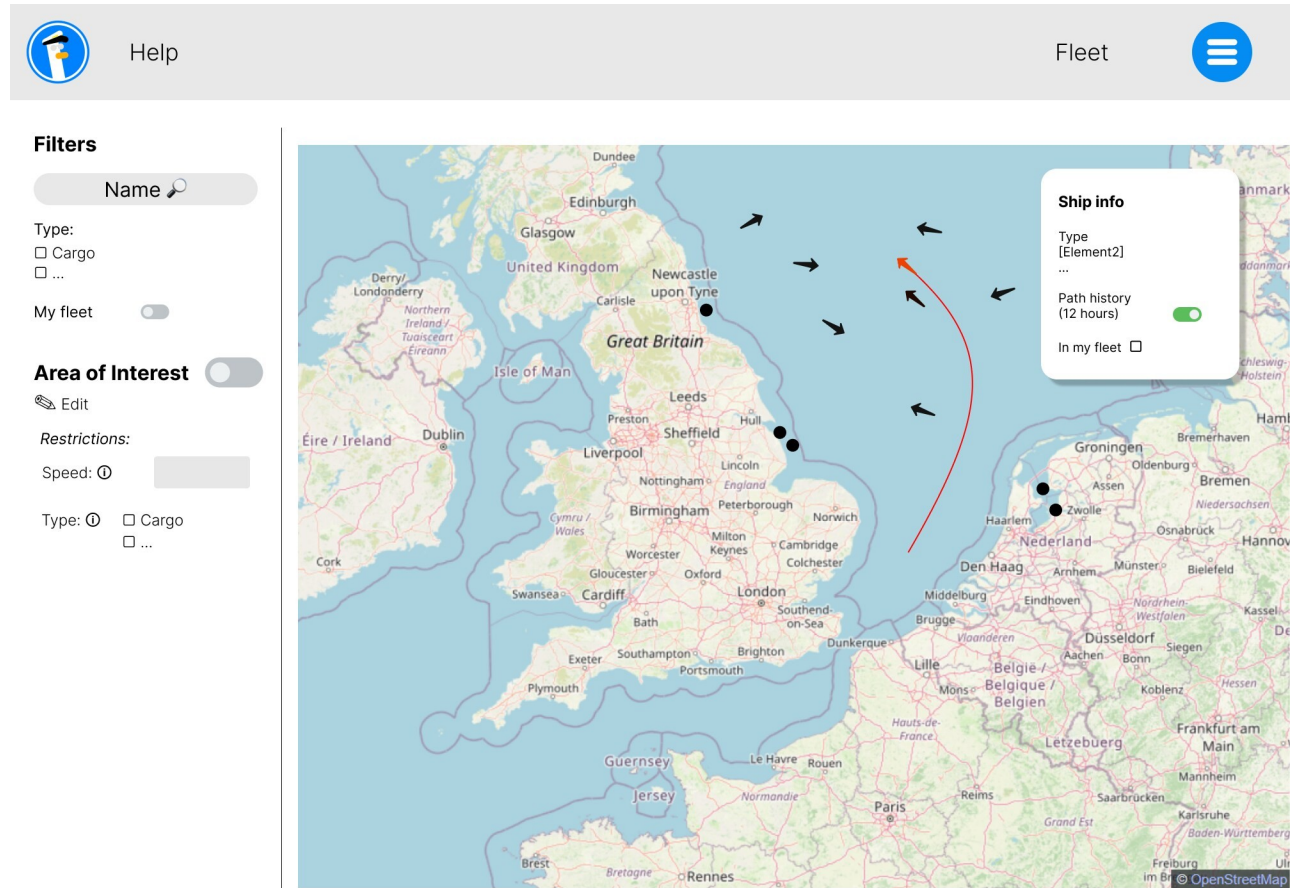


# Ασφάλεια

- HTTPS / WSS self-signed
  - Δημιουργία certificate με *keytool*
  - Springboot το φορτώνει μέσω *.p12*
  - Σε **όλα** τα endpoints

# Προβολή Πορείας Πλοίου

- Αίτημα πορείας μέσω GET στο REST API
- Leaflet.js για σχεδίαση στον χάρτη



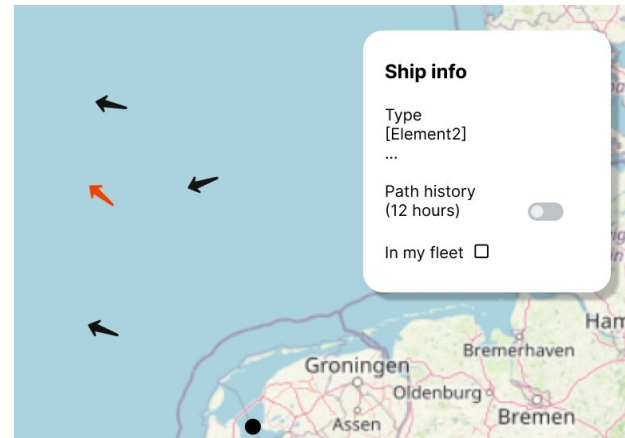
# Διαχείριση Στόλου

## 1. Καρτέλα Στόλου στο Navbar

- router
- GET το fleet του χρήστη
- παρουσίαση σε λίστα
- κουμπί αφαίρεσης απο στόλο

## 2. Λεπτομέρειες πλοίου στον Χάρτη

- GET λεπτομερειών
- κουμπί προσθήκης / αφαίρεσης απο στόλο





# Φιλτράρισμα Πλοίων


- 1) Αποστολή νέων φίλτρων μέσω websocket στο backend
- 2) Αποθήκευση φίλτρων του client στην μνήμη
- 3) Επαναποστολή όλων των πλοίων που πληρούν τα φίλτρα
- 4) Κανονική ροή με νέα φίλτρα

# Αναζήτηση Πλοίων

- Μπάρα αναζήτησης με βάση το όνομα
- Με την χρήση του Enter γίνεται GET request στο REST API, full text search στην βάση δεδομένων
- Η απόκριση αποτελεί λίστα με πλοία που πληρούν το κριτήριο του ονόματος, σε μορφή dropdown κάτω από την μπάρα αναζήτησης
- Όταν ο χρήστης επιλέξει πλοίο, το frontend το κεντράρει στον χάρτη, και εμφανίζεται πλαίσιο στα δεξιά με τις πληροφορίες του


# Διαχείριση Ζώνης Ενδιαφέροντος

- Section στο sidebar
- Ορισμός / Επεξεργασία περιοχής με leaflet και PUSH στο REST api
- Αποθήκευση Ζώνης στο database
- PostGIS στο backend για φιλτράρισμα χωρικών δεδομένων


 Help

Admin

Fleet




**Filters**

Name 

Type:  
☐ Cargo  
☐ ...


My fleet ☐

**Area of Interest** ☒

 Save

Restrictions:  
Speed:

Type: ☒ Cargo  
☐ ...



# Notifications Ζώνης Ενδιαφέροντος

- Φιλτράρισμα περιοχής με Post GIS
- Αποστολή μέσω websocket
- Καμπάνα στο Navbar εμφανίζει notification λίστα
  - dismiss all
  - dismiss διπλά σε κάθε notification

# Εξαγωγή Στατικών Δεδομένων

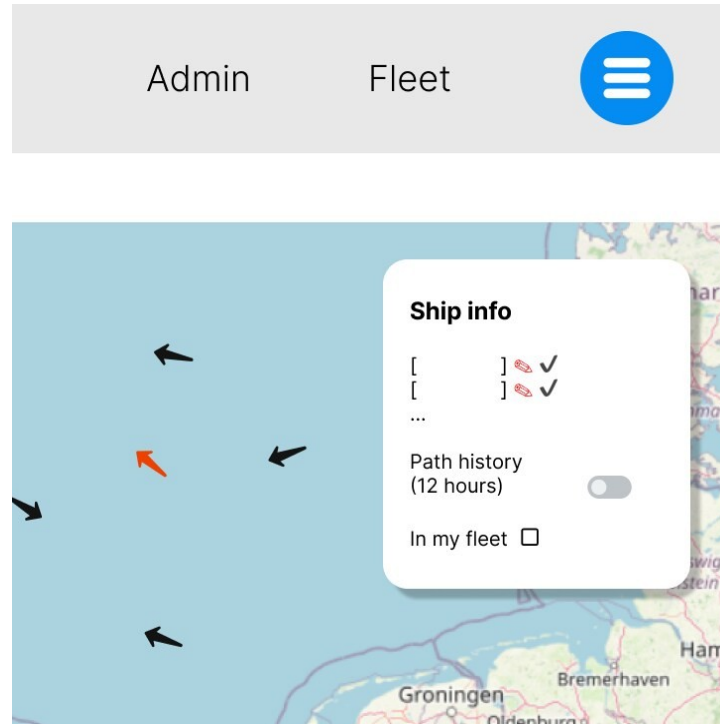
Navbar > Admin Καρτέλα:

- Κουμπί λήψης κάνει GET στο REST
- Λήψη .json με όλα τα στατικά δεδομένα

# Επεξεργασία Στατικών Δεδομένων

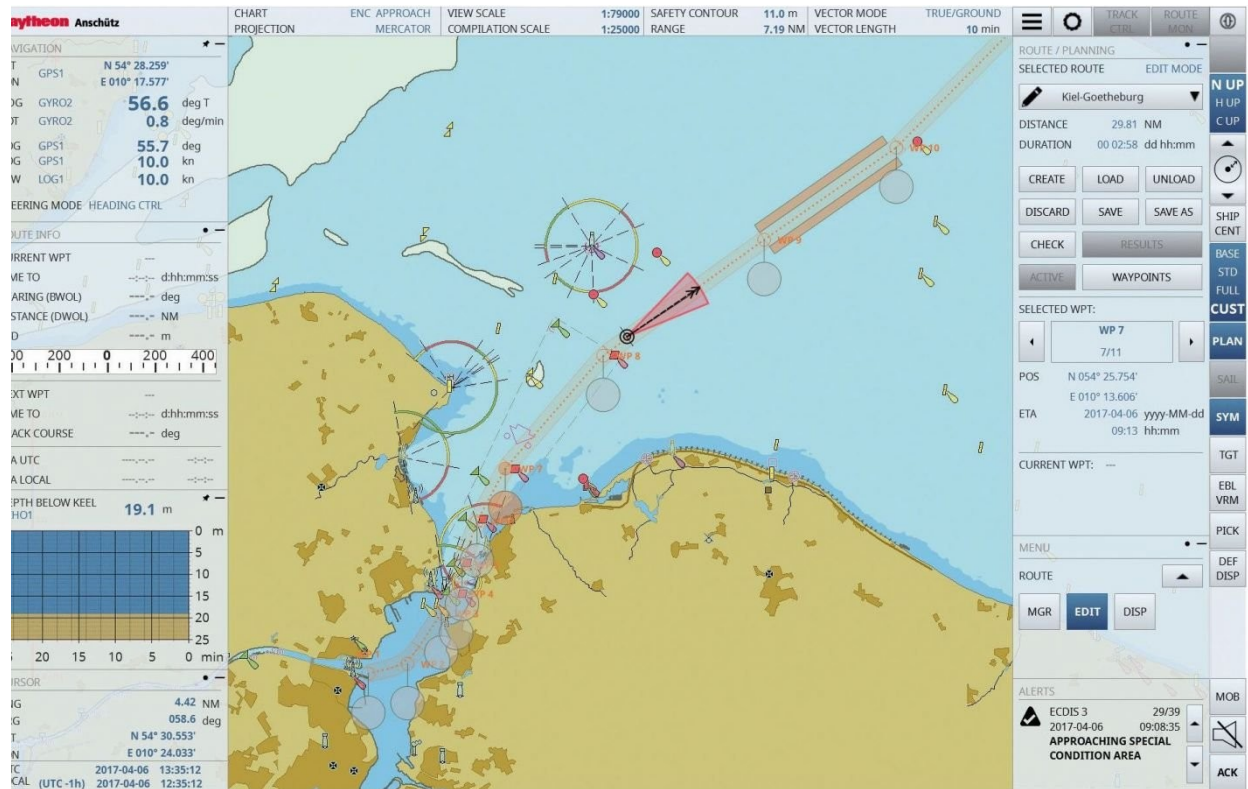
Χάρτης > Πλοίο >  
Λεπτομέρειες:

- Πεδία επεξεργασίας κάνουν
  - GET όρια [min, max] / επιλογές τιμών πεδίου
  - POST στο REST με τη νέα τιμή
- Ενημερώνεται η βάση



# Φιλικότητα προς το Χρήστη

- Οι παρόμοιοι με ήδη υπάρχοντα εργαλεία
- Λίγα κρυμμένα στοιχεία (hamburger menus)
- Σημειώσεις χρηστών ανά πλοίο
- Εύχρηστη Αναζήτηση
  - **SCM** **UM**  
να βρίσκει το **MSC MUMBAI VIII**
  - **KATERINA** **ST**  
να βρίσκει το **ST KATERINA**



# Κλιμακωσιμότητα

- Το backend στέλνει μόνο πλοία εντός την οθόνης
- Δεν κρατάμε όλα τα δεδομένα για το ιστορικό πορείας. Sampling ανά κάποιο  $\Delta t$ .
- Filtering στο backend που έχει περισσότερους πόρους: Δεν επιβαρύνεται ο client.