

1059633

Νικηφόρος-Γεώργιος Παπαγεωργίου
st1059633@ceid.upatras.gr

Αλέξανδρος Ξιάρχος
st1059619@ceid.upatras.gr

1059619

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ · ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ & ΣΥΣΤΗΜΑΤΑ ΣΤΟ ΠΑΓΚΟΣΜΙΟ ΙΣΤΟ

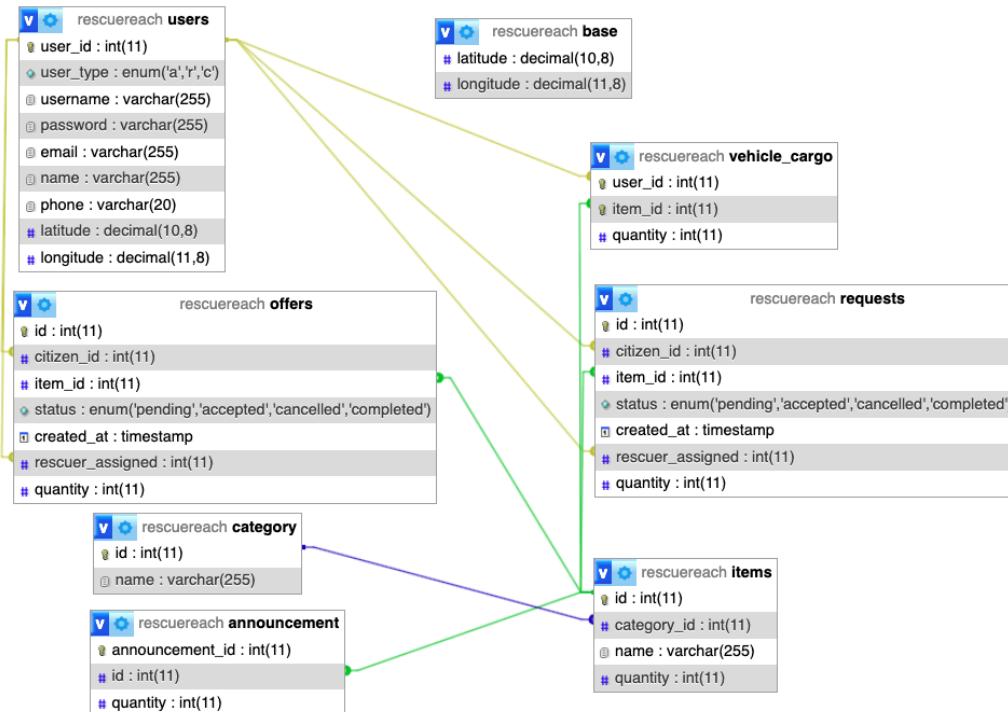
ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ · 2023 – 2024

ΠΕΡΙΕΧΟΜΕΝΑ

1	ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΗΣ	2
2	ΥΛΟΠΟΙΗΣΗ	2
2.1	LOGIN	2
2.1.1	login.php	3
2.2	REGISTER	3
2.2.1	signup.js	4
2.2.2	register_user.php	4
2.3	ADMIN	5
2.3.1	Dashboard	5
2.3.1.1	Quick Summary	5
2.3.1.2	Create Announcement	5
2.3.1.3	Create Rescuer Account	6
2.3.1.4	Announcements Summary	6
2.3.1.5	Graphs	6
2.3.2	Base	7
2.3.2.1	Add an item	8
2.3.2.2	Upload	8
2.3.3	Map	9
2.4	RESCUER	10
2.4.1	Home	10
2.4.2	My Tasks	12
2.4.3	Items in Car	13
2.4.4	Items in Base	14
2.5	USER	15
2.5.1	Home	15

1 ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΗΣ

Η σχεσιακή βάση μας είναι υλοποιημένη σε περιβάλλον XAMPP χρησιμοποιώντας MariaDB. Παρατίθεται το στιγμιότυπο που εξάχθηκε από το γραφικό περιβάλλον του PhpMyAdmin:



Σχήμα 1.1: Entity-Relationship Diagram

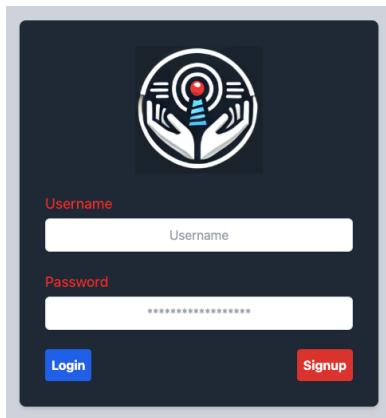
2 ΥΛΟΠΟΙΗΣΗ

Η εφαρμογή μας σχεδιάστηκε και υλοποιήθηκε κυρίως με χρήση `vanilla` τεχνικών, όπως `PHP` και `JavaScript`. Για τον χειρισμό των αιτημάτων προς τον διακομιστή, χρησιμοποιήθηκε η βιβλιοθήκη `jQuery`, η οποία διευκολύνει την αποστολή αιτημάτων στην `PHP` μέσω της `AJAX`.

Επιπλέον, για την κατασκευή των χαρτών στο `frontend`, αξιοποιήθηκε η βιβλιοθήκη `Leaflet`, που παρέχει διαδραστικούς και ευέλικτους χάρτες. Τέλος, για το `responsive design` της εφαρμογής χρησιμοποιήσαμε `Tailwind CSS` με `Flexbox` ώστε να έχουμε σωστή στοιχειοθέτηση σε πληθώρα αναλύσεων και συσκευών.

2.1 LOGIN

Ως `index.html` ορίζεται η σελίδα `Login`, στην οποία γίνεται η σύνδεση ή η εγγραφή των χρηστών-πολιτών.



Σχήμα 2.1: Φόρμα Login

Στη βάση δεδομένων έχουν ήδη δημιουργηθεί κάποιοι λογαριασμοί ποικιτών:

	Επεξεργασία	Αντιγραφή	Διαγραφή	user_id	user_type	username	password	email	name	phone
<input type="checkbox"/>	Επεξεργασία	Αντιγραφή	Διαγραφή	1	a	admin	admin	NULL	NULL	NULL
<input type="checkbox"/>	Επεξεργασία	Αντιγραφή	Διαγραφή	2	r	rescuer	rescuer	rescuer@gmail.com	rescuer	6912345567
<input type="checkbox"/>	Επεξεργασία	Αντιγραφή	Διαγραφή	3	c	nikf	nikhforos1!	nik@gmail.com	nikhforos	6945123321
<input type="checkbox"/>	Επεξεργασία	Αντιγραφή	Διαγραφή	4	r	rescuer2	rescuer!@#\$%	rescuer2@gmail.com	rescuer2	6912345678
<input type="checkbox"/>	Επεξεργασία	Αντιγραφή	Διαγραφή	7	r	rescuer3	rescuer3!	rescuer3@gmail.com	rescuer3	6912345566
<input type="checkbox"/>	Επεξεργασία	Αντιγραφή	Διαγραφή	8	c	alex	alex!@	alex@hotmail.com	Alexandros	6912345565
<input type="checkbox"/>	Επεξεργασία	Αντιγραφή	Διαγραφή	9	r	rescuer4	rescuer4!	rescuer4@gmail.com	rescuer4	6234221312
<input type="checkbox"/>	Επεξεργασία	Αντιγραφή	Διαγραφή	10	c	Aerakis	Aerakis	aerakis@gmail.com	Giannis Aerakis	6969696969
<input type="checkbox"/>	Επεξεργασία	Αντιγραφή	Διαγραφή	11	c	skasselakis	kassssss	skasselakis@gov.com	Stefanos Kasselakis	6911111111
<input type="checkbox"/>	Επεξεργασία	Αντιγραφή	Διαγραφή	12	c	Blondie420	TonYpeftino	karen000@gmail.com	Karen	6987654300

Σχήμα 2.2: πίνακας users

Η συνάρτηση login δέχεται τα ορίσματα των input fields με id = username και id = password, ελέγχεται αν έχουν πληροφορία [αν δεν έχουν, στέλνεται alert μέσω της SweetAlert], και αν έχουν, στέλνονται με ένα AJAX request στο login.php.

Σε περίπτωση επιτυχίας καλείται η handleSuccess(). Αυτή, στο αντικείμενο που επιστρέφεται¹, ελέγχει το user_type και κάνει το αντίστοιχο redirect [με ένα delay 2 sec] στην επόμενη σελίδα.

2.1.1 login.php

Αφού ανακτήσει το username και password από το AJAX request, στέλνει query στη βάση δεδομένων βάσει του username. Αν βρει, ελέγχει εξίσου και το password και στη συνέχεια δημιουργεί ένα array με τις πληροφορίες {username, user_id και user_type} του χρήστη, και ένα cookie με διάρκεια 30 ημερών.

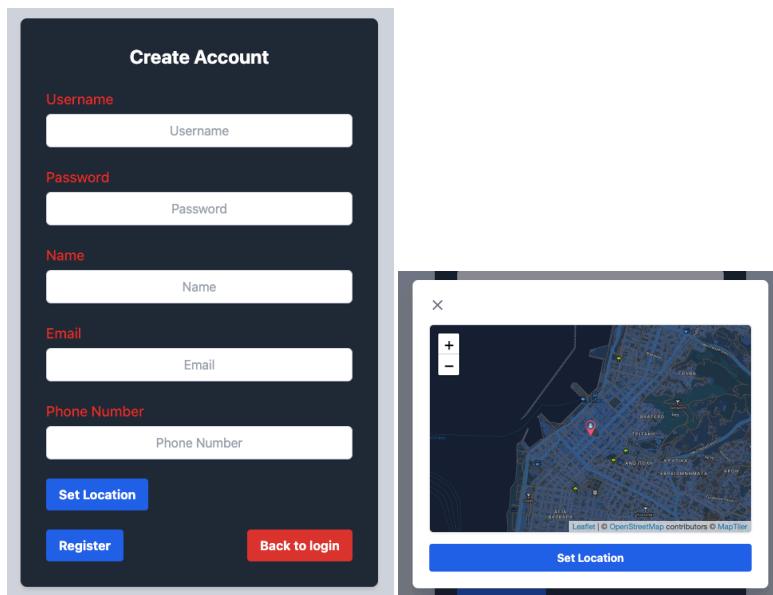
Εν τέλει, στέλνεται με echo ένα JSON αρχείο της μορφής

```
[{"username": <...>, "user_id": <...>, "user_type": <"a" ή "r" ή "c">}]
```

2.2 REGISTER

Στη σελίδα signup.html ορίζεται η φόρμα εγγραφής των χρηστών-ποικιτών:

¹redundant, αλλά ελέγχεται αν όντως επιστρέφεται αντικείμενο· αν όχι εμφανίζεται error



Σχήμα 2.3: Σελίδα Signup και modal προσδιορισμού τοποθεσίας

2.2.1 signup.js

Η συνάρτηση `signup()` δέχεται τα ορίσματα των input fields, ελέγχει αν έχουν πληροφορία [αν δεν έχουν, στέλνεται alert], και αν έχουν, στέλνει ένα AJAX request στο `register_user.php`.

Οι συναρτήσεις `openModal()` και `closeModal()`, εμφανίζουν και αποκρύπτουν το modal προσδιορισμού τοποθεσίας, το οποίο εμφανίζεται πατώντας στο κουμπί **Set Location**. Αυτό επιτυγχάνεται με την κατάλληλη αφαίρεση και προσθήκη κλάσεων του Tailwind CSS.

Στην εμφάνιση του **Set Location**, αρχικοποιείται ο χάρτης μέσω της `initializeMap()`. Ορίζεται η σταθερά `map` μέσω της Leaflet, με κεντραρισμένο `view` στο κέντρο της Πάτρας. Ορίζεται ο χάρτης που θα χρησιμοποιήσουμε ως συγκεκριμένο `tileLayer` (ο οποίος χάρτης εξάγεται μέσω API από το [maptiler.com](#)), και προστίθεται ένα `marker` (`userIcon` που αντιστοιχεί στην εικόνα `user.png`), το οποίο αρχικοποιείται με αρχική θέση στο κέντρο της Πάτρας [ίδιο `location` με την αρχικοποίηση του `view`]. Ελέγχεται το event του `click`, και σε εκείνη την περίπτωση αποθηκεύονται οι αλλαγμένες συντεταγμένες ως `value` στο `input` με `id = location`.

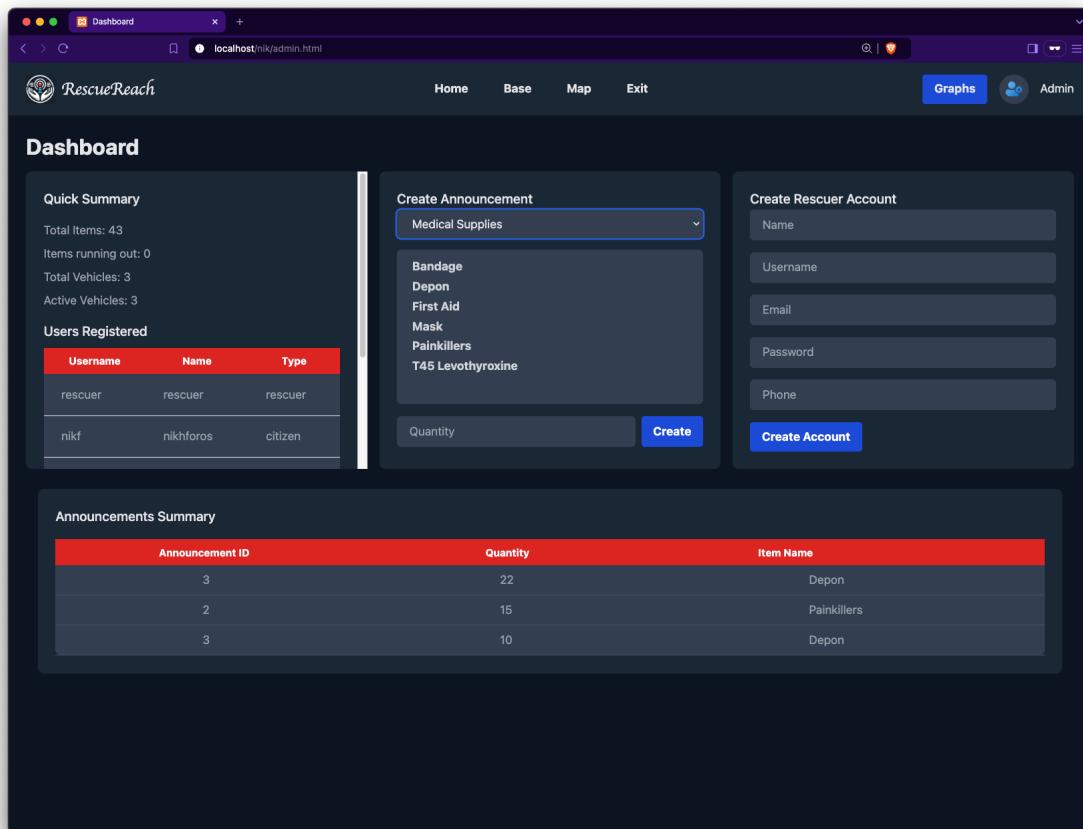
2.2.2 register_user.php

Αφού ανακτηθούν οι πληροφορίες, γίνονται `insert` στη βάση δεδομένων και γίνεται `echo` μήνυμα επιτυχίας. Στην περίπτωση της μεταβλητής της τοποθεσίας, είναι αναγκαία η διάσπασή της σε ξεχωριστές μεταβλητές για το `latitude` και `longitude`.

2.3 ADMIN

2.3.1 Dashboard

Η κεντρική σελίδα του διαχειριστή περιλαμβάνει τέσσερα διαφορετικά modules: ένα με μια καταγραφή των items, των vehicles και των χρηστών, ένα όπου μπορούμε να δημιουργήσουμε ανακοινώσεις, ένα που μπορούμε να δημιουργήσουμε ένα λογαριασμό διασώστη, και ένα με την καταγραφή όλων των ανακοινώσεων.



Σχήμα 2.4: Dashboard admin

Το αρχείο `dashboard.js` χειρίζεται τη συγκεκριμένη σελίδα. Στην αρχή καθούνται όμεσα οι απαραίτητες συναρτήσεις για την εμφάνιση των διαφόρων modules.

2.3.1.1 Quick Summary

Η συνάρτηση `loadDashboardSummary()` στέλνει ένα AJAX request στο `summary.php`, ζητώντας τον συνολικό αριθμό των items (`totalItems`), και τον αριθμό των items που τελειώνουν (`lowStockItems`), και επιστρέφονται σε JSON μορφή. Αντίστοιχα η `fetchActiveVehicles()` στέλνει ένα AJAX request στο `active_vehicles.php`, ζητώντας τον αριθμό των vehicles που είναι ενεργά (`activeVehicles`). Η `registeredUsers()` δέχεται δεδομένα από το `registered_users.php`, και μέσω της `populateTable()` εμφανίζει έναν πίνακα με τους εγγεγραμμένους χρήστες.

2.3.1.2 Create Announcement

Για τη φόρτωση των διαφορετικών κατηγοριών των items για τη δημιουργία της ανακοίνωσης, καθείται αρχικά η `loadCategoriesAndItems()` η οποία στέλνει ένα AJAX request στο `announchment.php`. Η PHP επιστρέφει ένα JSON αρχείο της μορφής:

```
{"categories": [id, name], "items": [id, name, category_id]}
```

Τα δεδομένα που επιστρέφονται χρησιμοποιούνται από τη `loadDashboardSummary()` για να γίνουν `populate` οι κατηγορίες της φόρμας. Αν ο χρήστης επιλέξει μια κατηγορία, η `filterItemsByCategory()` κάνει `populate` τη λίστα με τα αντίστοιχα items μέσω του JSON αρχείου.

Όταν πατηθεί το κουμπί **Create**, αποθηκεύονται σε μεταβλητές τα επιλεγμένα items, όπως επίσης και το πλήθος τους και στέλνονται στη `insert_announcement.php`, η οποία τα κάνει `insert` στη βάση. Σε περίπτωση επιτυχίας στέλνεται το αντίστοιχο μήνυμα.

2.3.1.3 Create Rescuer Account

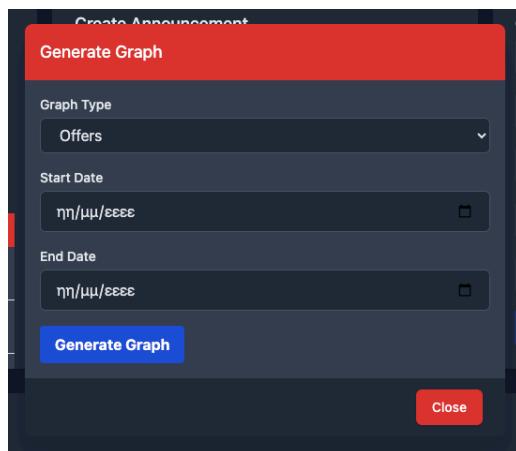
Όταν πατηθεί το κουμπί **Create Account**, καλείται η `createRescuerAccount()` η οποία μέσω Promise δημιουργεί μεταβλητές για κάθε value της φόρμας, τις οποίες στέλνει μέσω AJAX request στο `create_rescuer.php`. Η PHP ελέγχει αν υπάρχει ήδη τέτοιο username ή email, στέλνει κατάλληλα echos αν υπάρχουν, και αν δεν υπάρχουν κάνει `insert` τον νέο χρήστη στη βάση δεδομένων.

2.3.1.4 Announcements Summary

Ακολουθείται παρόμοια διαδικασία με τη δημιουργία του πίνακα των εγγεγραμμένων χρηστών. Χρησιμοποιείται η `loadAnnouncementSummary()` σε συνδυασμό με τη `get_ann.php`.

2.3.1.5 Graphs

Το κουμπί **Graphs** ενεργοποιεί ένα modal² με στατιστικά γραφήματα για τα requests και τα offers.



Σχήμα 2.5: Generate Graph model

Δημιουργούνται οι σταθερές `startDate`, `endDate`, και `graphType`. Μέσω της `fetchGraphData()` δημιουργείται ένα Promise ανάλογα με το αν το `graphType` είναι `offer` ή `request` προς τα `graph_offers.php` ή `graph_requests.php?startDate=${startDate}&endDate=${endDate}`.

Με το JSON αρχείο που επιστρέφεται σε κάθε περίπτωση, η `fetchGraphData()` μέσω της ChartJS δημιουργεί το αντίστοιχο γράφημα.

²με κατάλληλη προσθαφάριση κλάσεων του Tailwind

2.3.2 Base

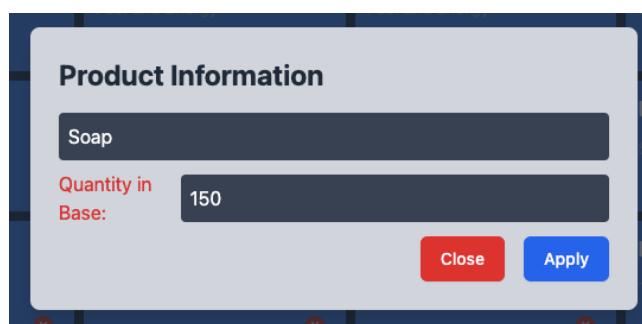
The screenshot shows a web-based application titled "Item Management" with a URL of "localhost/milk/base.html". The interface includes a sidebar with "RescueReach" logo and navigation links: Admin, Home, Base (selected), Map, and Exit. At the top right are buttons for "Search products...", "Add an Item", and "Upload". A dropdown menu "Filter by Category" is set to "all categories". The main area displays a 5x5 grid of product cards. Each card contains a product name, category, and a red delete icon. The categories represented are: Medical Supplies (T45 Levothyroxine, Painkillers, Depon, Bandage, First Aid); Special Items (Mask, Power Banks, Gasoline, Home Repair Tools, Kitchen Appliances); Personal Hygiene (Cleaning Supplies, Cutlery, Cups, Plates, Pots); Tools and Equipment (Toothbrush, Diapers, Towels, Toothpaste, Soap); Clothing and Cover (Repair Tools, Batteries, Flashlight, Gloves, Caps); Food (Socks, Shoes, Pants, Shirt, Blankets); Navigation Tools (Cereals, Snacks, Canned Goods, Water, GPS); and Sleep Essentials (Map, Compass, Earplugs, Heater, Matches).

Σχήμα 2.6: Σελίδα με τα αντικείμενα της Βάσης

Το αρχείο `base.js` χειρίζεται τη σελίδα `base.html`. Κατά την εμφάνιση της σελίδας δημιουργείται ένα AJAX request στο `item_categories.php`, το οποίο επιστρέφει όλες τις κατηγορίες και τα προϊόντα της βάσης. Συγκεκριμένα επιστρέφεται ένα JSON αρχείο της μορφής:

```
{"categories": [...], "products": [id, name, category_id, category]}
```

To population της σελίδας με τα αντικείμενα πραγματοποιείται μέσω της `updateProductGrid(products)`, η οποία περιέχει μια επανάληψη όπου προσθέτει divs στη HTML σελίδα. Περιέχονται Event Listeners για το hover και το click των διαφορετικών αντικειμένων.

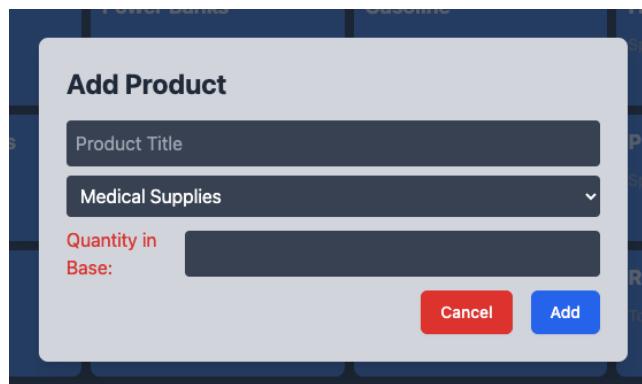


Σχήμα 2.7: Edit modal

Όταν κάνουμε hover εμφανίζεται ένα tooltip **Edit**, και αν το κλικάρουμε καλούνται οι πληροφορίες του αντικειμένου μέσω της `product_info.php`, και μπορούμε να τις τροποποιήσουμε. Η τροποποίηση πραγματοποιείται μέσω της `onProductFound()`, η οποία εμφανίζει το αντίστοιχο modal και καλεί την `update_product.php`.

Μέσω του κόκκινου κουμπιού X, το αντικείμενο διαγράφεται. Για να διαγραφτεί ένα αντικείμενο, καλείται με AJAX το `delete_product.php`.

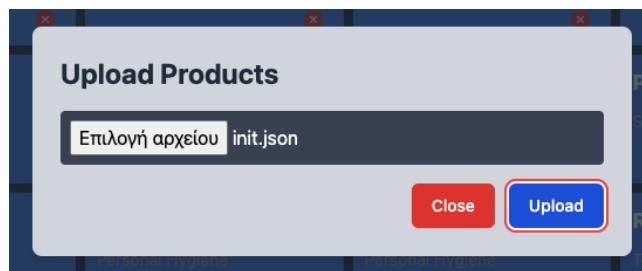
2.3.2.1 Add an item



Σχήμα 2.8: Add product modal

Το κουμπί **Add an item** ενεργοποιεί ένα modal με μια φόρμα για προσθήκη νέου προϊόντος. Για την επιλογή της κατηγορίας του προϊόντος χρησιμοποιείται το προηγούμενο JSON αρχείο, και όταν το καθορίσουμε και πατήσουμε **Add**, εκτελείται το `insert_product.php`. Μετά την προσθήκη του προϊόντος, γίνεται update το περιεχόμενο της βάσης εκτελώντας ξανά το `item_categories.php`.

2.3.2.2 Upload

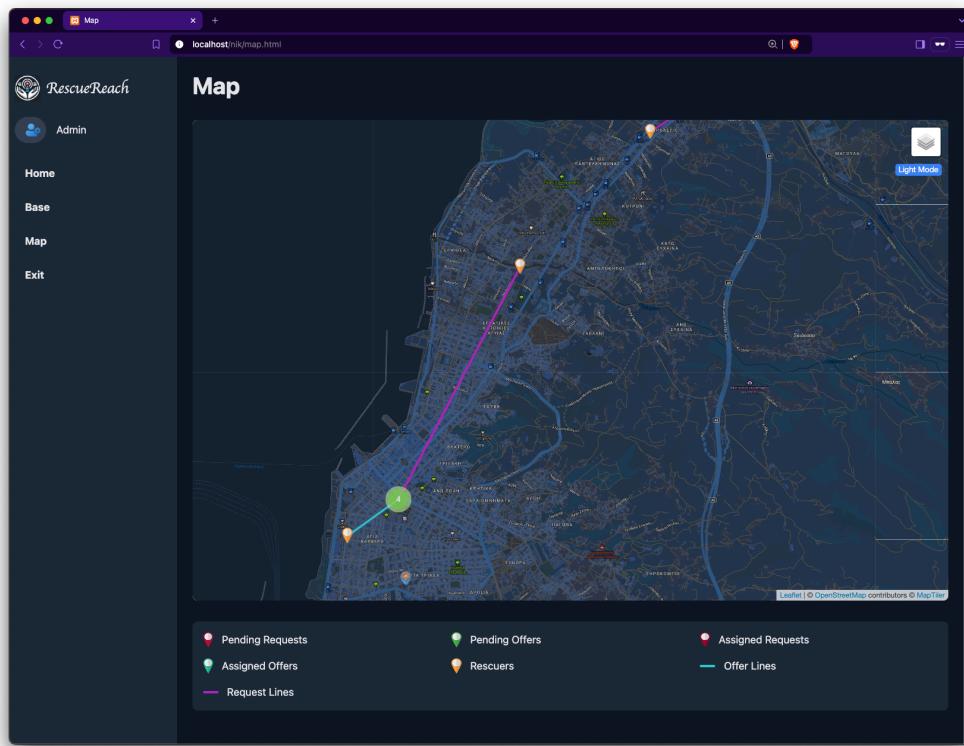


Σχήμα 2.9: Upload JSON modal

Το κουμπί **Upload** μέσω της `upload.js` ενεργοποιεί ένα modal, στο οποίο μπορούμε να ανεβάσουμε ένα JSON αρχείο και να αρχικοποιήσουμε οιλόκληρη τη βάση. Υπάρχει ένα Event Listener που ενεργοποιείται με το κλικ στο κουμπί **Upload**, το αρχείο αποθηκεύεται σε έναν reader, και γίνεται ένα AJAX request στο `upload.php`. Η PHP ελέγχει αν το προϊόν υπάρχει ήδη [ελέγχοντας το `id` του], και αν δεν υπάρχει, τη προσθέτει.

2.3.3 Map

Η σελίδα περιλαμβάνει μια οπτική αναπαράσταση των Requests, των Offer και των Rescuers σε έναν χάρτη.

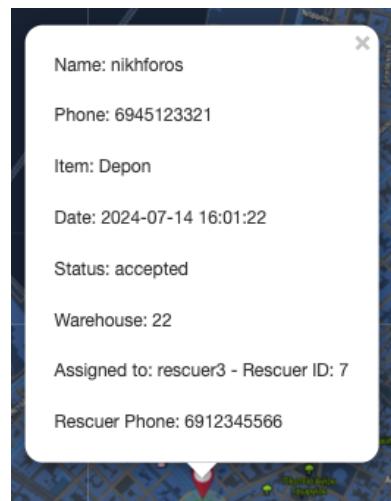


Σχήμα 2.10: Χάρτης admin

Κατά τη φόρτωση της σελίδας, στο map.js τρέχουν οι συναρτήσεις getOffers, getRequests, getRescuerData και getBaseLocation, οι οποίες στέλνουν AJAX requests σε αντίστοιχα αρχεία PHP, για την ανάκτηση των πληροφοριών από τη βάση. Επιπλέον, η getBaseLocation αρχικοποιεί ένα μετακινούμενο Leaflet marker στο χάρτη (που αναπαριστά την αποθήκη/βάση) στις συντεταγμένες που δέχεται από το PHP. Το marker μετακινείται μέσω της updateBaseLocation η οποία δημιουργεί τα κατάλληλα μηνύματα, και κάνει handle την ανανέωση της βάσης δεδομένων με τις νέες συντεταγμένες.

Μετά τη φόρτωση των δεδομένων, δημιουργούνται L.layerGroups markers που επίσης αντιστοιχίζονται στο legend του χάρτη. Δημιουργούνται δύο διαφορετικά tiles που αντιστοιχούν σε δύο διαφορετικούς χάρτες (light mode και dark mode), παρμένοι από το maptiler.com. Δημιουργείται ένα επιπλέον κουμπί στο legend του χάρτη μέσω της L.Control.extend για το toggle των δύο modes του χάρτη.

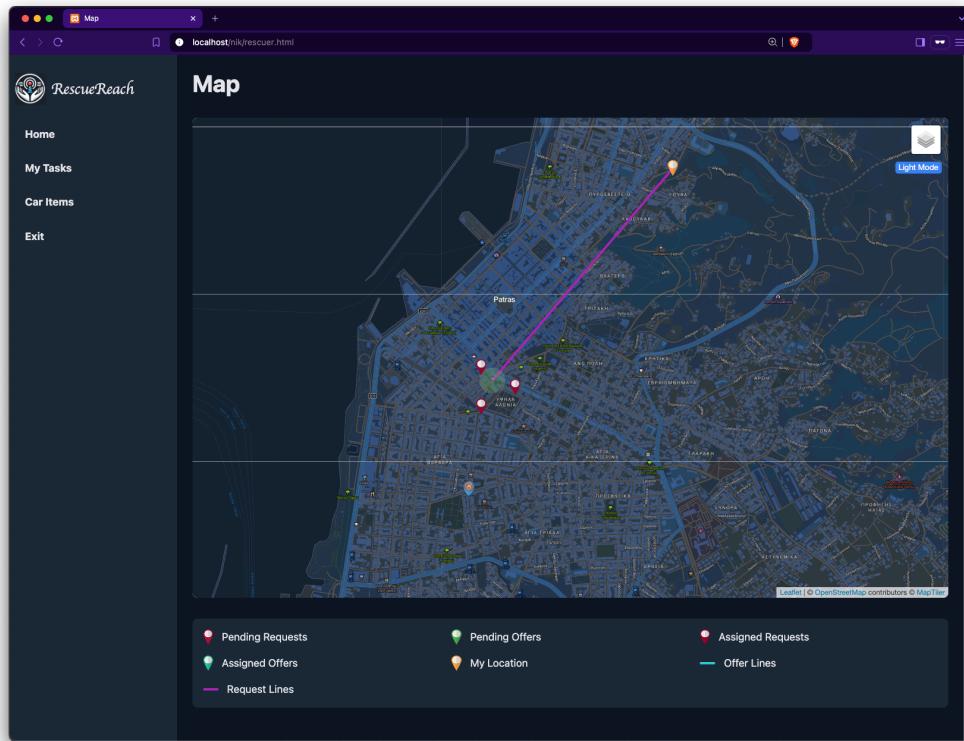
Η displayRequests() και displayOffers() κάνουν populate τον χάρτη και επίσης δημιουργούν τα συννεφάκια με πληροφορίες.



Σχήμα 2.11: Συννεφάκι marker

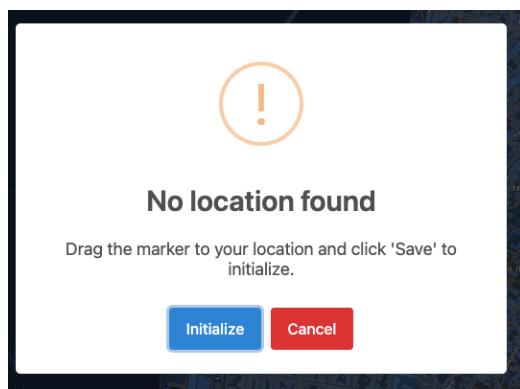
2.4 RESCUER

2.4.1 Home



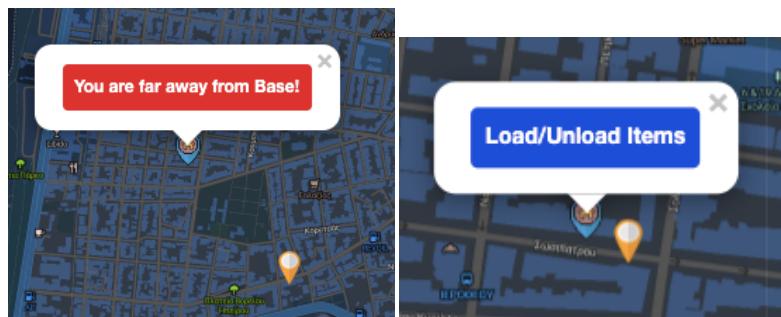
Σχήμα 2.12: Αρχική σελίδα rescuer

Η αρχική σελίδα του διασώστη περιλαμβάνει το χάρτη της πόλης, με τα markers που αντιστοιχούν στα requests και τα offers των πολιτών. Ο rescuer δεν μπορεί να δει τους υπόλοιπους rescuers. Αυτό επιτυγχάνεται μέσω της `logged_user` σταθεράς, που απομονώνει τον συγκεκριμένο χρήστη από τα δεδομένα που επιστρέφονται από τη βάση δεδομένων. Η διαδικασία εμφάνισης των markers είναι παρόμοια όπως στη σελίδα του admin.



Σχήμα 2.13: Αρχικοποίηση rescuer

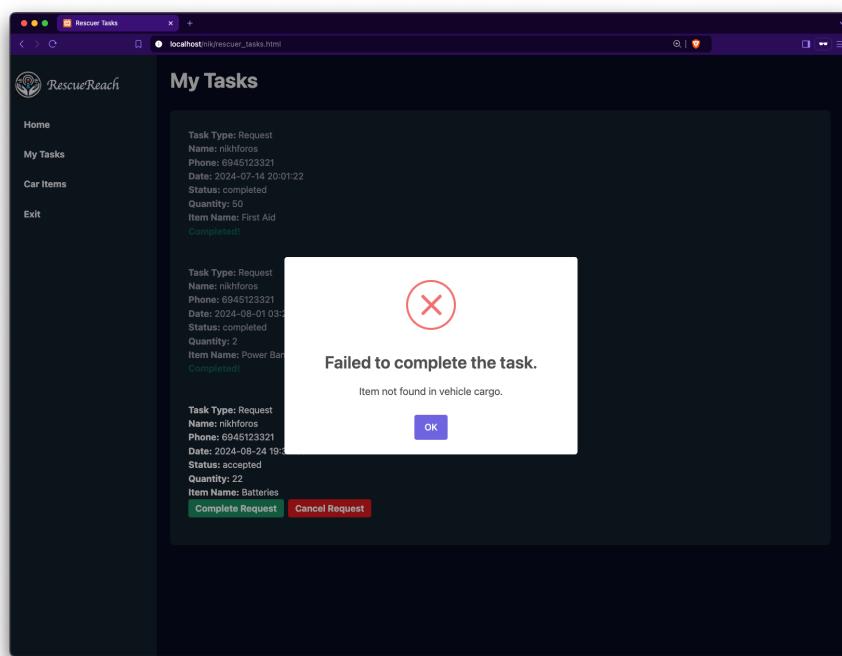
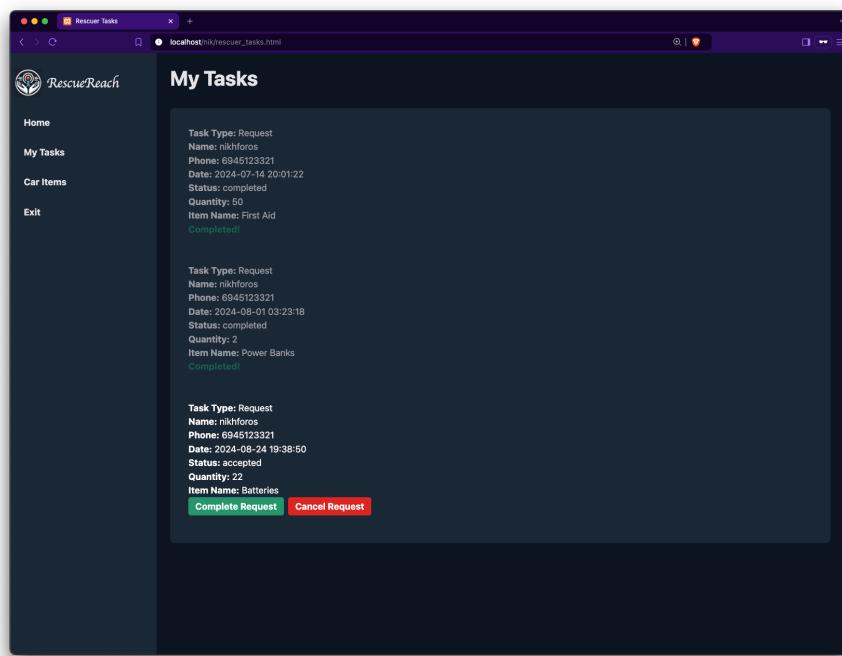
Μια διαφορά που προκύπτει στην `fetchRescuerLocation()` είναι κατά την αρχικοποίηση της τοποθεσίας του διασώστη μετά την εγγραφή του από τον admin.



Σχήμα 2.14: Ο διασώστης πρέπει να είναι σε ακτίνα 50m από τη βάση

Η `baseProximity()` ελέγχει την απόσταση του διασώστη από τη βάση και ανάλογα τροποποιεί το συννεφάκι της βάσης. Η `displayLines()` χρησιμοποιεί τις συντεταγμένες του διασώστη και των πομπιτών και δημιουργεί γραμμές (`L.polyline`) μεταξύ τους. Τέλος, η `takeTask()` ελέγχει με AJAX request στην `takeTask.php` πόσα tasks έχουν ήδη επιλεγεί από τους διασώστες, και αν είναι πάνω από 4, δεν επιτρέπει την επιλογή ενός νέου.

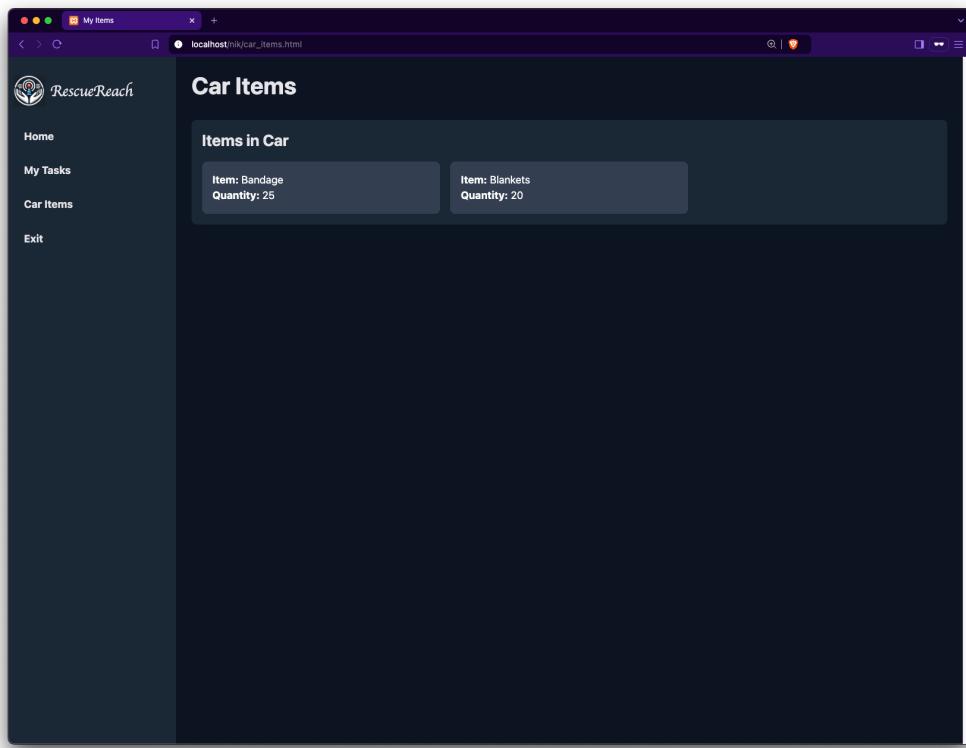
2.4.2 My Tasks



Σχήμα 2.15: Σελίδα με τα tasks του διασώστη

Κατά τη φόρτωση της σελίδας, γίνονται fetch τα tasks του διασώστη μέσω της `fetchRescuerTasks.php`, με συγκεκριμένο id ως `rescuer_location.php?user_id=$={user_id}`, και αναπαραστάνονται μέσω της `displayTasks.php`. Αν ο διασώστης βρίσκεται κοντά σε κάποιον πολίτη, εμφανίζονται και οι επιλογές για **Complete Request/Offer** ή **Cancel Request/Offer**. Η λειτουργικότητα αυτών των επιλογών καθορίζεται από τις `completeTask()` και `cancelTask()` αντίστοιχα.

2.4.3 Items in Car



Σχήμα 2.16: Inventory αντικειμένων διασώστη

Κατά τη φόρτωση της σελίδας, επιστρέφονται τα περιεχόμενα του αυτοκινήτου του διασώστη μέσω της `fetch_items.php`. Η σελίδα δημιουργεί τις κάρτες με τα αντικείμενα μέσω της `display_car_items.php`.

2.4.4 Items in Base

The screenshot shows a web application titled "Item Dashboard" from the "RescueReach" platform. On the left, there's a sidebar with links: Home, My Tasks, Item Dashboard (which is active), and Exit. The main area has a title "Items in Base" and a search bar. Below it is a grid of 12 items, each with a "Load to Car" button.

Item	Quantity	Action
T45 Levothyroxine	100	Load to Car
Painkillers	242	Load to Car
Depon	106	Load to Car
Bandage	200	Load to Car
First Aid	150	Load to Car
Mask	290	Load to Car
Power Banks	75	Load to Car
Gasoline	100	Load to Car
Home Repair Tools	100	Load to Car
Kitchen Appliances	50	Load to Car
Cleaning Supplies	100	Load to Car
Cutlery	150	Load to Car
Cups	200	Load to Car
Plates	200	Load to Car
Pots	100	Load to Car
Toothbrush	50	Load to Car
Diapers	100	Load to Car
Towels	200	Load to Car

Σχήμα 2.17: Μεταφορά αντικειμένων από/προς την βάση

Η σελίδα που εμφανίζεται όταν πατάμε στο **Load/Unload items**. Πάλι χρησιμοποιείται η `fetch_items.php`, και τα περιεχόμενα εμφανίζονται μέσω της `displayBaseItems.php` και της `display_car_items.php`. Οι συναρτήσεις δημιουργούν επιπλέον και τα κατάλληλα κουμπιά με αντίστοιχους Event Listeners που καλούν τα `loadItemToCar()` (`load_item.php`) και `unloadItemFromCar()` (`unload_item.php`).

2.5 | USER

2.5.1 Home

The screenshot shows the RescueReach user interface. At the top, it says "Welcome back, nikf". On the left, there's a sidebar with "Home" and "Exit" buttons. The main area has a "Create Request" form with a dropdown for "All Categories" and a search bar. Below that is a grid of items: Bandage, Batteries, Blankets, Canned Goods, Caps, Cereals, Cleaning Supplies, Compass, Cups, Cutlery, Depon, Diapers, Diesel, Earplugs, First Aid, Flashlight, Gloves, GPS, Heater, Home Repair Tools, Kitchen Appliances, Map, Mask, Matches, Mattresses, Painkillers, Pants, Plates, Pots, Power Banks, Repair Tools, Shirt, Shoes, Snacks, Soap, Socks, T45 Levothyroxine, Toothbrush, Toothpaste, Towels, and Water. A "Quantity" input field and a "Create" button are at the bottom of the request form. To the right, there's a "Quick Summary" section with "My Requests" and "My Offers". "My Requests" lists an item ID 3 (Depon, accepted, 106 units) and an item ID 5 (First Aid). "My Offers" lists an item ID 2 (Painkillers, cancelled, 242 units) and an item ID 3 (Depon). Below these sections is a table titled "Announcements" with columns "Item", "Quantity", and "Action". It lists three items: Depon (22), Painkillers (15), and Depon (10), each with a "Create Offer for [Item]" button.

Σχήμα 2.18: Αρχική σελίδα ποικίτη