https://github.com/xmerantzis/texnologia_logismikou

fuelpay™

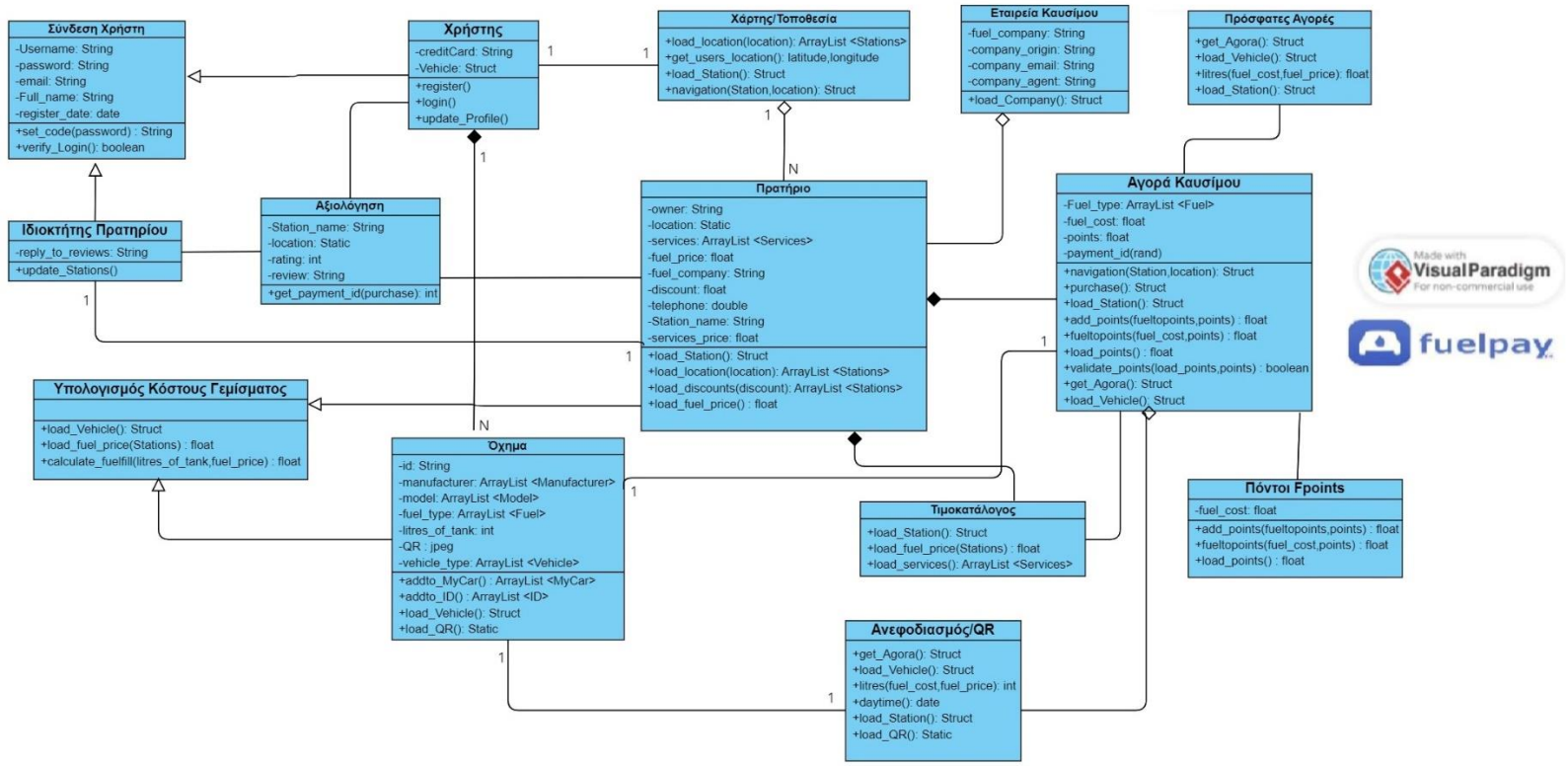
# Class-diagram-v0.1

**Το νέο σύστημα ανεφοδιασμού καυσίμων!**

https://github.com/xmerantzis/texnologia_logismikou

| ΟΝΟΜΑΤΕΠΩΝΥΜΟ | ΑΡΙΘΜΟΣ ΜΗΤΡΩΟΥ | EMAIL | Ρόλος στο παρόν κείμενο |
|---|---|---|---|
| ΘΑΝΟΣ ΚΑΠΝΙΑΣ | 1071112 | up1071112@upnet.gr | Co-Editor |
| ΧΡΗΣΤΟΣ ΜΕΡΑΝΤΖΗΣ | 1070936 | up1070936@upnet.gr | Co-Editor |
| ΠΑΝΤΕΛΗΣ ΜΑΚΡΥΓΙΑΝΝΗΣ | 1067526 | up1067526@upnet.gr | Co-Editor |
| ΤΣΙΝΤΖΕΛΗΣ ΔΗΜΗΤΡΙΟΣ | 1067370 | up1067370@upnet.gr | Co-Editor |

# Class diagram




## Σύνδεση Χρήστη:

Η κλάση Σύνδεση Χρήστη αναφέρεται στην είσοδο του χρήστη στη εφαρμογή και απαρτίζεται από το username(Username:String), τον κωδικό(password:String), το e-mail(e-mail:String), το πλήρες όνομα(Full_name:String) και την ημερομηνία εγγραφής(register_date:date).

***+set_code(password):*** Ορίζει την μεταβλητή password.

***+verify_Login()*** *(boolean):* Το σύστημα επαληθεύει αν ο κωδικός που δόθηκε είναι ο σωστός. Αν είναι σωστός επιστέφει true.

## Χρήστης:

Η κλάση Χρήστης κληρονομεί το όνομα χρήστη (Username:String), πλήρες όνομα (Full_name:String), email (e-mail:String) και απαρτίζεται από κάρτες πληρωμής (creditCard:String) και το όχημα (Vehicle:Struct).

***+register():*** Η μέθοδος επιτρέπει στον χρήστη να πραγματοποιήσει εγγραφή στο σύστημα.

***+login():*** Η μέθοδος login() επαληθεύει τα στοιχεία εισόδου και επιτρέπει στον χρήστη να πραγματοποιήσει είσοδο στο σύστημα.

***+update_Profile():*** Η μέθοδος update_Profile() λαμβάνει την δομή του λογαριασμού του χρήστη. Επεξεργάζεται και ενημερώνει τα στοιχεία του προσωπικού του προφίλ.

## Ιδιοκτήτης Πρατηρίου :

Η κλάση Ιδιοκτήτης Πρατηρίου αναφέρεται στην διαχείριση των πρατηρίων του ιδιοκτήτη μέσω της μεθόδου update_Stations(), και απαρτίζεται από τις απαντήσεις των αξιολογήσεων (reply_to_reviews:String).

***+update_Stations():*** Η μέθοδος λαμβάνει τις δομές των πρατηρίων για τον ιδιοκτήτη. Τροποποιεί τα στοιχεία του πρατηρίου(τιμές καυσίμων, τιμοκατάλογος, υπηρεσίες).

## Όχημα :

Περιέχει τον Τύπο οχήματος (αυτοκίνητο/μοτοσυκλέτα κτλ) (vehicle_type:ArrayList<Vehicle>), τον Αριθμό κυκλοφορίας(id:String), τον Κατασκευαστή (manufacturer:ArrayList<Manufacturer>), το Μοντέλο (model:ArrayList<Model>), τον Τύπο καυσίμου (fuel_type:ArrayList<Fuel>) καθώς και το Μέγεθος του ρεζερβουάρ (litres_of_tank:int) που εξασφαλίζει τον υπολογισμό της τιμής του γεμίσματος. Ακόμη σε κάθε όχημα χορηγείται ένα χαρακτηριστικό QR (QR:jpeg) που αφορά μόνο το συγκεκριμένο όχημα για την υλοποίηση του ανεφοδιασμού.

***+addto_MyCar():ArrayList<MyCar>:*** Η μέθοδος προσθέτει τη δομή του οχήματος, στην λίστα των οχημάτων του Χρήστη.

***+addto_ID():ArrayList(ID):*** Λαμβάνει από τη βάση δεδομένων του Υπουργείου Μεταφορών τις πινακίδες του οχήματος του χρήστη και επαληθεύει την ύπαρξή τους.

***+load_Vehicle:*** Η μέθοδος αυτή επιστρέφει το όχημα που επιλέγει ο Χρήστης. Ανατρέχει στην βάση δεδομένων των οχημάτων και ανακαλεί τα χαρακτηριστικά του (π.χ. μέγεθος ρεζερβουάρ, τύπος καυσίμου, κλπ)

## Πρατήριο :

Η Κλάση Πρατήριο αναφέρεται στα χαρακτηριστικά του πρατηρίου: Ιδιοκτήτης (owner:String), τοποθεσία (location:Static), παροχές (services:ArrayList<Services>), τιμές καυσίμων (fuel_price:float), επωνυμία πρατηρίου (fuel_company:String), εκπτώσεις (discount:float), τηλέφωνο (telephone:double), όνομα Πρατηρίου (Station_name:String), και τιμές παροχών (services_price:float).

***+load_Station:*** Το σύστημα ανατρέχει στη βάση δεδομένων των πρατηρίων και ανακαλεί τα στοιχεία του (τοποθεσία, είδος καυσίμων που παρέχει, υπηρεσίες, κλπ). Εμφανίζεται η τοποθεσία του πρατηρίου.

***+load_location(location)***: Λαμβάνει την τοποθεσία του πρατηρίου από την λίστα Πρατηρίων (βάση δεδομένων) και την επιστρέφει στον Χρήστη.

***+load_discounts(discount):*** Λαμβάνει τις εκπτώσεις του πρατηρίου από την λίστα Πρατηρίων (βάση δεδομένων) και τις επιστρέφει στον Χρήστη.

***+load_fuel_price(Stations):*** Η μέθοδος επιστρέφει από τη βάση δεδομένων τις τιμές των καυσίμων για το επιλεγμένο πρατήριο.

## Αγορά Καυσίμου :

Περιέχει τις μεθόδους (navigation(Station,users_location):Struct) για πλοήγηση, την καταχώρηση αγοράς (purchase(),get_Agora()), τις πληροφορίες πρατηρίου (load_Station():Struct). Ακόμη τις μεθόδους add_points(), fueltopoints(), load_points() , validate_points() για την διεκπεραίωση του συστήματος των πόντων. (load_Vehicle():Struct).

***+navigation(Station,users_location):*** Η μέθοδος λαμβάνει την τοποθεσία του χρήστη και του επιλεγμένου πρατηρίου. Ύστερα υπολογίζει την βέλτιστη διαδρομή προς πλοήγηση.

***+purchase():*** Η μέθοδος καταχωρεί τα στοιχεία της αγοράς όπως το είδος καυσίμου, τη συνολική τιμή αγοράς, τους πόντους Fpoints που τυχόν χρησιμοποιήθηκαν και το μοναδικό id αγοράς.

***+load_Station():***Το σύστημα ανατρέχει στη βάση δεδομένων των πρατηρίων και ανακαλεί τα στοιχεία του (τοποθεσία, είδος καυσίμων που παρέχει, υπηρεσίες, κλπ). Εμφανίζεται η τοποθεσία του πρατηρίου.

***+add_points(fueltopoints,points):*** Λαμβάνει τους καταχωρημένους στο σύστημα Fpoints του χρήστη και τους πόντους που προκύπτουν από τη νέα αγορά. Προσθέτει τους πόντους από την αγορά που συνέβη, στους ήδη καταχωρημένους.

***+fueltopoints(fuel_cost,points):*** : Η μέθοδος υπολογίζει και μετατρέπει το ποσό της αγοράς σε αντίστοιχους Fpoints.

***+load_points():*** Η μέθοδος αυτή φορτώνει το ποσό πόντων που είναι καταχωρημένοι στο Χρήστη.

***+validate_points(load_points,points):*** Η μέθοδος αυτή συγκρίνει τους πόντους που εισάγει ο Χρήστης για εξαργύρωση, με τους υπάρχοντες πόντους του χρήστη. Αν οι υπάρχοντες πόντοι επιτρέπουν την εξαργύρωση (αν οι πόντοι που επιχειρεί να χρησιμοποιήσει ο χρήστης υπάρχουν στον λογαριασμό του) επιστρέφει true.

***+get_Agora():*** λαμβάνει δομές (ιστορικό και πληροφορίες συναλλαγής-πρατηρίου) Αγορών.

***+load_Vehicle():*** Η μέθοδος αυτή επιστρέφει το όχημα που επιλέγει ο Χρήστης. Ανατρέχει στην βάση δεδομένων των οχημάτων και ανακαλεί τα χαρακτηριστικά του (π.χ. μέγεθος ρεζερβουάρ, τύπος καυσίμου, κλπ)

## Χάρτης:

Η κλάση «Χάρτης» περιλαμβάνει ένα Map που απεικονίζει τη τοποθεσία κάθε Πρατηρίου (load_location(location):ArrayList<Stations>), (load_Station():Struct) γύρω από τον χρήστη. Λαμβάνει και απεικονίζει επίσης, την τοποθεσία του χρήστη (get_users_location()) με δυνατότητα πλοήγησης του χρήστη στο Πρατήριο της αρέσκειάς του (navigation(Station,users_location):Struct).

***+load_location(location)***: Λαμβάνει την τοποθεσία του πρατηρίου από την λίστα Πρατηρίων (βάση δεδομένων) και την επιστρέφει στον Χρήστη.

***+get_users_location():*** Λαμβάνει τις γεωγραφικές συντεταγμένες του Χρήστη, εφόσον ο χρήστης έχει ενεργοποιημένες τις ρυθμίσεις τοποθεσίας. Καταχωρεί την τοποθεσία στην μεταβλητή users_location και την εμφανίζει.

***+load_Station():*** Το σύστημα ανατρέχει στη βάση δεδομένων των πρατηρίων και ανακαλεί τα στοιχεία του (τοποθεσία, είδος καυσίμων που παρέχει, υπηρεσίες, κλπ). Εμφανίζεται η τοποθεσία του πρατηρίου.

***+navigation(Station,users_location):*** Η μέθοδος λαμβάνει την τοποθεσία του χρήστη και του επιλεγμένου πρατηρίου. Ύστερα υπολογίζει την βέλτιστη διαδρομή προς πλοήγηση.

## Ανεφοδιασμός/QR :

Η κλάση αυτή θα αναφέρει αναλυτικές πληροφορίες για την αγορά του καυσίμου. Πιο αναλυτικά θα σημειώνεται το συνολικό κόστος καυσίμου (get_Agora():Struct), ο αριθμός κυκλοφορίας του οχήματος που έγινε ο ανεφοδιασμός (load_Vehicle():Struct), τα λίτρα που αγοράστηκαν από τον χρήστη (litres(fuel_cost,fuel_price):float), η ημερομηνία και ώρα που πραγματοποιήθηκε ανεφοδιασμός (daytime:date) καθώς και το πρατηρίου που έγινε η αγορά (load_Station():Struct).

***+get_Agora():*** Λαμβάνει δομές (ιστορικό και πληροφορίες συναλλαγής-πρατηρίου) Αγορών.

***+load_Vehicle():*** Η μέθοδος επιστρέφει το όχημα που επιλέγει ο Χρήστης. Ανατρέχει στην βάση δεδομένων των οχημάτων και ανακαλεί τα χαρακτηριστικά του (π.χ. μέγεθος ρεζερβουάρ, τύπος καυσίμου, κλπ)

***+litres(fuel_cost,fuel_price):*** Η μέθοδος αυτή λαμβάνει τις τιμές fuel_cost (τιμή ανά λίτρο) και fuel_price(τελική τιμή αγοράς). Κάνει την διαίρεση αυτών των δύο, και υπολογίζει τα λίτρα ανεφοδιασμού.

***+daytime():*** Μέθοδος που αποθηκεύει την ημερομηνία και την ώρα ανεφοδιασμού.

***+load_Station():*** Το σύστημα ανατρέχει στη βάση δεδομένων των πρατηρίων και ανακαλεί τα στοιχεία του (τοποθεσία, είδος καυσίμων που παρέχει, υπηρεσίες, κλπ). Εμφανίζεται η τοποθεσία του πρατηρίου.

***+load_QR():*** Μέθοδος η οποία επιστρέφει/εμφανίζει το προσωπικό QR του οχήματος του Χρήστη.

## Εταιρεία καυσίμου :

Περιλαμβάνει την μέθοδο +load_Company() και πληροφορίες Επωνυμίας.

***+load_Company:*** Η μέθοδος ανατρέχει στη βάση δεδομένων των πρατηρίων, και επιστρέφει τις πληροφορίες και τα χαρακτηριστικά (fuel_company, company_origin, company_email, company_agent) του πρατηρίου που επιλέγει ο Χρήστης.

## Πόντοι εφαρμογής - Fpoints:

Οι πόντοι εφαρμογής έχουν άμεση σχέση με τον Ανεφοδιασμό. Δέχονται σαν όρισμα το συνολικό ποσό Ανεφοδιασμού (fuel_cost():float), και υπολογίζει τους πόντους επιβράβευσης. Ακόμη περιλαμβάνει τους ήδη μαζεμένους πόντους που συνδέονται με τον Χρήστη και τους προσαυξάνει ανάλογα με την εξαργύρωση του χρήστη(add_points(fueltopoints,points):float).

***+add_points(fueltopoints,points):*** Λαμβάνει τους καταχωρημένους στο σύστημα Fpoints του χρήστη και τους πόντους που προκύπτουν από τη νέα αγορά. Προσθέτει τους πόντους από την αγορά που συνέβη, στους ήδη καταχωρημένους.

***+fueltopoints(fuel_cost,points):*** : Η μέθοδος υπολογίζει και μετατρέπει το ποσό της αγοράς σε αντίστοιχους Fpoints.

***+load_points():*** Η μέθοδος αυτή φορτώνει το ποσό πόντων που είναι καταχωρημένοι στο Χρήστη.

## Υπολογισμός Κόστους Γεμίσματος:

***+load_Vehicle():*** Η μέθοδος επιστρέφει το όχημα που επιλέγει ο Χρήστης. Ανατρέχει στην βάση δεδομένων των οχημάτων και ανακαλεί τα χαρακτηριστικά του (π.χ. μέγεθος ρεζερβουάρ, τύπος καυσίμου, κλπ)

***+load_fuel_price(Stations):*** Η μέθοδος επιστρέφει από τη βάση δεδομένων τις τιμές των καυσίμων για το επιλεγμένο πρατήριο.

***+calculate_fuelfill(litres_of_tank,fuel_price):*** Η μέθοδος ανατρέχει στα χαρακτηριστικά του αυτοκινήτου, λαμβάνει την χωρητικότητα του ρεζερβουάρ του οχήματος, και την τιμή καυσίμου σύμφωνα με το επιλεγμένο πρατήριο. Στη συνέχεια υπολογίζει την τιμή ανεφοδιασμού.

## Αξιολόγηση:

Απαρτίζεται από το όνομα του πρατηρίου (Station_name:String), από τη τοποθεσία του (location:Static), τη βαθμολογία (rating:int) και τις κριτικές του (review:String).

***+get_payment_id():*** Λαμβάνει τα payment id των αγορών του χρήστη και τα επαληθεύει. Αξιολόγηση γίνεται μόνο στα επαληθευμένα payment id του χρήστη. Αφού γίνει επαλήθευση ο χρήστης προσθέτει το σχόλιο του.

## Τιμοκατάλογος:

***+load_Station():*** Το σύστημα ανατρέχει στη βάση δεδομένων των πρατηρίων και ανακαλεί τα στοιχεία του (τοποθεσία, είδος καυσίμων που παρέχει, υπηρεσίες, κλπ). Εμφανίζεται η τοποθεσία του πρατηρίου.

***+load_fuel_price(Stations):*** Η μέθοδος αυτή επιστρέφει από τη βάση δεδομένων τις τιμές καυσίμων για το επιλεγμένο πρατήριο.

***+load_services():*** Η μέθοδος αυτή επιστρέφει από τη βάση δεδομένων την λίστα προϊόντων ArrayList <Services> και υπηρεσιών που προσφέρει το επιλεγμένο πρατήριο.

## Πρόσφατες Αγορές:

Παρέχει στον Χρήστη ένα ιστορικό Αγορών.
Έχει τις μεθόδους (get_Agora():Struct), (load_Vehicle():Struct), (litres(fuel_cost,fuel_price):float), (load_Station():Struct).

***+get_Agora():*** Λαμβάνει δομές (ιστορικό και πληροφορίες συναλλαγής-πρατηρίου) Αγορών.

***+load_Vehicle():*** Η μέθοδος αυτή επιστρέφει το όχημα που επιλέγει ο Χρήστης. Ανατρέχει στην βάση δεδομένων των οχημάτων και ανακαλεί τα χαρακτηριστικά του (π.χ. μέγεθος ρεζερβουάρ, τύπος καυσίμου, κλπ)

***+litres(fuel_cost,fuel_price):*** Η μέθοδος αυτή λαμβάνει τις τιμές fuel_cost (τιμή ανά λίτρο) και fuel_price(τελική τιμή αγοράς). Κάνει την διαίρεση αυτών των δύο, και υπολογίζει τα λίτρα ανεφοδιασμού.

***+load_Station():*** Το σύστημα ανατρέχει στη βάση δεδομένων των πρατηρίων ανακαλώντας τα στοιχεία του (τοποθεσία, είδος καυσίμων που παρέχει, υπηρεσίες, κλπ). Εμφανίζεται η τοποθεσία του πρατηρίου.

# Εργαλεία που χρησιμοποιήθηκαν

## Word

Χρησιμοποιήθηκε για την σύνταξη των κειμένων

## Visual Paradigm και Draw

Χρησιμοποιήθηκαν για την παραγωγή του Domain Model.