

Έγγραφο απαιτήσεων λογισμικού(SRS)

ΔΙΑΧΕΙΡΙΣΗ ΗΛΕΚΤΡΙΚΩΝ ΟΧΗΜΑΤΩΝ

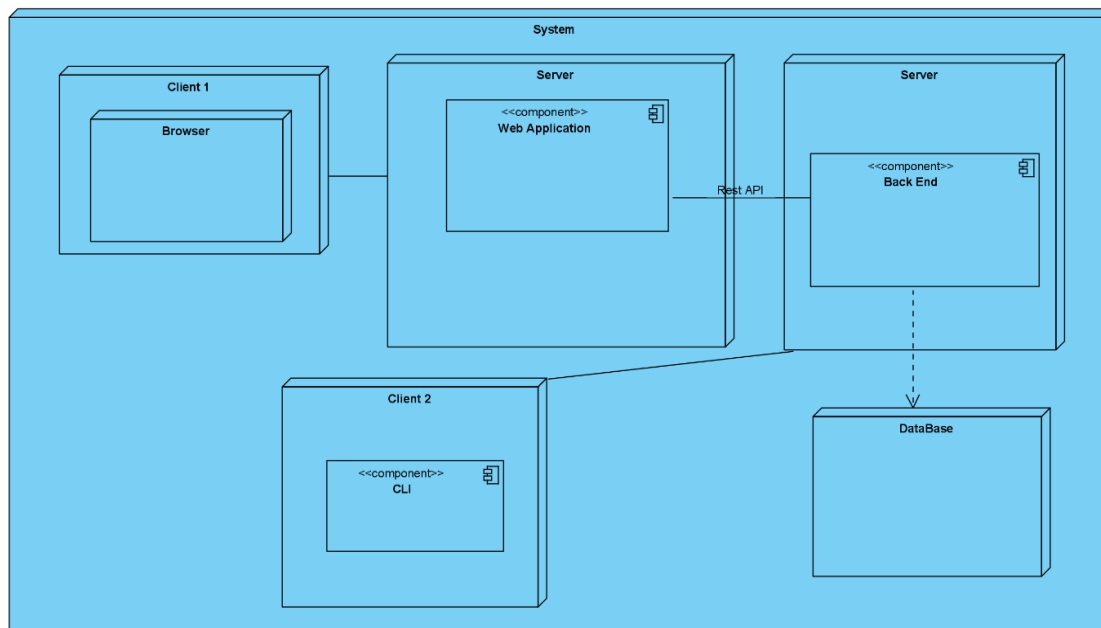
1. Εισαγωγή

1.1 Εισαγωγή: σκοπός του λογισμικού

Σκοπός του λογισμικού είναι να διευκολύνει τη διαδικασία φόρτισης ενός ηλεκτρικού αυτοκινήτου, τη διαχείριση σταθμών φόρτισης και τη παροχή πληροφοριών στους διάφορους εμπλεκόμενους.

1.2 Διεπαφές (interfaces)

1.2.1 Διεπαφές με εξωτερικά συστήματα



1.2.2 Διεπαφές με το χρήστη

- ✓ **Ιδιοκτήτης αυτοκινήτων:** Σε αυτή την κατηγορία εντάσσονται οι χρήστες που έχουν πρόσβαση στο front-end της εφαρμογής, στο REST-API και το CLI. Μέσω αυτών έχουν την δυνατότητα να βλέπουν διάφορες πληροφορίες που αφορούν τη χρήση των οχημάτων τους.
- ✓ **Δήμοι και ιδιοκτήτες χώρων στάθμευσης:** Αυτοί έχουν πρόσβαση στο front-end, στο REST-API και στο CLI της εφαρμογής. Μέσω του front-end μπορούν να δουν πληροφορίες και στατιστικά των σταθμών τους, ενώ μέσω του CLI μπορούν να τους επεξεργαστούν/δημιουργήσουν νέους.
- ✓ **Πάροχοι ηλεκτρικής ενέργειας:** Αυτοί έχουν πρόσβαση στο front-end της εφαρμογής, στο REST-API και στο CLI. Μέσω αυτών μπορούν να δουν πληροφορίες σχετικά με τους σταθμούς και τα σημεία φόρτισης που αφορούν την εταιρεία τους.

2. Αναφορές - Πηγές πληροφοριών

Ως πηγή πληροφοριών χρησιμοποιήθηκε το διαδικτυακό εγχειρίδιο: visual-paradigm.com

3. Προδιαγραφές απαιτήσεων λογισμικού

3.1 Περιπτώσεις χρήσης

3.1.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 1: Φόρτιση αυτοκινήτου

3.1.1.1 Χρήστες (ρόλοι) που εμπλέκονται

Οι κύριοι ρόλοι του use case είναι ο χρήστης που θέλει να φορτίσει το αυτοκίνητό του και οι χώροι στάθμευσης, οι οποίοι συλλέγουν πληροφορίες για την εκάστοτε φόρτιση. Για να επιτευχθούν τα παραπάνω χρειάζονται οι δευτερεύοντες ρόλοι. Αυτοί είναι μία βάση δεδομένων στην οποία αποθηκεύονται τα δεδομένα φόρτισης μέσω ενός server.

3.1.1.2 Προϋποθέσεις εκτέλεσης

Αναγνώριση οχήματος, δηλαδή ύπαρξη στη βάση δεδομένων.

3.1.1.3 Περιβάλλον εκτέλεσης

Το περιβάλλον στο οποίο εκτελείται η περίπτωση χρήσης είναι:

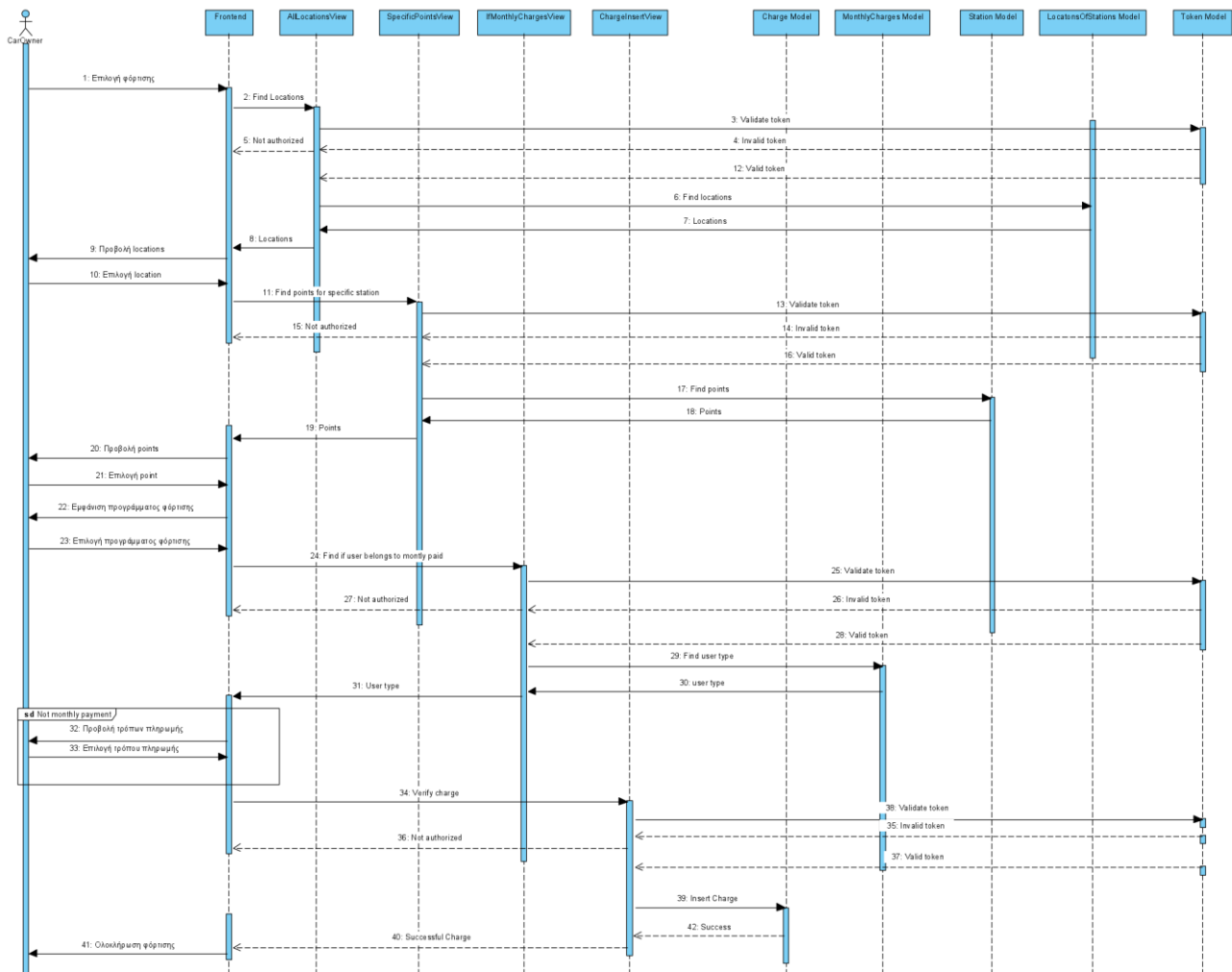
- SQL βάση δεδομένων
- Λειτουργικό σύστημα: Windows
- Front-End
- Web application

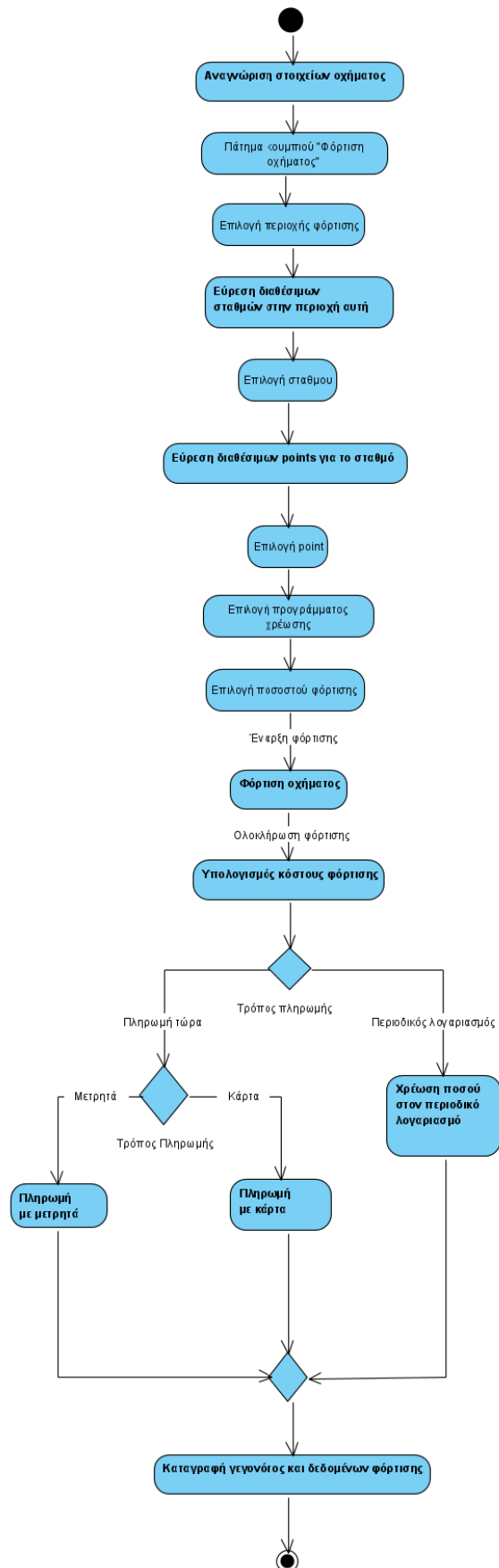
3.1.1.4 Δεδομένα εισόδου και 3.1.1.6 Παράμετροι

Δεδομένο εισόδου (Πεδίο)	Συνθήκη Εγκυρότητας
<i>Vehicle_id</i>	Πρέπει να υπάρχει καταχωρημένο το <i>vehicle_id</i> στη βάση δεδομένων
<i>Protocol</i>	Επιλογή προγράμματος χρέωσης (γρήγορη ή αργή φόρτιση)
<i>Energy_amount</i>	Επιλογή ποσότητα ηλεκτρικής ενέργειας για τη φόρτιση
<i>Payment_method</i>	Επιλογή χρέωσης μηνιαίου λογαριασμού ή άμεσης χρέωσης
<i>Card_number</i>	Εισαγωγή αριθμού κάρτας ώστε να επιτευχθεί η πληρωμή

3.1.1.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Περιγραφή με κείμενο (Βήμα 1, Βήμα 2 κλπ) και διαγράμματα UML αλληλουχίας (Sequence) και δραστηριοτήτων (Activity). Περιλαμβάνεται η συμπεριφορά σε απρόβλεπτες καταστάσεις και σφάλματα (εναλλακτικές ροές).





1. Ταυτοποίηση χρήστη.
 - 1.1. Σε περίπτωση μη εύρεσης του χρήστη έξοδος από τη φόρτιση.
2. Συμπλήρωση παραμέτρων.
3. Έναρξη φόρτισης.
 - 3.1. Εμφάνιση τελικού κόστους με την ολοκλήρωση της φόρτισης.
4. Αποστολή στοιχείων κάρτας.
 - 4.1. Πληρωμή και επιβεβαίωση πληρωμής

3.1.1.7 Δεδομένα εξόδου

1. Συλλογή δεδομένων οχήματος σε κάθε φόρτιση
2. Κόστος φόρτισης
3. Χρέωση ποσού στον περιοδικό λογαριασμό, εφόσον υπάρχει τέτοιος
4. Καταγραφή γεγονότων και δεδομένων φόρτισης

3.1.2 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 2: Επιλογή χώρου στάθμευσης για φόρτιση

3.1.2.1 Χρήστες (ρόλοι) που εμπλέκονται

Ο κύριος ρόλος του use case είναι ο χρήστης που ψάχνει σταθμό για να φορτίσει το όχημά του. Αυτό επιτυγχάνεται με τη χρήση μιας βάσης δεδομένων η οποία περιέχει στατιστικές πληροφορίες για όλους τους σταθμούς, όπως και ένας server ο οποίος μετεφέρει στο χρήστη αυτές τις πληροφορίες.

3.1.2.2 Προϋποθέσεις εκτέλεσης

Ύπαρξη σταθμών φόρτισης στη βάση.

3.1.2.3 Περιβάλλον εκτέλεσης

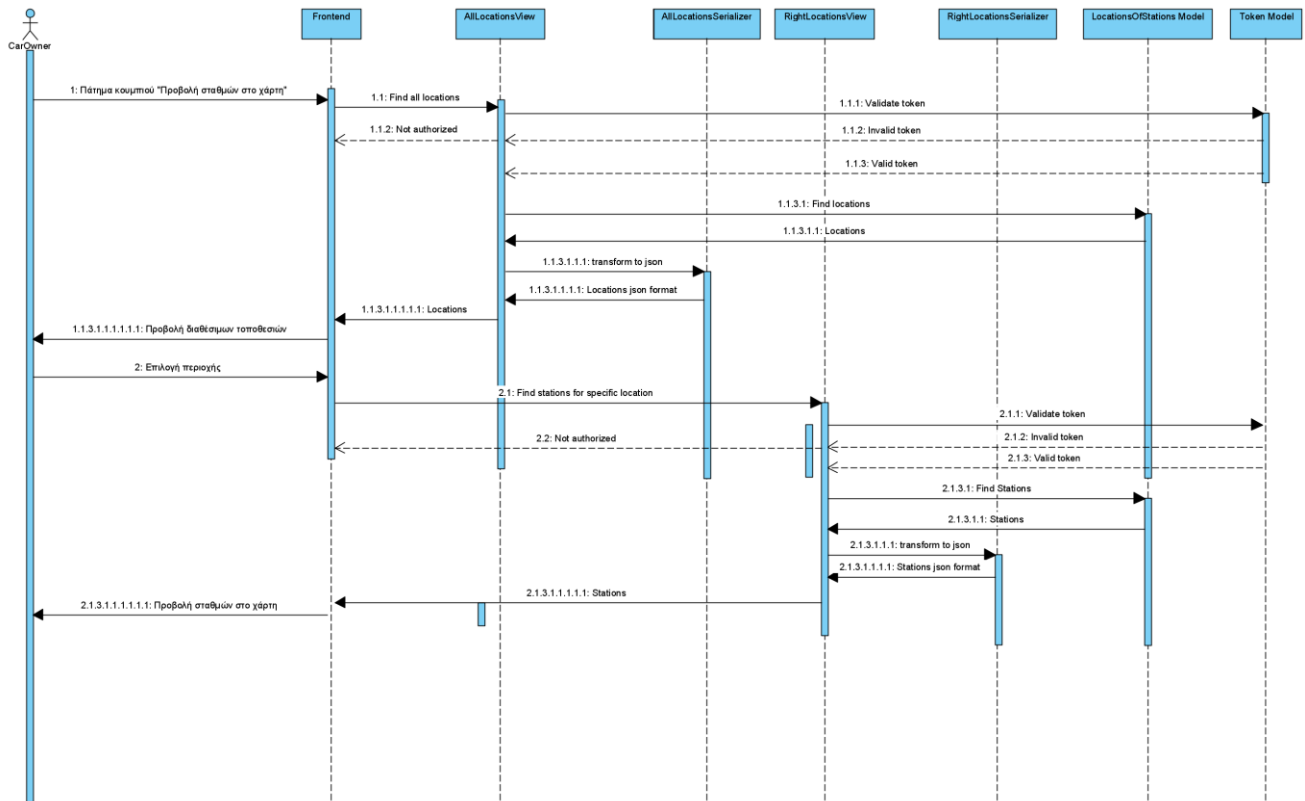
- SQL βάση δεδομένων
- Λειτουργικό σύστημα: Windows
- Front-End
- Web application

3.1.2.4 Δεδομένα εισόδου και 3.1.2.6 παράμετροι

Δεδομένο εισόδου (Πεδίο)	Συνθήκη Εγκυρότητας
Address_region	Ο χρήστης επιλέγει την περιοχή στην οποία θέλει να γίνει η φόρτιση
Ώρα	Επιλογή συγκεκριμένης ώρας της φόρτισης

3.1.2.6 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Περιγραφή με κείμενο (Βήμα 1, Βήμα 2 κλπ) και διαγράμματα UML αλληλουχίας (Sequence) και δραστηριοτήτων (Activity). Περιλαμβάνεται η συμπεριφορά σε απρόβλεπτες καταστάσεις και σφάλματα (εναλλακτικές ροές).





1. Ταυτοποίηση χρήστη.

1.1. Σε περίπτωση μη εύρεσης του χρήστη έξοδος από την επιλογή χώρου στάθμευσης.

2. Επιλογή περιοχής σταθμών

3. Πάτημα κουμπιού «Προβολή σταθμών στο χάρτη».

3.1. Ανάκτηση δεδομένων χώρων στάθμευσης από τη βάση, για τη συγκεκριμένη περιοχή.

3.1.1. Υπολογισμός κατάστασης σταθμών.

3.1.2. Επιστροφή

4. Προβολή σε χάρτη.

5. Επιλογή χώρου στάθμευσης.

3.1.2.7 Δεδομένα εξόδου

1. Πληρότητα και κατάσταση λειτουργίας σταθμού
2. Εκτιμώμενος χρόνος αναμονής
3. Προβολή σταθμών σε χάρτη

3.1.3 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 3: Έκδοση περιοδικού λογαριασμού

3.1.3.1 Χρήστες (ρόλοι) που εμπλέκονται

Ο κύριος ρόλος του use case είναι ο χρήστης που θέλει να δει τον μηνιαίο του λογαριασμό ή και να τον πληρώσει. Αυτό επιτυγχάνεται με τη χρήση μιας βάσης δεδομένων η οποία περιέχει τους περιοδικούς λογαριασμούς όλων των χρηστών, όπως και ένας server ο οποίος μετεφέρει στο χρήστη αυτές τις πληροφορίες.

3.1.3.2 Προϋποθέσεις εκτέλεσης

Να έχει επιλέξει ο χρήστης να χρεώνεται και να πληρώνει μέσω του περιοδικού λογαριασμού.

3.1.3.3 Περιβάλλον εκτέλεσης

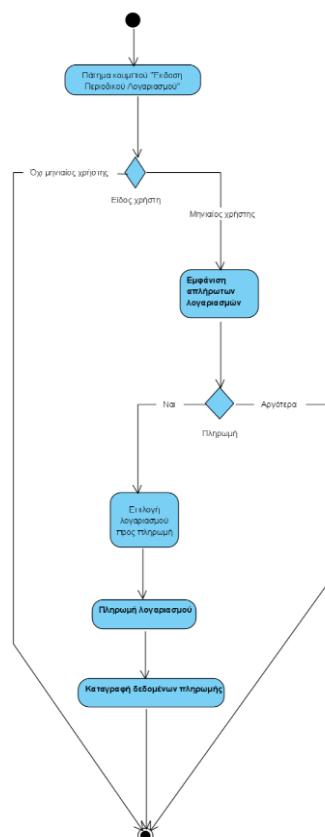
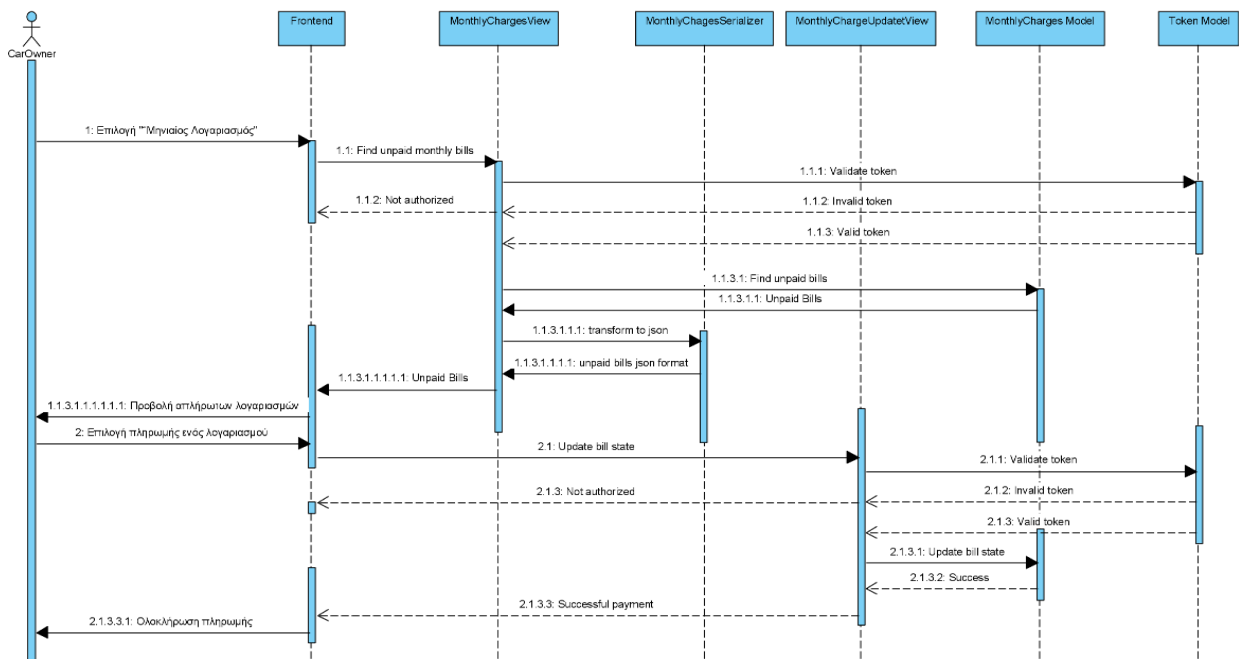
- SQL βάση δεδομένων
- Λειτουργικό σύστημα: Windows
- Front-End
- Web application

3.1.3.4 Δεδομένα εισόδου και 3.1.3.6 παράμετροι.

Δεδομένο εισόδου (Πεδίο)	Συνθήκη Εγκυρότητας
<i>CarOwner_id</i>	Επιλογή συγκεκριμένου ιδιοκτήτη αυτοκινήτου για τον περιοδικό λογαριασμό
<i>Ημερομηνία</i>	Ελέγχουμε αν είναι 1 ^η του μηνός
<i>Στοιχεία κάρτας</i>	Εφόσον πληρώσει τώρα ο χρήστης χρειάζεται να εισάγει τα στοιχεία της κάρτας του

3.1.3.6 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Περιγραφή με κείμενο (Βήμα 1, Βήμα 2 κλπ) και διαγράμματα UML αλληλουχίας (Sequence) και δραστηριοτήτων (Activity). Περιλαμβάνεται η συμπεριφορά σε απρόβλεπτες καταστάσεις και σφάλματα (εναλλακτικές ροές).



1. Ταυτοποίηση χρήστη.

1.1. Σε περίπτωση μη εύρεσης του χρήστη έξοδος από την πληρωμή περιοδικού λογαριασμού.

2. Εισαγωγή δεδομένων ημερομηνίας.

2.1. Για οποιαδήποτε ημερομηνία διαφορετική από 1^η του μηνός η διαδικασία τερματίζει.

3. Εμφάνιση απλήρωτου λογαριασμού.

4. Πληρωμή με κάρτα και επιβεβαίωση.

3.1.3.7 Δεδομένα εξόδου

1. Έκδοση μηνιαίου λογαριασμού

3.1.4 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 4: Διαχείριση σταθμών

3.1.4.1 Χρήστες (ρόλοι) που εμπλέκονται

Οι κύριοι ρόλοι του use case είναι οι δήμοι και οι ιδιοκτήτες ιδιωτικών χώρων στάθμευσης οι οποίοι διαχειρίζονται τους σταθμούς που τους ανήκουν. Αυτό επιτυγχάνεται με τη χρήση μιας βάσης δεδομένων η οποία περιέχει και μπορεί να ανανεώνει πληροφορίες σχετικά με τους σταθμούς και τους χώρους στάθμευσης.

3.1.4.2 Προϋποθέσεις εκτέλεσης

Ύπαρξη δικαιωμάτων χρήστη για τη διαχείριση σταθμών.

3.1.4.3 Περιβάλλον εκτέλεσης

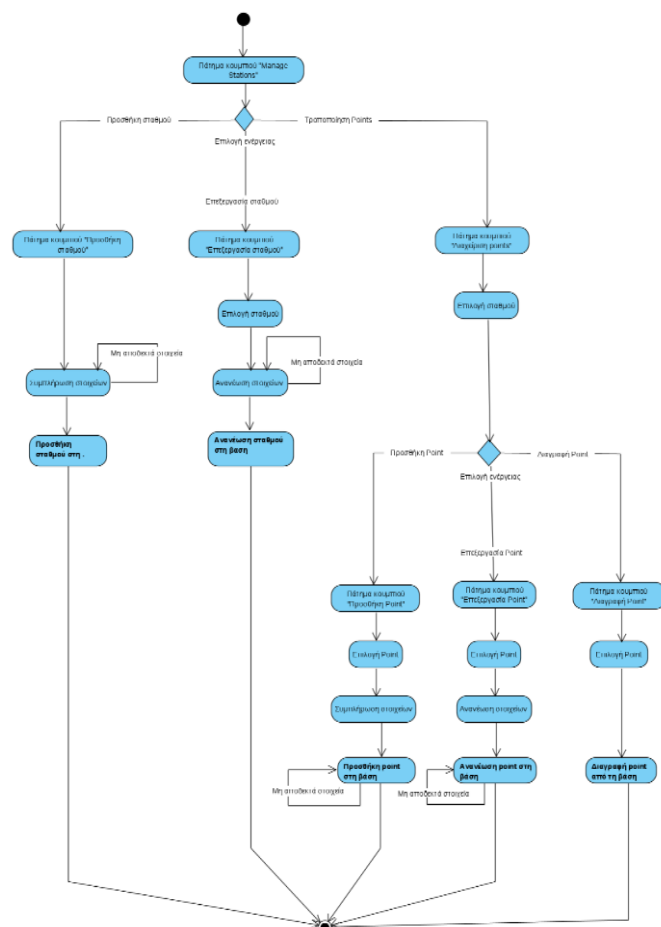
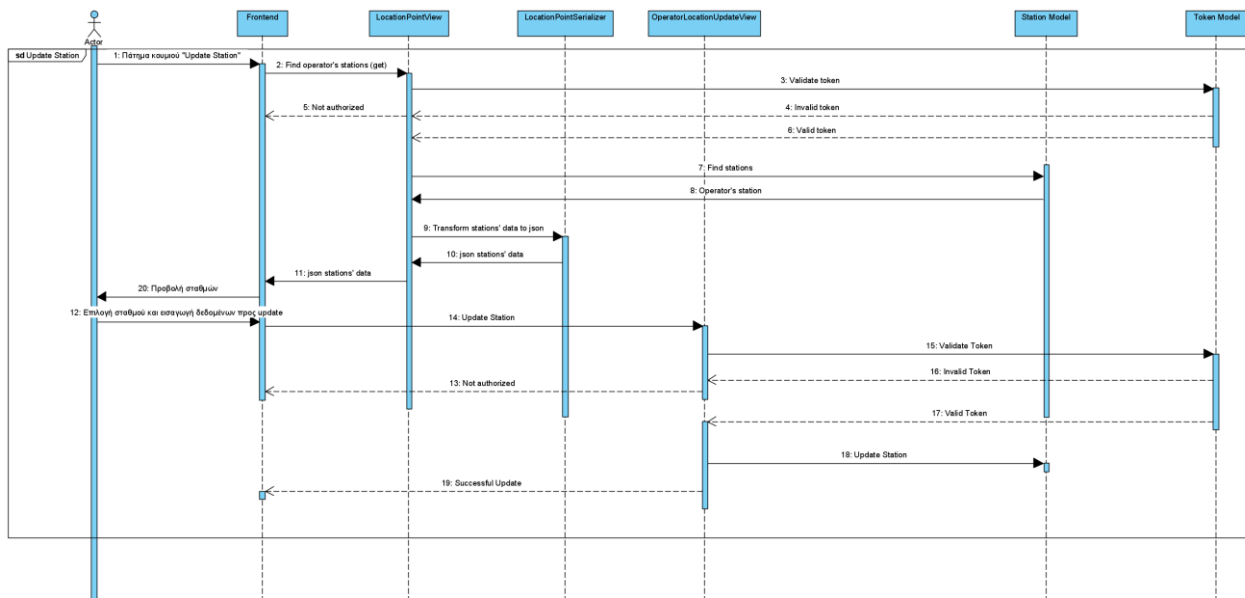
- SQL βάση δεδομένων
- Λειτουργικό σύστημα: Windows
- CLI

3.1.4.4 Δεδομένα εισόδου και 3.1.4.6 παράμετροι

Δεδομένο εισόδου (Πεδίο)	Συνθήκη Εγκυρότητας
User_id	Επιτρέπει στο χρήστη να αποκτήσει δικαιώματα διαχείρισης, εφόσον μπορεί να τα αποκτήσει
Station_id	Επιλογή σταθμού που θέλει να επεξεργαστεί ο χρήστης, εφόσον του ανήκει
Επεξεργασία	Επεξεργασία υπάρχοντος σταθμού
Δημιουργία	Δημιουργία νέου σταθμού

3.1.4.6 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Περιγραφή με κείμενο (Βήμα 1, Βήμα 2 κλπ) και διαγράμματα UML αλληλουχίας (Sequence) και δραστηριοτήτων (Activity). Περιλαμβάνεται η συμπεριφορά σε απρόβλεπτες καταστάσεις και σφάλματα (εναλλακτικές ροές).



1. Ταυτοποίηση χρήστη (ελέγχουμε αν ο χρήστης έχει δικαιώματα να επεξεργαστεί τους σταθμούς)

1.1 Σε περίπτωση που δεν έχει δικαιώματα επιστρέφουμε not-authorized.

2. Επιλογή του χρήστη αν θέλει να επεξεργαστεί ή να δημιουργήσει σταθμό.

2.1 Αν ο χρήστης επιλέξει να δημιουργήσει σταθμό εισάγει τα δεδομένα που αφορούν τον καινούργιο σταθμό

2.1.1 Το σύστημα εγκρίνει αν τα δεδομένα είναι επαρκή.

2.1.2 Το σύστημα προσθέτει τον νέο σταθμό στην βάση και εμφανίζει στον χρήστη επιτυχή ενέργεια

2.2 Αν ο χρήστης επιλέξει να επεξεργαστεί κάποιον σταθμό

2.2.1 Εμφανίζονται στον χρήστη οι σταθμοί που διαχειρίζεται και επιλέγει τον επιθυμητό.

2.2.1.1 Ο χρήστης συμπληρώνει τα καινούργια στοιχεία

2.2.1.2 Το σύστημα εγκρίνει αν τα δεδομένα επαρκούν

2.2.1.3 Ανανεώνεται η βάση δεδομένων

2.2.1.4 Εμφανίζεται στον χρήστη επιτυχή επεξεργασία σταθμού

3.1.4.7 Δεδομένα εξόδου

2. Ενημέρωση της βάσης δεδομένων σχετικά με τις αλλαγές που έγιναν.

3.1.5 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 5: ΣΤΑΤΙΣΤΙΚΑ

3.1.5.1 Χρήστες (ρόλοι) που εμπλέκονται

Οι κύριοι ρόλοι του use case είναι οι δήμοι, οι ιδιοκτήτες ιδιωτικών χώρων στάθμευσης και οι εταιρείες παραγωγής ηλεκτρικής ενέργειας οι οποίοι ενδιαφέρονται για διάφορα στατιστικά στοιχεία. Αυτό επιτυγχάνεται με τη χρήση μιας βάσης δεδομένων η οποία περιέχει τέτοιου είδους στατιστικά, όπως και ένας server ο οποίος μετεφέρει στους ενδιαφερόμενους αυτές τις πληροφορίες.

3.1.5.2 Προϋποθέσεις εκτέλεσης

Ύπαρξη γεγονότων σε στατιστική μορφή.

3.1.5.3 Περιβάλλον εκτέλεσης

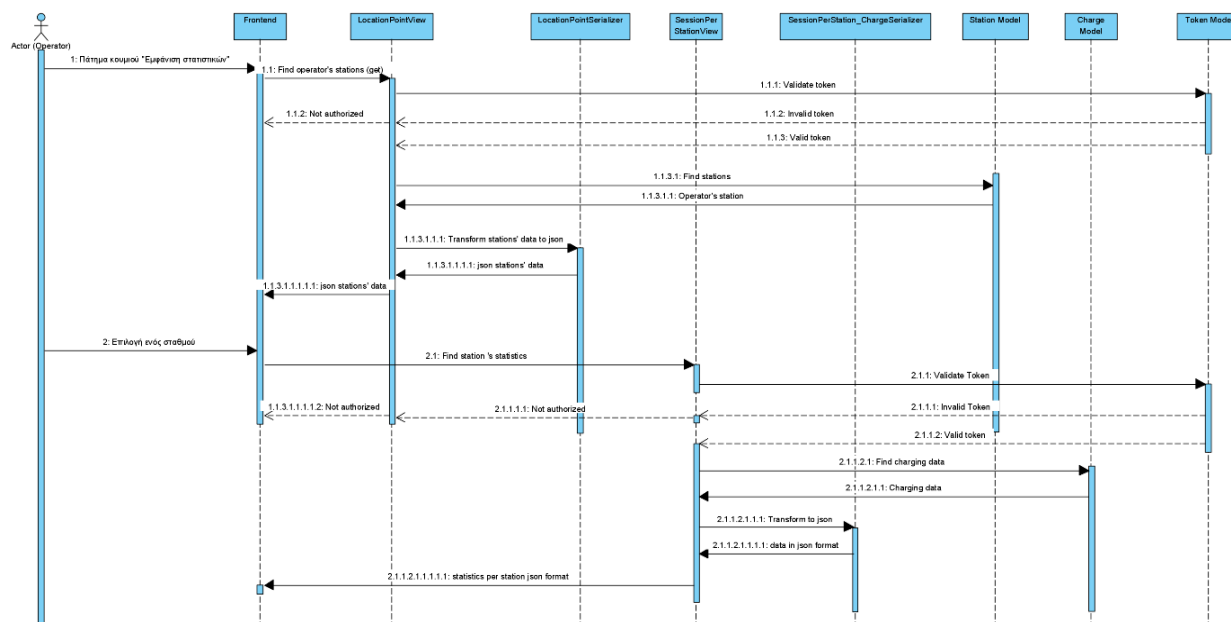
- SQL βάση δεδομένων
- Λειτουργικό σύστημα: Windows
- Front-End
- Web application

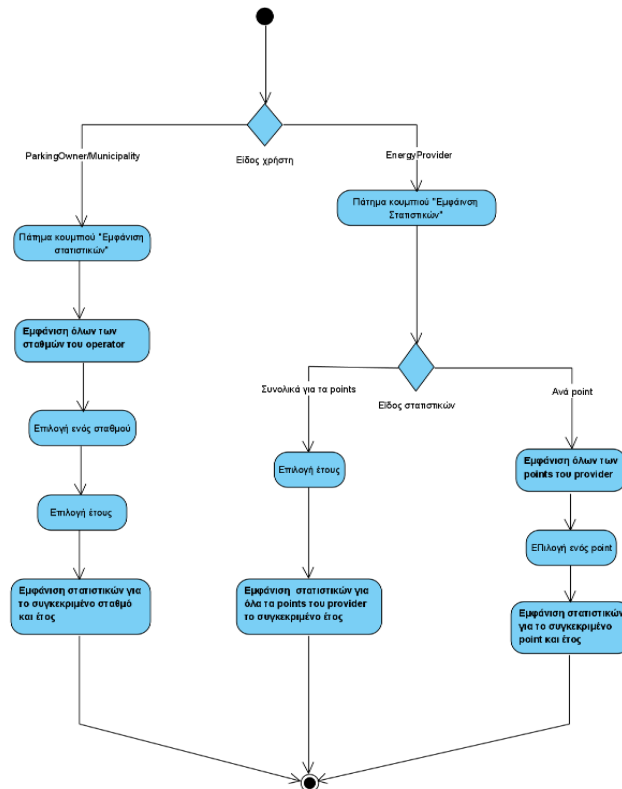
3.1.5.4 Δεδομένα εισόδου και 3.1.5.6 παράμετροι

Δεδομένο εισόδου (Πεδίο)	Συνθήκη Εγκυρότητας
<i>station_id</i>	Επιλογή σταθμού για απεικόνιση στοιχείων φόρτισης, εφόσον υπάρχει ο σταθμός
Χρονική περίοδος	Επιλογή χρονικής περιόδου απεικόνισης στατιστικών

3.1.5.6 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Περιγραφή με κείμενο (Βήμα 1, Βήμα 2 κλπ) και διαγράμματα UML αλληλουχίας (Sequence) και δραστηριοτήτων (Activity). Περιλαμβάνεται η συμπεριφορά σε απρόβλεπτες καταστάσεις και σφάλματα (εναλλακτικές ροές).





1. Πάτημα κουμπιού εμφάνιση στατιστικών

2. Ταυτοποίηση χρήστη (ελέγχουμε αν ο χρήστης έχει δικαιώματα να δει τα συγκεκριμένα στατιστικά)

2.1 Σε περίπτωση που δεν έχει δικαιώματα επιστρέφουμε not-authorized.

3. Επιλογή παραμέτρων (επιλογή σταθμού και χρονικής περιόδου που θέλουμε να εξάγουμε δεδομένα).

4. Εμφάνιση στατιστικών δεδομένων που αφορούν τις φορτίσεις και την κατανάλωση ενέργειας.

3.1.5.7 Δεδομένα εξόδου

3. Απεικόνιση δεδομένων και γεγονότων φόρτισης
4. Εμφάνιση στατιστικών στοιχείων φορτίσεων
5. Αναφορά κατανάλωσης ενέργειας

3.2 Απαιτήσεις επιδόσεων

Η εφαρμογή μας πρέπει να είναι συνεχώς διαθέσιμη και να μπορεί να δέχεται κλήσεις από πολλούς χρήστες ταυτόχρονα και με όλους τους διαθέσιμους τρόπους. Επίσης πρέπει να επιστρέφει τα δεδομένα χωρίς μεγάλη καθυστέρηση.

3.3 Απαιτήσεις οργάνωσης δεδομένων

3.3.1 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

Όλοι οι χρήστες θα έχουν πρόσβαση σε όλα τα δεδομένα, εκτός από τα δεδομένα για τους λογαριασμούς των υπόλοιπων χρηστών, με τον περιορισμό ότι δεν θα έχουν την δυνατότητα να τα τροποποιήσουν. Οι ιδιοκτήτες σταθμών και οι δημοί έχουν την επιπλέον δυνατότητα να τροποποιούν και δεδομένα.

3.4 Περιορισμοί σχεδίασης

Back-end

1. MySQL
2. Spring java
3. NPM
4. Node.js
5. Express.js

Build

1. Webpack

Testing

1. Spock

Front-end

1. Bootstrap
2. Create React App

3.5 Λοιπές απαιτήσεις

3.5.1 Απαιτήσεις διαθεσιμότητας λογισμικού

Το σύστημα πρέπει να είναι διαθέσιμο διαρκώς. Σε περίπτωση επανεκκίνησης θα πρέπει να είναι πάλι διαθέσιμο σε σύντομο χρονικό διάστημα.

3.5.2 Απαιτήσεις ασφάλειας

Η εφαρμογή πρέπει να είναι σύμφωνη με όλες τις σύγχρονες απαιτήσεις ασφαλείας. Κάποιες από αυτές είναι:

1. Υποστήριξη πρωτοκόλλου https για όλες τις διεπαφές μέσω self signed certificates.
2. Για τους κωδικούς πρόσβασης αποθηκεύονται μόνο οι τιμές κατακερματισμού τους.
3. Κάθε χρήστης έχει πρόσβαση στις λειτουργίες της εφαρμογής ανάλογα με το authentication του που προκύπτει μέσω user access token.

3.5.3 Απαιτήσεις συντήρησης

Πρέπει η βάση δεδομένων μας να έχει επαρκή και ικανό χώρο για να συντηρούνται τα ήδη υπάρχοντα δεδομένα μας, να αντέχει προσθήκη νέων όπως και να πετυχαίνει γρήγορη προσπέλαση αυτών. Θέλουμε η εφαρμογή μας να ανανεώνεται συνεχώς, προσθέτωντας καινούριες λειτουργίες και να γίνεται φιλική προς το χρήστη.