Έγγραφο απαιτήσεων λογισμικού (SRS)

ΠΡΟΣΑΡΜΟΓΗ ΤΟΥ ΑΝΤΙΣΤΟΙΧΟΥ ΕΓΓΡΑΦΟΥ ΤΟΥ ΠΡΟΤΥΠΟΥ ISO/IEC/IEEE 29148:2011

[Plug and Drive]

# Εισαγωγή

## 1.1 Εισαγωγή: σκοπός του λογισμικού

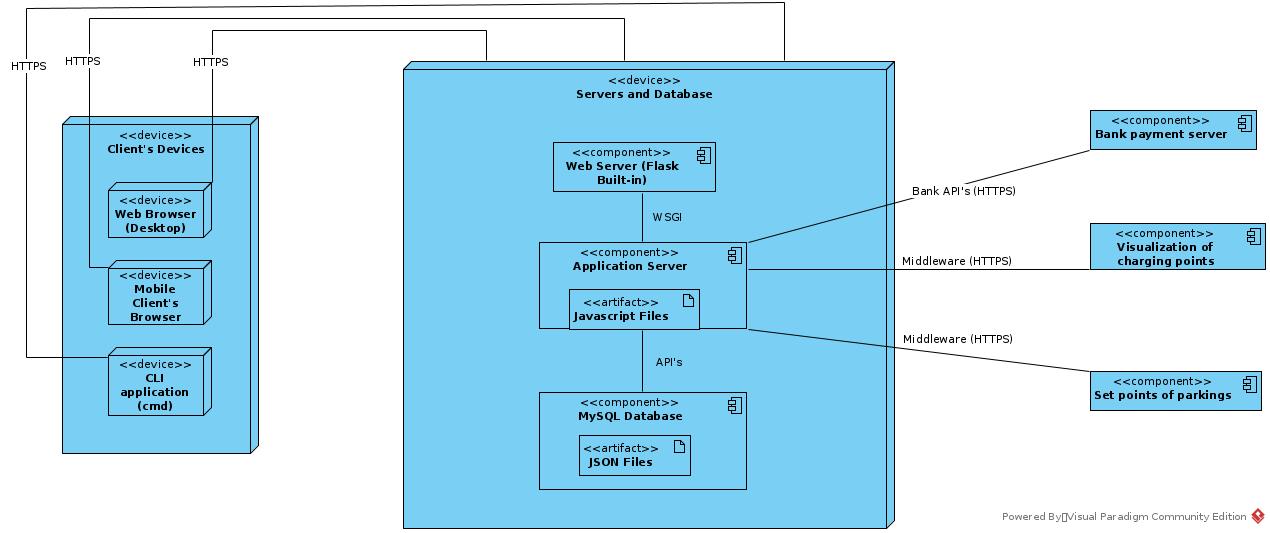
Το λογισμικό αυτό έχει σκοπό την υποστήριξη ενός γενικευμένου πληροφοριακού συστήματος , με στόχο την οργάνωση της φόρτισης των ηλεκτρικών οχημάτων. Οι εγγεγραμμένοι χρήστες της εφαρμογής μπορούν να απολάυσουν πληθώρα προνομοίων , ανάλογα με την ιδιότητά τους και τον σκοπό χρήσης της εφαρμογής.

Οι ιδιοκτήτες ηλεκτρικών οχημάτων, αφού επιλέξουν τον χώρο στάθμευσης και φόρτισης της επιλογής τους , πλέον θα μπορουν να αυτοματοποιήσουν τις διαδικασίες επιλογής προγράμματος φόρτισης και άμεσης πληρωμής, μέσω της εφαρμογής. Απο την άλλη οι κατασκευαστές ηλεκτρικών οχημάτων , θα έχουν στην διάθεση τους ενδιαφέροντα στατιστικά , χρήσιμα δεδομένα φόρτισης , καθώς και την ενεργειακή αποτιμήση των οχημάτων που έχουν διαθέσει στους καταναλωτές . Τέλος οι ιδιοκτήτες των σταθμών φόρτισης , θα μπορούν να διαχειρίζονται και να ενημερώνουν την εφαρμογή για πιθανές αλλαγές στους χώρους parking καθώς και στα “πακέτα προσφορών και επιβραβέυσεων” που διαθέτουν, ενώ ταυτόχρονα θα έχουν στην διάθεσή τους χρήσιμα στατιστικά στοιχεία και δεδομένα για τις φορτίσεις του κάθε οχήματος στους σταθμούς τους.

## 1.2 Διεπαφές (interfaces)

### 1.2.1 Διεπαφές με εξωτερικά συστήματα

Παραθέτουμε το UML Deployment Diagram:

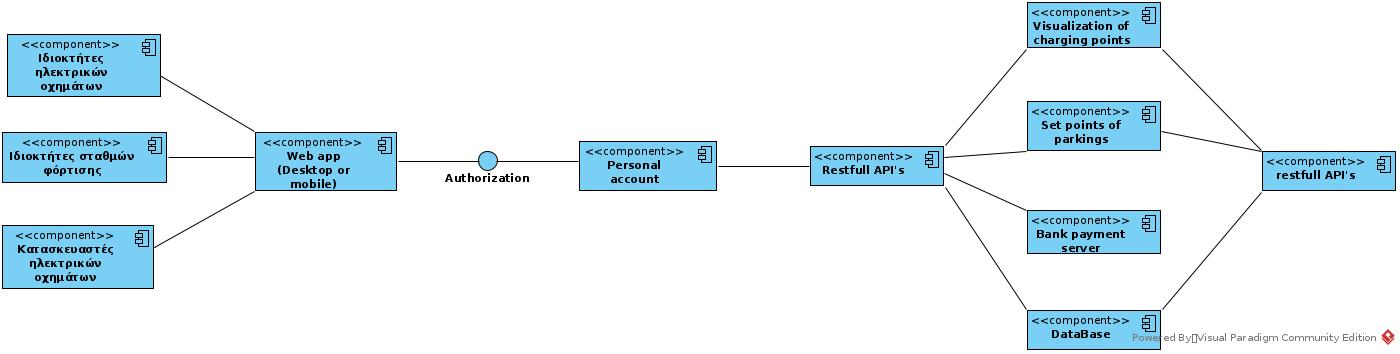


### 1.2.2 Διεπαφές με το χρήστη

Παραθέτουμε το UML Use Case Diagram:

### 

### + Component diagram



# Αναφορές - πηγές πληροφοριών

Αναφορά σε πηγές πληροφοριών στο μέτρο της αναγκαιότητας για την κατανόηση του συστήματος

ΝΑ ΜΗΝ ΣΥΜΠΛΗΡΩΘΕΙ ΑΝ ΔΕΝ ΑΠΑΙΤΕΙΤΑΙ

# Προδιαγραφές απαιτήσεων λογισμικού

## 3.1 Περιπτώσεις χρήσης

### 3.1.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 1: Φόρτιση Οχήματος

#### 3.1.1.1 Χρήστες (ρόλοι) που εμπλέκονται

Αναφορά στους ρόλους που αφορά η περίπτωση χρήσης

* Ιδιοκτήτης Αυτοκινήτου (Χρήστης)

#### 3.1.1.2 Προϋποθέσεις εκτέλεσης

Καταγραφή των συνθηκών που πρέπει να ισχύουν ώστε να μπορεί να εκτελεστεί η περίπτωση χρήσης

* Έγκυρη εγγραφή στην πλατφόρμα (Sign in) – Άπαξ
* Έγκυρη σύνδεση στην πλατφόρμα (Log in) – Κάθε φορά
* Έγκριση σταθμού φόρτισης (έλγχος χωρητικότητας δικτύου) – Κάθε φορά
* Έγκυρη τραπεζική συναλλαγή – Κάθε φορά

#### 3.1.1.3 Περιβάλλον εκτέλεσης

Η φόρτιση οχήματος πραγματοποιείται κατά βάση μέσα από την διεπαφή χρήστη/καταναλωτή του web based front end της πλατφόρμας. Ωστόσο, οι κλήσεις που δημιουργεί οδηγούν στην ανάμιξη του backend της πλατφόρμας, της βάσης δεδομένων, του συστήματος του σταθμού φόρτισης και του εξυπηρετητή της Τράπεζας.

#### 3.1.1.4 Δεδομένα εισόδου

Το σύστημα περιλαμβάνει τα εξής ζεύγη εισόδου-εξόδου και συνθήκες εγκυρότητας εξόδου ανάλογα με την κατάσταση της φόρτισης. Η έξοδος κάθε φορά λαμβάνεται υπό την προϋπόθεση ότι η είσοδος ήταν έγκυρη.

1. **Ταυτοποίηση χρήστη**

Είσοδος: Username, Password

Εγκυρότητα: Επιβεβαιώνεται από την πλατφόρμα με σύνδεση στη βάση δεδομένων.

Έξοδος: Αρχική σελίδα χρήστη,

* Προσωπικά στοιχεία χρήστη
* Στοιχεία Οχήματος
* Λίστα διαθέσιμων σταθμών φόρτισης με το αντίστοιχο κόστος φόρτισης (€/kWh) και τα υποστηριζόμενα τιμολόγια

1. **Επιλογή σταθμού φόρτισης**

Είσοδος: Επιλογή σταθμού

Εγκυρότητα: -

Έξοδος: Φόρμα συμπλήρωσης των επιθυμητών στοιχείων φόρτισης

1. **Αίτημα φόρτισης**

Είσοδος: Επιθυμητά στοιχεία φόρτισης

* Πρόγραμμα χρέωσης
* Επιθυμητό κόστος φόρτισης ή Επιθυμητό ποσό φόρτισης (σε kWh)
* Τρόπος πληρωμής

Εγκυρότητα:

* Έλεγχος από την πλατφόρμα ώστε το ποσό της φόρτισης να είναι μικρότερο ή ίσο από την χωρητικότητα της μπαταρίας. Αν είναι μεγαλύτερο, μειώνεται αυτόματα στα όρια της μπαταρίας.
* Έλεγχος από το σύστημα του σταθμού φόρτισης για την ικανότητα ανταπόκρισης του συστήματος ηλεκτρικής ενέργειας στο ύψος της ζητούμενης ποσότητας ενέργειας

Έξοδος: Σελίδα τραπεζικής πληρωμής

1. **Τραπεζική πληρωμή**

Είσοδος: Στοιχεία συναλλαγής

* Κόστος φόρτισης (εισάγεται αυτόματα σύμφωνα με το προηγούμενο βήμα)
* Ποσό φόρτισης (σε kWh) (εισάγεται αυτόματα σύμφωνα με το προηγούμενο βήμα)
* Αριθμός πιστωτικής κάρτας ή Όνομα paypal

Εγκυρότητα:

* Έλεγχος από τον εξυπηρετητή της τράπεζας για την ύπαρξη επαρκούς υπολοίπου λογαριασμού

Έξοδος: Σελίδα επιβεβαίωσης τραπεζικής πληρωμής

1. **Εξαγωγή δεδομένων φόρτισης**

Είσοδος: Επιθυμητός τύπος αρχείου (excel, pdf)

Εγκυρότητα: -

Έξοδος: Λήψη αρχείου δεδομένων φόρτισης

#### 3.1.1.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

1. **Ταυτοποίηση χρήστη**

Ο χρήστης εισάγει Username και Password. Σε περίπτωση αποτυχίας (εσφαλμένα στοιχεία), το σύστημα παρέχει κατάλληλο ενημερωτικό μήνυμα και ο χρήστης πραγματοποιεί εκ νέου προσπάθεια σύνδεσης. Εφόσον πραγματοποιηθεί επιτυχής σύνδεση, ο χρήστης μεταφέρεται στο home page της σελίδας φόρτισης. Αυτή περιλαμβάνει ένα πλαίσιο με όλα τα προσωπικά στοιχεία χρήστη (ονοματεπώνυμο, διεύθυνση κλπ.), ένα πλαίσιο με τα στοιχεία του οχήματος (μάρκα, μοντέλο, αριθμός κυκλοφορίας, κινητήρας, μπαταρία κλπ.) κι ένα πλαίσιο με τη λίστα όλων των διαθέσιμων σταθμών φόρτισης που συνεργάζονται με την πλατφόρμα. Για κάθε σταθμό φόρτισης αναγράφεται το κόστος φόρτισης (€/kWh) ανά κιλοβατώρα και το υποστηριζόμενο τιμολόγιο.

1. **Επιλογή σταθμού φόρτισης**

Ο χρήστης επιλέγει από τη λίστα των σταθμών εκείνον στον οποίο επιθυμεί να πραγματοποιήσει τη φόρτιση και όπου έχει συνδέση το αυτοκίνητό του. Αμέσως, μεταφέρεται στη σελίδα με τα στοιχεία της φόρτισης.

1. **Αίτημα φόρτισης**

Ο χρήστης πρέπει να συμπληρώσει μέσα σε μια φόρμα τα στοιχεία της φόρτισης. Αρχικά επιλέγει το πρόγραμμα χρέωσης. Κατόπιν, συμπληρώνει το ύψος της φόρτισεις. Ο χρήστης μπορεί να φορτίσει με βάση είτε το χρηματικό ποσό που επιθυμεί να διαθέσει (σε €) είτε το ποσό ενέργειας που θέλει να τροφοδοτήσει στο αυτοκίνητό του (σε kWh). Ωστόσο, η πλατφόρμα του επιτρέπει να συμπληρώσει μόνο το ένα από τα δύο πεδία διότι το σύστημα θα κάνει αυτόματα τον υπολογισμό του άλλου μεγέθους με βάση την τιμολογιακή πολιτική του σταθμού και του προγράμματος χρέωσης που δηλώθηκε στο προηγούμενο βήμα και θα τα εμφανίσει στην επόμενη σελίδα. Τέλος, επιλέγει τρόπο πληρωμής (τραπεζικός λογαριασμός ή paypal).

Μόλις υποβληθεί η φόρμα, το σύστημα θα φροντίσει να μειώσει το κόστος και το ύψος της φόρτισης σε περίπτωση που αυτό ξεπερνάει την μετρούμενη χωρητικότητα της μπαταρίας του αυτοκινήτου. Επίσης, γίνεται κι ένα ερώτημα στο σύστημα του σταθμού φόρτισης για να ελεγχθεί κατά πόσο βρίσκεται σε ετοιμότητα να ανταποκριθεί στην συγκεκριμένη ζήτηση ηλεκτρικής ενέργειας (ενδεχομένος να υπάρχουν επιπλοκές στο σύστημα παραγωγής, μεταφοράς και διανομής ηλεκτρικής ενέργειας που να καθιστούν απαγορευτική τη φόρτιση).

Εφόσον ο σταθμός εγκρίνει το αίτημα, ο χρήστης μεταφέρεται στη σελίδα τραπεζικής πληρωμής

1. **Τραπεζική πληρωμή**

Η σελίδα εμφανίζει το κόστος και το ποσό της φόρτισης σύμφωνα με τα στοιχεία που εισήχθησαν νωρίτερα (αν συμπλρηώθηκε κόστος, υπολογίζει ποσό ενέργειας και το αντίθετο) και κατόπιν των τροποποιήσεων που περιγράφηκαν στο προηγούμενο βήμα (αν η ζητούμενη ενέργεια ξεπερνάει την χωρητικότητα του αυτοκινήτου την μειώνει εμφανίζοντας και κατάλληλο μήνυμα). Αυτά τα πεδία δεν μπορούν να τροποποιηθούν παρά μόνο αν ο χρήστης επιστρέψει στην προηγούμενη σελίδα.

Το μόνο που συμπληρώνει ο χρήστης είναι ο αριθμός της πιστωτικής κάρτας ή το όνομα του paypal (το σύστημα αναμένει μόνο ένα εκ των δύο σύμφωνα με την επιλογή που έγινε στο προηγούμενο βήμα).

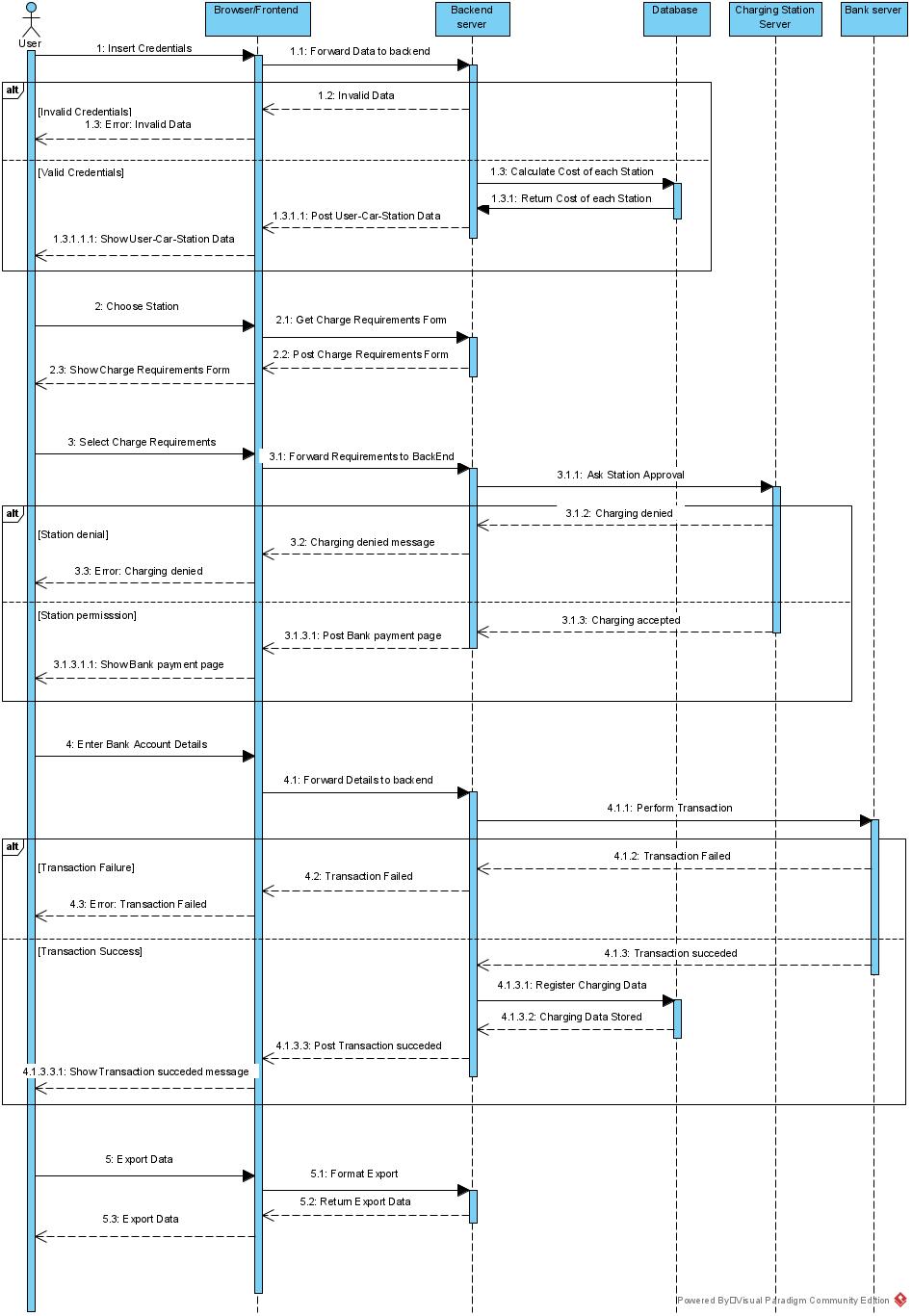
Μόλις γίνει υποβολή, η συναλλαγή προωθείται στον εξυπηρετητή της τράπεζας για τον έλεγχο ύπαρξης επαρκούς υπολοίπου στον λογαριασμό του χρήστη. Εφόσο, υπάρχει το απαιτούμενο απόθεμα, η συναλλαγή επιτυγχάνει, διαφορετικά τυπώνεται μήνυμα αποτυχίας.

1. **Εξαγωγή δεδομένων φόρτισης**

Εφόσον πραγματοποιηθεί ορθά η συναλλαγή με την τράπεζα, το σύστημα καταχωρεί αυτόματα στη βάση τη δεδομένα της φόρτισης και τα τυπώνει στην οθόνη του χρήστη.

Ο χρήστης, έχει την δυνατότητα να εξαγάγει αυτά τα δεδομένα επιλέγοντας τον τύπο του αρχείου (excel, pdf κλπ) με το οποίο επιθυμεί να γίνει η λήψη.

Η αλληλουχία των παραπάνω ενεργειών του χρήστη παρουσιάζεται στο παρακάτω ακολουθιακό διάγραμμα UML (UML Sequence Diagram):



#### 3.1.1.7 Δεδομένα εξόδου

1. **Ταυτοποίηση χρήστη**

Έξοδος: Αρχική σελίδα χρήστη,

* Προσωπικά στοιχεία χρήστη
* Στοιχεία Οχήματος
* Λίστα διαθέσιμων σταθμών φόρτισης με το αντίστοιχο κόστος φόρτισης (€/kWh) και τα υποστηριζόμενα τιμολόγια

1. **Επιλογή σταθμού φόρτισης**

Έξοδος: Φόρμα συμπλήρωσης των επιθυμητών στοιχείων φόρτισης

1. **Αίτημα φόρτισης**

Έξοδος: Σελίδα τραπεζικής πληρωμής

1. **Τραπεζική πληρωμή**

Έξοδος: Σελίδα επιβεβαίωσης τραπεζικής πληρωμής

1. **Εξαγωγή δεδομένων φόρτισης**

Έξοδος: Λήψη αρχείου δεδομένων φόρτισης

#### 3.1.1.8 Παρατηρήσεις

N/A.

### 3.1.2 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 2: Ενημέρωση (Διαχειριστών Parking)

#### 3.1.*2*.1 Χρήστες (ρόλοι) που εμπλέκονται

*Αναφορά στους ρόλους που αφορά η περίπτωση χρήσης*

Οι ρόλοι που εμπλέκονται είναι ένα υποσύνολο των διαχειριστών χώρων στάθμευσης οχημάτων, οι αναλυτές δεδομένων (Data Analysts). Εντούτοις, είναι χρήστες όπως όλοι οι υπόλοιποι αλλά με Privileged πρόσβαση σε συγκεκριμένα δεδομένα που αφορούν τους ιδιοκτήτες σταθμών (Parking) ηλεκτρικής ενέργειας.

#### 3.1.*2*.2 Προϋποθέσεις εκτέλεσης

*Καταγραφή των συνθηκών που πρέπει να ισχύουν ώστε να μπορεί να εκτελεστεί η περίπτωση χρήσης*

1. Βασική προϋπόθεση εκτέλεσης είναι η λειτουργία των υποσυστημάτων της εφαρμογής (backend και frontend) και η δυνατότητα πρόσβασης αυτών στη βάση δεδομένων.
2. Θα πρέπει ο εγγεγραμμένος χρήστης που επιθυμεί να έχει πρόσβαση στα δεδομένα των σταθμών, να έχει ταυτοποιηθεί από την υπηρεσία με χρήση κατάλληλων Username και Password.

#### 3.1.*2*.3 Περιβάλλον εκτέλεσης

*Αναφορά στο περιβάλλον στο οποίο εκτελείται η περίπτωση χρήσης. Πχ "διαδικτυακή διεπαφή χρήστη", "DBMS" κλπ*

Η ενημέρωση του αναλυτή δεδομένων γίνεται είτε μέσω της διαδικτυακής διεπαφής (online), από την σελίδα της πλατφόρμας είτε με την χρήση των REST API της εφαρμογής είτε με χρήση του CLI που παρέχεται σε εγκεκριμένους χρήστες.

#### 3.1.*2*.4 Δεδομένα εισόδου

*Καταγραφή δεδομένων εισόδου και εξόδου και συνθηκών εγκυρότητας αυτών.*

1. Ο χρήστης - Αναλυτής Δεδομένων καταχωρεί στο Login-Page τα στοιχεία του (username, password) σε μορφή string.
2. Ο χρήστης έπειτα διαλέγει την ενέργεια που επιθυμεί με χρήση του cursor του
3. Επιλέγει από drop down list με χρήση cursor τα απαραίτητα στοιχεία για τα δεδομένα που επιθυμεί.
4. Με χρήση του cursor διαλέγει αν επιθυμεί να προβάλει στην οθόνη του τα δεδομένα ή να τα λάβει τοπικά.
5. Τέλος, με χρηση cursor, ο χρήστης επιλέγει αν θέλει να αποσυνδεθεί ή να κάνει νέα κλήση δεδομένων.

#### 3.1.*2*.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).*

Βήμα 1: Ο χρήστης πραγματοποιεί είσοδο στην πλατφόρμα με χρήση στοιχείων ταυτοποίησης. Σε περίπτωση αποτυχίας, το σύστημα παρέχει κατάλληλο ενημερωτικό μήνυμα και ο χρήστης πραγματοποιεί εκ νέου προσπάθεια σύνδεσης.

Βήμα 2: Ο χρήστης επιλέγει την επιθυμητή ενέργεια μεταξύ των,

1. Υπολογισμός κατανάλωσης ενέργειας ανά λειτουργό σταθμού φόρτισης και χρονική περίοδο
2. Υπολογισμός κατανάλωσης ενέργειας ανά περιοχή και χρονική περίοδο
3. Απεικόνιση δεδομένων / γεγονότων φόρτισης σε διάγραμμα
4. Εμφάνιση αναλυτικών στοιχείων φορτίσεων για κάθε θέση φόρτισης για μια περίοδο
5. Καταγραφή γεγονότων και δεδομένων φόρτισης
6. Καταγραφή των οχημάτων που φορτίζουν σε κάθε σταθμό για δεδομένη χρονική περίοδο
7. Συλλογή δεδομένων οχήματος σε κάθε φόρτιση
8. Σύνδεση με υπηρεσίες πληρωμών
9. Σύστημα επιβράβευσης με πόντους
10. Υπολογισμός ζήτησης ενέργειας για διάφορα σενάρια οχημάτων και γεγονότων φόρτισης

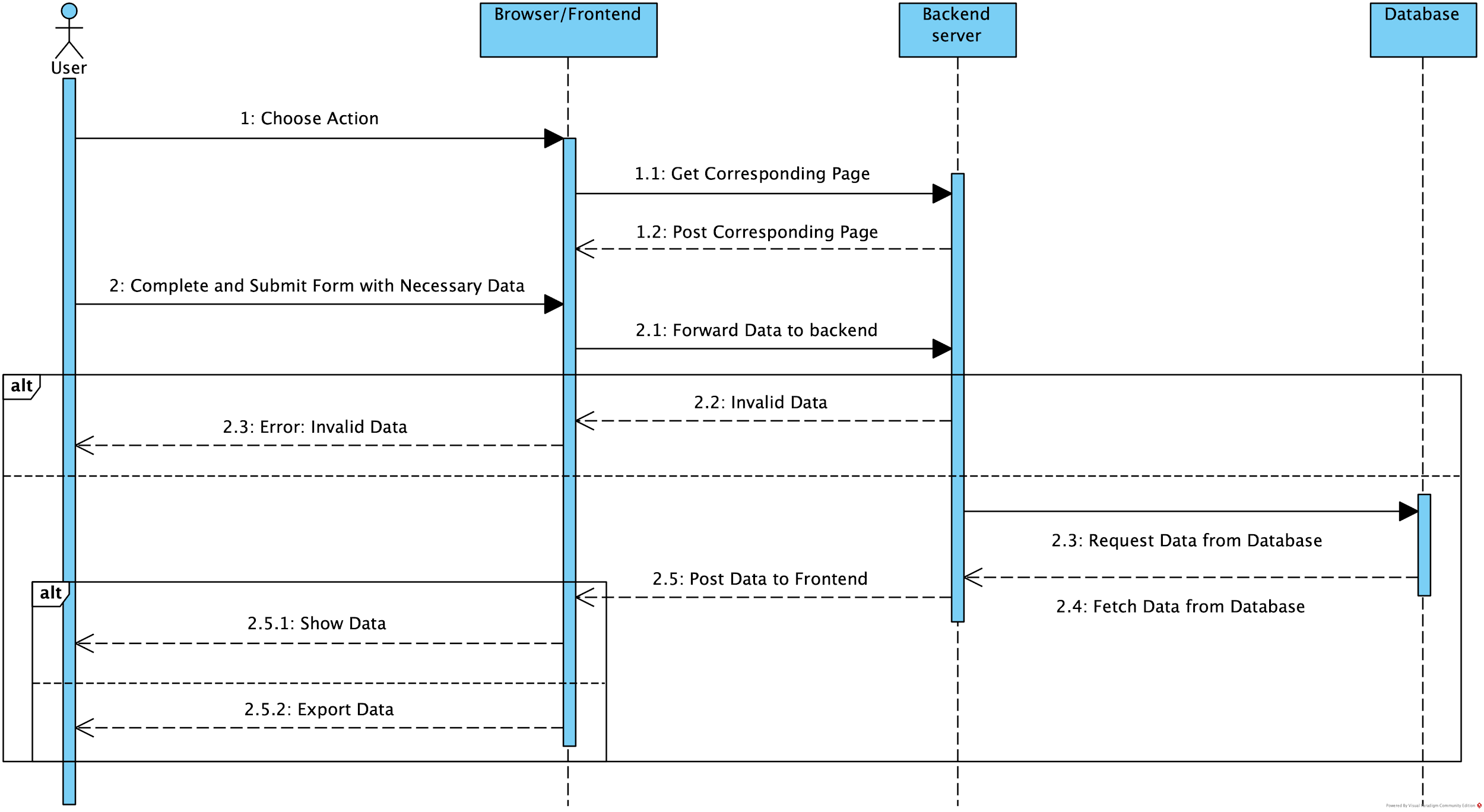
Βήμα 3: Στην σελίδα που εμφανίζεται ο χρήστης συμπληρώνει όλα τα απαραίτητα πεδία και επιλέγει υποβολή αυτών. Σε περίπτωση που κάποιο πεδίο μείνει κενό ή η τιμή κάποιου πεδίου δεν είναι έγκυρη το σύστημα παρέχει κατάλληλο μήνυμα λάθους και ο χρήστης πρέπει να συμπληρώσει ορθά όσα πεδία παρουσίασαν σφάλμα για να συνεχίσει.

Βήμα 5: Σε περίπτωση που τα πεδία συμπληρώθηκαν σωστά, το σύστημα κάνει με χρήση των Rest API μια κλήση στην βάση και ζητά τα δεδομένα που θέλει ο χρήστης.

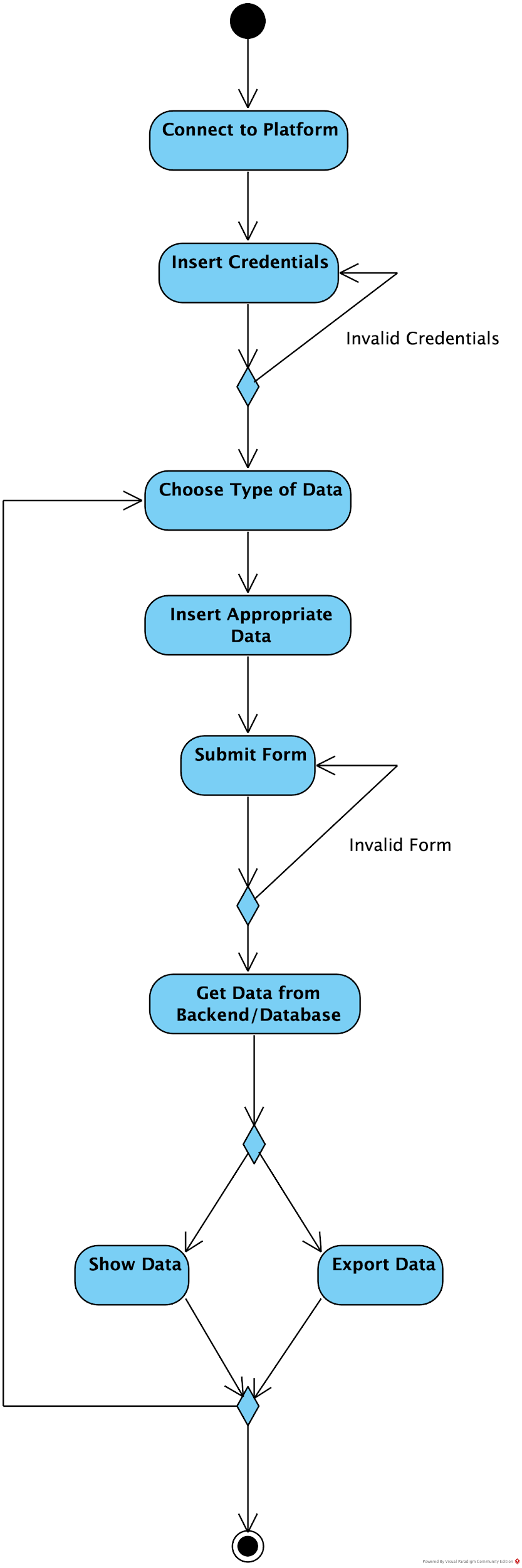
Βήμα 6: Ο χρήστης επιλέγει αν θέλει να τα προβάλει στην οθόνη του μέσω της εφαρμογής ή να τα κατεβάσει τοπικά με χρηση Button.

Βήμα 7: Τέλος, με χρηση cursor, ο χρήστης επιλέγει αν θέλει να αποσυνδεθεί ή να κάνει νέα κλήση δεδομένων.

Ακολουθεί το Sequence Diagram.



Ακολουθεί το Activity Diagram.



#### 3.1.*2*.7Δεδομένα εξόδου

Στην παραπάνω διαδικασία δεδομένα εξόδου αποτελούν τα γραφήματα - Πίνακες που δημιουργούνται σύμφωνα με την φόρμα που συμπλήρωσε ο χρήστης καθώς και τα μηνύματα επιτυχίας ή αποτυχίας που λαμβάνει ανάλογα με την έκβαση της δραστηριοτήτων (activities).

### 3.1.3 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 3: Διαχείριση σταθμών φόρτισης/ στάθμευσης (Διαχειριστών Parking)

#### 3.1.*3*.1 Χρήστες (ρόλοι) που εμπλέκονται

Όλοι οι εγγεγραμμένοι χρήστες που έχουν την ιδιότητα διαχειριστή μονάδων στάθμευσης/φόρτισης ηλεκτρικών οχημάτων

#### 3.1.*3*.2 Προϋποθέσεις εκτέλεσης

Βασική προϋπόθεση εκτέλεσης είναι η λειτουργία των υποσυστημάτων της εφαρμογής (backend και frontend) και η δυνατότητα πρόσβασης αυτών στη βάση δεδομένων. Επιπλέον, απαραίτητη είναι η ιδιότητα του εγγεγραμένου χρήστη ως διαχειριστή.

#### 3.1.*3*.3 Περιβάλλον εκτέλεσης

Ο χρήστης επιλέγει τον σταθμό, την επιθυμητή ενέργεια (προσθήκη, διαγραφή, τροποποίηση) που θέλει να επιτελέσει σε αυτόν καθώς και τα απαραίτητα δεδομένα που απαιτούνται στις περιπτώσεις της προσθήκης και της τροποποίησης. Στην συνέχεια αυτά ελέγχονται από το REST API και αν πληρούν τις προϋποθέσεις καταχωρούνται στη βάση δεδομένων.

#### 3.1.*3*.4 Δεδομένα εισόδου

Ο χρήστης συμπληρώνει ως δεδομένα εισόδου την ενέργεια που έχει σκοπό να εκτελέσει, τον αντίστοιχο σταθμό στάθμευσης/φόρτισης καθώς και τα σχετικα δεδομένα σε περίπτωση προσθήκης νέου ή τροποποίησης υπάρχοντος σταθμού.

#### 3.1.*3*.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Βήμα 1: Ο εγγεγραμμένος χρήστης επισκέπτεται την σελίδα διαχείρισης σταθμών στάθμευσης και φόρτισης ηλεκτρικών οχημάτων.

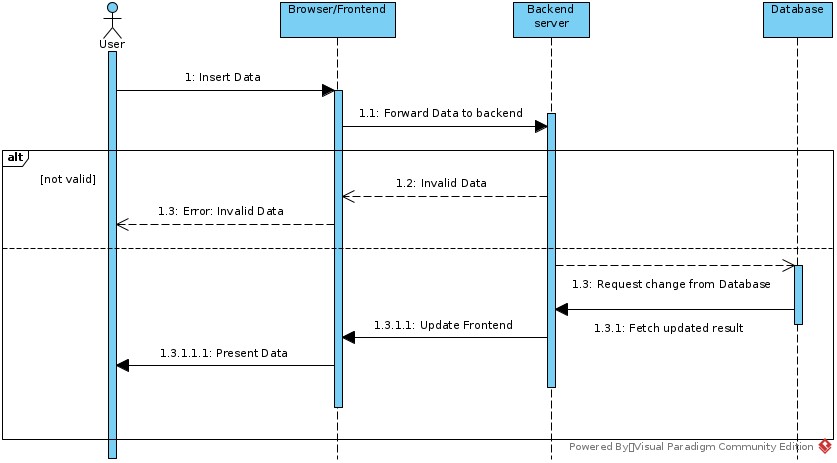
Βήμα 2: Ο χρήστης συμπληρώνει την επιθυμητή ενέργεια, επιλέγει τον αντίστοιχο σταθμό σε περίπτωση διαγραφής ή τροποποίησης και παρέχει τα απαραίτητα δεδομένα σε περίπτωση εισαγωγής ή τροποποίησης.

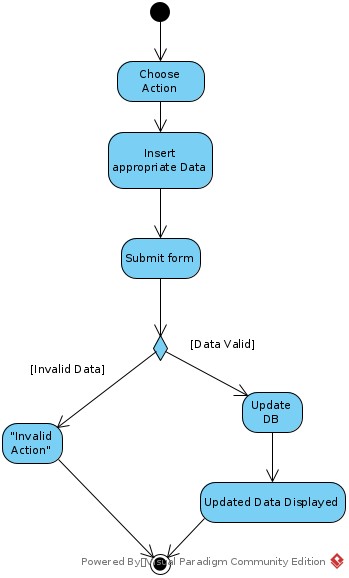
Βήμα 3: Ο χρήστης με τη χρήση κάποιας λειτουργίας του frontend (π.χ button “Καταχώρηση ενέργειας”) αποστέλει τα επιθυμητά στοιχεία στο REST API

Βήμα 4: Το REST API με την σειρά του ελέγχει αν έχουν συμπληρωθεί ορθά τα δεδομένα και τα προωθέι στην βάση δεδομένων

Βήμα 4: Αν τα στοιχεία τηρούν τις προϋποθέσεις και η καταχώριση ολοκληρώνεται επιτυχώς, αποστέλεται σχετικό μήνυμα στον χρήστη. Σε αντίθετη περίπτωση ο χρήστης λαμβάνει μήνυμα αποτυχίας.

Ακολουθεί το Sequence Diagram:



Ακολουθεί το Activity Diagram

#### 3.1.*3*.7Δεδομένα εξόδου

Στην παραπάνω διαδικασία δεδομένου εξόδου αποτελούν οι μεταβολές στην βάση δεδομένων καθώς και τα μηνύματα επιτυχίας ή αποτυχίας που λαμβάνει ο χρήστης ανάλογα με την έκβαση της καταχώρισης.

### 3.1.4 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 4: Ενημέρωση κατασκευαστών ηλεκτρικών οχημάτων

#### 3.1.4.1 Χρήστες (ρόλοι) που εμπλέκονται

Αναφορά στους ρόλους που αφορά η περίπτωση χρήσης

Οι συνεργαζόμενες εταιρείες κατασκευής ηλεκτρικών οχημάτων, μέσω κάποιων εκπροσώπων.

#### 3.1.4.2 Προϋποθέσεις εκτέλεσης

Καταγραφή των συνθηκών που πρέπει να ισχύουν ώστε να μπορεί να εκτελεστεί η περίπτωση χρήσης

* Να λειτουργούν ορθά τα υποσυστήματα της εφαρμογής (backend και frontend) και η δυνατότητα πρόσβασης αυτών στη βάση δεδομένων.
* Να έχει δημιουργηθεί εταιρικό προφίλ για την εκάστοτε εταιρεία κατασκευής ηλεκτρικών οχημάτων.
* Ο εκπρόσωπος της εταιρείας να ταυτοποιηθεί και να συνδεθεί στην πλατφόρμα με τα κατάλληλα στοιχεία.

#### 3.1.4.3 Περιβάλλον εκτέλεσης

Αναφορά στο περιβάλλον στο οποίο εκτελείται η περίπτωση χρήσης. Πχ "διαδικτυακή διεπαφή χρήστη", "DBMS" κλπ

Ο εκπρόσωπος της εταιρείας αποκτά πρόσβαση στα δεδομένα που τον αφορούν και ενημερώνεται είτε μέσω της αντίστοιχης διαδικτυακής διεπαφής. Η εφαρμογή επικοινωνεί μέσω του REST API με τη διεπαφή αυτή και με τη βάση δεδομένων και με βάση τις εισόδους του χρήστη πραγματοποιεί τις κατάλληλες ενέργειες και εμφανίζει τα τελικά αποτελέσματα.

#### 3.1.4.4 Δεδομένα εισόδου

Καταγραφή δεδομένων εισόδου και εξόδου και συνθηκών εγκυρότητας αυτών.

* Τα στοιχεία ταυτοποίησης (λογαριασμός, κωδικός) του χρήστη κατά τη σύνδεση στην πλατφόρμα. Για να υπάρξει επιτυχής σύνδεση θα πρέπει τα στοιχεία αυτά να είναι έγκυρα, δηλαδή ο λογαριασμός να υπάρχει και να έχει τα κατάλληλα προνόμια και ο κωδικός να είναι σωστός.
* Η επιλογή, μέσω drop down λίστας, των δεδομένων που επιθυμεί να δει ο χρήστης.
* Η (δυνητική) επιλογή, μέσω drop down λίστας, των κριτηρίων με βάση τα οποία ο χρήστης επιθυμεί να φιλτράρει τα δεδομένα.
* Η (δυνητική) επιλογή για κατέβασμα των δεδομένων.

#### 3.1.4.5 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).

Βήμα 1. Ο χρήστης εισάγει τα στοιχεία ταυτοποίησής του στην πλατφόρμα. Το σύστημα επαληθεύει τα στοιχεία αυτά και σε περίπτωση επιτυχίας εμφανίζει στη διεπαφή χρήστη τις επιλογές δεδομένων. Σε περίπτωση αποτυχίας το σύστημα επιστρέφει μήνυμα λάθους.

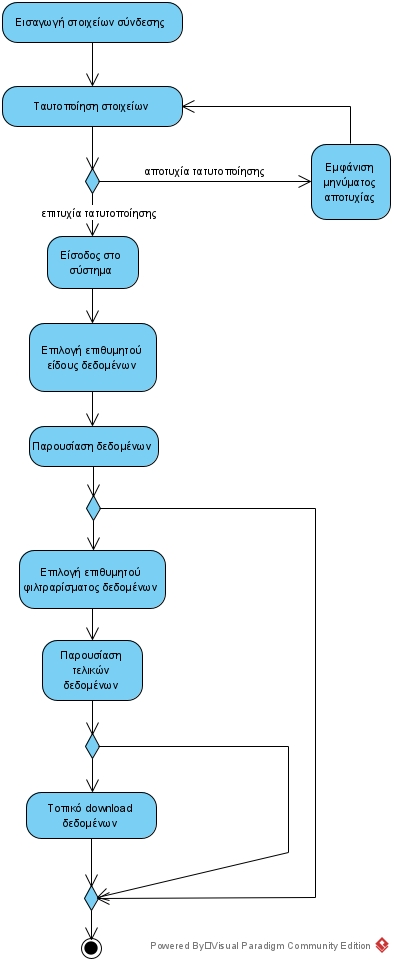
Βήμα 2. Ο χρήστης επιλέγει το είδος των δεδομένων που επιθυμεί να δει μεταξύ των παρακάτω:

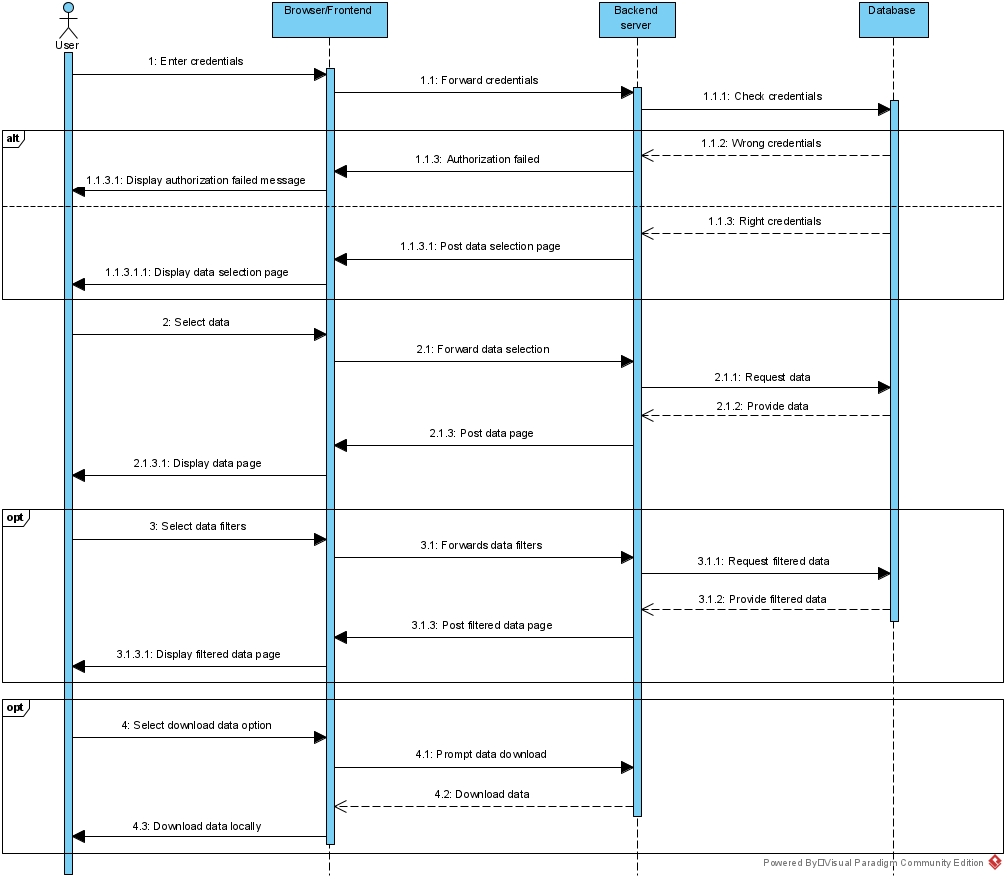
* Ενέργεια που καταναλώθηκε ανά κατηγορία οχήματος για δεδομένη χρονική περίοδο
* Ενεργειακό κόστος κάθε οχήματος ανά χιλιόμετρο και χρονική περίοδο
* Γεγονότα φόρτισης των οχημάτων του κατασκευαστή ανά περιοχή και περίοδο

Το σύστημα επικοινωνεί με τη βάση δεδομένων και εμφανίζει στη διεπαφή χρήστη τα κατάλληλα δεδομένα.

Βήμα 3. Ο χρήστης επιλέγει, εφόσον το επιθυμεί, κριτήρια με βάση τα οποία θέλει να φιλτράρει τα δεδομένα. Το σύστημα επικοινωνεί με τη βάση δεδομένων και εμφανίζει στη διεπαφή χρήστη τα νέα δεδομένα.

Βήμα 4. Ο χρήσης επιλέγει, αν το επιθυμεί, να κατεβάσει τα δεδομένα τοπικά στον υπολογιστή του. Το σύστημα πραγματοποιεί το download.





#### 3.1.4.7 Δεδομένα εξόδου

Διαγράμματα UML αλληλουχίας για την παραγωγή δεδομένων εξόδου. Ως δεδομένα εξόδου νοούνται όλα τα δεδομένα του συστήματος τα οποία δημιουργούνται ή μεταβάλλονται κατά την εκτέλεση (αν υπάρχουν τέτοια)

Δεδομένα εξόδου της παραπάνω διαδικασίας αποτελούν τα γραφήματα και οι πίνακες με τα δεδομένα που εμφανίζονται στη διεπαφή χρήστη, καθώς και τα αρχεία με τα δεδομένα τα οποία ο χρήστης δύναται να κατεβάσει. Επιπλέον, έξοδο αποτελεί και το πιθανό μήνυμα λάθους σε περίπτωση αποτυχίας σύνδεσης.

#### 3.1.4.8 Παρατηρήσεις

Ο,τι δεν εντάσσεται στα προηγούμενα, εφόσον υπάρχει

N/A

### 3.1.5 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 5: Πληρωμή

#### 3.1.*5*.1 Χρήστες (ρόλοι) που εμπλέκονται

Αναφορά στους ρόλους που αφορά η περίπτωση χρήσης

Συμμετέχει αποκλειστικά ο ιδιοκτήτης του αυτοκινήτου, ο οποίος επιθυμεί να πραγ-ματοποιήσει μία πληρωμή.

#### 3.1.*5*.2 Προϋποθέσεις εκτέλεσης

Καταγραφή των συνθηκών που πρέπει να ισχύουν ώστε να μπορεί να εκτελεστεί η περίπτωση χρήσης

Απαιτείται ο χρήστης να είναι εγγεγραμμένος και να διαθέτει μοναδικό Username και Password, προκειμένου να χρησιμοποιήσει την εφαρμογή. Μόνο μέσω του προσωπι-κού αυτού λογαριασμού μπορεί να διαχειριστεί και να εκτελέσει την πληρωμή. Γιαυ-τό αρχικά πραγματοποιείται σύνδεση.

#### 3.1.*5*.3Περιβάλλον εκτέλεσης

Αναφορά στο περιβάλλον στο οποίο εκτελείται η περίπτωση χρήσης. Πχ "διαδικτυακή διεπαφή χρήστη", "DBMS" κλπ

Η προσθήκη γεγονότος πληρωμής πραγματοποιείται από τη διαδικτυακή διεπαφή με το χρήστη, από τη σελίδα νέας πληρωμής ή μέσω του API της εφαρμογής.

#### 3.1.*5*.4Δεδομένα εισόδου

Καταγραφή δεδομένων εισόδου και εξόδου και συνθηκών εγκυρότητας αυτών.

* Υπάρχουν τρία επίπεδα εισόδου, με αντίστοιχα πεδία συμπλήρωσης στοιχείων, των οποίων η εγκυρότητα επιβεβαιώνεται από την πλατφόρμα. Στο πρώτο σημείο ο χρή-στης εισάγει τα προσωπικά του στοιχεία προκειμένου να συνδεθεί στην εφαρμογή, δηλαδή το Username και το Password. Στο δεύτερο σημείο ο χρήστης εισάγει τα δε-δομένα της πληρωμής. Συγκεκριμένα συμπληρώνει ημερομηνία πληρωμής, id οχήμα-τος, σημείο φόρτισης, καθώς και ποσό πληρωμής, αναλόγως τον λογαρισμό που θέ-λει να εκδόσει. Ο λογαριασμός μπορεί να είναι στιγμιαίος, αλλά και περιοδικός, δη-λαδή να αφορά τα έξοδα ενός οχήματος σε διάστημα ενός μήνα. Στο τρίτο, τελευταίο και προαιρετικό σημείο, ο χρήστης εισάγει τα δεδομένα της credit κάρτας του, στην περίπτωση που επιθυμεί να πληρώσει με αυτόν τον τρόπο. Η εγκυρότητα των παρα-πάνω στοιχείων ελέγχεται με τον εξής τρόπο :
* Sign In : Από τη βάση δεδομένων των εγγεγραμμένων χρηστών
* Payment Data : Από τους αντίστοιχους περιορισμούς που επιβάλλουμε, για παράδειγμα η ημερομηνία πληρωμής πρέπει να έχει την μορφή dd/mm/yy

Credit Card : Από τη βάση δεδομένων της αντίστοιχης τράπεζας

#### 3.1.*5*.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).

Αρχικά ο χρήστης συνδέεται μέσω του προσωπικού του λογαριασμού στην εφαρμο-γή, συμπληρώνοντας έγκυρα τα πεδία του Username και του Password. Στη συνέχεια, του παρέχονται δύο λειτουργίες :

1. Πραγματοποίηση νές πληρωμής
2. Παρακολούθηση παρελθοντικών πληρωμών

Στην δεύτερη και απλή επιλογή, ο χρήστης εύκολα πληροφορείται πλήρως για τα δε-δομένα των παλαιότερων πληρωμών του, και φυσικά μπορεί να επιστρέψει στο αρ-χικό μενού. Στην πρώτη επιλογή, η πληρωμή μπορεί να είναι είτε στιγμιαία, δηλαδή να αφορά απλοκλειστικά και μόνο μια φόρτιση που μόλις ολοκληρώθηκε, είτε να έχει περιοδικό χαρακτήρα, δηλαδή να σχετίζεται με την μηνιαία συνδρομή που διαθέτει ο χρήστης. Για κάθε μία από τις δύο περιπτώσεις, είναι αναγκαία η συμπλήρωση των δεδομένων την πληρωμής, δηλαδή :

1. Ημερομηνία πληρωμής
2. Όχημα (id)
3. Σημείο φόρτισης (πρατορείο)
4. Ποσό πληρωμής

Τέλος, ο χρήστης έχει τη δυνατότητα να πληρώσει μέσω μετρητών, κάρτας, ή bonus πόντων. Κατά την πληρωμή με κάρτα, είναι απαραίτητο να συμπληρωθούν τα αντί-στοιχα υποχρεωτικά πεδία.

#### 

#### 

#### 3.1.*5.*7 Δεδομένα εξόδου

Διαγράμματα UML αλληλουχίας για την παραγωγή δεδομένων εξόδου. Ως δεδομένα εξόδου νοούνται όλα τα δεδομένα του συστήματος τα οποία δημιουργούνται ή μεταβάλλονται κατά την εκτέλεση (αν υπάρχουν τέτοια)

Με την επιτυχή συμπλήρωση των στοιχείων της πληρωμής από το χρήστη, προστίθε-ται στη βάση δεδομένων ένα νέο γεγονός πληρωμής. Κατά την επιλογή παρακολού-θησης όλων των παρελθοντικών πληρωμών, φορτώνονται στο χρήστη όλα τα γεγονό-τα πληρωμών που τον αφορούν, καθώς ίσως και κάποια στατιστικά διαγράμματα αυ-τών.

#### 3.1.*5*.8 Παρατηρήσεις

Ο,τι δεν εντάσσεται στα προηγούμενα, εφόσον υπάρχει

N/A

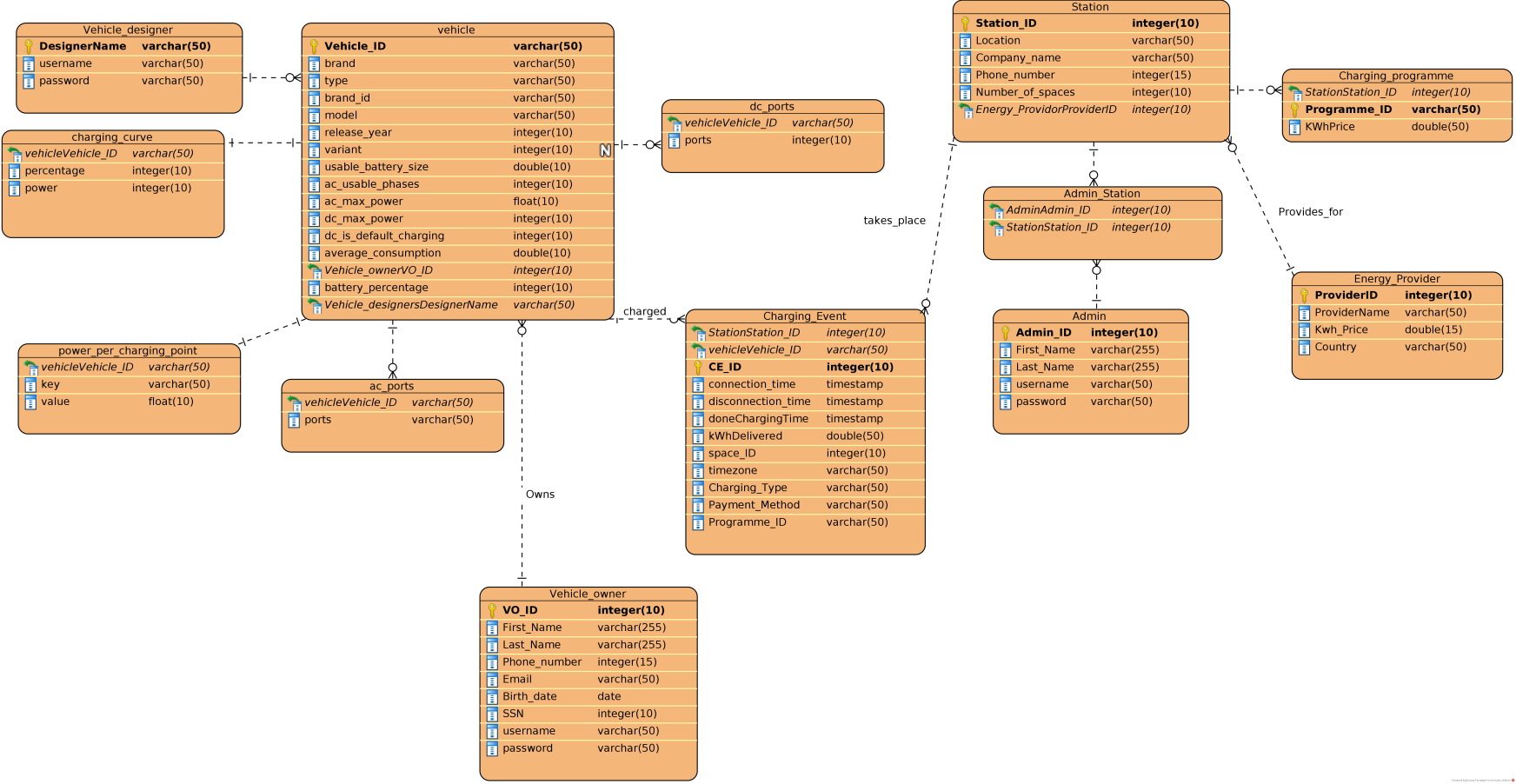
## 3.2 Απαιτήσεις επιδόσεων

*Ποσοτική τεκμηρίωση μέτρων και κριτηρίων επιθυμητών επιδόσεων με αναφορά στα ποσοτικά χαρακτηριστικά εισόδων και φορτίου του λογισμικού.*

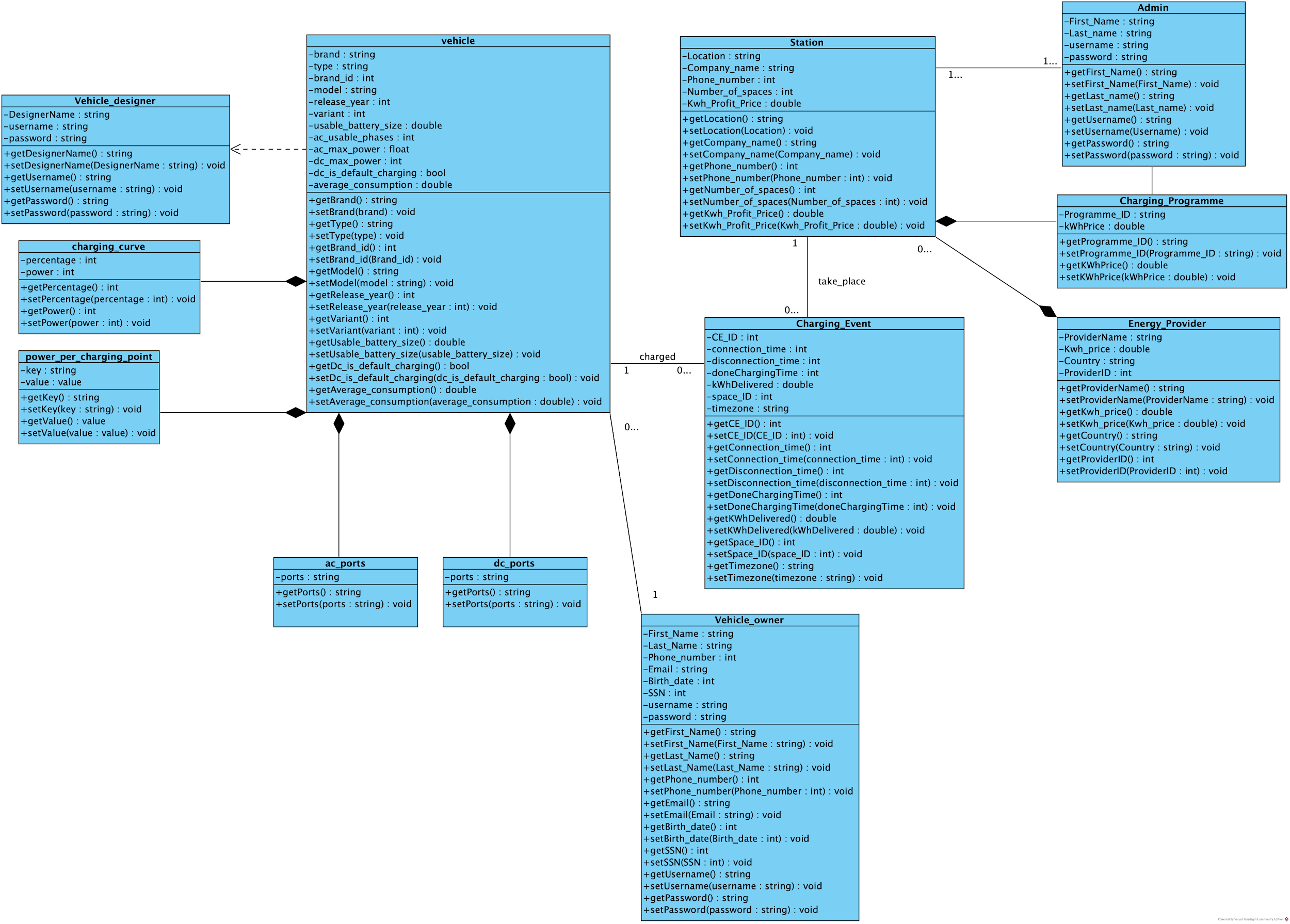
Το λογισμικό μας θα είναι συνεχώς διαθέσιμο σε αυτούς που επιθυμούν να το χρησιμοποιήσουν. Αναμένουμε σε ώρες αιχμής φορτίο 500 ταυτόχρονα συνδεδεμένων χρηστών τους οποίους και μπορούμε να διαχειριστούμε. Επιπλέον, το REST API μας μπορεί να καλύψει 50 requests το δευτερόλεπτο προς όλα τα resources. Όσον αφορά τους χρόνους απόκρισης, η σύνδεση χρήστη δε θα πρέπει να υπερβαίνει τα 2 δευτερόλεπτα, ενώ οι εμφανίσεις των δεδομένων στα αντίστοιχα use cases δε θα πρέπει να υπερβαίνουν τα 3 δευτερόλεπτα.

## 3.3 Απαιτήσεις οργάνωσης δεδομένων

Entity Relation Diagram



Class Diagram



### 3.3.1 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

*Απαιτήσεις πρόσβασης και περιορισμοί.*

Κάθε τύπος πρόσβασης (Parking - Data Analyst, Parking - Admin,...) έχει περιορισμένη πρόσβαση στα δεδομένα αναλόγως τα Privileges που του ανήκουν. Όσων αφορά τα προσωπικά δεδομένα όλων των χρηστών (username, password,...), είναι ορατά μόνο από τους διαχειριστές της πλατφόρμας και φυλάσσονται σε ξεχωριστή βάση δεδομένων από αυτήν που έχουν πρόσβαση οι χρήστες.

## 3.4 Περιορισμοί σχεδίασης

*Λεπτομερής τεχνική τεκμηρίωση των περιορισμών σχεδίασης οι οποίοι επιβάλλονται από απαιτήσεις συμμόρφωσης σε πρότυπα, κανονισμούς, ή άλλους περιορισμούς του έργου. Περιλαμβάνεται η πολιτική ονοματολογίας οντοτήτων δεδομένων και πεδίων. Τέτοιοι περιορισμοί μπορεί να επιβάλλονται από τη χρήση βιβλιοθηκών, frameworks, περιβαλλόντων ανάπτυξης κλπ*

*Front-End*

●Για τη μορφοποίηση, τη διαδραστικότητα και τη γενική διαμόρφωση της ιστοσελίδας θα γίνει η χρήση  Javascript, η οποία συνεπάγεται κάποιους περιορισμούς στην ταχύτητα, ανάλογα με τη χρήση της, καθώς και τη μεταβλητότητα που μπορεί να παρουσιάσει σε διαφορετικούς browsers.

● Στην συγγραφή και οργάνωση του κώδικα Javascript αξιοποιούμε τη βιβλιοθήκη React, η οποία μας παρέχει μεγαλύτερη ευκολία στην δημιουργία του front-end σε σύγκριση με απλό κώδικα Javascript. Η ευκολία αυτή προκύπτει από τη μορφοποίηση και τις διευκολύνσεις που προσφέρει το React επιτρέποντας μεγαλύτερη ταχύτητα συγγραφής του κώδικα καθώς και την ύπαρξη υποστήριξης για αναδυόμενα ερωτήματα λόγω της ευρείας χρήσης της. Επίσης, η χρήση του αντικειμενοστραφούς προγραμματισμού εντός του πλαισίου της React προσφέρει κάποια επιπλέον πλεονεκτήματα όπως η ξεκάθαρη οργάνωση και δομή του κώδικα, η ευελιξία μέσω του πολυμορφισμού κλπ.

Back-End

● Χρήση της γλώσσας Java και Apache Groovy για την υλοποίηση του back-end της εφαρμογής.

● Χρήση του Git (και πιο συγκεκριμένα της πλατφόρμας Github) για την διαδικασία version control του συστήματος.

● Χρήση του εργαλείου αυτόματου χτισίματος Gradle για αυτοματισμό της επίλυσης εξαρτήσεων λογισμικού εντός του project και αυτόματη εκτέλεση των τεστ λειτουργικότητας.

● Σαν test framework για τις κλάσεις που απαιτείται χρησιμοποιείται το Spock λόγω της ευελιξίας του. Ενδεχόμενη χρήση και του γνωστού framework JUnit.

● Συγγραφή δοκιμαστικού (test) κώδικα πριν τον κώδικα κλάσης όπου αυτός απαιτείται.

## 3.5 Λοιπές απαιτήσεις

### 3.5.1 Απαιτήσεις διαθεσιμότητας λογισμικού

*Η εφαρμογή είναι επιθυμητό να είναι διαθέσιμη οποιαδήποτε ώρα της ημέρας για όλα τα είδη χρηστών.*

* Χρήση αντιγράφων της βάσης δεδομένων και εναλλακτικών servers οι οποίοι θα υποστηρίζουν την κίνηση της πλατφόρμας σε περίπτωση συντήρησης ενός server. Αυτό είναι επιθυμητό σε όλες τις φάσεις υιοθέτησης της πλατφόρμας καθώς στο αρχικό στάδιο οι χρήστες θα είναι λίγοι αλλά οι ανάγκες σε αλλαγές και βελτιώσεις του λογισμικού πολλές ενώ στη συνέχεια η βάση χρηστών θα είναι μεγαλύτερη.
* Συνέπεια στην δυνατότητα πρόσβασής της.
* Συμφωνία με πάροχο υπηρεσιών διαδικτύου, ο οποίος μπορεί να παρέχει πλήρη διαθεσιμότητα μέσω των δικτυακών εγκαταστάσεών του στο διαδίκτυο.

### 3.5.2 Απαιτήσεις ασφάλειας

*Τεκμηρίωση απαιτήσεων ασφαλείας*

*Απαιτούνται:*

* Ασφαλή κανάλια για τη σύνδεση των χρηστών.
* Χρήση HTTPS και SSL certificate.
* Αυτόματη έξοδος χρηστών ανενεργών για μεγάλο χρονικό διάστημα.
* Κωδικοί πρόσβασης και άλλα ευαίσθητα προσωπικά δεδομένα δεν θα πρέπει να εμφανίζονται ή να αποθηκεύονται σε υπολογιστές χρηστών μέσω cookies ή με οποιοδήποτε άλλον τρόπο.
* Οι κωδικοί πρόσβασης θα πρέπει να είναι κρυπτογραφημένοι στη  βάση δεδομένων.

### 3.5.3 Απαιτήσεις συντήρησης

*Τεκμηρίωση απαιτήσεων συντήρησης*

* Λόγω της στατικής φύσης των δεδομένων, αρκεί να μπορεί το σύστημα να τροφοδοτηθεί από κάποιο backup αποθηκευτικό χώρο.
* Συντήρηση του συστήματος θα εκτελείται κάθε 3 μήνες και σε έκτακτες περιστάσεις. Η συχνότητα αυτή υπόκειται σε αλλαγές ανάλογα με τον αριθμό χρηστών και την απόδοση του συστήματος.