Έγγραφο απαιτήσεων λογισμικού(SRS)

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

**ElectronCharge**

# Εισαγωγή

## Εισαγωγή: σκοπός του λογισμικού

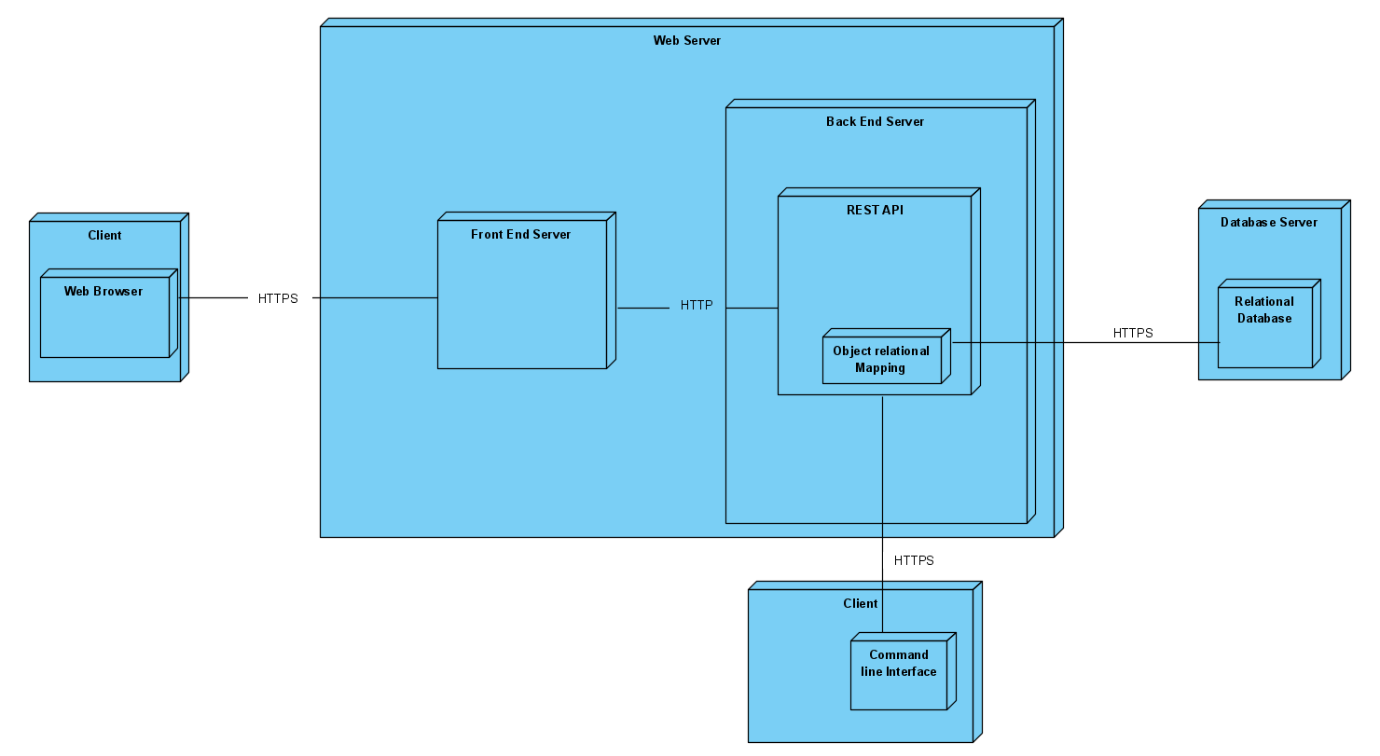
Σκοπός μας είναι η ανάπτυξη μιας εφαρμογής που απευθύνεται στους χρήστες-οδηγούς ηλεκτρικών οχημάτων με στόχο την ολοκληρωμένη παροχή υπηρεσιών για την φόρτιση του οχήματος τους

## 1.2 Διεπαφές (interfaces)

### 1.2.1 Διεπαφές με εξωτερικά συστήματα

Το σύστημα χρησιμοποιεί ως διεπαφή με τη βάση δεδομένων το πρότυπο κλήσεων REST API. Back End και Front End server κατοικούν στον ίδιο web server για τα πλαίσια της εφαρμογής και μεταξύ Back End και Βάσης δεδομένων χρησιμοποείται ένα επίπεδο Object relational mapping τύπου SQLAlchemy. Για βάση δεδομένων χρησιμοποιούμε τo σχεσιακό DBMS MySQL το οποίο ακολουθεί το πρωτόκολλο MySQL Protocol μεταξύ MySQL client και MySQL server. Το μορφότυπο ανταλλαγής δεδομένων μέσω του RESTAPI είναι προκαθορισμένα JSON και προαιρετικά CSV ενώ για την εξουσιοδότηση των χρηστών (authorization) χρησιμοποιούνται JSON Web tokens. Τέλος η μεταφορά δεδομένων πάνω απ το διαδίκτυο γίνεται με HTTPS ενώ η μεταφορά δεδομένων μεταξύ Back End και Front End γίνεται μέσω απλού HTTP

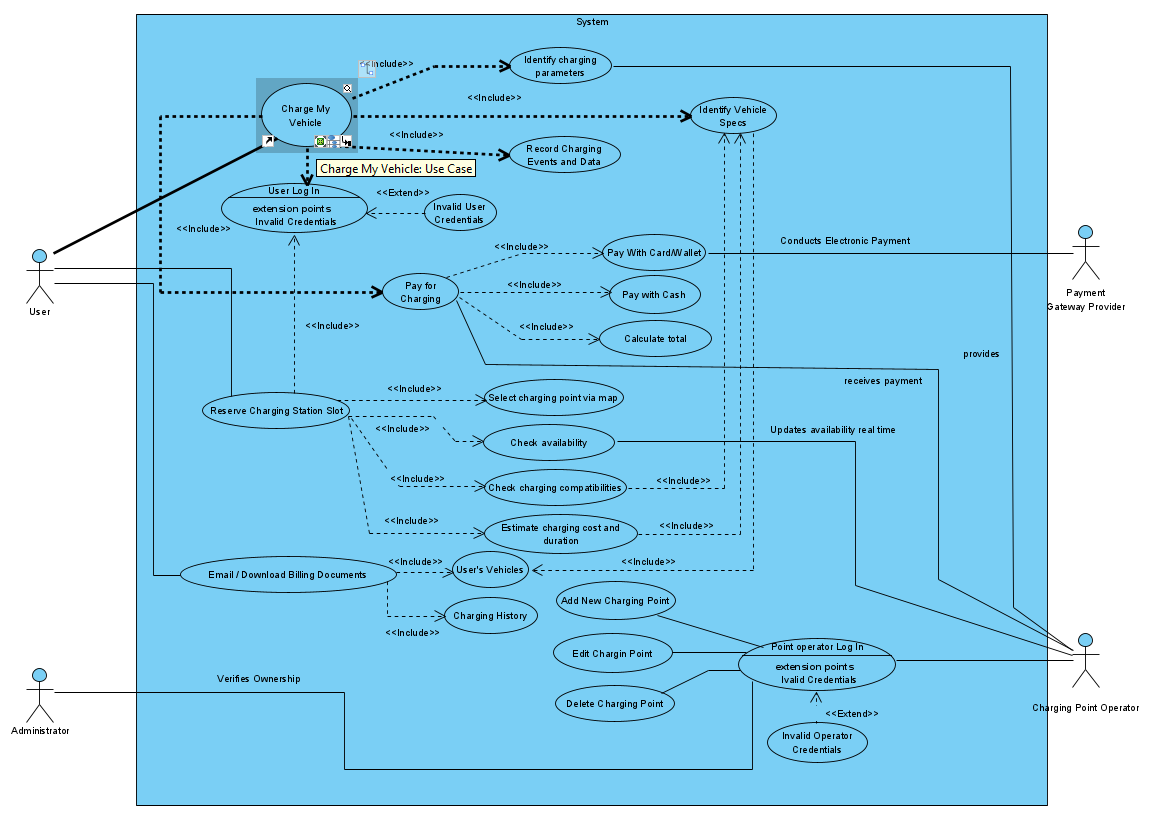
Αυτά οπτικοποιούνται στο παρακάτω UML Deployment diagram:



Το λογισμικό που υλοποιούμε χρησιμοποιεί επίσης react google maps (διεπαφή με εξωτερικό σύστημα), ώστε να απεικονίζονται στο χρήστη τα διαθέσιμα charging points στο χάρτη.

### 1.2.2 Διεπαφές με το χρήστη

Προδιαγραφή διεπαφών με το χρήστη. Μοντέλο Use Case (UML).



Ο χρήστης (είτε ως χρήστης είτε ως administrator) χρησιμοποιεί ως διεπαφή την επιθυμητή εφαρμογή περιήγησης στο διαδίκτυο (Web application). Μέσω αυτής μπορεί να περιηγηθεί και να αξιοποιήσει τις λειτουργίες της εφαρμογής (front end και cli) .

# Αναφορές - πηγές πληροφοριών

Ν/Α.

# Προδιαγραφές απαιτήσεων λογισμικού

## 3.1 Περιπτώσεις χρήσης

### 3.1.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 1: Επεξεργασία δεδομένων charging point

3.1.1.1 Χρήστες (ρόλοι) που εμπλέκονται

Χειριστές (Operators) σημείων φόρτισης, Διαχειριστές συστήματος (για πιστοποίηση αδειών)

3.1.1.2 Προϋποθέσεις εκτέλεσης

Ο χειριστής σημείου θα πρέπει να καταχωρήσει valid στοιχεία ώστε να υποβληθεί επιτυχώς το αίτημα για την ανανέωση στοιχείων υπάρχοντος σημείου, την προσθήκη νέου ή τη διαγραφή

3.1.1.3 Περιβάλλον εκτέλεσης

Ο χειριστής σημείου συμπληρώνει τα στοιχεία στη διαδικτυακή διεπαφή του συμπεριλαμβάνοντας και την άδεια σημείου φόρτισης και στη συνέχεια αυτά ελέγχονται από το REST API. Αν είναι valid σύμφωνα με ορισμένους περιορισμούς καταχωρούνται στην βάση δεδομένων ως pending αιτήματα.

3.1.1.4 Δεδομένα εισόδου

Ο ιδιοκτήτης καταχωρεί το αίτημά του.

Αν πρόκειται για προσθήκη σημείου συμπληρώνει την κατάλληλη φόρμα με τα στοιχεία του σημείου και του τίτλου ιδιοκτησίας.

Αν πρόκειται για ενημέρωση σημείου ο χειριστής επεξεργάζεται είτε την φόρμα με τα στοιχεία του σημείου είτε τον τίτλο ιδιοκτησίας.

Αν πρόκειται για διαγραφή σημείου φόρτισηςκάνει το αντίστοιχο αίτημα επιλέγοντας απ τα διαθέσιμα σε αυτόν σημεία.

3.1.1.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Η αλληλουχία ενεργειών περιγράφεται και οπτικά με τα διαγράμματαUML που παρατίθενται.

Βήμα 1: Επιλογή επεξεργασίας υπάρχοντος σημείου φόρτισης ή δημιουργίας νέου

Βήμα 2α: Συμπλήρωση φόρμας για δημιουργία νέου σημείου ή εισαγωγή των στοιχείων από αρχείο

Βήμα 3α : Συμπλήρωση τίτλου ιδιοκτησίας

Βήμα 4α : Ο ιδιοκτήτης με το button “Submit” αποστέλλει τα επιθυμητά αρχεία στο REST API

Βήμα 5α: Το REST API αποθηκεύει τα αρχεία στη βάση δεδομένωνως pending

Βήμα 6α : Εμφανίζεται στον χειριστή σελίδα - μήνυμα επιτυχούς καταχώρησης αιτήματος και επιστρέφει στην αρχική οθόνη.

Βήμα 2β :Επεξεργασία υπάρχοντος σημείου – Επιλογή διαγραφής ή ανανέωσης των πληροφοριών

Βήμα 3β1: Αίτημα διαγραφής σημείου

Βήμα 4β1: Ο ιδιοκτήτης με το button “Confirm Deletion” αποστέλλει το αίτημα στο REST API

Βήμα 4β2: Το REST API με τη σειρά του διαγράφει από τη βάση δεδομένων τα στοιχεία του σημείου

Βήμα 3β2: Αίτημα ανανέωσης πληροφοριών

Βήμα 4β2: Ο ιδιοκτήτης επιλέγει να επεξεργαστεί τη φόρμα με τα στοιχεία του σημείου

Βήμα 5β2: Το REST API αναζητά στη βάση δεδομένων το αντίστοιχο υπάρχον αρχείο και το εμφανίζει σε επεξεργάσιμη μορφή στον ιδιοκτήτη

Βήμα 6β2: Ο ιδιοκτήτης επεξεργάζεται τα στοιχεία του σημείου στη φόρμα και με το κουμπί“Submit” αποστέλλει τα επιθυμητά αρχεία στο REST API

Βήμα 7β2: Το REST API ελέγχει ότι τα στοιχεία που υπέβαλε ο ιδιοκτήτης είναι σωστά και τα καταχωρεί στη βάση δεδομένων αλλιώς τον ανακατευθύνει στην επιλογή της επεξεργασίας στοιχείων σημείου

Βήμα 8β2 : Εμφανίζεται στο ιδιοκτήτη σελίδα - μήνυμα επιτυχούς καταχώρησης αιτήματος και επιστρέφει στην αρχική οθόνη.

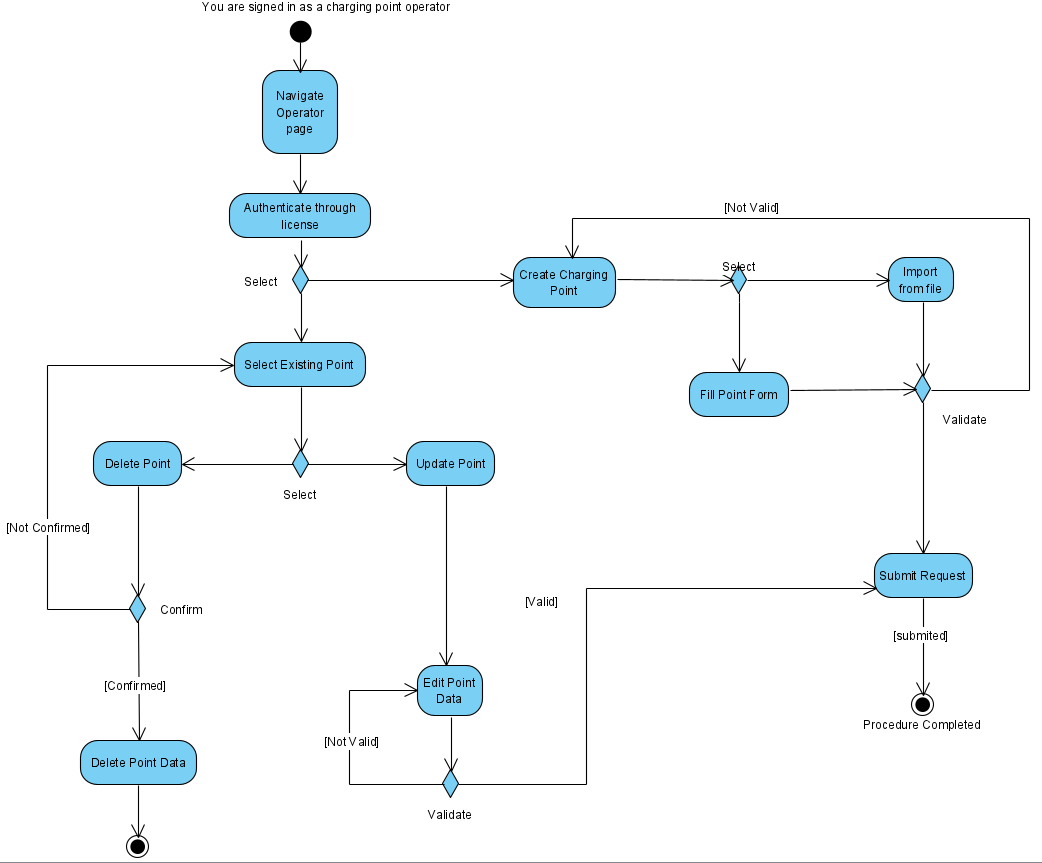
Βήμα 4β2’ : Ο ιδιοκτήτης επιλέγει να επεξεργαστεί τον τίτλο ιδιοκτησίας

Βήμα 5β2’: Το REST API αναζητά στη βάση δεδομένων το αντίστοιχο υπάρχον αρχείο και το εμφανίζει σε επεξεργάσιμη μορφή στον ιδιοκτήτη

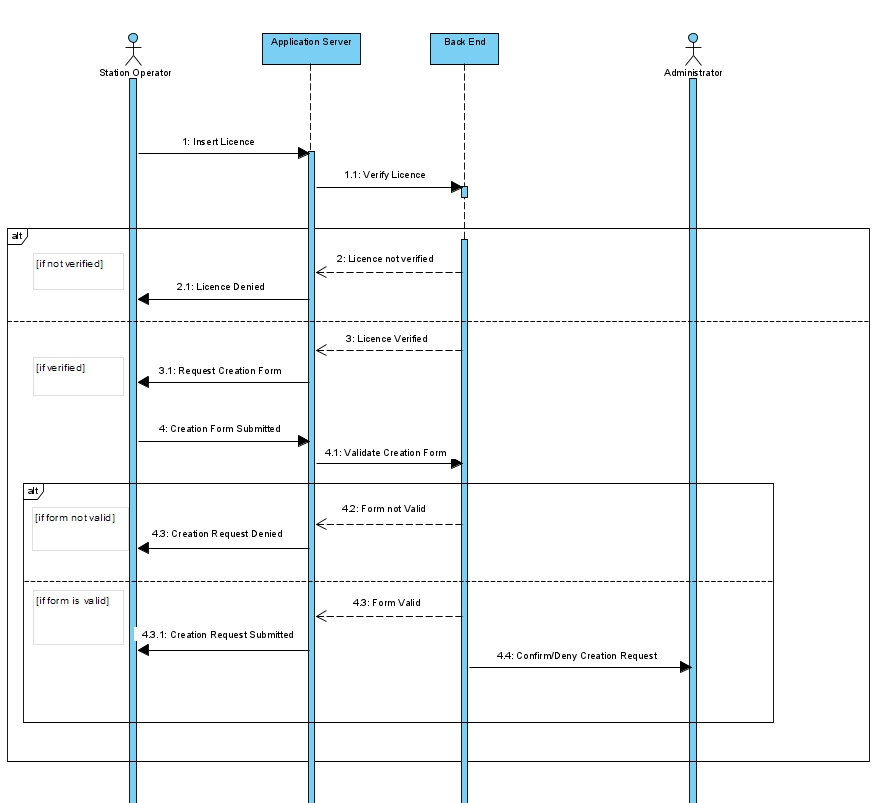
Βήμα 6β2’: Ο ιδιοκτήτης επεξεργάζεται τις πληροφορίες και με το κουμπί «Submit” αποστέλλει τα επιθυμητά αρχεία στο REST API

Βήμα 7β2’: Το REST API αποθηκεύει τα αρχεία στη βάση δεδομένων ως pending

Βήμα 8β2 : Εμφανίζεται στον ιδιοκτήτη σελίδα - μήνυμα επιτυχούς καταχώρησης αιτήματος και επιστρέφει στην αρχική οθόνη.



#### 3.1.1.7 Δεδομένα εξόδου

Παρατίθεται το αντίστοιχο διάγραμμα UML

Δεδομένα εξόδου τη συγκεκριμένης περίπτωσης χρήσης αποτελούν το πεδίο του σημείου στη βάση δεδομένων και όποιο μήνυμα προκύψει για την αποτυχία ή την επιτυχία της καταχώρησης του αιτήματος.

### 3.1.2 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 2: Κράτηση slot για φόρτιση

#### 3.1.2.1 Χρήστες (ρόλοι) που εμπλέκονται

Μόνο εγγεγραμμένοι χρήστες έχουν τη δυνατότητα να κάνουν κράτηση για φόρτιση.

#### 3.1.2.2 Προϋποθέσεις εκτέλεσης

Για να εκτελεστεί σωστά η περίπτωση χρήσης θα πρέπει να υπάρχουν διαθέσιμοι σταθμοί και ο χρήστης να ακολουθήσει την προβλεπόμενη ακολουθία ενεργειών.

#### 3.1.2.3 Περιβάλλον εκτέλεσης

Ο χρήστης μέσω της διαδικτυακής διεπαφής του χρήστη εκχωρεί την κράτηση slot σταθμού φόρτισης. Στη συνέχεια τα δεδομένα του reservation καταχωρούνται στο DBMS.

#### 3.1.2.4 Δεδομένα εισόδου

Ο χρήστης καλείται να επιλέξει από το χάρτη το σημείο φόρτισης, το EV από τα συνδεδεμένα στο λογαριασμό του που θέλει να φορτίσει καθώς και το slot που επιθυμεί.

#### 3.1.2.5 Παράμετροι

Απαιτείται από το χρήστη η είσοδος στην εφαρμογή με τον προσωπικό του λογαριασμό και η ύπαρξη τουλάχιστον ενός καταχωρημένου ηλεκτρικού οχήματος. Οι υπόλοιποι παράμετροι εμφανίζονται στο User Interface.

#### 3.1.2.6 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Η αλληλουχία ενεργειών περιγράφεται και οπτικά με τα διαγράμματαUML που παρατίθενται.

Βήμα 1: Πλοήγηση στο χάρτη της γραφικής διεπαφής αφού έχει προηγηθεί Log In ( βλ. Παράμετροι ).

Βήμα 2: Επιλογή σημείου φόρτισης ηλεκτρικών οχημάτων από τα διαθέσιμα.

Βήμα 3: Από τις διάφορες λεπτομέρειες σχετικά με το σημείο φόρτισης ratings/connectors etc … επιλογή “Charge my EV here”.

Βήμα 4: Επιλογή από τα διαθέσιμα ηλεκτρικά οχήματα ( Στην κοινή περίπτωση του ενός οχήματος μεταβαίνουμε στο επόμενο βήμα ).

Βήμα 5: Συγκρίνονται από το front-end της εφαρμογής οι connectors του επιλεγμένου σημείου με τους συμβατούς connectors του επιλεγμένου οχήματος. Εάν υπάρχει τουλάχιστον ένας συμβατός connector προχωράμε στο Βήμα 6 αλλιώς ενημερώνουμε με μήνυμα σφάλματος και οπισθοδρομούμε στο βήμα 3 ή 4 ανάλογα με το πλήθος των οχημάτων.

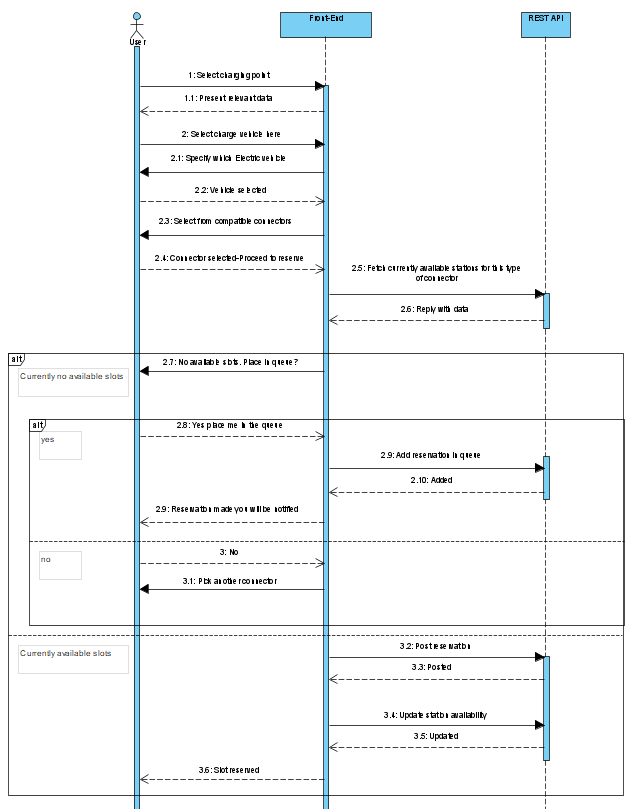
Βήμα 6: Επιλογή ενός από τους διαθέσιμους connectors και παράλληλη συνοπτική παρουσίαση λειτουργιών και τιμολόγησης.

Βήμα 7: Επιλογή Proceed to reserve slot για τον έλεγχο διαθέσιμων θέσεων στο σημείο φόρτισης. Εάν υπάρχουν διαθέσιμες θέσεις τότε η κράτηση γίνεται και ειδοποιείται ο χρήστης ότι η κράτηση του παραμένει έγκυρη για ένα ορισμένο χρονικό διάστημα. Εάν δεν υπάρχουν την παρούσα στιγμή διαθέσιμες θέσεις τότε προχωράμε στο βήμα 8.

Βήμα 8: Διερώτηση του χρήστη σχετικά με το εάν θέλει να τοποθετηθεί σε ουρά προτεραιότητας παρουσιάζοντας του παράλληλα τον εκτιμώμενο χρόνο αναμονής. Εάν ο χρήστης επιλέξει την περίπτωση της αναμονής τότε η θέση δεσμεύεται και του παρουσιάζεται μήνυμα ότι θα ειδοποιηθεί από την υπηρεσία σχετικά με τη διαθεσιμότητα της κράτησης του. Εάν αρνηθεί την περίπτωση της αναμονής επιστρέφει στο βήμα 6.

#### 

#### 3.1.2.7 Δεδομένα εξόδου

Παρατίθενται τα αντίστοιχα UML διαγράμματα.

Δεδομένο εξόδου τη συγκεκριμένης περίπτωσης χρήσης αποτελεί η οντότητα της κράτησης η οποία καταχωρείται στη βάση δεδομένων μέσω POST request απ το front end και η ανανέωση των διαθέσιμων θέσεων σε ένα σταθμό φόρτισης εφόσον η κράτηση έγινε χωρίς αναμονή στην ουρά με απλό PUT request. Στην περίπτωση της ουράς σε ένα πραγματικό σύστημα θα χρησιμοποιούσαμε ένα ξεχωριστό Application server για την παροχή αυτής της λειτουργίας όμως αυτό είναι εκτός πλαισίου της εργασίας.

### 3.1.3 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 3: Έκδοση αναλυτικού λογαριασμού

#### 3.1.3.1 Χρήστες (ρόλοι) που εμπλέκονται

Μόνο οι εγγεγραμμένοι χρήστες

#### 3.1.3.2 Προϋποθέσεις εκτέλεσης

Η περίπτωση χρήσης έχει μοναδική προυπόθεση να ακολουθηθούν σωστά τα βήματα από το UserInterface.

#### 3.1.3.3 Περιβάλλον εκτέλεσης

Ο ιδιοκτήτης κάνει αίτημα για έκδοση λογαριασμού στη διαδικτυακή διεπαφή του και στη συνέχεια το REST API ανακτά τα αντίστοιχα στοιχεία που βρίσκονται στη βάση δεδομένων και τα εμφανίζει στην οθόνη.

#### 3.1.3.4 Δεδομένα εισόδου

Ο ιδιοκτήτης καταχωρεί το αίτημά του, επιλέγει χρονική περίοδο, συγκεκριμένο όχημα και τύπο λογαριασμού που επιθυμεί να εκδοθεί (π.χ αναλυτικός λογαριασμός, διαγράμματα, στατιστικά εξόδων) επίσης ως δεδομένο εισόδου μπορεί να συμπεριληφθεί και το πεδίο του χρήστη στη βάση δεδομένων.

#### 3.1.3.5 Παράμετροι

Ο χρήστης θα πρέπει να είναι εγγεγραμμένος, να έχει κάνει login να έχει καταχωρημένο τουλάχιστον ένα όχημα στο λογαριασμό του και να υπάρχουν καταχωρημένα transactions.

#### 3.1.3.5 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).

Η αλληλουχία ενεργειών περιγράφεται και οπτικά με τα διαγράμματα UML που παρατίθενται.

Βήμα 1: Επιλογή του οχήματος από την λίστα οχημάτων του χρήστη και συγκεκριμένων φίλτρων που αφορούν το λογαριασμό. (Χρονική περίοδος, τρόπος πληρωμής, charging point κλπ)

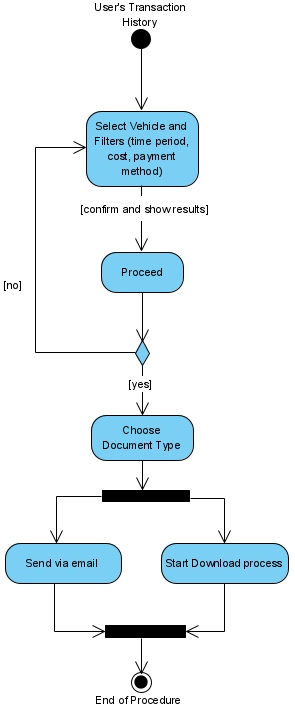
Βήμα 2: Αν ο χρήστης επιβεβαιωσει τις παραπάνω επιλογές αποστέλλονται τα δεδομένα στο REST API. Αλλιώς, επιστροφή στο βήμα 1.

Βήμα 3: Το REST API με βάση τις επιλογές του χρήστη και τον τύπο αρχείου που έχει επιλεγεί (pdf, excel) προσπελάζει τη βάση δεδομένων στο πεδίο του χρήστη και

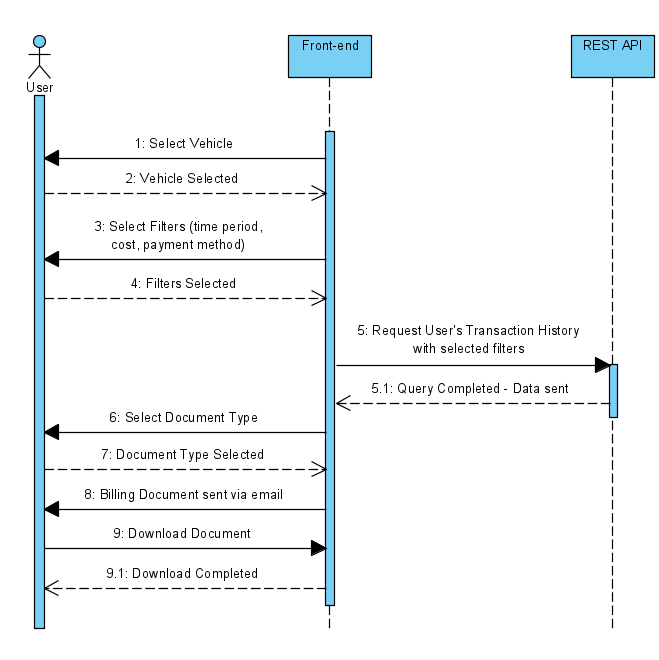
Α. Επιστρέφει τον αναλυτικό λογαριασμό προς εκτύπωση

Β. Αποστέλλει το ίδιο αρχείο στο email του χρήστη

Στον περιοδικό λογαριασμό θα περιέχονται αναλυτικά στοιχεία των συναλλαγών του χρήστη όπως οι φόροι που έχουν παρακρατηθει, αναλυτικά δεδομένα για κάθε φόρτιση καθώς και τα προσωπικά στοιχεία του χρήστη και του οχήματος που αφορά ο λογαριασμός.



#### 3.1.1.7 Δεδομένα εξόδου

Παρατίθεται το αντίστοιχο διάγραμμα UML

Δεδομένα εξόδου τη συγκεκριμένης περίπτωσης χρήσης αποτελούν η σελίδα του λογαριασμού (κείμενο, γράφημα, report) στην οθόνη του χρήστη και το αντίστοιχο έγγραφο που αποστέλλεται με email.

### 3.1.4 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 2: Φόρτιση οχήματος

#### 3.1.4.1 Χρήστες (ρόλοι) που εμπλέκονται

Μόνο εγγεγραμμένοι χρήστες έχουν τη δυνατότητα να εκτελέσουν μία διαδικασία φόρτισης

#### 3.1.4.2 Προϋποθέσεις εκτέλεσης

Για να εκτελεστεί σωστά η περίπτωση χρήσης θα πρέπει ο χρήστης να ακολουθήσει την προβλεπόμενη ακολουθία ενεργειών και να υπάρξει επιβεβαιωμένη συναλλαγή για την πληρωμή

#### 3.1.4.3 Περιβάλλον εκτέλεσης

Ο χρήστης μέσω της διαδικτυακής διεπαφής του χρήστη εκχωρεί την κράτηση slot σημείου φόρτισης. Στη συνέχεια τα δεδομένα του reservation καταχωρούνται στο DBMS

#### 3.1.4.4 Δεδομένα εισόδου

Ο χρήστης καλείται να επιλέξει από το χάρτη το σημείο φόρτισης, το EV από τα συνδεδεμένα στο λογαριασμό του που θέλει να φορτίσει καθώς και το slot που επιθυμεί

#### 3.1.4.5 Παράμετροι

Ο χρήστης θα πρέπει να είναι εγγεγραμμένος, να έχει κάνει login να έχει καταχωρημένο τουλάχιστον ένα όχημα στο λογαριασμό του και να είναι συνδεδεμένο το όχημα σε ένα charging station.Επίσης για να ολοκληρωθεί η φόρτιση θα πρέπει να γίνει επιτυχώς η πληρωμή από το χρήστη ή ο χρήστης να έχει στο λογαριασμένου κάποιο προπληρωμένο πακέτο φορτίσεων.

#### 3.1.4.6 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Για τη συγκεκριμένη περίπτωση χρήσης έχουμε θεωρήσει πως σημείο «εκκίνησης» είναι η κατάσταση στην οποία το EV είναι plugged in σε κάποιον σταθμό φόρτισης

Βήμα 1: Επιλογή προγράμματος από τη λίστα με τα διαθέσιμα προγράμματα

Βήμα 2: Ο χρήστης με το button ”Start Charging” αποστέλλει το αίτημα για εκκίνηση της φόρτισης και το EV «κλειδώνεται» στο σσταθμό μέχρι την περάτωση της πληρωμής

Βήμα 3:

Α 1. Το σύστημα αναγνωρίζει τα στοιχεία του οχήματος και τα αποστέλλει στο REST API.

2. Το REST API αποθηκεύει στη βάση δεδομένων τα στοιχεία του οχήματος στο πεδίο του σταθμού φόρτισης για το ιστορικό φορτίσεων

Β.(Ταυτόχρονα) Τυπώνεται στην οθόνη μήνυμα με τον εκτιμώμενο χρόνο φόρτισης που απομένει με βάση το επιλεγμένο πρόγραμμα

Γ.(Ταυτόχρονα) Τυπώνεται στην οθόνη μήνυμα με το εκτιμώμενο συνολικό κόστος της φόρτισης και με το έως τώρα κόστος. Τα στοιχεία εμφανίζονται κατευθείαν με βάση τα δεδομένα του σταθμού χωρίς την παρεμβολή του REST API.

Δ. 1.(Ταυτόχρονα) Τα δεδομένα της φόρτισης (ημερομηνία, ώρα, επιλεγμένο πρόγραμμα) αποστέλλονται στο REST API.

2. To REST API αποθηκεύει τα παραπάνω δεδομένα τόσο στο πεδίο για τις φορτίσεις του χρήστη όσο και στο πεδίο του ιστορικού φορτίσεων του σταθμού

Βήμα 4:

Βήμα 5: Εμφάνιση συνολικού κόστους

Βήμα 6: Επιλογή τρόπου πληρωμής μεταξύ μετρητών και κάρτας (φυσικά αυτό το βήμα είναι απαραίτητο ώστε να ολοκληρωθεί η φόρτιση και να «ξεκλειδωθεί» το όχημα).]

Βήμα 7: Επιλογή έκδοσης τιμολογίου ή απόδειξης

Βήμα 8: Το REST API αποστέλλει τα δεδομένα της χρέωσης μέσω email στο χρήστη

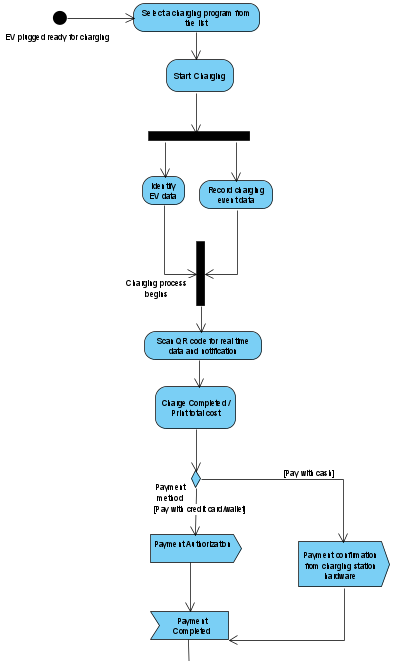
Βήμα 9:

Βήμα 10: Επιλογή χρήστη για βαθμολογία του σταθμού ή υπενθύμιση αργότερα

Βήμα 11: Αν ο χρήστης επέλεξε τη βαθμολόγηση, αποστολή δεδομένων βαθμολογίας (αιρθμητική αξιολόγηση στα 5 ή/και κείμενο κριτικής) στο REST API.

Βήμα 12: Το REST API αποθηκεύει τα δεδομένα της αξιολόγησης στη βάση δεδομένων στο πεδίο του σταθμού στις κριτικές

Βήμα 13: Εκτύπωση μηνύματος επιτυχούς φόρτισης και επιστροφή στην αρχική σελίδα



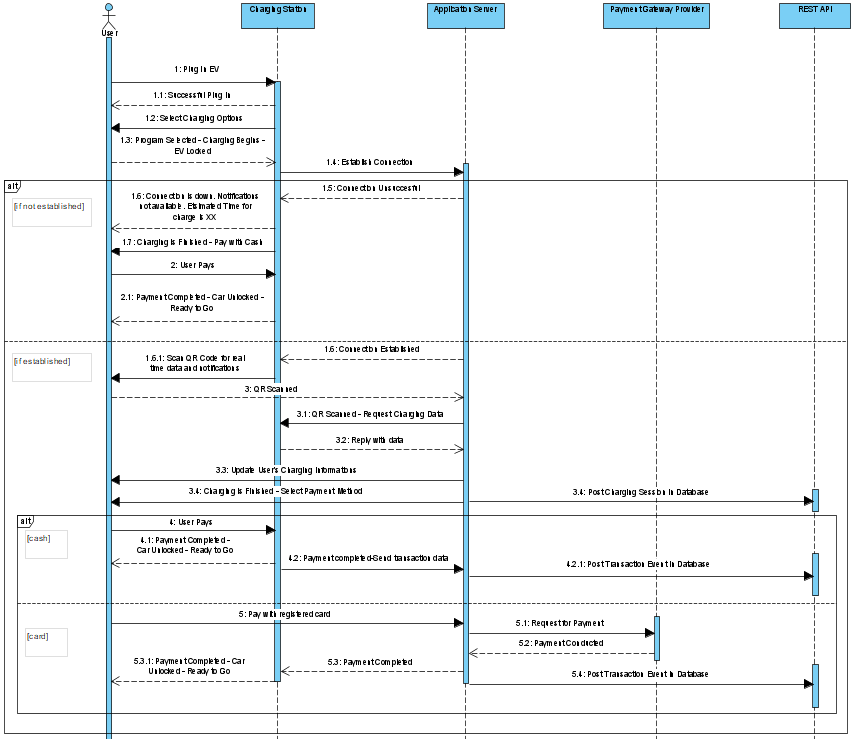
Συνεχίζει απο κάτω

#### 

#### 3.1.4.7 Δεδομένα εξόδου

Δεδομένα εξόδου της συγκεκριμένης περίπτωσης χρήσης αποτελούν τα διάφορα μηνύματα προς το χρήστη, τα δεδομένα που είναι διαθέσιμα κατά τη διάρκεια της φόρτισης (πχ εκτιμώμενο κόστος, επίπεδο φόρτισης, εκτιμώμενο κόστος). Επίσης, η οντότητα της φόρτισης με τα δεδομένα της (πχ ημερομηνία, ώρα, χρονική διάρκεια, πληροφορίες για το EV) καταχωρείται στη βάση δεδομένων μέσω POST request.

Παρατίθενται τα αντίστοιχα UML διαγράμματα.



## 3.2 Απαιτήσεις επιδόσεων

## 3.3 Απαιτήσεις οργάνωσης δεδομένων

### 3.3.1 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

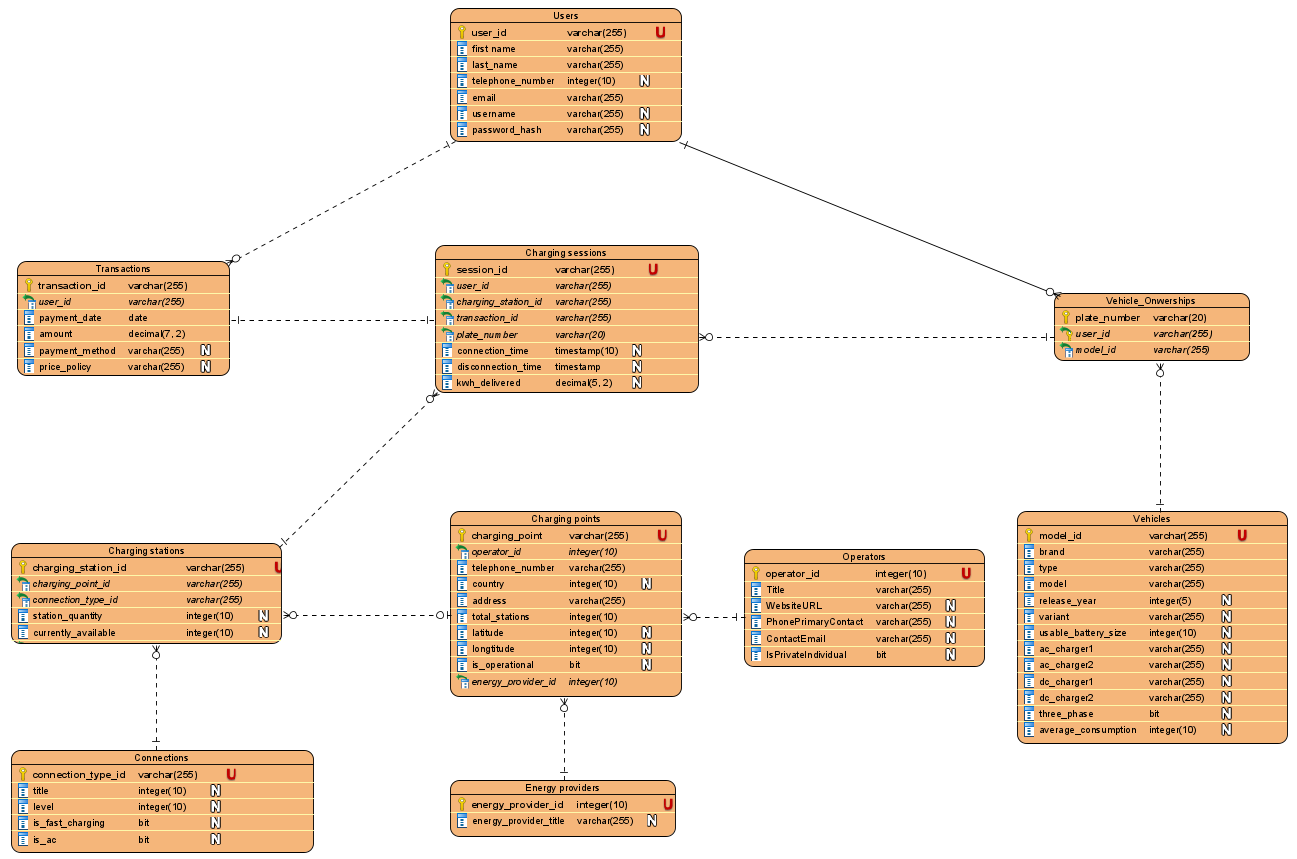
Απαιτήσεις πρόσβασης και περιορισμοί.

Κανένας χρήστης εγγεγραμμένος ή μη δεν έχει προσβαση στα δεδομένα της βάσης δεδομένων παρά μόνο αν αυτά αφορούν το λογαριασμό του. Βέβαια μπορούν να κάνουν αιτήματα προς τους admins (όπως και οι ιδιοκτήτες charging points πχ) ώστε να καταχωρούνται επιπλέον δεδομένα στη βάση.

Οι admins μόνο έχουν πρόσβαση στα δεδομένα και μπορούν να μεταβάλλουν το περιεχόμενό τους στη βάση δεδομένων. Φυσικά και εκείνοι περιορίζονται από κρυπτογράφηση στα προσωπικά δεδομένα τόσο των χρηστών όσο και των ιδιοκτητών σταθμών ή λοιπών φορέων και εταιρειών.

## 3.4 Περιορισμοί σχεδίασης

## ER Diagram



## 3.5 Λοιπές απαιτήσεις

### 3.5.1 Απαιτήσεις διαθεσιμότητας λογισμικού

Τεκμηρίωση απαιτήσεων διαθεσιμότητας

Η εφαρμογή μας αφορά λειτουργίες που είναι επιτακτικό να είναι διαθέσιμες κάθε στιγμή. Είναι εκ των ων ουκ ανευ απαίτηση του συστήματος να είναιδιαθέσιμο κάθε στιγμή τόσο στους χρήστες (κυρίως) όσο και στους administrators προς ενημέρωση και άντληση πληροφοριών

### 3.5.2 Απαιτήσεις ασφάλειας

Τεκμηρίωση απαιτήσεων ασφαλείας

Φυσικά θα πρέπει οι κωδικοί των χρηστών να αποθηκεύονται στη βάση δεδομένων αφού κρυπτογραφηθούν. Οι απαιτήσεις ασφάλειας είναι πολύ υψηλές αφού το σύστημα διαχειρίζεται σοβαρές και πολλές πληροφορίες σημαντικών και μεγάλων φορέων, εταιρειών και μετόχων.

### 3.5.3 Απαιτήσεις συντήρησης

Τεκμηρίωση απαιτήσεων συντήρησης

Απαιτούνται τακτικά backups στο database ώστε να διασφαλίζεται η ύπαρξη των δεδομένων ακόμα και σε ακραίες περιπτώσεις κατάρρευσης του συστήματος.

Ο web server μας είναι πιθανόν να χρειαστεί restart μετά από κάποια updates που ίσως υλοποιηθούν.