Έγγραφο απαιτήσεων λογισμικού (SRS)

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

ΣΗΜΕΙΩΣΗ: Το στυλ κειμένου (style) "Description" χρησιμοποιείται για να σας δώσει σύντομες οδηγίες για κάθε ενότητα. Το κείμενο που θα συμπληρώσετε πρέπει να γραφεί με το στυλ "Normal"

[ΤΙΤΛΟΣ ΕΦΑΡΜΟΓΗΣ ΛΟΓΙΣΜΙΚΟΥ]

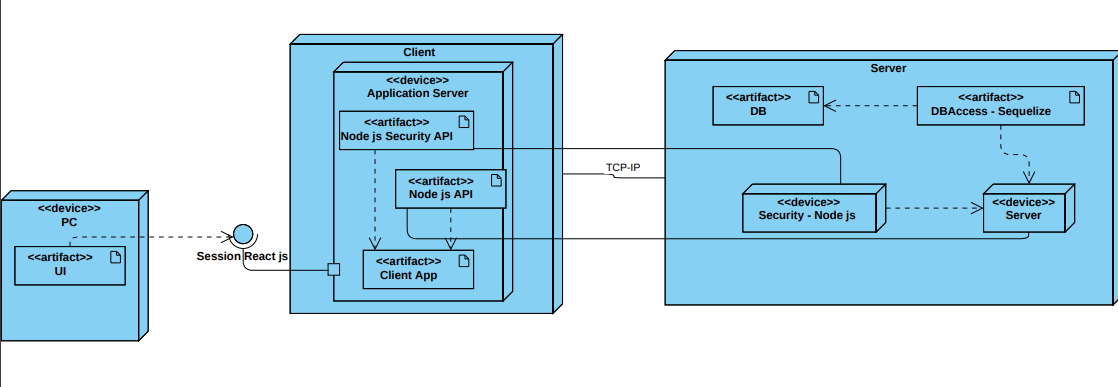
# Εισαγωγή

## 1.1 Εισαγωγή: σκοπός του λογισμικού

Το λογισμικό μας έχει σκοπό να δίνει όσες περισσότερες δυνατότητες γίνεται στον χρήστη για να μπορεί να φορτίζει το ηλεκτρικό του αμάξι με αρκετή άνεση, χωρίς δυσκολίες και χάσιμο χρόνου, παρέχοντας ταυτόχρονα στατιστικά και πληροφορίες για τη φόρτιση του αμαξιού του. Ταυτόχρονα, το λογισμικό συλλέγει στοιχεία για το κάθε αμάξι, σεβόμενο τα προσωπικά δεδομένα κάθε χρήστη, με σκοπό να παρέχει στους παροχείς σταθμών φόρτισης αυτά τα στατιστικά, τα οποία θα μπορούν να χρησιμοποιούν για να βελτιώσουν τις υπηρεσίες τους και το επιχειρησιακές τους στρατηγικές.

## 1.2 Διεπαφές (interfaces)

### 1.2.1 Διεπαφές με εξωτερικά συστήματα



### 1.2.2 Διεπαφές με το χρήστη

Χρήστης: Πρόκειται για τον χρήστη του front-end υποσυστήματος ή/και του RESTful API

τόσο για την πλοήγηση, αναζήτηση και ανάκτηση των δεδομένων και στατιστικών του οχήματος/οχημάτων του. Επίσης, είναι απαραίτητη η εγγραφή τους στο σύστημα για οποιαδήποτε χρήση του, όπως το να μπορεί ο χρήστης να βλέπει τα διαθέσιμα πακέτα προσφορών φόρτισης καθώς και το ιστορικό του.

Πάροχος σταθμού φόρτισης. Πρόκειται για τον χρήστη του front-end υποσυστήματος ή/και του RESTful API τόσο για την πλοήγηση, αναζήτηση και ανάκτηση των δεδομένων. Είναι υποχρεωτικό να είναι εγγεγραμμένος στο σύστημα, καθώς έχει ειδικές άδειες για να μπορεί να βλέπει τα στατιστικά των φορτίσεων και των οχημάτων γενικά.

Διαχειριστής: Πρόκειται για τον χρήστη του back-end υποσυστήματος που είναι υπεύθυνος για τη διαχείριση των λογαριασμών χρηστών της πρώτης κατηγορίας και της δεύτερης κατηγορίας μέσω του CLI και του API. Έχει τη δυνατότητα ανάθεσης ρόλων σε χρήστες, δημιουργίας και διαγραφής των λογαριασμών τους, προσθήκη νέων δεδομένων στην βάση δεδομένων και τροποποίησης των καταχωρήσεων που βρίσκονται στο σύστημα όταν αυτό κρίνεται απαραίτητο. Επίσης, έχει όλα τα δικαιώματα εγγεγραμμένου χρήστη και παρόχου σταθμού φόρτισης.

# 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

 Λεπτομερής προδιαγραφή των λειτουργιών του λογισμικού σε επίπεδο περιπτώσεων χρήσης. Ο αριθμός των περιπτώσεων χρήσης ανάλογα με τον αριθμό των μελών της ομάδας σύμφωνα με την εκφώνηση.

### 3.1.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 1: Ο χρήστης εισέρχεται στο σύστημα για να δει τους διαθέσιμους σταθμούς στο σύστημα.

#### 3.1.1.1 Χρήστες (ρόλοι) που εμπλέκονται

Σε αυτή την περίπτωση χρήσης, εμπλέκονται ο χρήστης. Συγκεκριμένα, για τον χρήστη έχει γίνει επιβεβαίωση ταυτότητας, δηλαδή έχει συνδεθεί στο σύστημα.

AVAILABE

#### 3.1.1.2 Προϋποθέσεις εκτέλεσης

Μοναδική προϋπόθεση είναι για τον χρήστη να συνδεθεί με τα μοναδικά του credentials στο σύστημα ώστε να έχει πρόσβαση στις πληροφορίες.

#### 3.1.1.3 Περιβάλλον εκτέλεσης

Η συγκεκριμένη περίπτωση χρήσης εκτελείται μέσω της διαδικτυακής διεπαφής χρήστη. Επίσης, υπάρχει η βάση δεδομένων από την οποία λαμβάνονται τα δεδομένα και το RESTful API που μεσολαβεί μεταξύ βάσης και χρήστη τόσο για την ταυτοποίηση όσο και για την παροχή των δεδομένων.

#### 3.1.1.4 Δεδομένα εισόδου

Κατά τη διαδικασία εγγραφής στο σύστημα θα πρέπει με ασφάλεια να αποθηκεύονται τα προσωπικά στοιχεία του χρήστη (username, password, email). To e-mail πρέπει να είναι μοναδικό για κάθε χρήστη στο σύστημα.

Επίσης, όταν κάνει log in, τα δεδομένα εισόδου του(username, password) θα μεταβιβάζονται στο σύστημα. Το συγκεκριμένο ζεύγος username, password θα πρέπει να είναι ορθά, έτσι ώστε να τα δεχτεί το σύστημα.

Ως δεδομένα εξόδου, ο χρήστης θα έχει τη δυνατότητα να δει το όνομα, την διεύθυνση και το τηλέφωνο το διαθέσιμων σταθμών και εφόσον επιλέξει κάποιον από αυτούς να δει και τους διαθέσιμους φορτιστές.

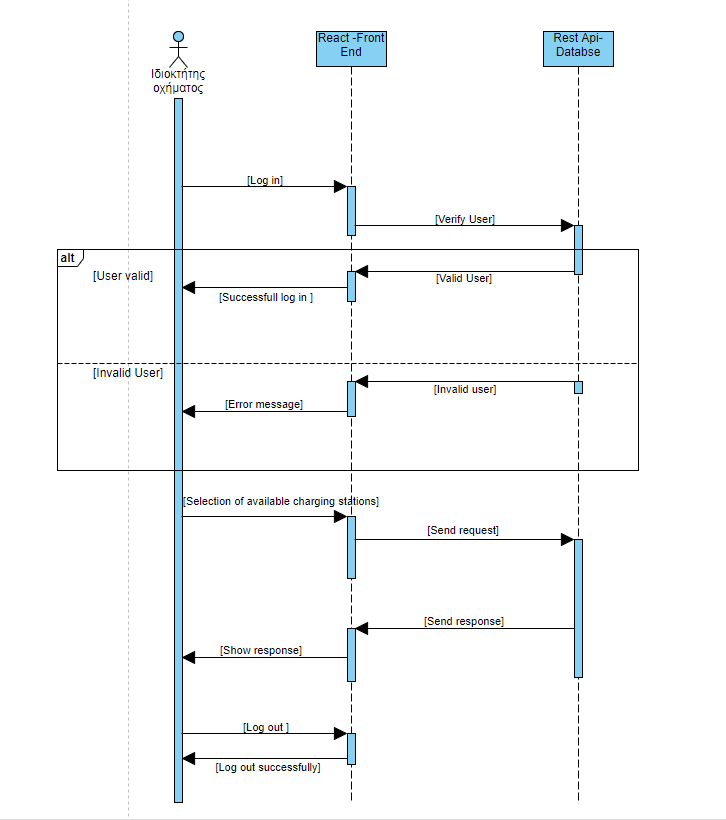
#### 3.1.1.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

1. Ο χρήστης εισάγει τα στοιχεία του (username, password) για να συνδεθεί στο σύστημα, άμα είναι ήδη εγγεγραμμένος χρήστης. Σε περίπτωση που δεν είναι εγγεγραμμένος πρέπει να επικοινωνήσει με τον διαχειριστή τους συστήματος ή έναν υπεύθυνο σταθμού φόρτισης.

2. Αν είναι ήδη εγγεγραμμένος και τα στοιχεία αυτά υπάρχουν ήδη στη βάση, τότε συνδέεται επιτυχώς. Αλλιώς, του ζητείται να ξανά πληκτρολογήσει τα στοιχεία του.

3. Από την αρχική οθόνη της εφαρμογής ο χρήστης επιλέγει να δει τις πληροφορίες των τους διαθέσιμων σταθμών φόρτισης.

4. Ο χρήστης αποσυνδέεται.



#### 3.1.1.7 Δεδομένα εξόδου

Δεδομένα εξόδου κατά την επιτυχία της σύνδεσης, κατά την επιλογή προβολής των διαθέσιμων σταθμών αλλά και κατά την αναφορά κάποιας κακής χρήσης.

#### 3.1.1.8 Παρατηρήσεις

Ο,τι δεν εντάσσεται στα προηγούμενα, εφόσον υπάρχει

### 3.1.2 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 2: Συγκέντρωση στατιστικών από τις φορτίσεις των οχημάτων των χρηστών για χρήση από τον πάροχο σταθμού φόρτισης

#### 3.1.2.1 Χρήστες (ρόλοι) που εμπλέκονται

Στη συγκεκριμένη περίπτωση χρήσης, συμμετέχει ένας διαχειριστής του σταθμού, ο οποίος είναι υπεύθυνος για την εγκυρότητα της συλλογής των δεδομένων για τους χρήστες που χρησιμοποιούν τον συγκεκριμένο σταθμό φόρτισης και αφότου βεβαιωθεί ότι έχουν συλλεχτεί με ασφάλεια, χωρίς να υπάρχει κάποια παραβίαση προσωπικών δεδομένων και χωρίς να έχει παρουσιαστεί κάποιο λάθος τα στέλνει μπορεί να έχει πρόσβαση σε αυτά μέσω της ιστοσελίδας της εφαρμογής και να τα στείλει στην κατάλληλη ομάδα ανάλυσης δεδομένων. (Συγκεκριμένα, στην πραγματικότητα μεγάλων σταθμών φόρτισης αυτά στέλνονται στο data intelligence/analytics department, στο οποίο κάποιος υπάλληλος/ομάδα υπαλλήλων θα τα αναλύει, με σκοπό να χρησιμοποιηθούν για να βελτιώσουν τις υπηρεσίες τους και το επιχειρησιακές τους στρατηγικές και να κερδίσουν κάποιο edge σε σχέση με τον ανταγωνισμό).

#### 3.1.2.2 Προϋποθέσεις εκτέλεσης

Όπως είναι λογικό, οφείλεται φυσικά να γίνεται καταγραφή δεδομένων σε μία βάση, έτσι ώστε να είναι δυνατή η συγκεκριμένη περίπτωση χρήσης. Επίσης, επιβάλλεται να μην υπάρχει παραβίαση προσωπικών δεδομένων και να είναι δυνατή η αποστολή των στατιστικών σε κάθε πάροχο σταθμού φόρτισης.

#### 3.1.2.3 Περιβάλλον εκτέλεσης

Η συγκεκριμένη περίπτωση χρήσης εκτελείται μέσω της διαδικτυακής διεπαφής χρήστη(frontend). Επίσης, υπάρχει η βάση δεδομένων στην οποία περνούν τα δεδομένα για κάθε φόρτιση που λαμβάνει χώρα και μέσω της οποίας γίνεται η ταυτοποίηση και ένα RESTful API που συνδέει την βάση με το frontend παρέχοντας ταυτόχρονα ασφάλεια.

#### 3.1.2.4 Δεδομένα εισόδου

Κάθε φορά που κάποιος χρήστης θα χρησιμοποιεί τον σταθμό, θα συλλέγονται στατιστικά για τη φόρτιση που πραγματοποιήθηκε στη βάση δεδομένων. Ύστερα, θα ομαδοποιούνται έτσι ώστε να μπορούν να εκτιμηθούν και προτού σταλθούν στην αντίστοιχη ομάδα ανάλυσης (δεδομένα εξόδου), θα ελέγχονται για την εγκυρότητά τους από τον διαχειριστή του σταθμού.

#### 3.1.2.5 Παράμετροι

Βασική παράμετρος είναι το ζεύγος username-password του διαχειριστή σταθμού να είναι έγκυρα για να μπορεί να εισέλθει στην πλατφόρμα. Επίσης, οφείλεται να διαβεβαιωθεί ότι δεν υπάρχει παραβίαση προσωπικών δεδομένων των χρηστών.

#### 3.1.2.6 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

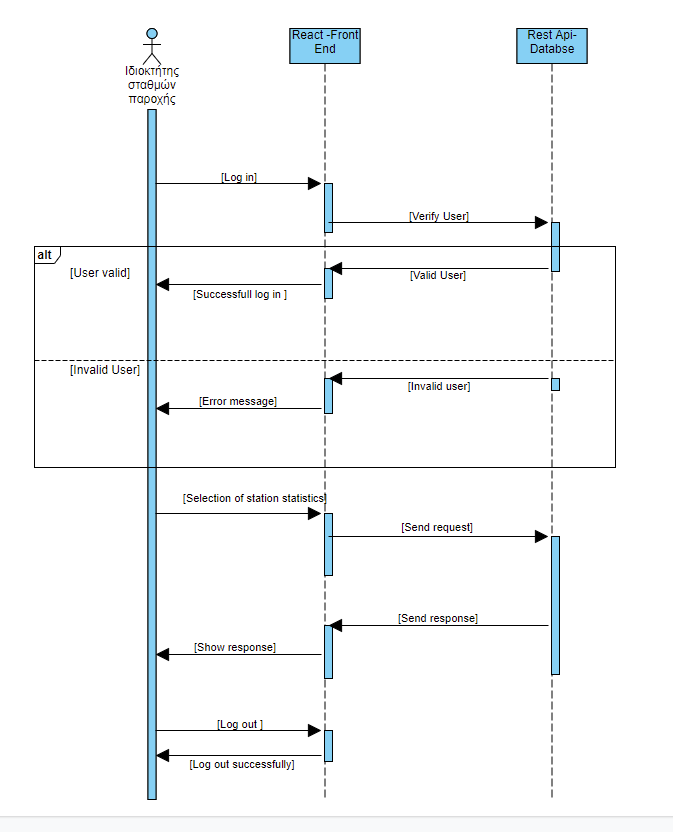
1. Ο διαχειριστής σταθμού εισάγει τα στοιχεία του (username, password) για να συνδεθεί στο σύστημα, άμα είναι ήδη εγγεγραμμένος. Σε περίπτωση που δεν είναι, πρέπει να έρθει σε επικοινωνία με τον administrator, ώστε να του δημιουργήσει λογαριασμό με τα κατάλληλα privileges.

2. Αν είναι ήδη εγγεγραμμένος και τα στοιχεία αυτά υπάρχουν ήδη στη βάση, τότε συνδέεται επιτυχώς. Αλλιώς, του ζητείται να ξανά πληκτρολογήσει τα στοιχεία του.

3. Ο διαχειριστής σταθμού διαλέγει σε ποια στατιστικά χρειάζεται να έχει πρόσβαση.

4. Το σύστημα του επιστρέφει τα συγκεκριμένα στατιστικά.

5. Ο χρήστης αποσυνδέεται.



#### 3.1.2.7 Δεδομένα εξόδου

Δεδομένα εξόδου είναι τα στατιστικά τα οποία παρέχονται στον διαχειριστής σταθμού, αλλά και οποιαδήποτε αναφορά κάποιας κακής χρήσης μπορεί να προκύψει.

#### 3.1.2.8 Παρατηρήσεις

Ο,τι δεν εντάσσεται στα προηγούμενα, εφόσον υπάρχει

. . .

(η ενότητα 3.1.Χ.1 - 3.1.Χ.8 επαναλαμβάνεται για όλες τις περιπτώσεις χρήσης που συμπεριλαμβάνονται στο έγγραφο, όπως απαιτούνται από τις ομάδες ανάλογα με τον αριθμό των ατόμων)

## 3.2 Απαιτήσεις επιδόσεων

Πρέπει το λογισμικό μας να είναι συνεχώς διαθέσιμο σε αυτούς που επιθυμούν να το χρησιμοποιήσουν. Επομένως, πρέπει να μπορεί να εξυπηρετήσει το αναμενόμενο πλήθος χρηστών, ειδικά σε ώρες αιχμής, τόσο των online χρηστών, οι οποίοι εξ αποστάσεως θα θέλουν να ελέγξουν τα στατιστικά των φορτίσεων που έχουν κάνει στο παρελθόν, τόσο και των διαχειριστών σταθμών και το διαχειριστή του συστήματος.

## 3.3 Απαιτήσεις οργάνωσης δεδομένων

### 3.3.1 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

Κάθε χρήστης θα έχει πρόσβαση σε διαφορετικά δεδομένα. Ο συνδεδεμένος χρήστης (logged-in user) θα έχει έξτρα πρόσβαση σε στατιστικά προηγούμενών του φορτίσεων, πληροφορίες για τους διαθέσιμους σταθμούς τις πλατφόρμας και τους φορτιστές κάθε μίας και ακόμη πρόσβαση σε διαθέσιμα πακέτα και προσφορές.

Ο διαχειριστής των σταθμών θα έχει πρόσβαση στα στατιστικά της βάσης δεδομένων, χωρίς να έχει τη δυνατότητα να τροποποιήσει κάποιο από αυτά ή να δει τα προσωπικά δεδομένων των απλών χρηστών.

Ο διαχειριστής (administrator) θα έχει την μεγαλύτερη πρόσβαση από όλους τους χρήστες, καθώς θα μπορεί να δει και να επεξεργαστεί όλα τα δεδομένα των χρηστών, των σταθμών και όλης της βάσης δεδομένων όπως να ελέγξει τα διάφορα logs του λογισμικού καθώς και να κάνει αλλαγές στην λειτουργία και την δομή του συστήματος.

## 3.4 Περιορισμοί σχεδίασης

## Στη συνέχεια αναφέρονται τα **τεχνολογικά πρότυπα** που χρησιμοποιήθηκαν κατά την ανάπτυξη της εφαρμογής:

Back end

* Node.js v14.15.2
* Express Framework
* Mysql Database
* Sequalize

Front-end

* React 17.0.1

CLI

* Node.js v14.15.2
* Commander
* Inquirer

Testing

* Mocha
* Chai

Τα **πρότυπα συγγραφής κώδικα** που χρησιμοποιήθηκαν από όλη την ομάδα κατά την ανάπτυξη της εφαρμογής είναι:

* Back end: Node.js REST API coding standard  [JSLint](http://www.javascriptlint.com/)  και ακολουθήθηκε Model , Controller , Routing model και έπειτα τις διευκρινίσεις τις εκφώνησης.
* Frontend: React coding Standard Practices.
* CLI: Commander Standard βασισμένο στο github page και ακολουθώντας και τις οδηγίες τις εκφώνησης με σκοπό να το κάνουμε όσο πιο απλό και λειτουργικό γίνεται.

Τα **πρότυπα συνεισφοράς** με τα οποία επιτεύχθηκε ο συντονισμός και η συνεργατικότητα στα πλαίσια της ομάδας είναι:

* Development με branches στο Github
* Kanban Scheduling με project boards

## 3.5 Λοιπές απαιτήσεις

### 3.5.1 Απαιτήσεις διαθεσιμότητας λογισμικού

Η εφαρμογή δε θα είναι σε θέση να απαντήσει με δεδομένα σε περίπτωση μη διαθεσιμότητας του REST API.

### 3.5.2 Απαιτήσεις ασφάλειας

Η σύνδεση του χρήστη στη σελίδα της εφαρμογής θα πρέπει να είναι ασφαλής σύμφωνα με το ανάλογο πρωτόκολλο, ενώ και οι κωδικοί θα πρέπει να αποθηκεύονται σε ασφαλή βάση δεδομένων.

3.5.3 Απαιτήσεις συντήρησης

Η διαχείριση του συστήματος θα είναι υπεύθυνη για την ενημέρωση και συντήρησή του. Οι απαιτήσεις αυτές έγκειται κυρίως στα διαχειριστικά(διασφάλιση ορθής πληροφορίας και ενασχόληση με θέματα των clients), αλλά και σε θέματα ενημέρωσης του ίδιου του λογισμικού του συστήματος.