Έγγραφο απαιτήσεων λογισμικού (SRS)

# Εισαγωγή

## 1.1 Εισαγωγή: σκοπός του λογισμικού

Σκοπός του λογισμικού είναι η παροχή δεδομένων σχετικά με την πρόβλεψη ζήτησης και την κατανάλωση ηλεκτρικής ενέργειας, καθώς και με τον τρόπο και την ποσότητα παραγωγής. Συγκεκριμένα, τα δεδομένα διατίθενται ανά χώρα και αν συγκεκριμένες χρονικές στιγμές.

## 1.2 Διεπαφές (interfaces)

### 1.2.1 Διεπαφές με εξωτερικά συστήματα

Το λογισμικό θα επικοινωνεί με εξωτερικά συστήματα μέσω του API που διαθέτει. Συγκεκριμένα, εξωτερικά συστήματα θα μπορούν να επικοινωνούν με τον σερβερ της εφαρμογής χρησιμοποιώντας ένα url και δίνοντας δεδομένα με μέθοδο post ή get ανάλογα την ενέργεια.

### 1.2.2 Διεπαφές με το χρήστη

Οι χρήστες θα μπορούν να επικοινωνήσουν με την εφαρμογή μέσω ενός command line interface το οποίο συνδέεται μέσω του Αpi με τον server.

# Αναφορές - πηγές πληροφοριών

Όλες τα δεδομένα που χρησιμοποιούνται και παρέχονται από την εφαρμογή προέρχονται από την entso-e.

# Προδιαγραφές απαιτήσεων λογισμικού

## 3.1 Περιπτώσεις χρήσης

### 3.1.1 Αποστολή δεδομένων για το πραγματικό συνολικό φορτίο μέσω CLI

#### 3.1.1.1 Χρήστες (ρόλοι) που εμπλέκονται

Εταιρίες ή ιδιώτες που επιθυμούν να αποκτήσουν πρόσβαση στα στατιστικά αγοραπωλησιών ηλεκτρικής ενέργειας της entso-e.

#### 3.1.1.2 Προϋποθέσεις εκτέλεσης

Προκειμένου να εκτελεστεί η εν λόγω περίπτωση χρήσης πρέπει ο χρήστης πρώτα να έχει εγγραφεί στο σύστημα.

#### 3.1.1.3 Περιβάλλον εκτέλεσης

Command line interface το οποίο συνδέεται με το REST-API της εφαρμογής.

#### 3.1.1.4 Δεδομένα εισόδου

Τα δεδομένα εισόδου θα πρέπει να είναι το token που παράγεται κατά την σύνδεση του χρήστη στο σύστημα, το οποίο στέλνεται ως custom header, το όνομα της περιοχής, το resolution, δηλαδή τη συχνότητα με την οποία εμφανίζονται τα δεδομένα(π.χ. ανά 15,30,60 λεπτά) και έπειτα ανάλογα με το τι εύρους δεδομένα επιθυμεί πρέπει να δώσει την κατάλληλη ημερομηνία. Συγκεκριμένα, αν επιθυμεί δεδομένα για όλο το χρόνο πρέπει να δώσει μόνο το έτος, αν θέλει για τον μήνα πρέπει να δώσει και τον μήνα και αν θέλει μόνο για μία ημέρα πρέπει να δώσει και την ημέρα. Για την εγκυρότητα πρέπει να είναι έγκυρο το token, δηλαδή να αντιστοιχεί σε κάποιον διαπιστευμένο χρήστη που έχει κάνει login στην εφαρμογή. Επίσης, πρέπει ο χρήστης να έχει διαθέσιμες ερωτήσεις(quota) για να κάνει στο σύστημα. Τέλος, πρέπει τα δεδομένα που ζητάει να υπάρχουν και να μην αφήσει κενό κανένα πεδίο από τα παραπάνω.

Τέλος, δεν υπάρχουν καινούργια δεδομένα εξόδου. Η έξοδος είναι τα δεδομένα που υπάρχουν στην βάση στον πίνακα ActualTotalLoad είτε σε κάποιο άθροισμα είτε μεμονωμένα ανάλογα με την χρονική περίοδο που ζητάει ο χρήστης. Το μόνο που αλλάζει στην βάση είναι ο αριθμός ερωτήσεων του χρήστη, που μειώνεται κατά 1 μετά από κάθε κλήση στο σύστημα.

#### 3.1.1.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Αρχικά, ο χρήστης κάνει login και το σύστημα του επιστρέφει το token το οποίο είναι encoded string που περιέχει το apikey και το username του χρήστη. Έπειτα, το cli, έχοντας αποθηκεύσει το apikey δίνει τη δυνατότητα στον χρήστη να εκτελέσει μια από τις προκαθορισμένες κλήσεις στο σύστημα. Στην συνέχεια, ο χρήστης επιλέγει ενέργεια και εισάγει τα απαραίτητα δεδομένα. Το cli μετά αντιστοιχεί τα παραπάνω δεδομένα στο κατάλληλο url του API αφού πρώτα ελέγξει ότι κανένα πεδίο δεν είναι κενό. Στην περίπτωση που κάποιο πεδίο είναι κενό ζητά από τον χρήστη να το εισάγει προτού το στείλει στην εφαρμογή. Με αυτόν τον τρόπο δεν μειώνονται αχρείαστα ερωτήσεις του χρήστη. Αν όλα τα δεδομένα έχουν δοθεί τότε το cli επικοινωνεί με το API. Τότε, το API ελέγχει την εγκυρότητα του token και ύστερα τον αριθμό ερωτήσεων που μπορεί να κάνει ο χρήστης. Αν όλα είναι σωστά η εφαρμογή ελέγχει τα δεδομένα αλλιώς στέλνει κατάλληλο error code. Στην περίπτωση που κάποιο δεδομένο εισόδου είναι λάθος ή δεν έχει δοθεί επιστρέφει κατάλληλο error code. Όταν τελειώσουν οι έλεγχοι η εφαρμογή συντάσει το query που θα κάνει στην βάση. Αν η βάση επιστρέψει δεδομένα το API θα τα στείλει στο cli και αυτό θα τα επιστρέψει στον χρήστη. Αν δεν υπάρχουν θα στείλει κατάλληλο error code. Όπως και να έχει το app θα μειώσει κατά ένα τον αριθμό ερωτήσεων του χρήστη (quotas).

#### 3.1.1.7 Δεδομένα εξόδου

Ουσιαστικά δεδομένα εξόδου είναι τα δεδομένα που ζήτησε ο χρήστης από τη βάση δεδομένων του συστήματος και τα οποία του επιστρέφονται μέσω του cli, είτε τα error codes σε περίπτωση που δεν πληρούνται οι συνθήκες επιστροφής δεδομένων.

### 3.1.2 Ενημέρωση βάσης από διαχειριστή

#### 3.1.2.1 Χρήστες (ρόλοι) που εμπλέκονται

Η εν λόγω περίπτωση χρήσης αφορά στον διαχειριστή του συστήματος που μπορεί να είναι είτε κάποιος δημόσιος φορέας παροχής ηλεκτρικής ενέργειας(π.χ ΔΕΔΔΗΕ) είτε κάποια ιδιωτική εταιρία.

#### 3.1.2.2 Προϋποθέσεις εκτέλεσης

Ο χρήστης να διαθέτει λογαριασμό διαχειριστή και να έχει έγκυρα δεδομένα σε μορφή csv.

#### 3.1.2.3 Περιβάλλον εκτέλεσης

Command line interface το οποίο συνδέεται με το api της εφαρμογής.

#### 3.1.2.4 Δεδομένα εισόδου

Τα δεδομένα εισόδου είναι το αρχείο csv που περιέχει τα νέα δεδομένα, ο πίνακας στον οποίο θα αποθηκευτούν και το token παράγεται όταν συνδέεται ο admin. Το αρχείο csv πρέπει να είναι γραμμένα σε μορφή που να μπορεί να καταλάβει το api. Το token πρέπει να είναι έγκυρο.

Τα δεδομένα εξόδου είναι τα νέα δεδομένα που αποθηκεύονται στην βάση.

#### 3.1.2.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Αρχικά, ο διαχειριστής κάνει login και το σύστημα του επιστρέφει το token που παράγεται. Έπειτα, το cli, έχοντας αποθηκεύσει το apikey ρωτάει τον χρήστη τι ενέργεια θέλει να κάνει. Στην συνέχεια, ο χρήστης επιλέγει να κάνει update την βάση και εισάγει το αρχείο με τα δεδομένα. Το cli μετά ανοίγει το αρχείο και το συντάσει σαν κείμενο στο link που θα στείλει στο api. Στην περίπτωση που το αρχείο δεν ανοίγει ή υπάρχει κάποιο άλλο σφάλμα επιστρέφει αντίστοιχο μήνυμα στον χρήστη. Αν όλα τα δεδομένα έχουν δοθεί τότε το cli επικοινωνεί με το app. Τότε, το app ελέγχει την εγκυρότητα του token. Αν δεν είναι έγκυρο επιστρέφει error. Όταν τελειώσουν οι έλεγχοι η εφαρμογή συντάσει το query που θα κάνει στην βάση. Αν η βάση δεν επιστρέψει error πάει να πει ότι ανανεώθηκε επιτυχώς. Αλλιώς, υπάρχει σφάλμα.

#### 3.1.2.6 Δεδομένα εξόδου

Τα δεδομένα εξόδου μπορεί να είναι είτε κάποιο μήνυμα επιτυχούς καταχώρησης συνοδευόμενο από τον κώδικα 200 - ΟΚ, είτε κάποιο μήνυμα σφάλματος μαζί με τον αντίστοιχο κώδικα.

## 3.2 Απαιτήσεις επιδόσεων

Το λογισμικό θα πρέπει να είναι σε θέση να διεκπεραιώνει πολλά αιτήματα ταυτόχρονα διατηρώντας ταυτόχρονα την βάση consistent χωρίς πολλά stalls. Συγκεκριμένα, θέλουμε η εφαρμογή να έχει υψηλό δείκτη αιτημάτων ανά μονάδα χρόνου και να μπορεί να ανταπεξέλθει όταν χρειάζεται να γίνει εισαγωγή στοιχείων στην βάση.

## 3.3 Απαιτήσεις οργάνωσης δεδομένων

### 3.3.1 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

Θέλουμε να υπάρχει δυνατότητα πρόσβασης στα δεδομένα απ’ τους διαπιστευμένους

χρήστες και δυνατότητα τροποποίησης απ’ τους διαχειριστές του συστήματος.

## 3.4 Περιορισμοί σχεδίασης

Για την ανάπτυξη του λογισμικού χρησιμοποιήθηκε το flask framework. Στην συγκεκριμένη εφαρμογή δεν υπήρξαν περιορισμοί.

## 3.5 Λοιπές απαιτήσεις

### 3.5.1 Απαιτήσεις διαθεσιμότητας λογισμικού

Το λογισμικό θα πρέπει να είναι διαθέσιμο σε όλους όσους έχουν εγγραφή και έχουν πρόσβαση στο internet καθώς και στο cli που επικοινωνεί με την εφαρμογή.

### 3.5.2 Απαιτήσεις ασφάλειας

Το σύστημα θα πρέπει να προστατεύει τους λογαριασμούς των χρηστών και τα προσωπικά τους στοιχεία. Επιπλέον, θα πρέπει να εμποδίζει την πρόσβαση με δικαιώματα διαχειριστή σε οποιονδήποτε δεν είναι εγκεκριμένο πάροχος ηλεκτρικής ενέργειας ή κάποιος ειδικός του τομέα. Αυτό συμβαίνει για να εξασφαλιστεί η ορθότητα των δεδομένων της βάσης.

### 3.5.3 Απαιτήσεις συντήρησης

Το λογισμικό θα συντηρείται ώστε να προσαρμόζεται σε τυχόν αλλαγές στα πρότυπα της αγοράς. Επίσης, θα απαιτείται συντήρηση στην βάση δεδομένων με διαγραφή χρηστών που δεν χρησιμοποιούν την εφαρμογή αλλά και την αλλαγή δικαιωμάτων όποτε χρειάζεται.