

Έγγραφο απαιτήσεων λογισμικού (SRS)

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

[Διαλειτουργικότητα στα δρόδια αυτοκινητοδρόμων]

1. Εισαγωγή

1.1 Εισαγωγή: σκοπός του λογισμικού

Ανάπτυξη ενός λογισμικού που στοχεύει στην διαχείριση της διαλειτουργικότητας των σταθμών των διόδων όσον αφορά τους πολλαπλούς λειτουργούς των αυτοκινητόδρομων. Συγκεκριμένα, δίνεται η δυνατότητα σε όλους τους χρήστες οι οποίοι είναι κάτοχοι ενός πάσου - tag να περνάνε από δρόμους οποιουδήποτε λειτουργού, ασχέτως ποιος λειτουργός είναι ο πάροχος του εκάστοτε tag, χωρίς να δημιουργούνται προβλήματα μεταξύ των λειτουργών. Ειδικότερα, με κάθε διέλευση ενός χρήστη-κάτοχου tag γίνεται μία τοπική αποθήκευση του χρηματικού ποσού που δόθηκε, του λειτουργού – παρόχου του tag και του λειτουργού – παρόχου του σταθμού από τον οποίο έγινε η διέλευση. Μετά από ένα χρονικό διάστημα, γίνεται υπολογισμός των οφειλών μεταξύ όλων των λειτουργών και οι αντίστοιχες πληρωμές πραγματοποιούνται.

Από την πλευρά της υλοποίησης, μια βάση δεδομένων κρατά την συσχέτιση μεταξύ κάθε id του οχήματος και του id κάθε πομποδέκτη, καθώς επίσης και όλα τα χρήσιμα δεδομένα για την λειτουργία του λογισμικού. Στη βάση δεδομένων καταγράφονται επίσης και τα δεδομένα κάθε τοπικής διέλευσης, ενώ αποθηκεύονται και οι συνολικές οφειλές μεταξύ λειτουργών και όσα στοιχεία των tag μας ενδιαφέρουν. Τα παραπάνω, αποτελούν το back-end του λογισμικού. Τέλος, οι δυνατότητες που παρέχονται σε κάθε χρήστη είναι προσβάσιμες μέσω web app το οποίο επικοινωνεί με το back-end. Μέσω του CLI επικοινωνούμε, με τη χρήση του terminal, με το API και τη βάση.

1.2 Διεπαφές (interfaces)

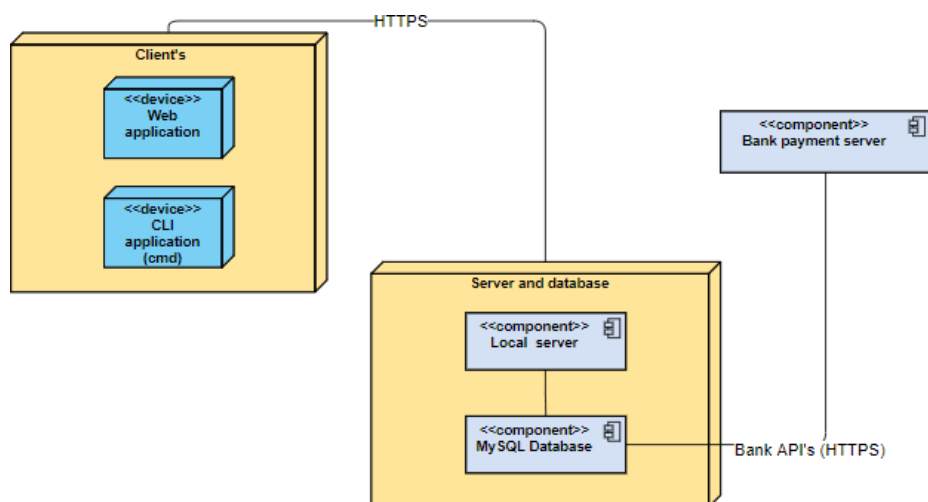
1.2.1 Διεπαφές με εξωτερικά συστήματα

Το σύστημα θα χρησιμοποιεί API's των παρακάτω εξωτερικών συστημάτων:

- 7 συστήματα λειτουργών - operators των αυτοκινητόδρομων
- Hardware της πύλης
- Τραπεζικό σύστημα
- Υπουργείο συγκοινωνιών

Εσωτερικά, έχουμε τις παρακάτω διεπαφές:

- Apache http Web Server (επικοινωνία web app με front-end)
- REST API's (επικοινωνία front-end με back-end)
- MySQL queries (επικοινωνία back-end με database)



1.2.2 Διεπαφές με το χρήστη

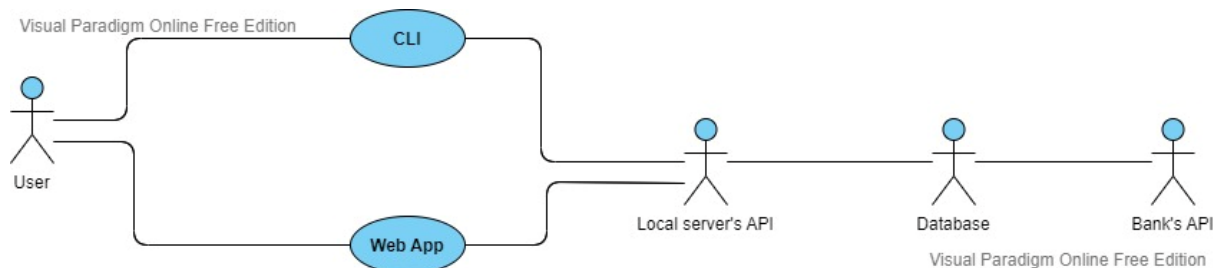
Η επαφή με τον χρήστη γίνεται μέσω Command Line Interface(CLI) και μέσω ενός frontend app για το οποίο έχει χρησιμοποιηθεί HTML, CSS, PHP και JavaScript. Η έξοδος των δεδομένων στην εφαρμογή Frontend γίνεται μέσω ενός RESTful API γραμμένο σε JavaScript (nodeJS).

RESTful API: Το API έχει περιορισμένη πρόσβαση στην βάση δεδομένων ώστε όταν δέχεται τα διάφορα HTTP requests (π.χ. get, put, post κ.α.) να μπορεί να εκτελέσει την ενέργεια που του ζητήθηκε.

Command Line Interface: Το CLI αποτελεί επαφή των χρηστών με την εφαρμογή μέσω του οποίου γίνονται κλήσεις για την εκτέλεση των βασικών λειτουργιών της εφαρμογής.

Front-End Application: Στο front-end ο χρήστης και οι διαχειριστές θα μπορούν να έχουν πρόσβαση στις διενέργειες που εκτελεί το λογισμικό.

Παρακάτω παρατίθεται το Use Case UML diagram:



2. Αναφορές - πηγές πληροφοριών

N/A

3. Προδιαγραφές απαιτήσεων λογισμικού

3.1 Περιπτώσεις χρήσης

3.1.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 1: **Διέλευση οχήματος από τα διόδια**

3.1.1.1 *Χρήστες (ρόλοι) που εμπλέκονται*

- *Χρήστης - Οδηγός του οχήματος*
- *Λειτουργός του εκάστοτε σταθμού διοδίων*

3.1.1.2 *Προϋποθέσεις εκτέλεσης*

- Επιτυχής αναγνώριση του πάσου
- Έλεγχος επαρκούς υπολοίπου

3.1.1.3 *Περιβάλλον εκτέλεσης*

- Διεπαφή CLI
- Database

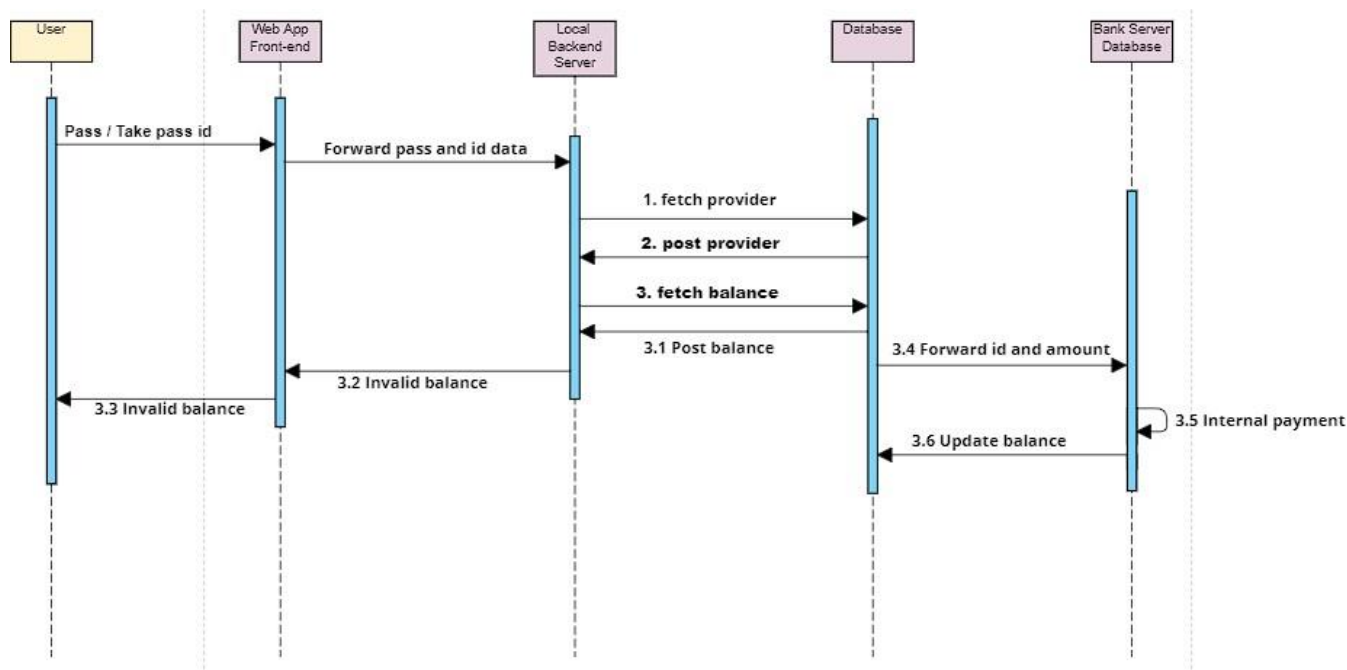
3.1.1.4 *Δεδομένα εισόδου*

- *Tag Id*
- *Ημερομηνία διέλευσης*
- *Id σταθμού των διοδίων*
- *Χρηματικό Αντίτιμο*
 - *Πρέπει το υπόλοιπο να επαρκεί για να πραγματοποιηθεί η διέλευση*

3.1.1.5 *Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά*

1. Λήψη του Tag Id, του παρόχου του και του Id του σταθμού διοδίων
2. Λήψη του αντίτιμου και του υπολοίπου του tag (σε περίπτωση ανεπαρκούς υπολοίπου του tag θα επιστρέφεται μήνυμα σφάλματος)
3. Αφαίρεση του αντιτίμου
4. Καταγραφή του ποσού για τον πάροχο

To UML Sequence Diagram:



3.1.1.6 Δεδομένα Εξόδου

Πραγματοποιείται καινούρια εγγραφή διέλευσης στη βάση και ενημερώνεται το χρηματικό υπόλοιπο του πάσου.

3.1.1.7 Παρατηρήσεις

N/A

3.2.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 2: Καταγραφή και υπολογισμός οφειλών μεταξύ 2 λειτουργιών των αυτοκινητόδρομων

3.1.2.1 Χρήστες (ρόλοι) που εμπλέκονται

- Λειτουργοί των αυτοκινητόδρομων

3.1.2.2 Προϋποθέσεις εκτέλεσης

- Λήψη του tag Id κατά τη διέλευση
- Λήψη παρόχου του tag και του λειτουργού του αυτοκινητόδρομου κατά τη διέλευση
- Λήψη του αντιτίμου διέλευσης με τις απαραίτητες πληροφορίες

3.1.2.3 Περιβάλλον εκτέλεσης

- Front-end (CLI ή Web App)
- Back-end server's API
- Database

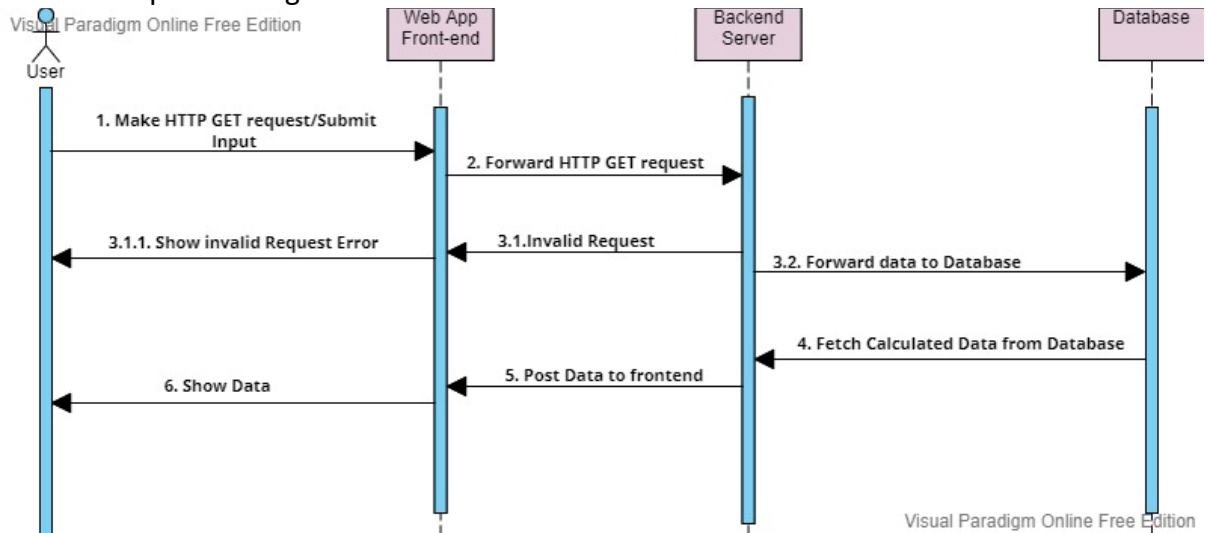
3.1.2.4 Δεδομένα εισόδου

- Id λειτουργού αυτοκινητόδρομου 1
- Id λειτουργού αυτοκινητόδρομου 2
- Αρχή περιόδου
- Τέλος περιόδου

3.1.2.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

1. Γίνεται αίτημα από τον λειτουργό για την εμφάνιση των οφειλών
2. Το σύστημα λαμβάνει τα στοιχεία εισόδου
3. Πραγματοποιεί έλεγχο για όλες τις διελεύσεις που αφορούν τους συγκεκριμένους χειριστές οδών για την περίοδο που δόθηκε και υπολογίζεται το τελικό ποσό.
4. Επιστρέφεται το ποσό στο front-end.

To UML Sequence Diagram:



3.1.2.6 Δεδομένα εξόδου

Εμφάνιση του ποσού οφειλής μεταξύ του λειτουργού 1 και 2 και σε περίπτωση πληρωμής αυτού, ενημερώνεται και το table transactions στη βάση δεδομένων.

3.1.2.7 Παρατηρήσεις

N/A

3.3.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 3: Υπολογισμός οφειλών προς έναν λειτουργό αυτοκινητόδρομου

3.1.3.1 Χρήστες (ρόλοι) που εμπλέκονται

- Λειτουργοί των αυτοκινητόδρομων

3.1.3.2 Προϋποθέσεις εκτέλεσης

- Λήψη του tag Id κατά τη διέλευση

- Λήψη παρόχου του tag και του λειτουργού του αυτοκινητόδρομου κατά τη διέλευση
- Λήψη του αντιτίμου διέλευσης με τις απαραίτητες πληροφορίες

3.1.3.3 Περιβάλλον εκτέλεσης

- Front-end (CLI ή Web App)
- Back-end server's API
- Database

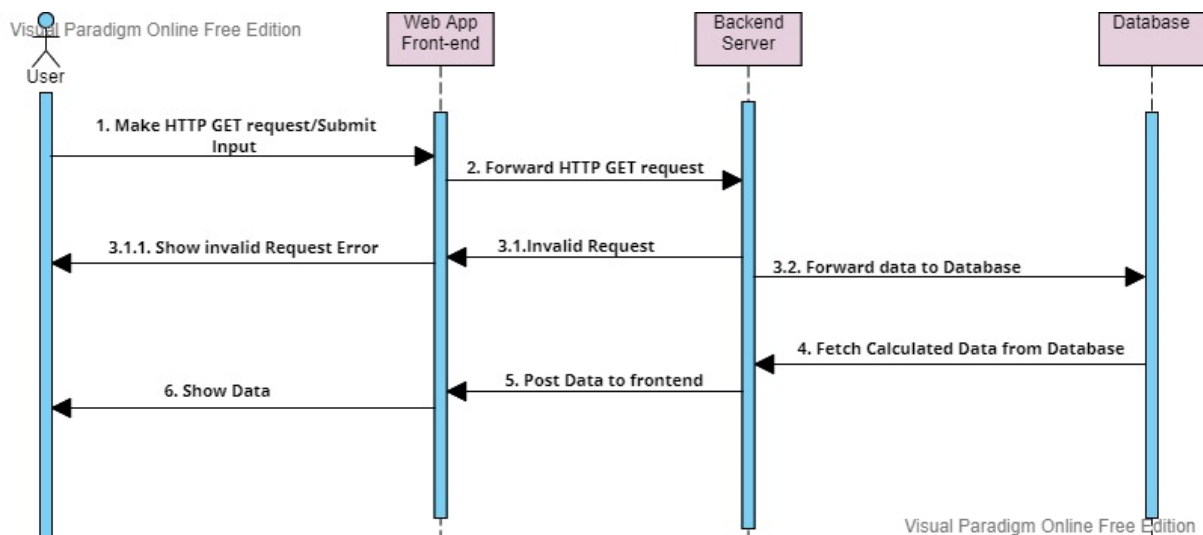
3.1.3.4 Δεδομένα εισόδου

- Id λειτουργού αυτοκινητόδρομου
- Αρχή περιόδου
- Τέλος περιόδου

3.1.3.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

1. Γίνεται αίτημα από τον λειτουργό για την εμφάνιση των οφειλών από τους υπόλοιπους λειτουργούς
2. Το σύστημα λαμβάνει τα στοιχεία εισόδου
3. Πραγματοποιεί έλεγχο για όλες τις διελεύσεις που αφορούν τους συγκεκριμένους χειριστές οδών για την περίοδο που δόθηκε και υπολογίζεται το τελικό ποσό.
4. Επιστρέφεται το ποσό στο front-end.

To UML Sequence Diagram:



3.1.3.6 Δεδομένα εξόδου

Εμφάνιση του ποσού οφειλής για τον λειτουργό και σε περίπτωση πληρωμής αυτού, ενημερώνεται και το table transactions στη βάση δεδομένων.

3.1.3.7 Παρατηρήσεις

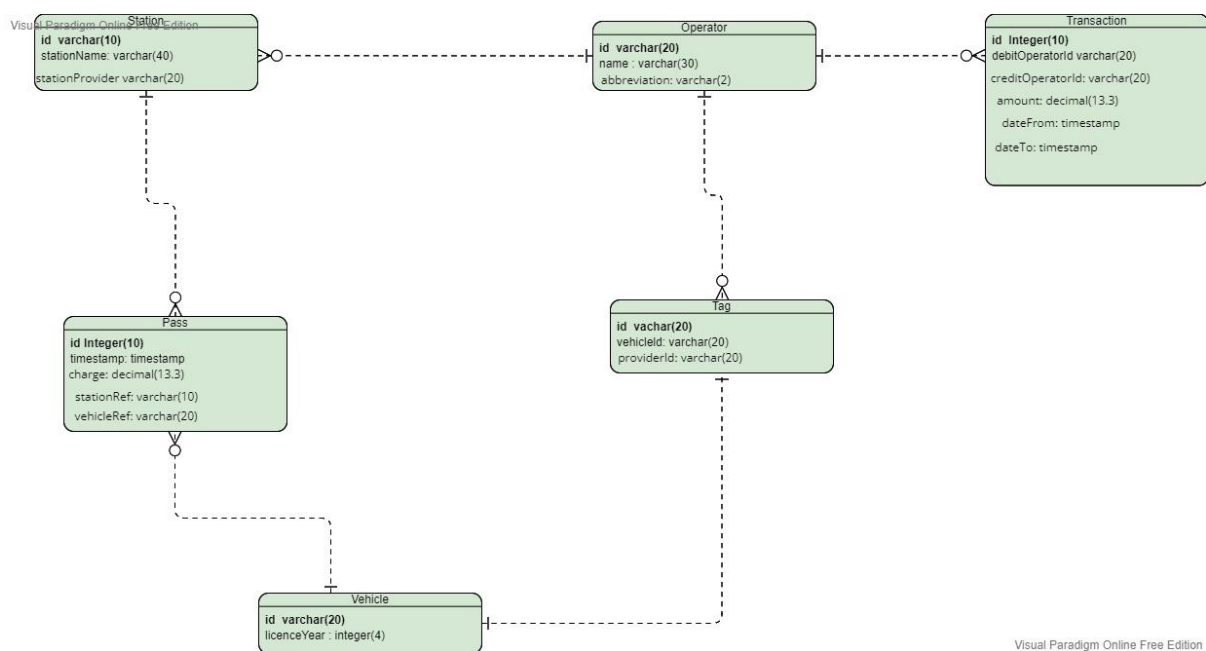
N/A

3.2 Απαιτήσεις επιδόσεων

Η βασικότερη απαίτηση είναι η συνεχής λειτουργία του λογισμικού. Επιπλέον, πρέπει σε περιόδους έντονων μετακινήσεων να ανταποκρίνεται αποδοτικά προς όλους τους χρήστες και για όλες τις λειτουργίες που προσφέρει. Όσον αφορά την διεπαφή με τον χρήστη, θέλουμε το API να ανταποκρίνεται σε ένα σημαντικό πλήθος ταυτόχρονων αιτημάτων. Η ενέργεια της συσχέτισης του tag id του χρήστη με τα υπόλοιπα στοιχεία του πάσου που είναι απαραίτητα κάθε φορά πρέπει να παρέχεται σε ένα μικρό χρονικό διάστημα, μη αντιληπτό στο χρήστη. Τέλος, κάθε περίπτωση χρήσης πρέπει να ολοκληρώνεται σε δευτερόλεπτα.

3.3 Απαιτήσεις οργάνωσης δεδομένων

Entity Relational Diagram:



3.3.1 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

Τα προσωπικά δεδομένα κάθε tag είναι ορατά μόνο στους διαχειριστές της πλατφόρμας, στους χρήστες είναι όρατα μόνο όσα τους αφορούν και τα εμπλεκόμενα συστήματα έχουν πρόσβαση μόνο σε όσα δεδομένα κρίνονται απαραίτητα για την ολοκλήρωση της εκάστοτε ενέργειας.

3.5 Λοιπές απαιτήσεις

3.5.1 Απαιτήσεις διαθεσιμότητας λογισμικού

Το τελικό λογισμικό είναι απαραίτητο να είναι διαθέσιμο όλες τις ώρες της ημέρας για όλους τους χρήστες. Το παραπάνω εξασφαλίζεται με τους εξής τρόπους:

- Σε περίπτωση που κάποιος server βρεθεί unreachable (είτε λόγω τεχνικού προβλήματος είτε λόγω συντήρησης) θα πρέπει ένας διαφορετικός server ο οποίος περιέχει ακριβές

αντίγραφα των αρχείων του αρχικού να συνδέεται στο δίκτυο στη θέση του.

- Να υπάρχει μονίμως κατάλληλη και αξιόπιστη σύνδεση στο διαδίκτυο με δυνατότητα εξυπηρέτησης μεγάλου φορτίου.

3.5.2 Απαιτήσεις ασφάλειας

Οι ενέργειες που έχει την δυνατότητα να κάνει ο χρήστης γίνονται μέσω του web app. Επομένως είναι απαραίτητη η χρήση HTTPS και SSL certificate για την ασφάλεια των αιτημάτων στο δίκτυο.

Στην βάση προστατεύονται τα προσωπικά δεδομένα των χρηστών παραχωρώντας τα ανάλογα δικαιώματα ανάγνωσης και εγγραφής