Έγγραφο απαιτήσεων λογισμικού (SRS)

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

DiodeIO

# Εισαγωγή

## 1.1 Εισαγωγή: σκοπός του λογισμικού

Οι σύγχρονοι σταθμοί διοδίων σε μεγάλους αυτοκινητόδρομους αποτελούν κάτι πολύ διαδεδομένο στη σημερινή εποχή και εμπλέκουν ένα πλήθος διαφορετικών εταιρειών. Σκοπός μας είναι η ανάπτυξη ενός λογισμικού που να εξασφαλίζει τη διαλειτουργικότητα μεταξύ πολλαπλών εταιρειών εκμετάλλευσης διοδίων της χώρας. Οι εταιρείες διαχείρισης αυτοκινητοδρόμων θα παρέχουν στην εφαρμογή μας δεδομένα σχετικά με γεγονότα διελεύσεων που πραγματοποιούνται στους σταθμούς διοδίων που τους ανήκουν. Στη συνέχεια μετά την κατάλληλη επεξεργασία των δεδομένων αυτών, οι εταιρείες θα ενημερώνονται από το λογισμικό μας για τα ποσά που οφείλουν ή τους οφείλουν άλλες εταιρείες. Επιπλέον, δίνεται η δυνατότητα σε κάθε εμπλεκόμενο μέλος να συνδεθεί στην εφαρμογή και να αντλήσει πληθώρα δεδομένων σχετικά με την επισκεψιμότητα των σταθμών, τις τραπεζικές συναλλαγές που έχουν πραγματοποιηθεί, τη συμφόρηση της κυκλοφορίας των οδικών δικτύων της χώρας κ.λπ.

## 1.2 Διεπαφές (interfaces)

### 1.2.1 Διεπαφές με εξωτερικά συστήματα

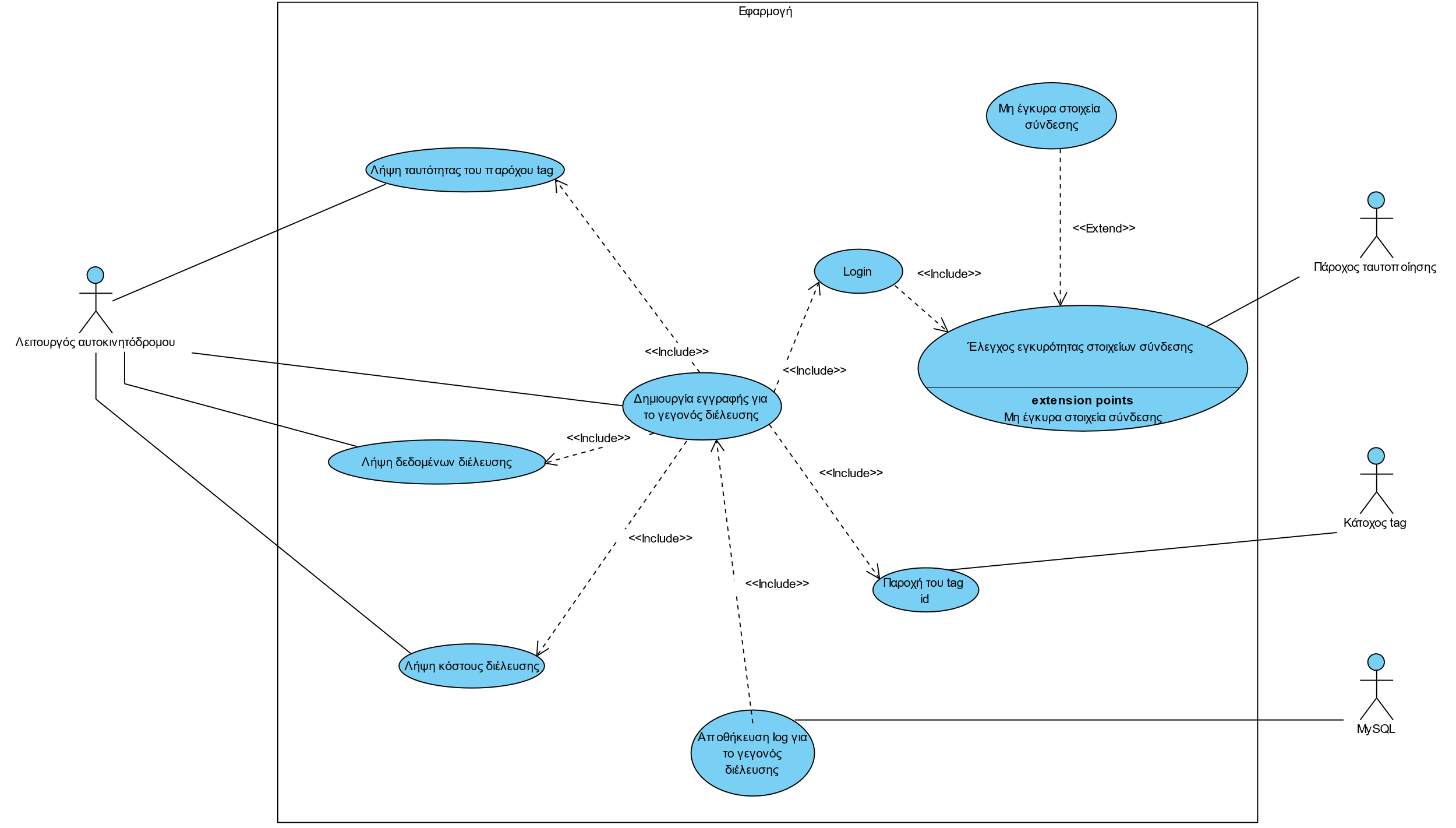
Τα εξωτερικά συστήματα με τα οποία θα έρχεται σε επαφή η εφαρμογή για την ανάκτηση και την προώθηση δεδομένων περιλαμβάνουν:

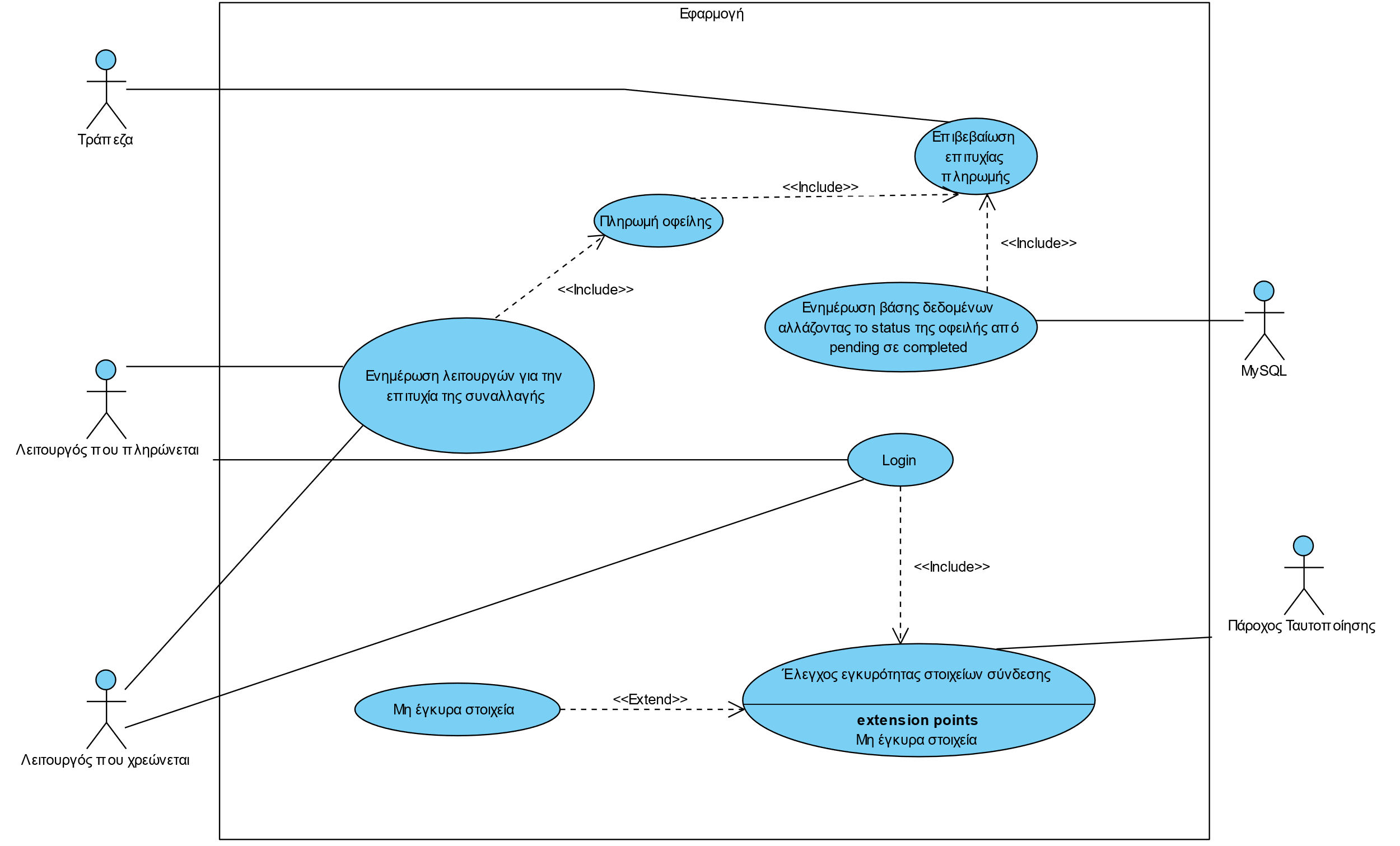
* Τα API των διαφόρων διαχειριστών αυτοκινητοδρόμων, μέσω των οποίων θα ανακτούμε δεδομένα σχετικά με τα γεγονότα διελεύσεων και φορτίσεων στους σταθμούς διοδίων
* Τις διεπαφές των Παρόχων Υπηρεσιών Πληρωμών ώστε να αποστέλλονται οι διακανονισμοί που υπολογίζονται από την εφαρμογή μας και να δρομολογούνται οι πληρωμές ανάμεσα στους λειτουργούς
* Το API της υπηρεσίας που μας παρέχει τον χάρτη το οποίο θα χρησιμοποιήσουμε για την οπτικοποίηση των εξαγόμενων, από το λογισμικό μας, στατιστικών

Οι παραπάνω διεπαφές υλοποιούνται στο διαδίκτυο μέσω RESTful API’s τα οποία καταναλώνει η εφαρμογή μας στέλνοντας HTTP Requests στα εκτεθειμένα τους endpoints. Η ανταλλαγή δεδομένων με αυτή την τεχνολογία πραγματοποιείται με πακέτα JSON και είναι κρυπτογραφημένη με πρωτόκολλο HTTPS, υπογεγραμμένο με Self-Signed Certificate.

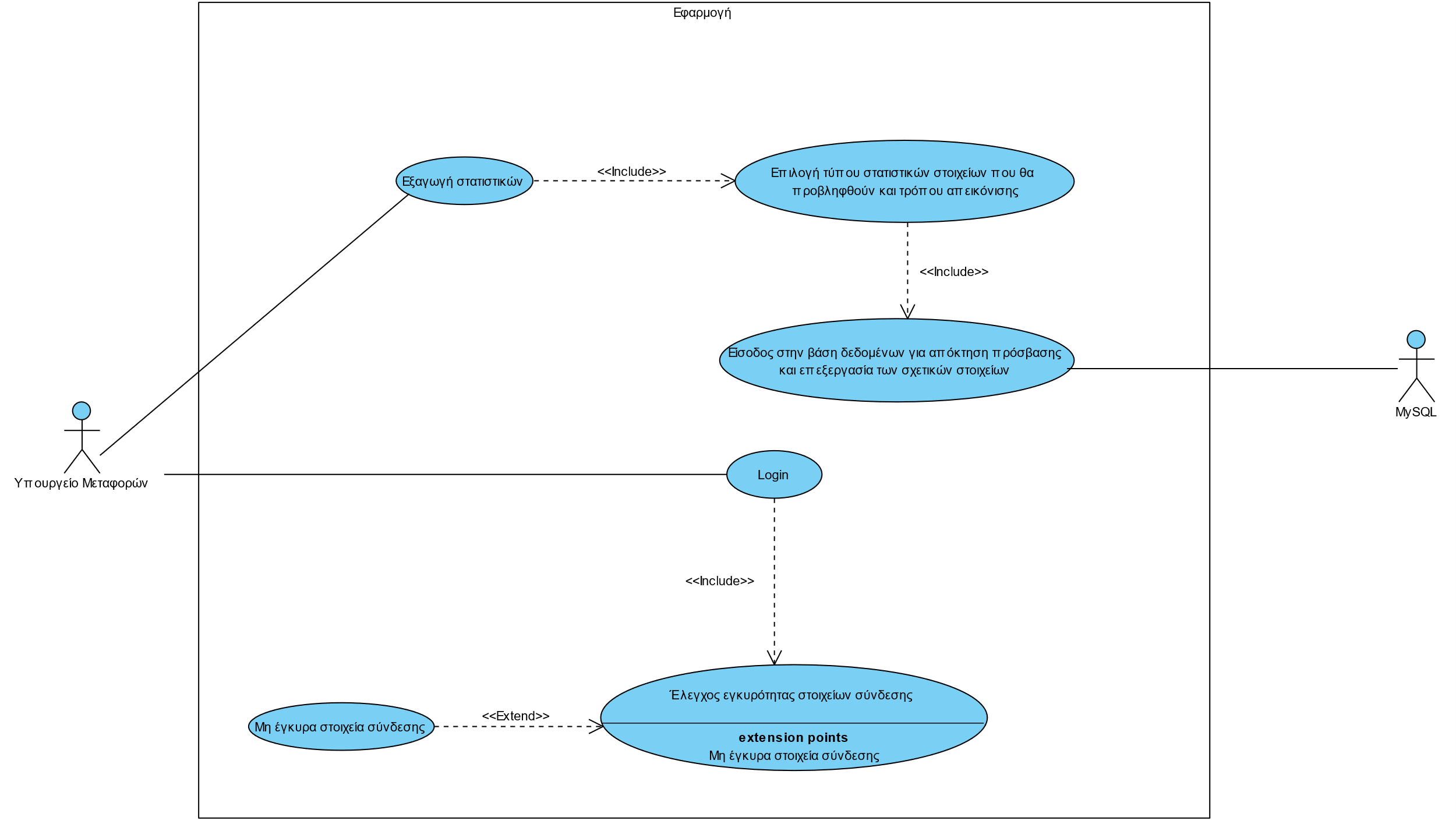
### 1.2.2 Διεπαφές με το χρήστη

Τα δύο βασικά υποσυστήματα τα οποία παρέχουν τη δυνατότητα στα εμπλεκόμενα μέλη να αλληλεπιδρούν με την εφαρμογή είναι ένα Command Line Interface (CLI) και ένα Frontend Web Application. Συγκεκριμένα, οι δύο αυτές διεπαφές θα υποστηρίζουν διάφορες λειτουργίες επεξεργασίας και προβολής δεδομένων που παρέχονται από τον κύριο server του λογισμικού. Τέλος, σημειώνεται πως τρίτες εφαρμογές μπορούν να χρησιμοποιήσουν απευθείας τις λειτουργίες του API μας, στέλνοντας απευθείας αιτήματα στα κατάλληλα endpoints.

**UML Use Case διάγραμμα για διέλευση οχήματος**

****

**UML Use Case διάγραμμα για διεκπεραίωση συναλλαγών**

**UML Use Case διάγραμμα για προβολή στατιστικών στοιχείων**

# Αναφορές - πηγές πληροφοριών

Στην εμβέλεια της δικής μας εφαρμογής δεν χρησιμοποιείται συγκεκριμένος πάροχος υπηρεσιών πληρωμών. Παρά ταύτα μπορούμε να προτείνουμε τα εξής Payment API που μπορούν να υλοποιήσουν τις τραπεζικές συναλλαγές μεταξύ των λειτουργών:

* Stripe: <https://stripe.com>
* Google Pay API: <https://developers.google.com/pay/api>

Επιπλέον, για την αναπαράσταση των εξαγόμενων στατιστικών σε μορφή χάρτη μπορούμε να χρησιμοποιήσουμε κάποιο από τα εξής έτοιμα API:

* Mapbox: [https://www.mapbox.com](https://www.mapbox.com/)
* Google Maps Platform: <https://developers.google.com/maps>

# Προδιαγραφές απαιτήσεων λογισμικού

## 3.1 Περιπτώσεις χρήσης

### 3.1.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 1: ΔΙΕΛΕΥΣΗ ΟΧΗΜΑΤΟΣ

#### 3.1.1.1 Χρήστες (ρόλοι) που εμπλέκονται

* Εταιρείες διαχείρισης αυτοκινητοδρόμων
* Η εφαρμογή μας

#### 3.1.1.2 Προϋποθέσεις εκτέλεσης

* Ανίχνευση του πομποδέκτη από τον μηχανισμό του σταθμού
* Προσπάθεια διέλευσης οχήματος
* Καταγραφή δεδομένων διέλευσης:
  + Αναγνωριστικό του οχήματος
  + Ημερομηνία και ώρα
  + Αντίτιμο διέλευσης
  + Σταθμός διέλευσης
  + Εταιρεία που υπάγεται ο πομποδέκτης
* Αποστολή της εγγραφής στην εφαρμογή μας
* Καταχώρηση του γεγονότος στη βάση δεδομένων της εφαρμογής

#### 3.1.1.3 Περιβάλλον εκτέλεσης

Μέσω της διεπαφής των λειτουργών των αυτοκινητοδρόμων με την εφαρμογή μας θα γίνεται η μεταφορά των δεδομένων ενός γεγονότος διέλευσης και θα καταχωρείται αυτόματα στη βάση δεδομένων της εφαρμογής.

#### 3.1.1.4 Δεδομένα εισόδου

* **Ταυτοποίηση Operator**: Κάθε συνεργαζόμενος λειτουργός θα εισέρχεται στην εφαρμογή με μοναδικό όνομα χρήστη και κωδικό
* **Δεδομένα γεγονότος διέλευσης**: Η καταγραφή που θα αποστέλλεται στην εφαρμογή μας θα έχει την μορφή της εξής τούπλας  
  (tag\_id, timestamp, point\_ref, amount, operator\_tag, operator\_toll)
  + tag\_id: Το αναγνωριστικό του πομποδέκτη
  + point\_ref: Το αναγνωριστικό του σταθμού διοδίων
  + amount: Το ποσό πληρωμής
  + operator\_tag: Το αναγνωριστικό του παρόχου του πομποδέκτη
  + operator\_toll: Το αναγνωριστικό της εταιρείας που υπάγεται ο σταθμός διοδίων

#### 3.1.1.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

**Βήμα 1:** Είσοδος στην εφαρμογή με τα credentials του λειτουργού πράγμα που συνοδεύεται από τον έλεγχο εγκυρότητας των στοιχείων

**Βήμα 2:** Επεξεργασία των δεδομένων ενός γεγονότος διέλευσης στη μορφή που απαιτείται

**Βήμα 3:** Αποστολή γεγονότος διέλευσης στην εφαρμογή

**Βήμα 4:** Έλεγχος εγκυρότητας των δεδομένων για το εάν βρίσκονται στη σωστή μορφή

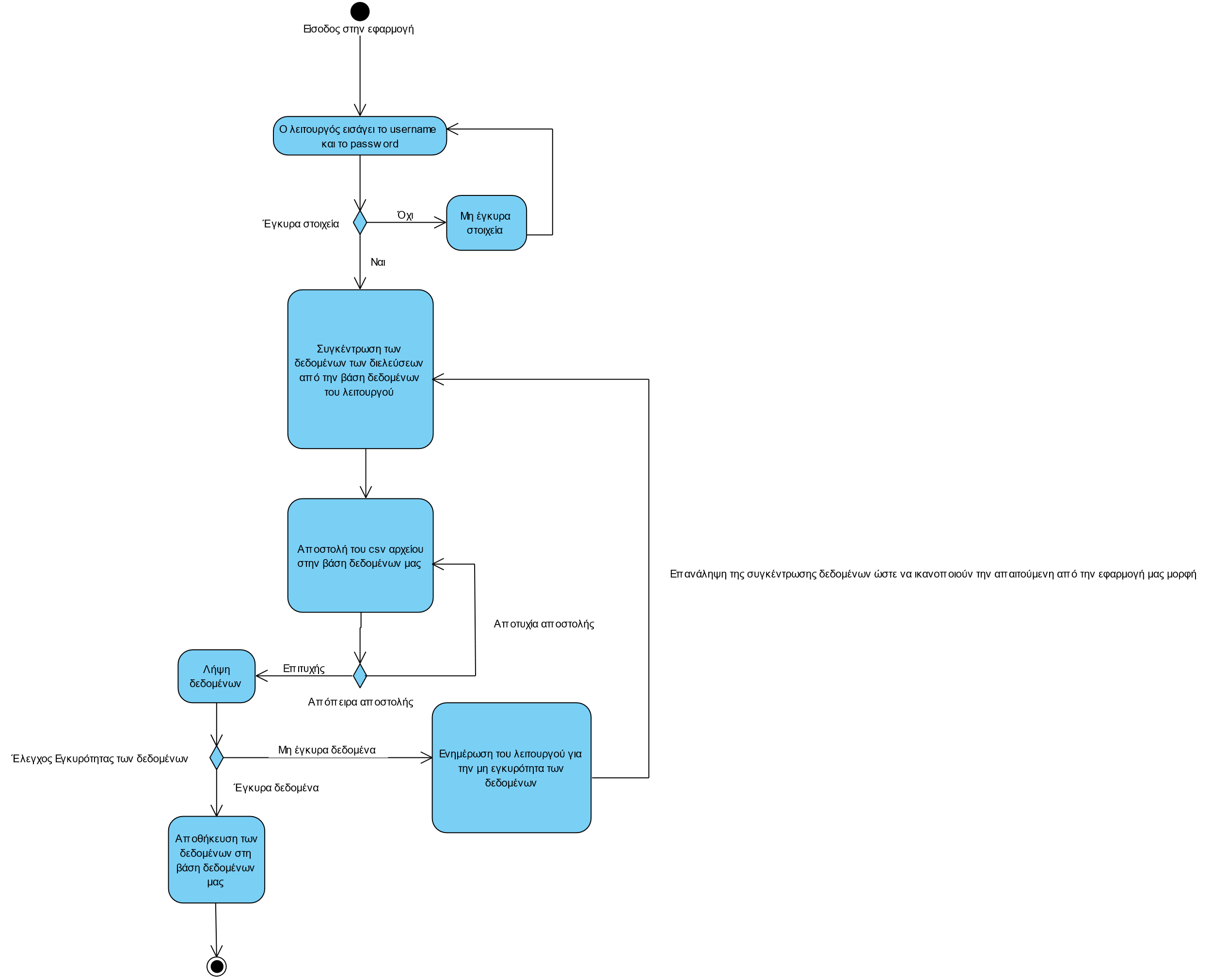
**Βήμα 5:** Ενημέρωση του χειριστή για την εγκυρότητα των δεδομένων

**Βήμα 6:** Εάν τα δεδομένα είναι έγκυρα, αποθήκευση της καταγραφής στη βάση δεδομένων

#### 3.1.1.7 Δεδομένα εξόδου

Ενημέρωση του operator για την επιτυχή καταχώρηση των δεδομένων της διέλευσης (ή των διελεύσεων) στην βάση δεδομένων της εφαρμογή μας. Σε περίπτωση αποτυχίας εμφάνιση κατάλληλου μηνύματος στον χρήστη σχετικά με το πρόβλημα που προέκυψε (π.χ. μη έγκυρα δεδομένα ή εσωτερικό σφάλμα της εφαρμογής)

#### 3.1.1.8 Παρατηρήσεις

N/A

### 3.1.2 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 2: ΕΞΟΦΛΗΣΗ ΟΦΕΙΛΩΝ

#### 3.1.2.1 Χρήστες (ρόλοι) που εμπλέκονται

* Οι Payment Service Providers
* Ο λειτουργός που πληρώνεται καθώς και ο λειτουργός που πιστώνεται
* Η εφαρμογή μας

#### 3.1.2.2 Προϋποθέσεις εκτέλεσης

* Οι πληρωμές μεταξύ δύο εταιρειών θα γίνονται μετά την παρέλευση ενός προσυμφωνηθέντος χρονικού διαστήματος
* Η εφαρμογή να παρέχει έγκυρα δεδομένα (δηλαδή να έχει υπολογίσει σωστά τις οφειλές) στους παρόχους υπηρεσιών πληρωμών ώστε να εκτελούνται ομαλά οι συναλλαγές

#### 3.1.2.3 Περιβάλλον εκτέλεσης

Η εξόφληση των οφειλών μεταξύ δύο εταιρειών πραγματοποιείται στο υπολογιστικό σύστημα του αρμόδιου Payment Service Provider. Αν η συναλλαγή είναι επιτυχής, η τράπεζα οφείλει να ενημερώσει και το δικό μας σύστημα για την επιτυχία της και έτσι να αλλάξει την κατάσταση της δεδομένης οφειλής σε ολοκληρωμένη.

#### 3.1.2.4 Δεδομένα εισόδου

* **Παράμετροι Διακανονισμού**: Οι παράμετροι που θα καθορίσουν τον υπολογισμό του διακανονισμού μεταξύ δύο εταιρειών είναι προκαθορισμένοι στη βάση δεδομένων του συστήματος και είναι της μορφής:

(operator\_id\_1, operator\_id\_2, date\_from, date\_to)

* + operator\_id\_1, operator\_id\_2: Τα αναγνωριστικά των εμπλεκόμενων εταιρειών
  + date\_from, date\_to: Ημερομηνίες αρχής και τέλους διακανονισμού

#### 3.1.2.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

**Βήμα 1**: Επιλογή εκκρεμούσας οφειλής μεταξύ δύο εταιρειών για το προκαθορισμένο χρονικό διάστημα από το λογισμικό

**Βήμα 2**: Υπολογισμός της οφειλής: Εύρεση λειτουργού που πιστώνεται και λειτουργού που πληρώνεται. Τα στοιχεία της οφειλής οργανώνονται στη μορφή:  
(operator\_credited, operator\_debited, date\_from, date\_to, amount)

**Βήμα 3**: Αποστολή των στοιχείων συναλλαγής στην τράπεζα

**Βήμα 4**: Αναμονή για σήμα έγκρισης και ολοκλήρωσης της συναλλαγής

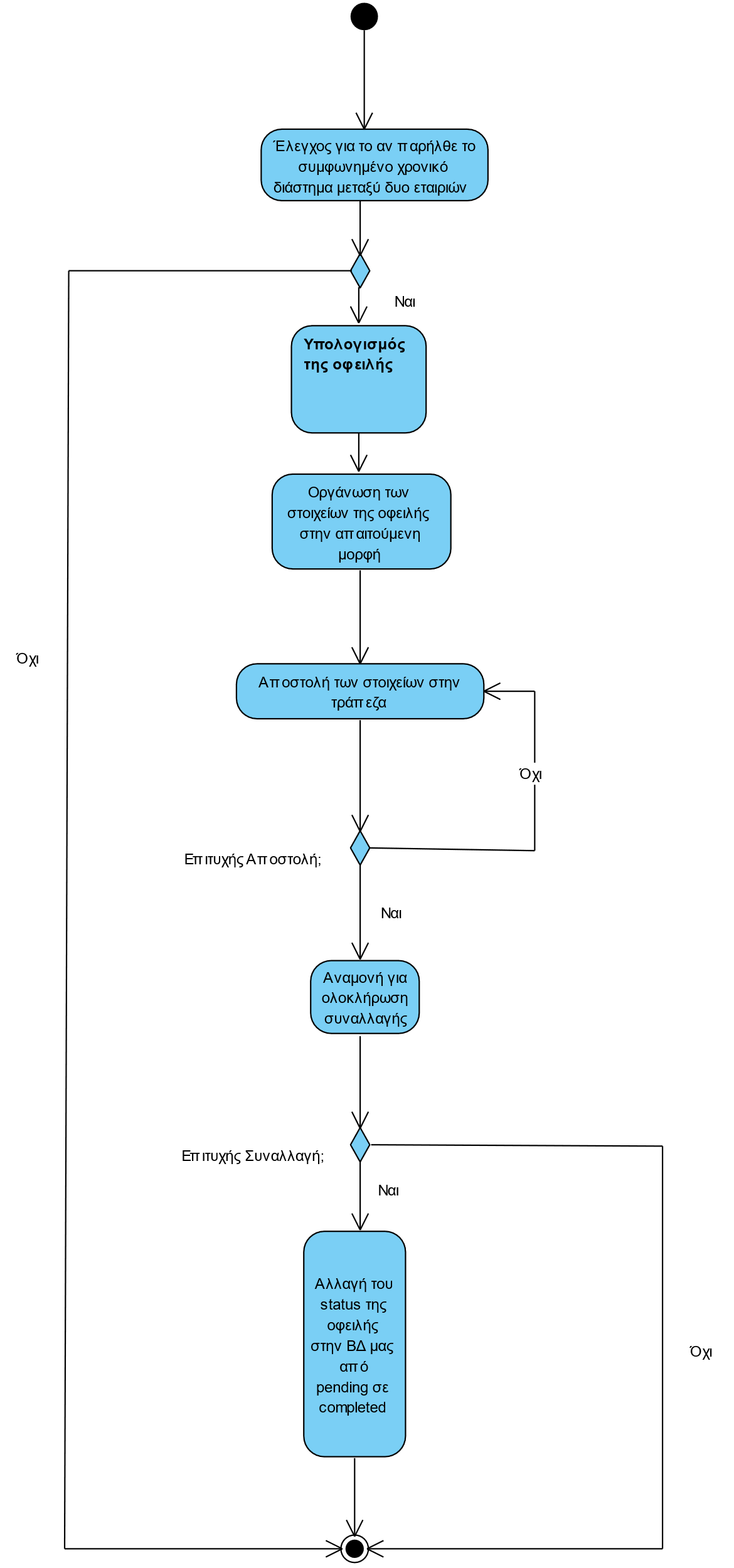
**Βήμα 5**: Αν ολοκληρώθηκε με επιτυχία η συναλλαγή, ανανέωση της οφειλής στη βάση δεδομένων από εκκρεμούσα σε ολοκληρωμένη

#### 3.1.2.7 Δεδομένα εξόδου

Μετά την αποστολή των στοιχείων οφειλής, η τράπεζα ενημερώνει την εφαρμογή μας σχετικά με την επιτυχία ή όχι της συναλλαγής. Επιπλέον, η εγγραφή στη βάση δεδομένων της εφαρμογής ανανεώνεται σε ολοκληρωμένη, με βάση την απόφαση της τράπεζας.

#### 3.1.2.8 Παρατηρήσεις

N/A



### 3.1.3 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 3: ΣΤΑΤΙΣΤΙΚΑ ΔΕΔΟΜΕΝΑ ΓΙΑ ΤΟ ΥΠΟΥΡΓΕΙΟ ΜΕΤΑΦΟΡΩΝ

#### 3.1.3.1 Χρήστες (ρόλοι) που εμπλέκονται

* Το Υπουργείο Μεταφορών
* Η εφαρμογή μας

#### 3.1.3.2 Προϋποθέσεις εκτέλεσης

Πρέπει να υπάρχουν όσο το δυνατόν περισσότερες συνεργαζόμενες εταιρείες που χρησιμοποιούν την εφαρμογή ώστε να έχει νόημα η εξαγωγή γενικών στατιστικών και προβλέψεων για τις διελεύσεις των οχημάτων. Επιπλέον, είναι απαραίτητο τα δεδομένα σχετικά με τα γεγονότα διελεύσεων να αποθηκεύονται για μεγάλα χρονικά διαστήματα στην βάση, προκειμένου να είναι δυνατή η εξαγωγή μακροχρόνιων στατιστικών και προβλέψεων.

#### 3.1.3.3 Περιβάλλον εκτέλεσης

Το κύριο περιβάλλον στο οποίο θα προβάλλονται στατιστικά δεδομένα σε λογαριασμούς του Υπουργείου θα είναι η διαδικτυακή εφαρμογή όπου θα οπτικοποιούνται τα αναλυτικά στοιχεία που θα παράγει το API σε μορφή διαγραμμάτων (π.χ. piecharts, histograms, πίνακες) ή σε μορφή χάρτη. Επιπλέον, μπορεί κανείς μέσω της CLI εφαρμογής ή απευθείας από το API να ανακτήσει κάποια από τα δεδομένα καθώς παρέχονται λειτουργίες που αφορούν την ανάλυση διελεύσεων με βάση δεδομένο σταθμό, χρονικό διάστημα κ.λπ.

#### 3.1.3.4 Δεδομένα εισόδου

Στα δεδομένα εισόδου συγκαταλέγονται:

* Ταυτοποίηση Χρήστη Υπουργείου Μεταφορών: Κάθε χρήστης θα εισέρχεται στην εφαρμογή με μοναδικό όνομα χρήστη και κωδικό
* H χρονική περίοδος, ο σταθμός διέλευσης, η εταιρεία – εταιρείες ενδιαφέροντος

#### 3.1.3.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

**Βήμα 1**: Είσοδος στην εφαρμογή με τα credentials του Υπουργείου πράγμα που συνοδεύεται από τον έλεγχο εγκυρότητας των στοιχείων

**Βήμα 2**: Επιλογή τύπου στατιστικού που επιθυμεί ο χρήστης να προβάλλει:

* **PassesPerStation**: Ανάλυση διελεύσεων για δεδομένο σταθμό διοδίων και συγκεκριμένη περίοδο
* **PassesAnalysis**: Ανάλυση γεγονότων διέλευσης που πραγματοποιήθηκαν με tag ενός παρόχου σε σταθμούς διοδίων ενός διαφορετικού παρόχου σε συγκεκριμένη χρονική περίοδο
* **PassesCost**: Πλήθος διελεύσεων που πραγματοποιήθηκαν σε συγκεκριμένη χρονική περίοδο με πομποδέκτη ενός παρόχου σε σταθμούς διοδίων ενός διαφορετικού παρόχου
* **Προβολή Χάρτη / Ιστογράμματος / Διαγράμματος** που περιλαμβάνει τους σταθμούς διοδίων και το πλήθος των διελεύσεων που έχουν πραγματοποιηθεί σε αυτούς για δεδομένη χρονική περίοδο

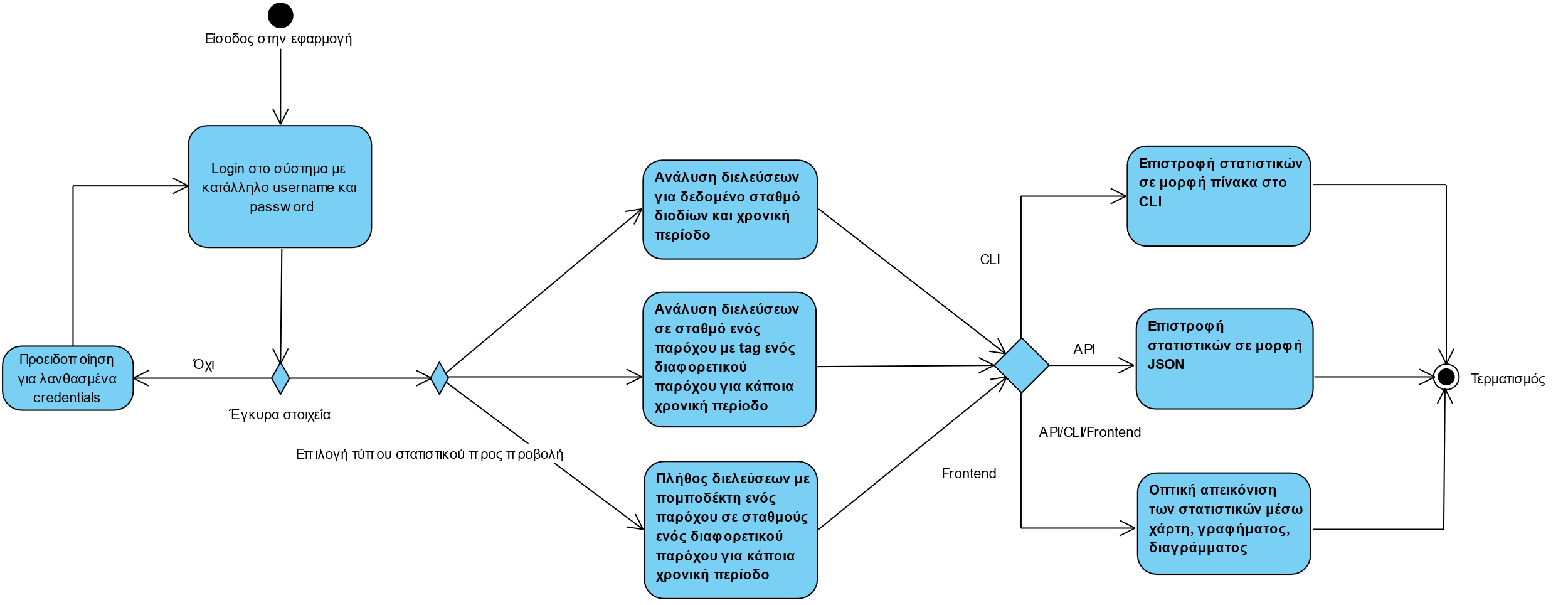
**Βήμα 3**: Επανάληψη του βήματος 2 εάν ο χρήστης επιθυμεί να δει κι άλλα αναλυτικά στοιχεία

#### 3.1.3.7 Δεδομένα εξόδου

Στα δεδομένα εξόδου συγκαταλέγονται τα εξής:

* Στατιστικά δεδομένα (σε ευανάγνωστη μορφή)
* Δεδομένα ανά σταθμό, χρονική περίοδο, εταιρεία σε μορφή JSON ή CSV
* Προβλέψεις σχετικά με τη συμφόρηση του οδικού δικτύου σε μία περιοχή και κάποια χρονική στιγμή

#### 3.1.3.8 Παρατηρήσεις

N/A

## 3.2 Απαιτήσεις επιδόσεων

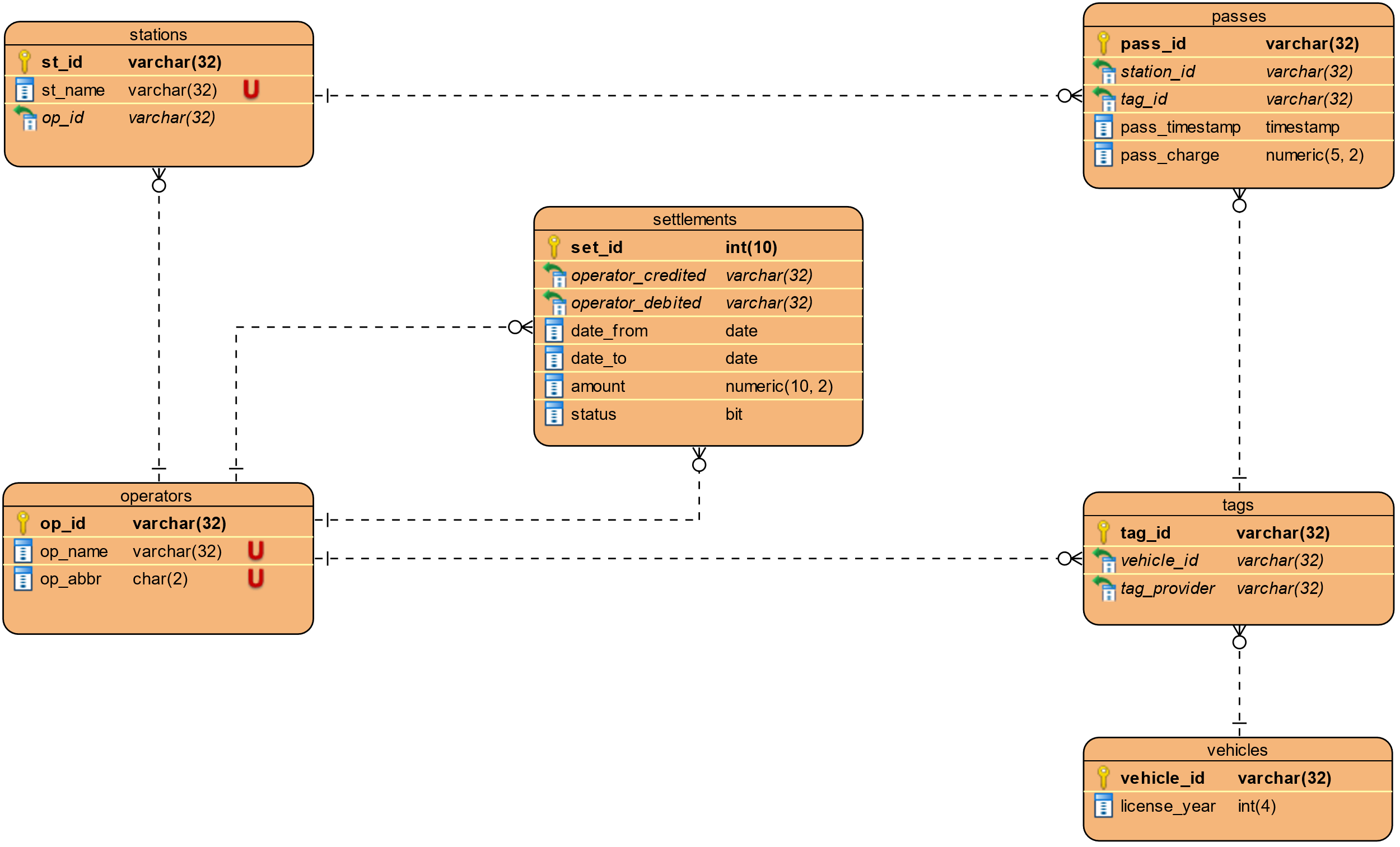
Δεδομένης της συνεχόμενης λειτουργίας των συνεργαζόμενων αυτοκινητοδρόμων και της μεγάλης κίνησης στο οδικό δίκτυο της χώρας θα πρέπει η εφαρμογή μας, ιδιαίτερα τις ώρες κυκλοφορικής αιχμής, να βρίσκεται συνδεδεμένη στο διαδίκτυο και έτοιμη να λάβει ένα προς ένα τα γεγονότα διελεύσεων και να τα επεξεργαστεί σε χρόνο της τάξης των millisecond, ώστε να γλιτώσουμε πολύτιμους υπολογιστικούς πόρους. Σε περίπτωση που ο κεντρικός server δεν είναι online, απαιτείται ο συγχρονισμός των δεδομένων μας με τα νέα γεγονότα διελεύσεων πράγμα που χρειάζεται πολύ χρόνο να ολοκληρωθεί, καθώς καλούμαστε να επεξεργαστούμε μεγάλο όγκο εγγραφών μονομιάς.

Επιπλέον, είναι κοινό συμφέρον των συνεργαζόμενων εταιρειών, η εφαρμογή να είναι δεκτική στην ενσωμάτωση νέων εταιρειών στο σύστημα πράγμα που καθιστά την ανάγκη για κλιμάκωση και εξοικονόμηση των υπολογιστικών μας πόρων ακόμα πιο επιτακτική.

Όσον αφορά την εμπειρία των χρηστών της Frontend διεπαφής, η οποία περιλαμβάνει την παρουσίαση υπολογιζόμενων στατιστικών και δεδομένων οφειλών με φιλικό τρόπο, είναι σημαντικό οι χρόνοι απόκρισης του API, του επιλεγμένου συστήματος διαχείρισης της βάσης δεδομένων αλλά και των third party API να είναι στιγμιαίοι. Αυτό προϋποθέτει την βελτιστοποίηση της υλοποίησης της εφαρμογής από μεριάς μας καθώς επίσης και τη χρήση non-blocking backend τεχνολογιών που μπορούν να λάβουν μαζικά αιτήματα και να τα χειριστούν ομαλά, ακόμα και σε στιγμές που ο server πρέπει να ικανοποιήσει πολλούς χρήστες ταυτόχρονα.

## 3.3 Απαιτήσεις οργάνωσης δεδομένων

### 3.3.1 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

Το σχεσιακό διάγραμμα της βάσης δεδομένων που θα χρησιμοποιηθεί παρατίθεται παρακάτω:

Πρόσβαση σε δεδομένα της εφαρμογής έχουν όλα τα εμπλεκόμενα μέλη – χρήστες:

* Διαχειριστές – Προγραμματιστές: Απόλυτη πρόσβαση στα δεδομένα της εφαρμογής. Έχουν τη δυνατότητα, είτε απευθείας μέσα από τη βάση δεδομένων είτε μέσω των διαχειριστικών routes του API να τροποποιήσουν ή να διαγράψουν τις εγγραφές της βάσης δεδομένων
* Λειτουργοί αυτοκινητοδρόμων: Δεν έχουν άμεση πρόσβαση στα δεδομένα της εφαρμογής καθώς αυτά εμπλέκουν και άλλες εταιρείες και δεδομένα οχημάτων που δεν σχετίζονται με την εταιρεία. Όμως έχουν πρόσβαση σε δεδομένα οφειλών και στατιστικών δεδομένων που τους αφορούν
* Υπουργείο Μεταφορών: Αντίστοιχα με τους λειτουργούς, δεν μπορούν να τροποποιήσουν με κάποιον τρόπο τα δεδομένα του λογισμικού, παρά μόνο να δουν στατιστικά που σχετίζονται με τη γενικότερη εικόνα του ενδοχώριου οδικού δικτύου
* Τράπεζες και λοιποί Πάροχοι Υπηρεσιών Πληρωμών: Η αρμοδιότητα τους είναι να εκπληρώνουν συναλλαγές που υπολογίζονται στο υπολογιστικό μας σύστημα. Συνεπώς έχουν πρόσβαση στις οφειλές μεταξύ των εταιρειών. Οι τράπεζες έχουν πρόσβαση και σε δεδομένα των εταιρειών που ξεφεύγουν από τις ανάγκες της εφαρμογής μας όπως τη ρευστότητα της κάθε εταιρείας και είναι υπεύθυνη, αν η συναλλαγή που πρέπει να εκπληρώσει ακυρωθεί για οποιονδήποτε λόγο, να ενημερώσει τους λειτουργούς και την εφαρμογή μας

## 3.5 Λοιπές απαιτήσεις

### 3.5.1 Απαιτήσεις διαθεσιμότητας λογισμικού

Η εφαρμογή που επιχειρούμε να αναπτύξουμε, δεδομένης της συνεχόμενης λειτουργίας των αυτοκινητοδρόμων και της συμφόρησης που υπάρχει σε περιοχές του οδικού δικτύου της χώρας, είναι κρίσιμο να μπορεί να ανταποκριθεί άμεσα στα αιτήματα των χρηστών της και να διαθέτει επαρκείς υπολογιστικούς πόρους (μνήμη αποθήκευσης δεδομένων, επεξεργαστική ισχύ). Συνεπώς, είναι λογικό να απαιτήσουμε η εφαρμογή να είναι διαθέσιμη ανά πάσα στιγμή, όσο αυτό επιτρέπεται, ώστε να μπορούμε με σύγχρονο τρόπο να χειριστούμε γεγονότα διελεύσεων και να μη χρειάζεται να ανανεώνουμε τη βάση δεδομένων μας κάθε φορά που το σύστημα είναι εκτός σύνδεσης. Σημαντικό είναι επίσης η διάρκεια που το σύστημα θα είναι offline, λόγω συντήρησης ή έλλειψης δικτύου, να είναι όσο το δυνατόν μικρότερη ώστε να μην χρειάζεται να γίνουν μεγάλες επιδιορθώσεις στα αποθηκευμένα δεδομένα.

### 3.5.2 Απαιτήσεις ασφάλειας

Τα γεγονότα διελεύσεων ανά πομποδέκτη και οι διακανονισμοί των οφειλών μεταξύ των εταιρειών είναι δεδομένα τα οποία πρέπει να εκτίθενται μόνο σε εμπλεκόμενα μέλη που τα αφορούν (εταιρείες μπορούν να ελέγξουν τις οφειλές τους με οποιαδήποτε άλλη εταιρεία) και να αποκρύπτονται από τους υπόλοιπους χρήστες. Επίσης, είναι απαραίτητη η χρήση ασφαλούς συστήματος διαχείρισης χρηστών που χρησιμοποιεί κρυπτογράφηση για την αποθήκευση κωδικών πρόσβασης και η ανταλλαγή δεδομένων να γίνεται μέσω ασφαλούς σύνδεσης SSL.