Έγγραφο απαιτήσεων λογισμικού (SRS)

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

**<IntelliQ>**

# Εισαγωγή

## 1.1 Εισαγωγή: σκοπός του λογισμικού

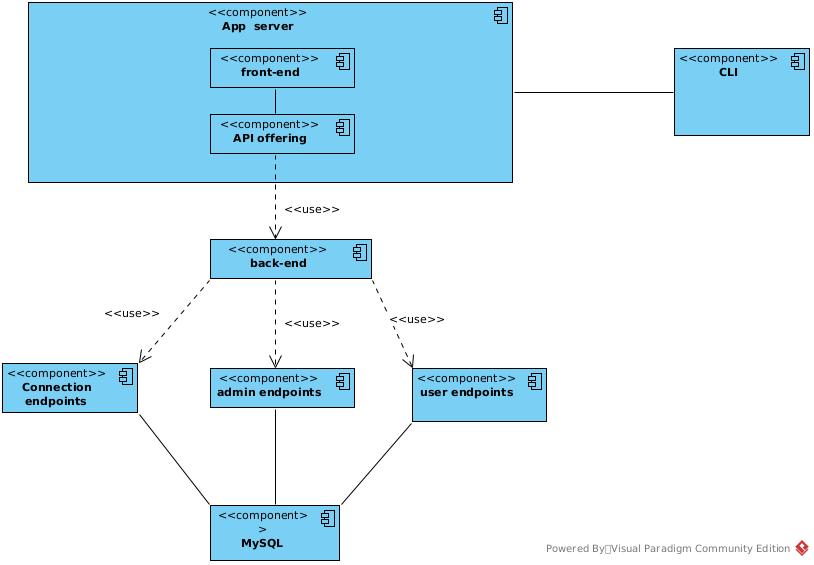
Αντικείμενο αυτής της εφαρμογής είναι η διαχείριση "έξυπνων ερωτηματολογίων" για την πραγματοποίηση online ερευνών (surveys) κάθε είδους. Ως "έξυπνο", χαρακτηρίζεται ένα ερωτηματολόγιο όταν η κάθε επόμενη ερώτηση και οι απαντήσεις της δύναται να καθορίζεται από την απάντηση της προηγούμενης.

## 1.2 Διεπαφές (interfaces)

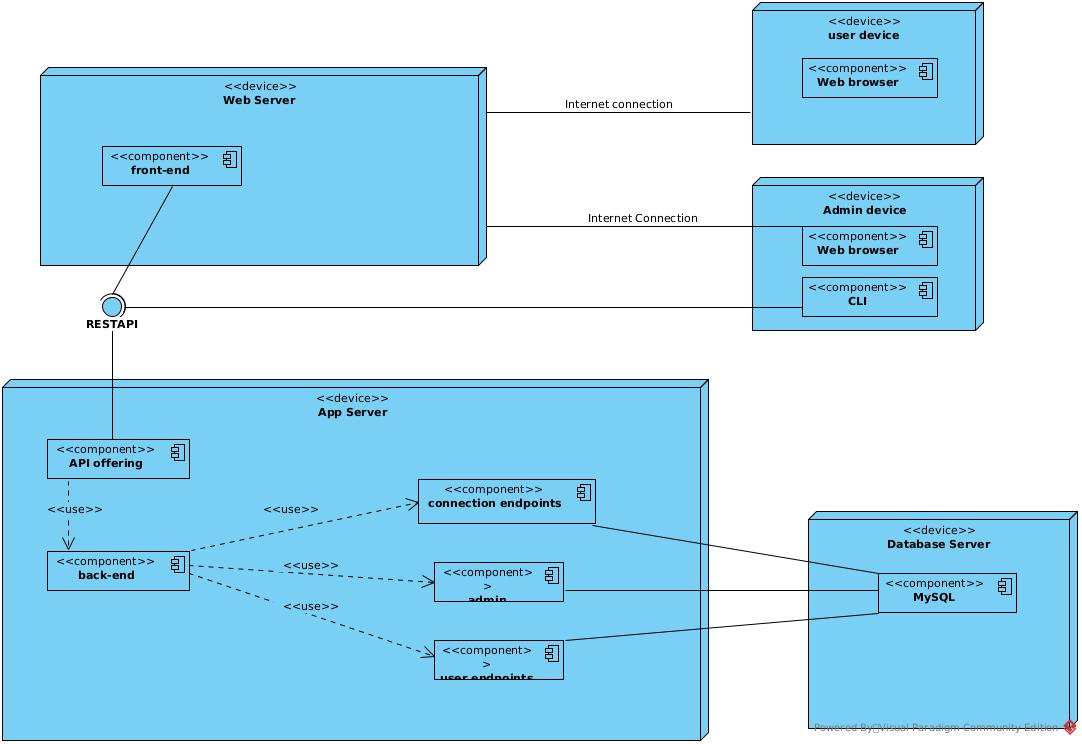
### *1.2.1 Διεπαφές με εξωτερικά συστήματα*

* MySQL server : Είναι το σύστημα διαχείρισης της βάσης δεδομένων που χρησιμοποιούμε για την εφαρμογή.
* React Web Server (επικοινωνία web app με front-end) .
* REST API’s (επικοινωνία front-end με back-end).

Παραθέτουμε το UML Component Diagram:



Και το Deployment Diagram:



### *1.2.2 Διεπαφές με το χρήστη*

* Frond-End Application : Προσφέρει δυνατότητες επιλογής και απάντησης ερωτηματολογίων.
* Βack-end : το οποίο θα υποστηρίζει λειτουργίες διαχείρισης της δομής των ερωτηματολογίων, λήψης, καταχώρησης και εξαγωγής απαντήσεων.
* CLI (Command Line Interface) : Λειτουργεί ως client του REST API που παρέχεται από το back-end υποσύστημα, προσφέροντας στο χρήστη της τη δυνατότητα να εκτελεί λειτουργίες μέσω ενός Shell .

# Αναφορές - πηγές πληροφοριών

Για την κατανόηση του συστήματος, διαβάστε το API doc και τα README στο github repo μας.

# 3. Προδιαγραφές απαιτήσεων λογισμικού

## 3.1 Περιπτώσεις χρήσης

Diagram

Description automatically generated Παραθέτουμε το use case diagram :

3.1.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 1: Δημιουργία Ερωτηματολογίου από τον διαχειριστή.

#### 3.1.1.1 Χρήστες (ρόλοι) που εμπλέκονται

* Ο διαχειριστής .

#### 3.1.1.2 Προϋποθέσεις εκτέλεσης

* Ο διαχειριστής να κάνει successful log in.
* Να έχει καταχωρήσει ένα έγκυρο .json αρχείο που πληρεί της προδιαγραφές που θέτει ο δημιουργός της εφαρμογής.

#### 3.1.1.3 Περιβάλλον εκτέλεσης

* CLI (Command Line Interface)
* Frontend
* Database

#### 3.1.1.4 Δεδομένα εισόδου

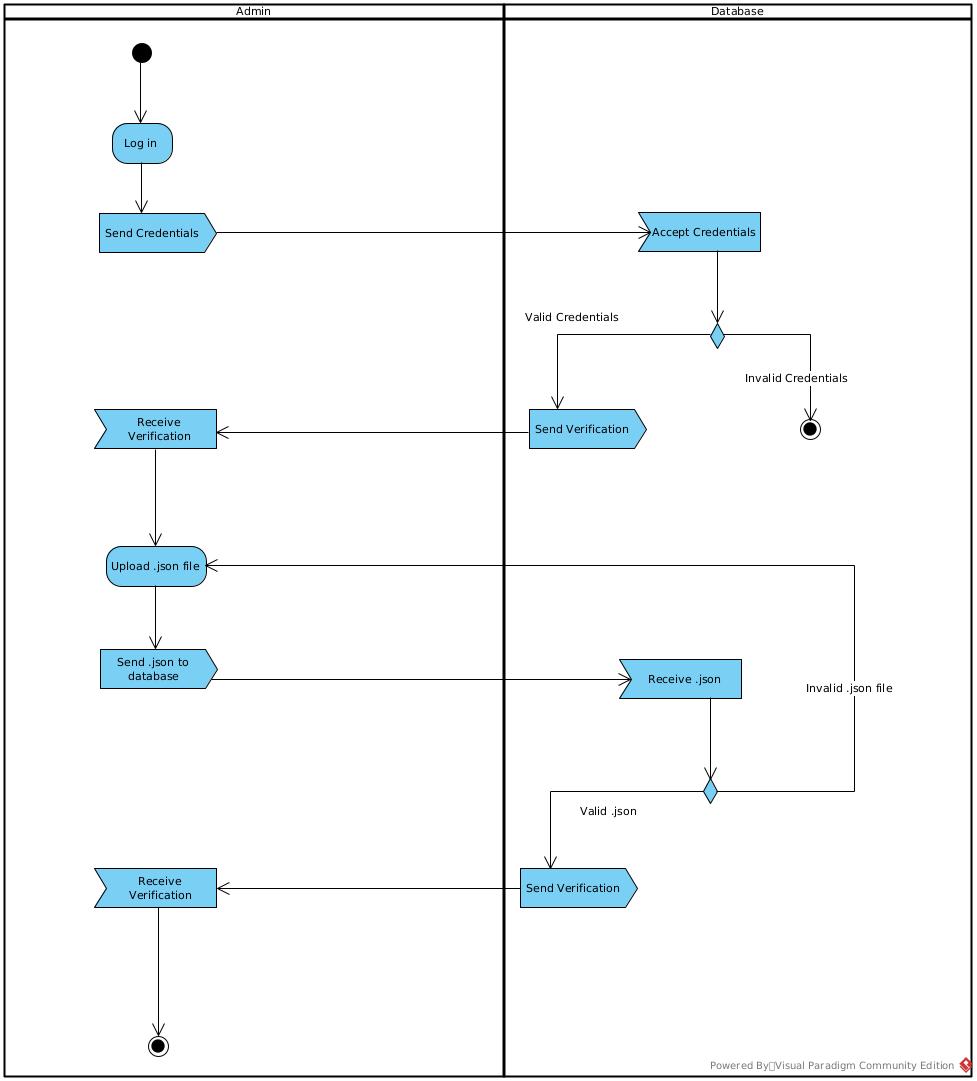
#### Ταυτοποίηση admin . Επιβεβαίωση των στοιχείων του με εκείνα που είναι καταχωρημένα στην βάση δεδομένων.

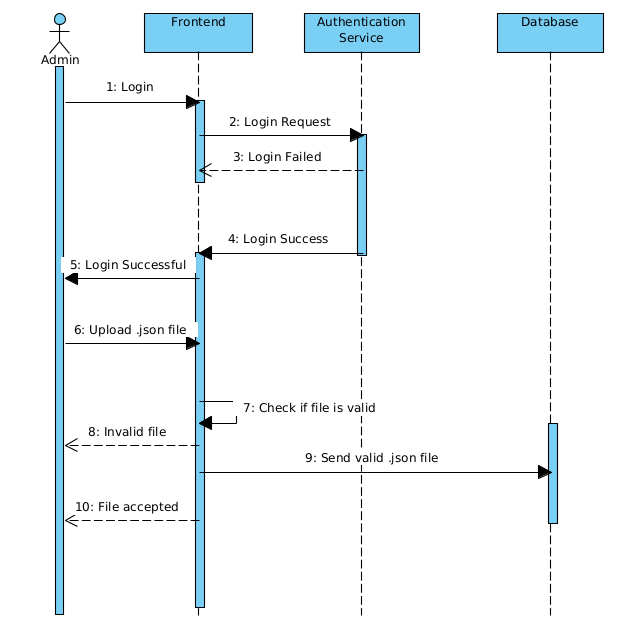
* Έγκυρο αρχείο .json (αν το αρχείο δεν πληροί τις προδιαγραφές δεν μπορεί να δημιουργηθεί το ερωτηματολόγιο).

#### 3.1.1.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

* Βήμα 1 : Ο admin κάνει log in.Τα στοιχεία του ελέγχονται και διασταυρώνονται με εκείνα που είναι αποθηκευμένα στην βάση δεδομένων.
* Βήμα 2 : Αν το log in επιτύχει ο admin μπορεί να ανεβάζει το .json αρχείο που περιέχει το ερωτηματολόγιο που θέλει να δημιουργήσει.
* Βήμα 3 : Γίνεται έλεγχος του .json αρχείου. Aν είναι επιτυχής, δίνεται μήνυμα επιβεβαίωσης της δημιουργίας του ερωτηματολογίου.

Παρακάτω παρατίθενται το activity και το sequence diagram που αντιστοιχεί σε αυτή την περίπτωση χρήσης. (\*Tο διάγραμμα φαίνεται καλύτερα στο αρχείο .vpp).





#### 3.1.1.6 Δεδομένα εξόδου

* Εμφανίζει μήνυμα επιβεβαίωσης στέλνοτας απόκριση HTTP “200 OK” σε περίπτωση επιτυχίας, αλλιώς εμφανίζει μήνυμα λάθους Bad Request.

### 3.1.2 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 2 : Συμπλήρωση ερωτηματολογίων.

#### 3.1.2.1 Χρήστες (ρόλοι) που εμπλέκονται

* Ο χρήστης που απαντά το ερωτηματολόγιο.
* Ο χρήστης μπορεί να είναι οποιοσδήποτε ανώνυμος ή επώνυμος .

#### 3.1.2.2 Προϋποθέσεις εκτέλεσης

* Ο χρήστης να έχει κάνει επιτυχημένο login (ακόμα και ο ανώνυμος χρήστης κάνει login αυτοματοποιημένα από το σύστημα).

#### 3.1.2.3 Περιβάλλον εκτέλεσης

* Front-end web application : Το γραφικό περιβάλλον που γίνεται η απάντηση του ερωτηματολογίου.
* Database : Αποθήκευση απαντήσεων.

#### Δεδομένα εισόδου

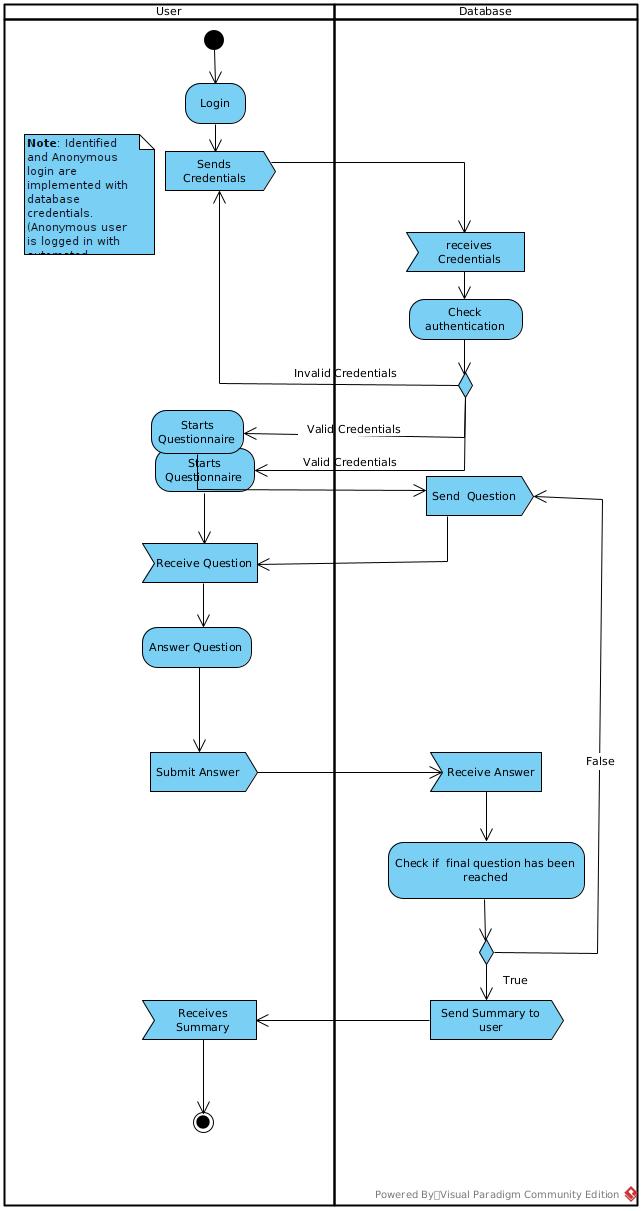
#### Ταυτοποίηση χρήστη.

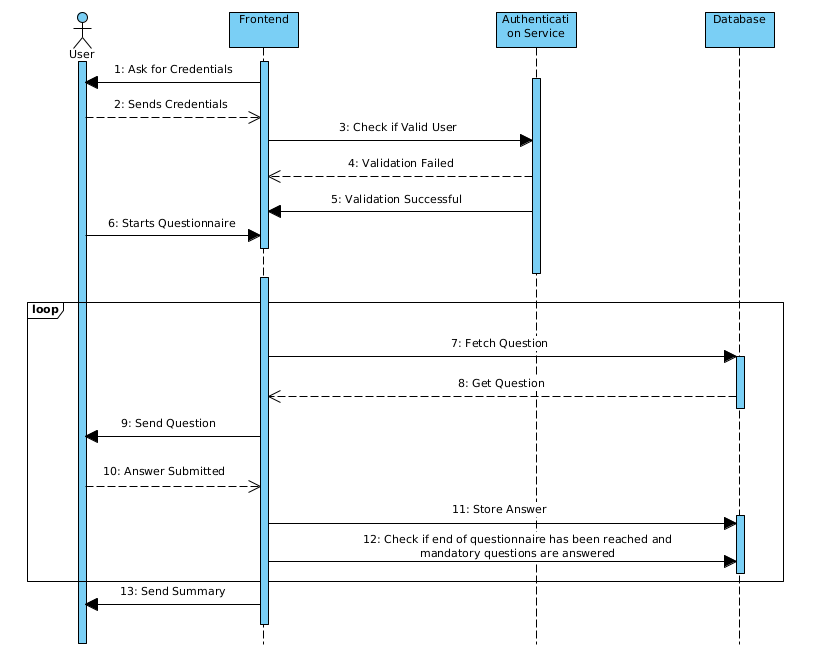
* Απαντήσεις χρήστη στις ερωτήσεις του ερωτηματολογίου.

#### 3.1.2.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

* Βήμα 1: Ο χρήστη κάνει log in και ελέγχεται η ορθότητα των στοιχείων του.
* Σε περίπτωση μη διασταύρωσης με τα στοιχεία της βάσης ζητείτε εκ νέου από τον χρήστη να κάνει log in.
* Βήμα 2: Ο χρήστης επιλέγει ποιο από τα διαθέσιμα ερωτηματολόγια θέλει να απαντήσει.
* Βήμα 3: Η πρώτη ερώτηση εμφανίζεται και ο χρήστης απαντάει. Αν η ερώτηση έχει οριστεί από τον διαχειριστή ως υποχρεωτική ο χρήστης θα πρέπει να απαντήσει για να μπορέσει το επιλέξει “Επόμενο” και η απάντηση του αποθηκεύεται στην βάση. Σε άλλη περίπτωση, μπορεί να παραδώσει το ερωτηματολόγιο (αφού η επόμενη ερώτηση εξαρτάται από την απάντηση της προηγούμενης, κάναμε την παραδοχή ότι πρώτες εμφανίζονται οι υποχρεωτικές ερωτήσεις και όταν αυτές τελειώσουν εμφανίζονται οι προαιρετικές ερωτήσεις και ο χρήστης μπορεί να παραδώσει το ερωτηματολόγιο απαντώντας σε μερικές από αυτές ή όχι).
* Βήμα 4: Επαναλαμβάνεται το Βήμα 3 μέχρι να τελειώσει το ερωτηματολόγιο.
* Βήμα 5: Το ερωτηματολόγιο τελειώνει και εμφανίζεται η σύνοψη των απαντήσεων στον χρήστη.

Παρακάτω παρατίθενται το activity και το sequence diagram που αντιστοιχεί σε αυτή την περίπτωση χρήσης. (\*Tο διάγραμμα φαίνεται καλύτερα στο αρχείο .vpp).





#### 3.1.2.6 Δεδομένα εξόδου

* Σύνοψη των απαντήσεων στον χρήστη.
* Οι απαντήσεις αποθηκεύονται στην βάση.

### 3.1.2 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 3: Εξαγωγή απαντήσεων και στατιστικών δεδομένων

#### 3.1.3.1 Χρήστες (ρόλοι) που εμπλέκονται

Ο χρήστης που έχει πρόσβαση στα δεδομένα, το ιστορικό και την εξαγωγή τους από την βάση δεδομένων, ο administrator.

#### 3.1.3.2 Προϋποθέσεις εκτέλεσης

Επιτυχές log in από την πλευρά του administrator. Πρόσβαση στην βάση δεδομένων.

#### 3.1.3.3 Περιβάλλον εκτέλεσης

DMBS για την εξαγωγή δεδομένων και back end app server για την εκτέλεση των queries και την δημιουργία στατιστικών στοιχείων.

#### 3.1.3.4 Δεδομένα εισόδου

Αρχικά είναι τα στοιχεία εισόδου του admin. Στην συνέχεια, επιλογή ανάμεσα στα 3 διαθέσιμα actions:

Extract diagram, View History, ή Statistical Analysis.

#### 3.1.1.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Βήμα 1: Log in του admin και έλεγχος ορθότητας των στοιχείων.

Βήμα 2: Ο Admin επιλέγει ένα ερωτηματολόγιο για να ακολουθήσει κάποιο από τα επόμενα actions.

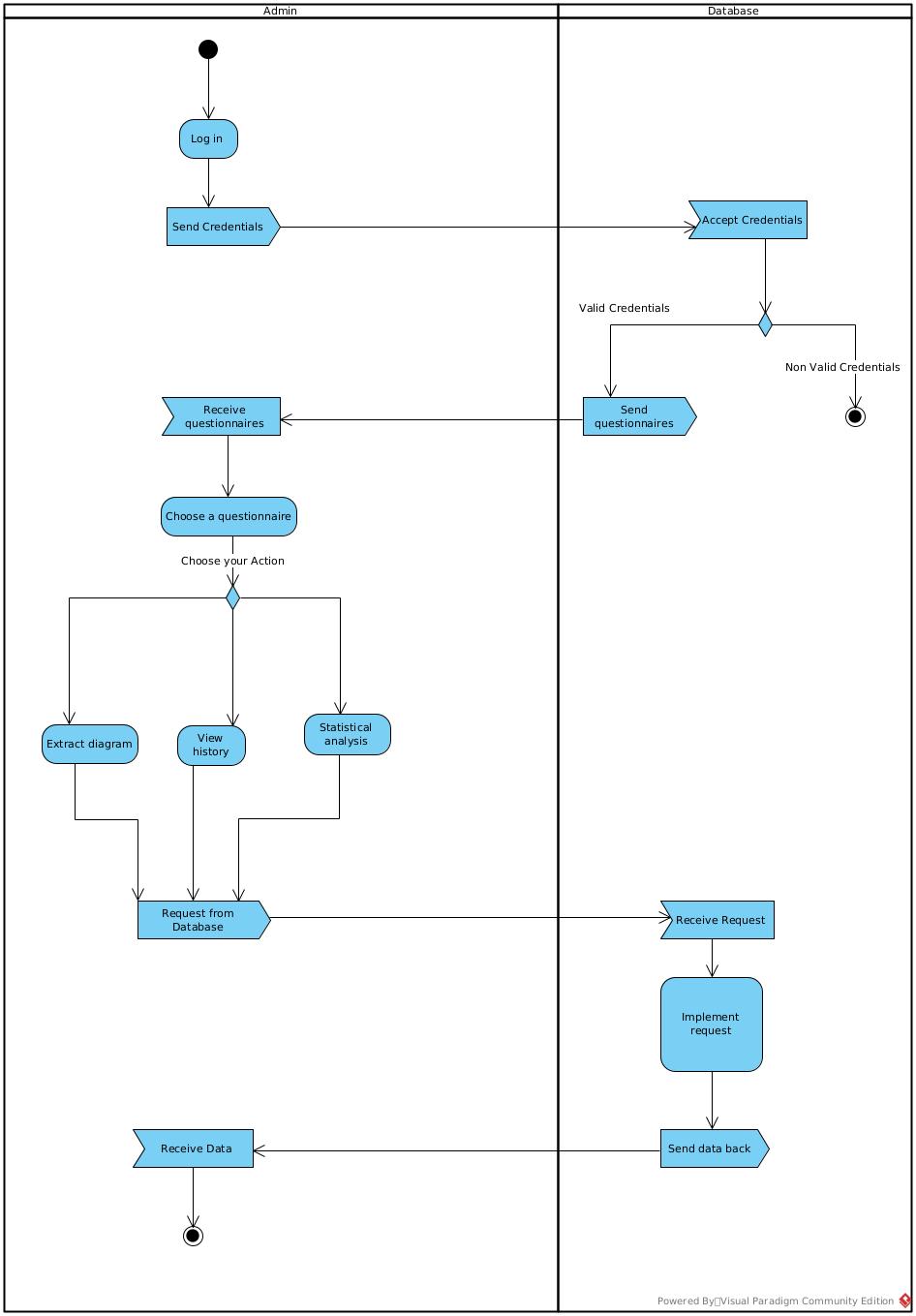
Βήμα 3: Επιλογή του επιθυμητού action (Extract diagram, View History, ή Statistical Analysis.)

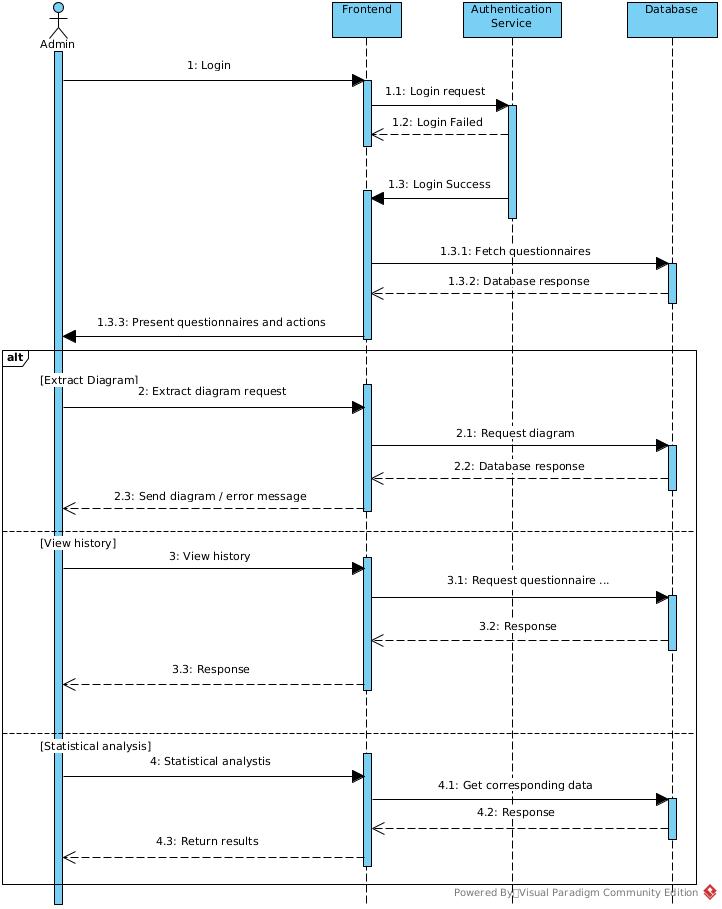
Βήμα 3.1: Επιλογή action: Extract diagram, ζητούνται τα data από την Database και ο admin λαμβάνει ανάλογη απάντηση. (γράφημα ή μήνυμα λάθους)

Βήμα 3.2: Επιλογή action: View History, ζητούνται από την Database το Ιστορικό των απαντήσεων σε μορφή .json ή csv.

Βήμα 3.3: Επιλογή action: Statistical Analysis, επιστροφή στατιστικής ανάλυσης των απαντήσεων του ερωτηματολογίου.

Παρακάτω παρατίθενται το activity και το sequence diagram που αντιστοιχεί σε αυτή την περίπτωση χρήσης. (\*Tο διάγραμμα φαίνεται καλύτερα στο αρχείο .vpp).





#### 3.1.3.6 Δεδομένα εξόδου

* Για το πρώτο Action επιστρέφεται διάγραμμα ανάλογα με τις απαντήσεις του ερωτηματολογίου.
* Για το δεύτερο Action επιστρέφεται το ιστορικό των ερωτηματολογίων.
* Για το τέταρτο Action επιστρέφει την στατιστική ανάλυση των απαντήσεων.

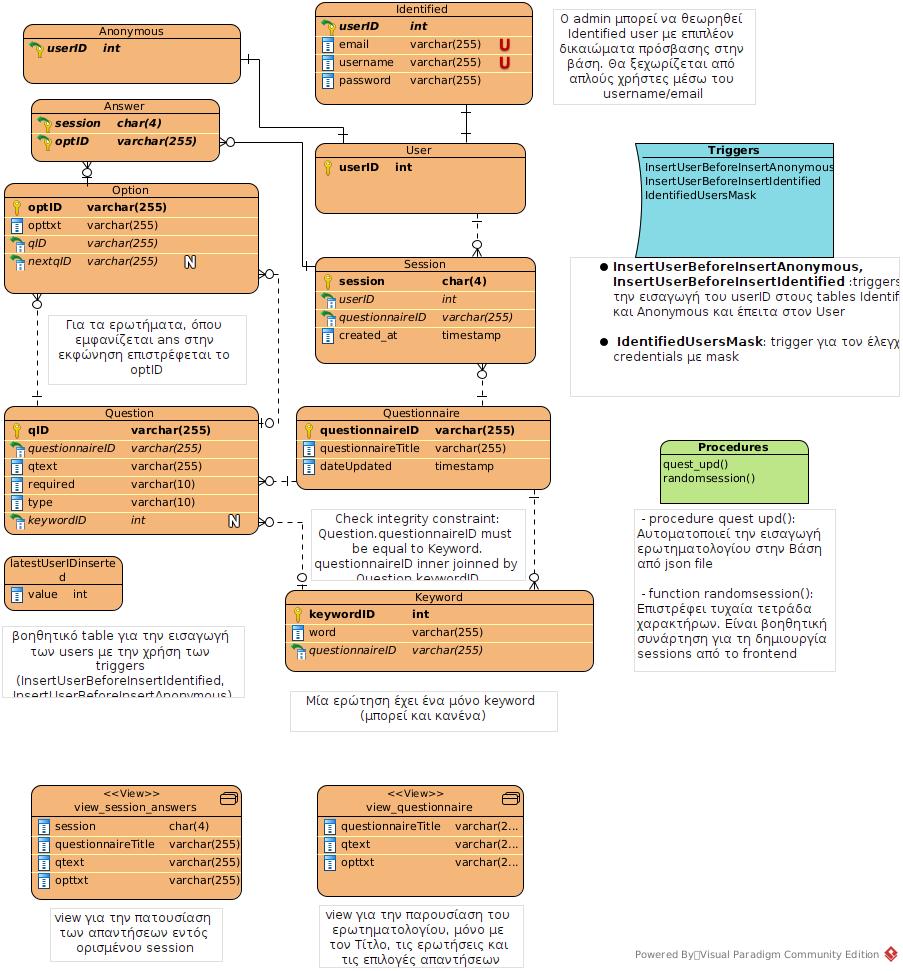
## *3.2 Απαιτήσεις επιδόσεων*

* Η υποστήριξη πολλαπλών χρηστών ταυτόχρονα. To API να ανταποκρίνεται σε ένα μεγάλο αριθμό ταυτόχρονων αιτημάτων.

## 3.3 Απαιτήσεις οργάνωσης δεδομένων

(Αντί για ER έχουμε ανεβάσει το relational)

Relational Diagram: (\*Tο διάγραμμα φαίνεται καλύτερα στο αρχείο .vpp)



Diagram

Description automatically generatedAPI/Class diagram:

### 3.3.1 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

Η εφαρμογή υποστηρίζει δύο είδη χρηστών, επώνυμους και ανώνυμους χρήστες, αφού επιτρέπεται επώνυμη και ανώνυμη απάντηση των ερωτηματολογίων. Ο κάθε admin είναι επώνυμος χρήστης με διακριτικό στο email και στο username του.

Για τους επώνουμους χρήστες επιτρέπονται μόνο emails @identified.com. Η επιλογή της συγκεκριμένης μάσκας είναι ενδεικτική , μπορούμε με τον ίδιο τρόπο να προσθέσουμε πολλά μάσκες email πχ ntua.gr κ.λπ.

Επειδή η επόμενη ερώτηση που εμφανίζεται σε κάποιον χρήστη εξαρτάται από την απάντησή του στην προηγούμενη ερώτηση αποφασίσαμε ότι σε κάθε ερωτηματολόγιο οι υποχρεωτικές ερωτήσεις εμφανίζονται πρώτες. Αφού ο χρήστης απαντήσει σε αυτές μπορεί να παραδώσει το ερωτηματολόγιο ή να συνεχίσει να απαντάει τις όποιες προαιρετικές ερωτήσεις υπάρχουν.

Στην Βάση όσον αφορά τα ερωτηματολόγια αποθηκεύονται ο τίτλος τους, και η πιο πρόσφατη ημερομηνία επεξεργασίας τους, οι ερωτήσεις του καθενός ερωτηματολογίου μαζί με το χαρακτηριστικό required/optional και τον τύπο της και τα keywords του ερωτηματολογίου.

Κάθε ερωτηματολόγιο μπορεί να έχει πολλά keywords, ενώ κάθε ερώτηση μπορεί να έχει το πολύ ένα keyword.

Κάθε ερώτηση μπορεί να έχει πολλαπλές επιτρεπόμενες απαντήσεις (options) και η επόμενη ερώτηση που θα εμφανιστεί στον χρήστη εξαρτάται με την απάντηση που έχει δώσει στην τρέχουσα ερώτηση.

Κάθε φορά που κάποιος χρήστης ολοκληρώνει την απάντηση κάποιου ερωτηματολογίου, δημιουργείται ένα νέο entry στην σχέση Session, που συνδέει τον χρήστη με το ερωτηματολόγιο που απάντησε.

Στο κάθε Session αποθηκεύονται οι επιλογές/απαντήσεις του χρήστη μέσω της σχέσης Answer, ώστε να γνωρίζουμε την απάντηση του χρήστη σε κάθε ερώτηση.

## *3.5 Λοιπές απαιτήσεις*

### 3.5.1 Απαιτήσεις διαθεσιμότητας λογισμικού

* Θα πρέπει η εφαρμογή να είναι προσβάσιμη από όλους τους περιηγητές ιστού.
* Να υπάρχουν αντίγραφα ασφαλείας με δυνατότητα ανάκτησης δεδομένων σε περίπτωση δυσλειτουργίας του συστήματος.

### 3.5.2 Απαιτήσεις ασφάλειας

* Η χρήση HTTPS είναι απαραίτητη για την ασφάλεια των αιτημάτων στο διαδίκτυο.
* Προσωπικά δεδομένα όπως το όνομα, mail και κυρίως το password του user δεν πρέπει να εμφανίζονται στον διαχειριστή.