

Έγγραφο απαιτήσεων λογισμικού (SRS)

ΤΙΤΛΟΣ ΕΦΑΡΜΟΓΗΣ ΛΟΓΙΣΜΙΚΟΥ: intelliQ

1. Εισαγωγή

1.1 Εισαγωγή: σκοπός του λογισμικού

Ο κύριος άξονας του λογισμικού είναι η διαμόρφωση και απάντηση ερωτηματολογίων. Ένα βασικό χαρακτηριστικό του είναι η δυνατότητα σχεδίασης “έξυπνων” ερωτηματολογίων, στα οποία η κάθε απάντηση του χρήστη θα καθορίζει την επόμενη ερώτηση που θα του τεθεί.

Ο απλός και αποτελεσματικός τρόπος συμπλήρωσης ερωτηματολογίων, σε συνδυασμό και με την ανωνυμία της όλης διαδικασίας, προσφέρει τη βέλτιστη εμπειρία στο χρήστη.

Η διαχείριση των ερωτηματολογίων και η επεξεργασία των συλλεγόμενων δεδομένων θα γίνεται εύκολα από τον admin, μέσω ενός *command-line interface*. Θα υπάρχει και η επιπλέον δυνατότητα για εξαγωγή δεδομένων σε μορφή csv ή json.

Η χρησιμότητα του λογισμικού άπτεται τόσο στο κομμάτι της διεξαγωγής ανώνυμων ερευνών, όσο και στη σχεδίαση ερωτηματολογίων για προσωπικό σκοπό, που απευθύνονται σε μικρό και συγκεκριμένο δείγμα.

1.2 Διεπαφές (interfaces)

1.2.1 Διεπαφές με εξωτερικά συστήματα

Το σύστημα στην πλήρη λειτουργία του θα χρησιμοποιεί τις εξής διεπαφές:

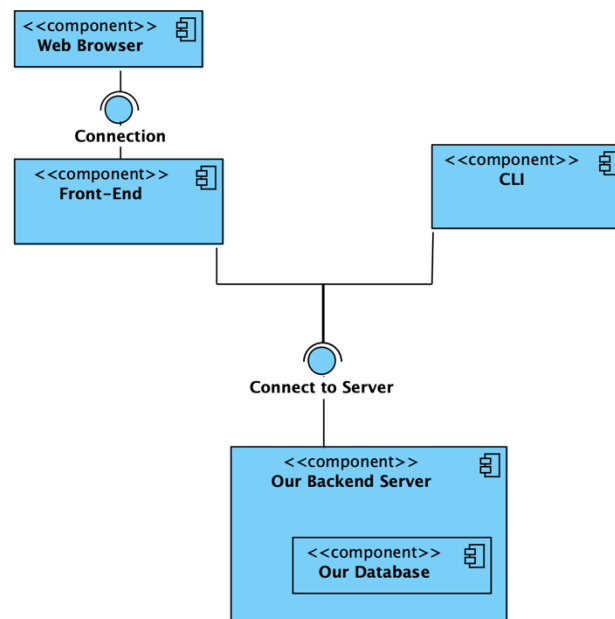
Εξωτερικά:

- Web Browser του χρήστη (Επικοινωνία με την υπηρεσία μας)
- CLI

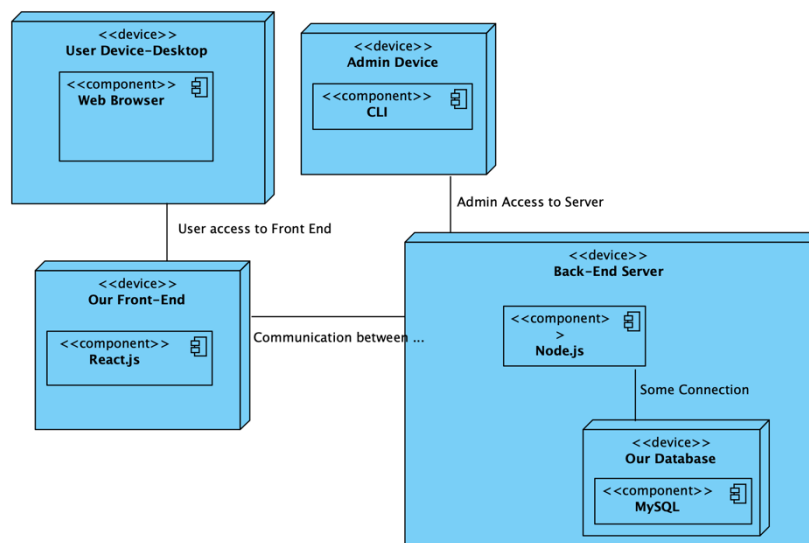
Εσωτερικά:

- React Web Server (για την επικοινωνία *client's browser* με *front-end*)
- Node.js & Rest API's (για την επικοινωνία *front-end* με *back-end* καθώς και *CLI* με *back-end*)
- MySQL (για την επικοινωνία *back-end* με *database*)

Ακολουθεί το **UML Component Diagram** :



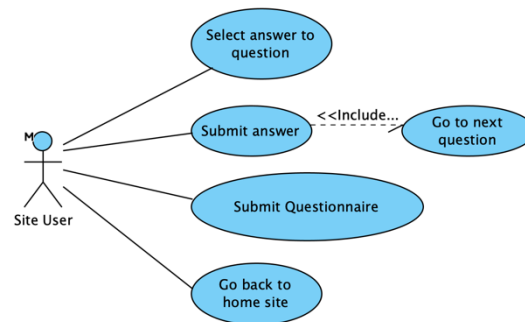
Ακολουθεί το **UML Deployment Diagram** :



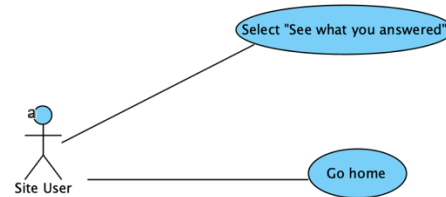
1.2.2 Διεπαφές με το χρήστη

Ακολουθούν τα **UML Use Case Diagrams**:

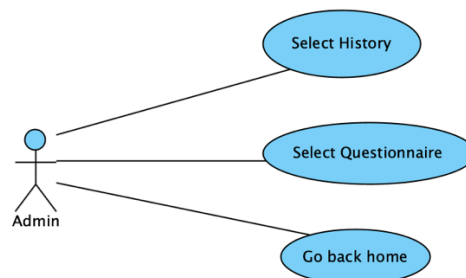
Answer Question:



View summary of the answers:



View history of answers:



2. Αναφορές - πηγές πληροφοριών

N/A

3. Προδιαγραφές απαιτήσεων λογισμικού

3.1 Περιπτώσεις χρήσης

3.1.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 1: Απάντηση Ερωτηματολογίου

3.1.1.1 Χρήστες (ρόλοι) που εμπλέκονται

- Χρήστες που συμμετέχουν στις έρευνες απαντώντας τα ερωτηματολόγια (User)

3.1.1.2 Προϋποθέσεις εκτέλεσης

- Δεν είναι απαραίτητο το login του χρήστη

3.1.1.3 Περιβάλλον εκτέλεσης

- Διεπαφή χρήστη του web based front-end
- Back-end server
- Database server

3.1.1.4 Δεδομένα εισόδου

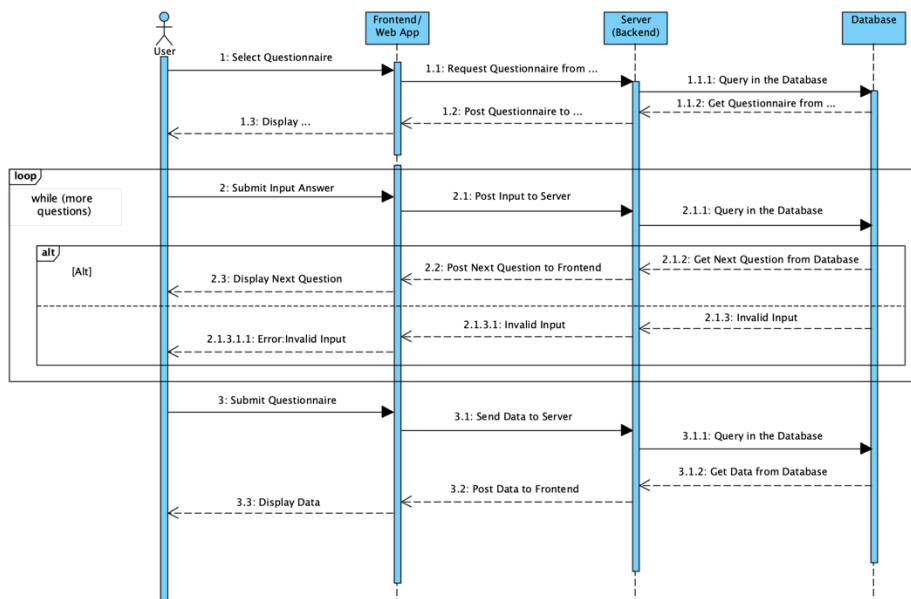
Απαντήσεις χρηστών

3.1.1.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

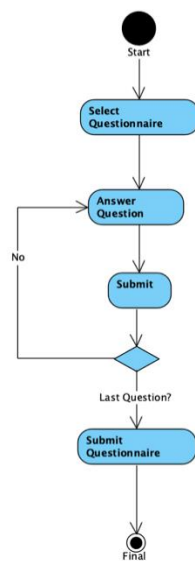
Περιγραφή με κείμενο (Βήμα 1, Βήμα 2 κλπ) και διαγράμματα UML αλληλουχίας (Sequence) και δραστηριοτήτων (Activity). Περιλαμβάνεται η συμπεριφορά σε απρόβλεπτες καταστάσεις και σφάλματα (εναλλακτικές ροές).

1. Είσοδος στο σύστημα
2. Επιλογή ερωτηματολογίου προς απάντηση
3. Απάντηση και υποβολή για κάθε ερώτηση
4. Ολοκλήρωση μετά και την υποβολή της απάντησης στην τελευταία ερώτηση

UML Sequence Diagram:



UML Activity Diagram:



3.1.1.6 Δεδομένα εξόδου

Διαγράμματα UML αλληλουχίας για την παραγωγή δεδομένων εξόδου. Ως δεδομένα εξόδου νοούνται όλα τα δεδομένα του συστήματος τα οποία δημιουργούνται ή μεταβάλλονται κατά την εκτέλεση (αν υπάρχουν τέτοια)

Το σύστημα επιστρέφει τα παρακάτω HTTP status codes ανάλογα την κλήση.

1. Επιτυχής υποβολή κάθε απάντησης (200- Success)
2. Λανθασμένοι παράμετροι (400- Bad request)
3. Γενικό σφάλμα (500- Server Error)
4. Κενή απάντηση (402- No data)

3.1.2 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 2: Προβολή σύνοψης των απαντήσεων

3.1.2.1 Χρήστες (ρόλοι) που εμπλέκονται

- Χρήστης που απαντάει το ερωτηματολόγιο

3.1.2.2 Προϋποθέσεις εκτέλεσης

- Επιτυχής ολοκλήρωση του ερωτηματολογίου

3.1.2.3 Περιβάλλον εκτέλεσης

- Διεπαφή χρήστη του web based front-end
- Back-end server
- Database server

3.1.2.4 Δεδομένα εισόδου

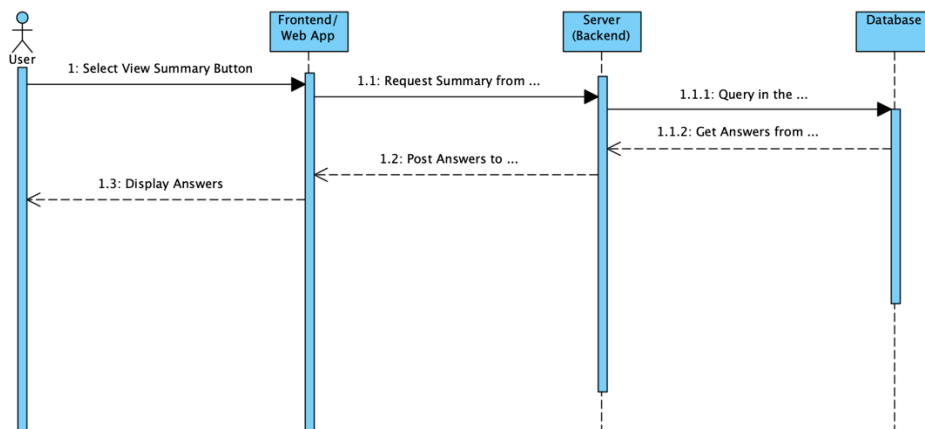
N/A

3.1.2.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

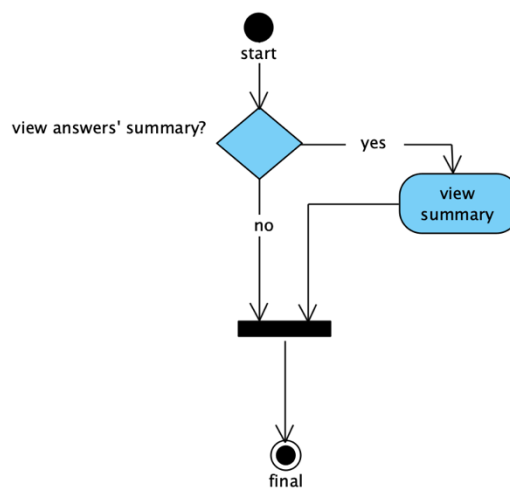
Περιγραφή με κείμενο (Βήμα 1, Βήμα 2 κλπ) και διαγράμματα UML αλληλουχίας (Sequence) και δραστηριοτήτων (Activity). Περιλαμβάνεται η συμπεριφορά σε απρόβλεπτες καταστάσεις και σφάλματα (εναλλακτικές ροές).

1. Είσοδος στο σύστημα
2. Συμπλήρωση του ερωτηματολογίου
3. Επιλογή της δυνατότητας σύνοψης των απαντήσεων που δόθηκαν

UML Sequence Diagram:



UML Activity Diagram:



3.1.2.6 Δεδομένα εξόδου

Το σύστημα επιστρέφει 200 σε επιτυχής κλήση ή error 500 σε περίπτωση γενικού σφάλματος (Internet server error)

3.1.3 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 3: Δυνατότητα θέασης ιστορικού από τον διαχειριστή

3.1.3.1 Χρήστες (ρόλοι) που εμπλέκονται

- Διαχειριστής του ερωτηματολογίου

3.1.3.2 Προϋποθέσεις εκτέλεσης

- Ύπαρξη απαντημένου ερωτηματολογίου

3.1.3.3 Περιβάλλον εκτέλεσης

- Διεπαφή χρήστη του web based front-end
- Back-end server
- Database server

3.1.3.4 Δεδομένα εισόδου

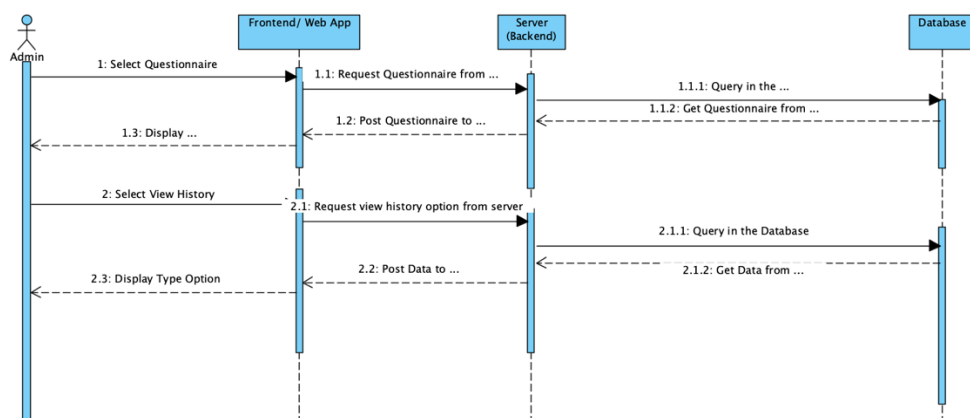
N/A

3.1.3.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

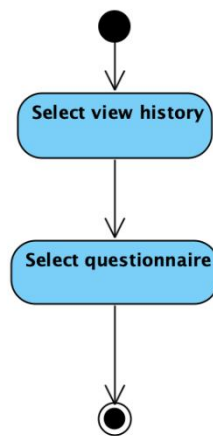
Περιγραφή με κείμενο (Βήμα 1, Βήμα 2 κλπ) και διαγράμματα UML αλληλουχίας (Sequence) και δραστηριοτήτων (Activity). Περιλαμβάνεται η συμπεριφορά σε απρόβλεπτες καταστάσεις και σφάλματα (εναλλακτικές ροές).

1. Είσοδος στο σύστημα με διαπίστευση διαχειριστή (ID)
2. Επιλογή επιθυμητού ερωτηματολογίου
3. Επιλογή ερώτησης για προβολή ιστορικού απαντήσεων

UML Sequence Diagram:



UML Activity Diagram:



3.1.3.6 Δεδομένα εξόδου

Διαγράμματα UML αλληλουχίας για την παραγωγή δεδομένων εξόδου. Ως δεδομένα εξόδου νοούνται όλα τα δεδομένα του συστήματος τα οποία δημιουργούνται ή μεταβάλλονται κατά την εκτέλεση (αν υπάρχουν τέτοια)

3.2 Απαιτήσεις επιδόσεων

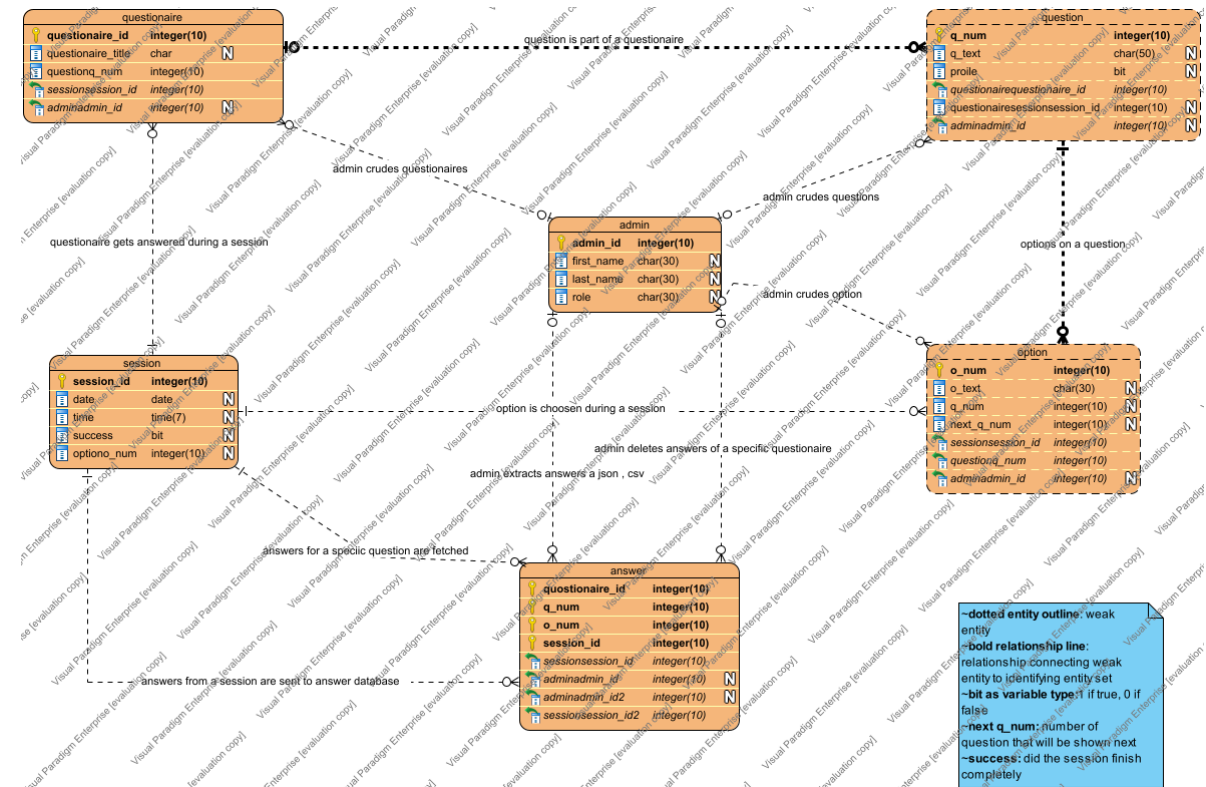
Ποσοτική τεκμηρίωση μέτρων και κριτηρίων επιθυμητών επιδόσεων με αναφορά στα ποσοτικά χαρακτηριστικά εισόδων και φορτίου του λογισμικού.

3.3 Απαιτήσεις οργάνωσης δεδομένων

Ερωτηματολόγια μπορούν να εισαχθούν από τον Admin με χρήση json αρχείου, με την προϋπόθεση τα δεδομένα να τηρούν μια συγκεκριμένη μορφή, σύμφωνα με τα παραδείγματα json στο φάκελο backend-api/endpoints/admin του Github. Τα δεδομένα του πίνακα Session (energy στη βάση) εισάγονται αυτόματα, με κάθε επιτυχή σύνδεση χρήστη. Οι δοσμένες απαντήσεις επίσης εισάγονται αυτόματα στη βάση, κατά τη συμπλήρωση ερωτηματολογίου.

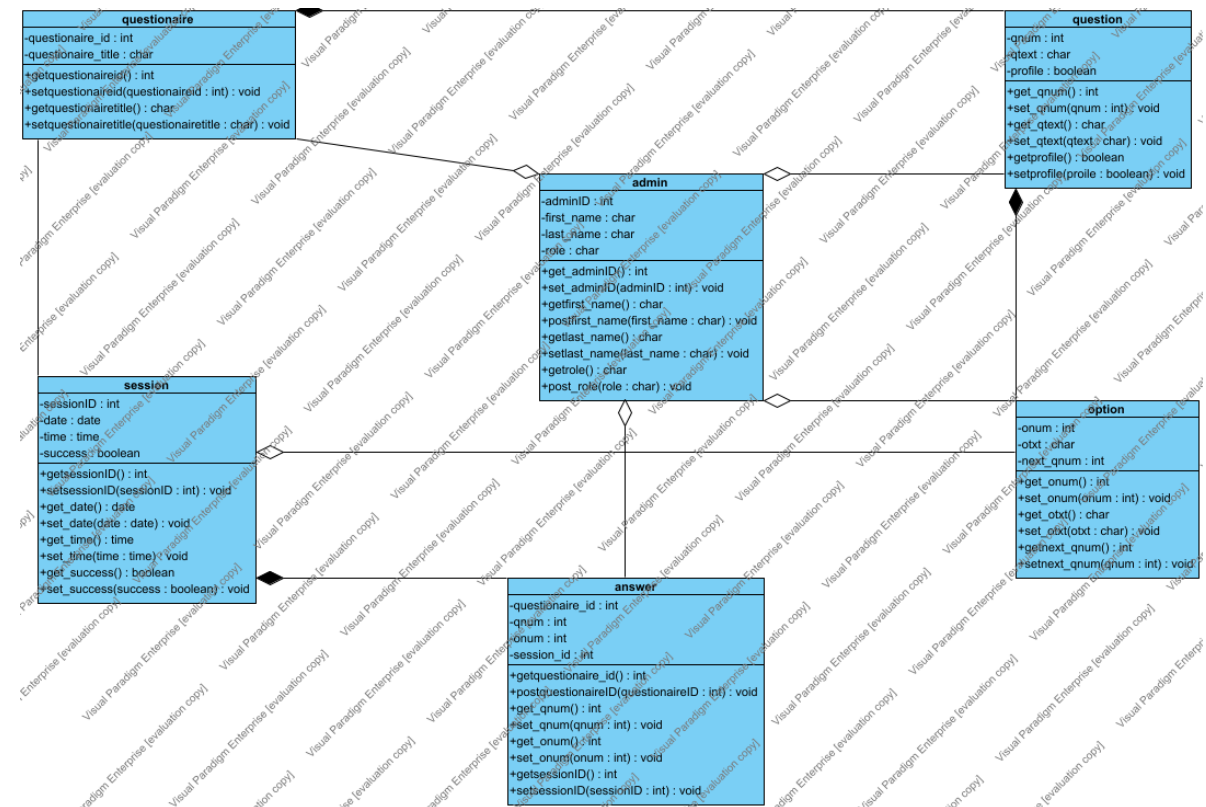
Ακολουθεί το **UML ER Diagram**:

Entity Relationship Diagram



Ακολουθεί το UML Class Diagram :

Class Diagram



3.3.1 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

Κάθε χρήστης της εφαρμογής (Admin, User) έχει πρόσβαση σε συγκεκριμένες λειτουργίες/δεδομένα ανάλογα με το ρόλο του. Δεν έχει υλοποιηθεί κάποιου είδους login που να επικυρώνει την πρόσβαση.

3.4 Περιορισμοί Σχεδίασης

Back-end: Χρήση Javascript, και συγκεκριμένα node.js, χρήση της βιβλιοθήκης Express και του εργαλείου npm.

Front-end: Χρήση Javascript και της βιβλιοθήκης React, με ενσωμάτωση κώδικα CSS.

CLI: Χρήση node.js

Άλλα: Χρήση Git και Visual Paradigm (για τη συγγραφή του SRS και τη σχεδίαση των διαγραμμάτων).

3.5 Λοιπές απαιτήσεις

3.5.1 Απαιτήσεις διαθεσιμότητας λογισμικού

Τεκμηρίωση απαιτήσεων διαθεσιμότητας

3.5.2 Απαιτήσεις ασφάλειας

Τεκμηρίωση απαιτήσεων ασφαλείας

- Υλοποίηση συστήματος login για διαπίστευση διαχειριστών.
- Χρήση HTTPS και SSL certificate.