Έγγραφο απαιτήσεων λογισμικού (SRS)

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

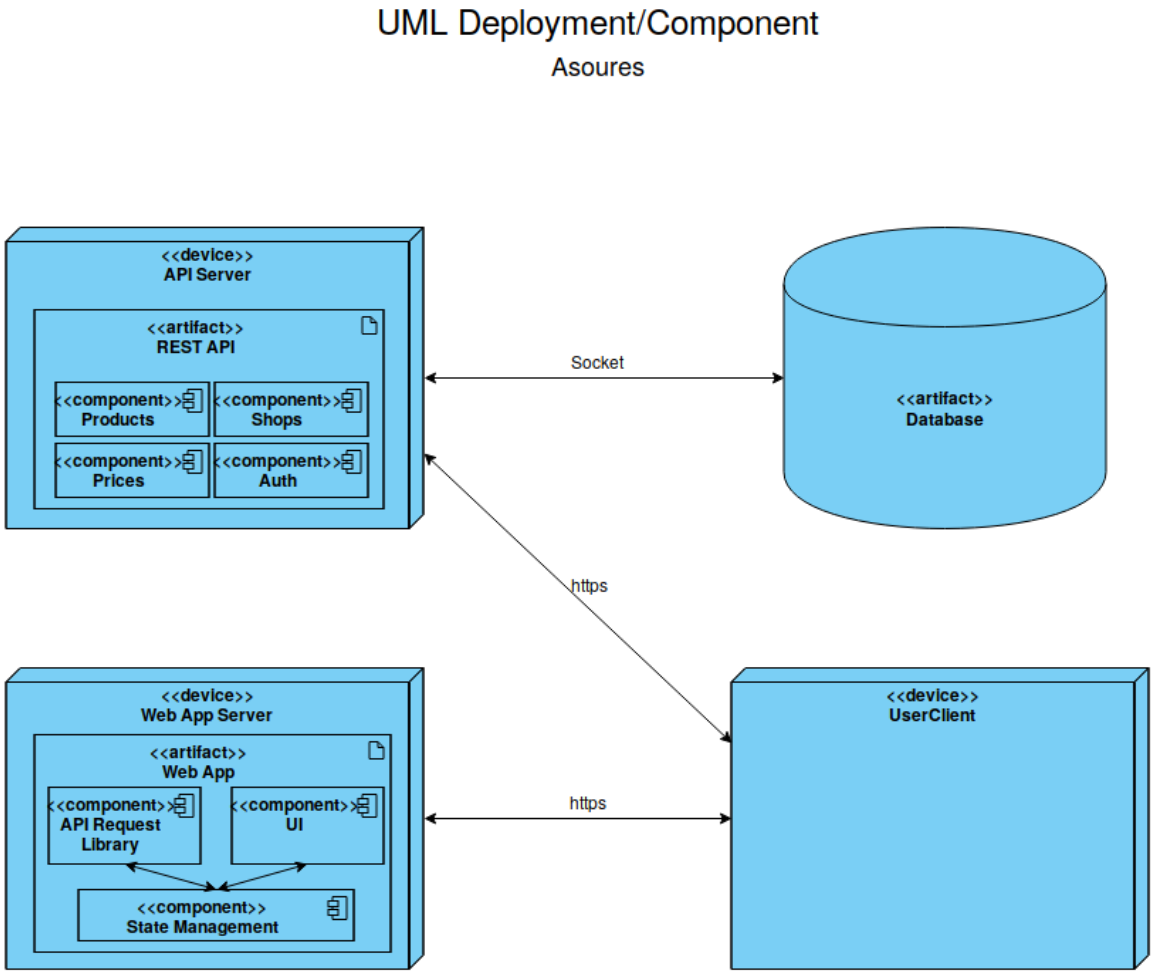
Παρατηρητήριο Τιμών Κομμωτηρίων

1. Εισαγωγή

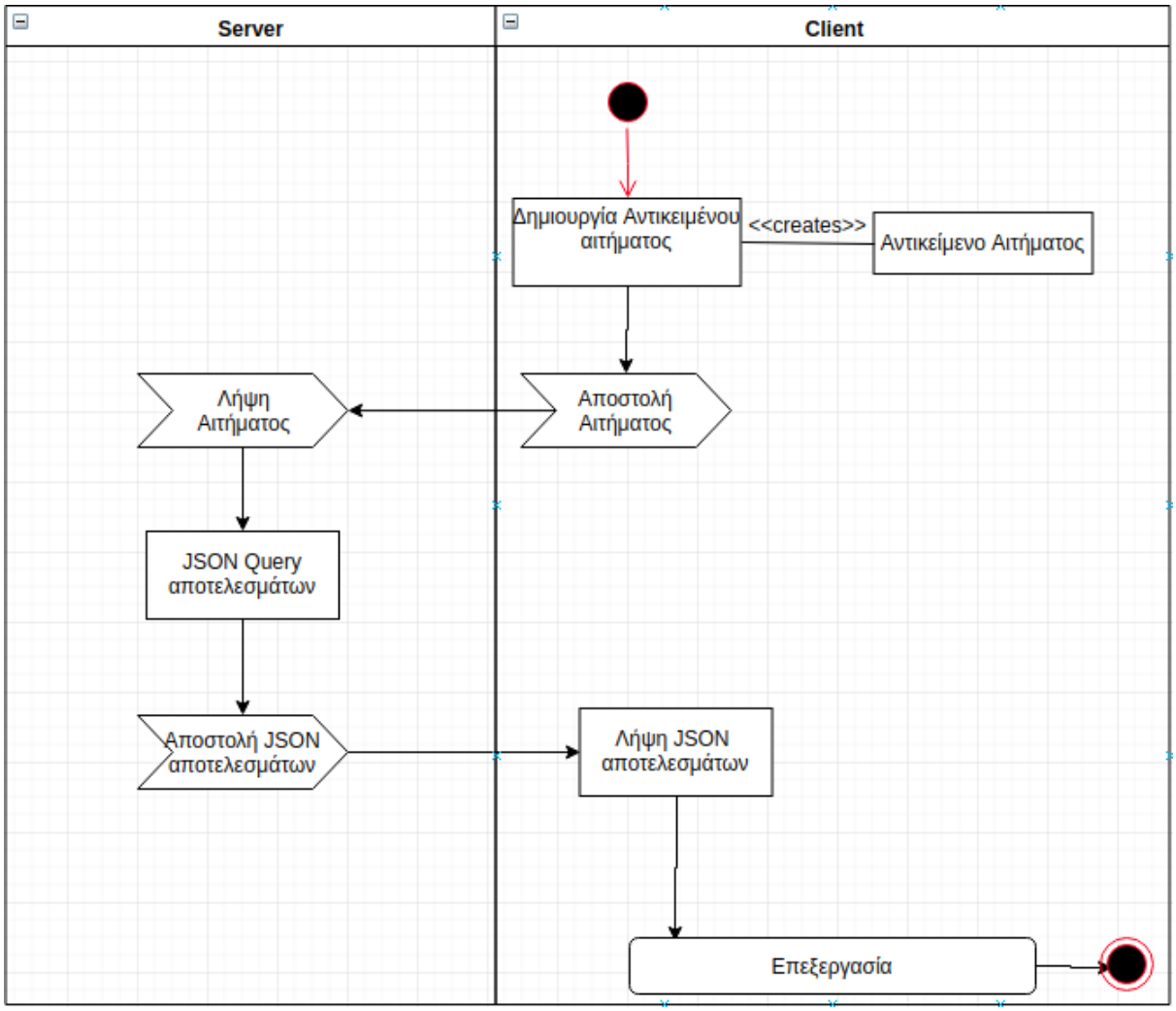
1.1 Εισαγωγή: σκοπός του λογισμικού

Σκοπός του συστήματος είναι η παροχή πληροφοριών στους χρήστες σχετικά με κομμωτήρια μιας περιοχής της επιλογής τους, όπως τοποθεσία, τιμές-προσφορές, διαφορετικές επιλογές κόμμωσης κλπ καθώς και η διευκόλυνση τους κατά την κράτηση σε κάποιο κομμωτήριο η οποία θα γίνεται διαδικτυακά, γρήγορα και εύκολα. Ταυτόχρονα, οι χρήστες μπορούν να συγκρίνουν τις επιλογές τους με βάση το κριτήριο που αυτοί επιθυμούν, προκειμένου να καταλήξουν στην κατάλληλη για αυτούς επιλογή. Τέλος, τα δεδομένα αυτά παρέχονται και μέσω προδιαγεγραμμένου API σε οποιονδήποτε ενδιαφερόμενο τρίτο.

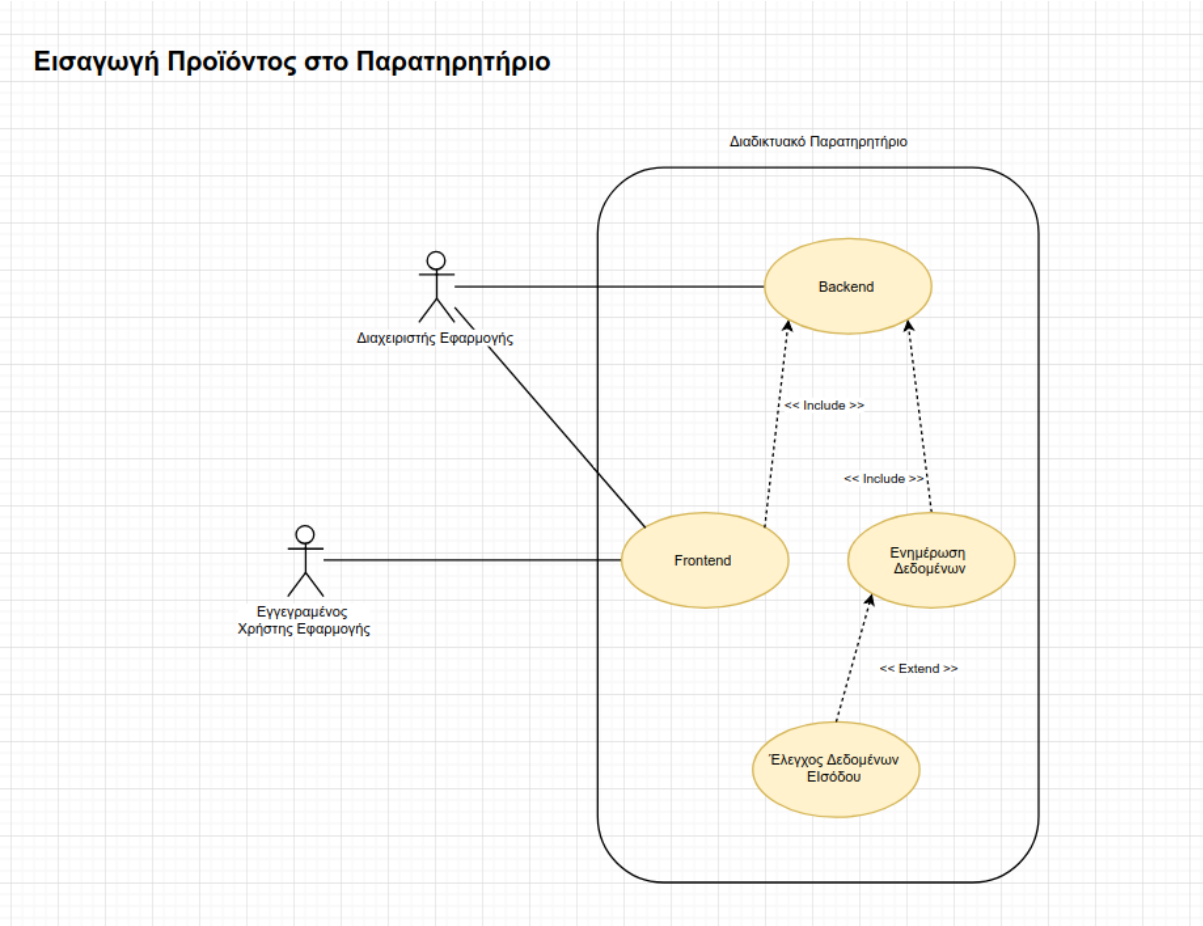
1.2 Επισκόπηση του λογισμικού

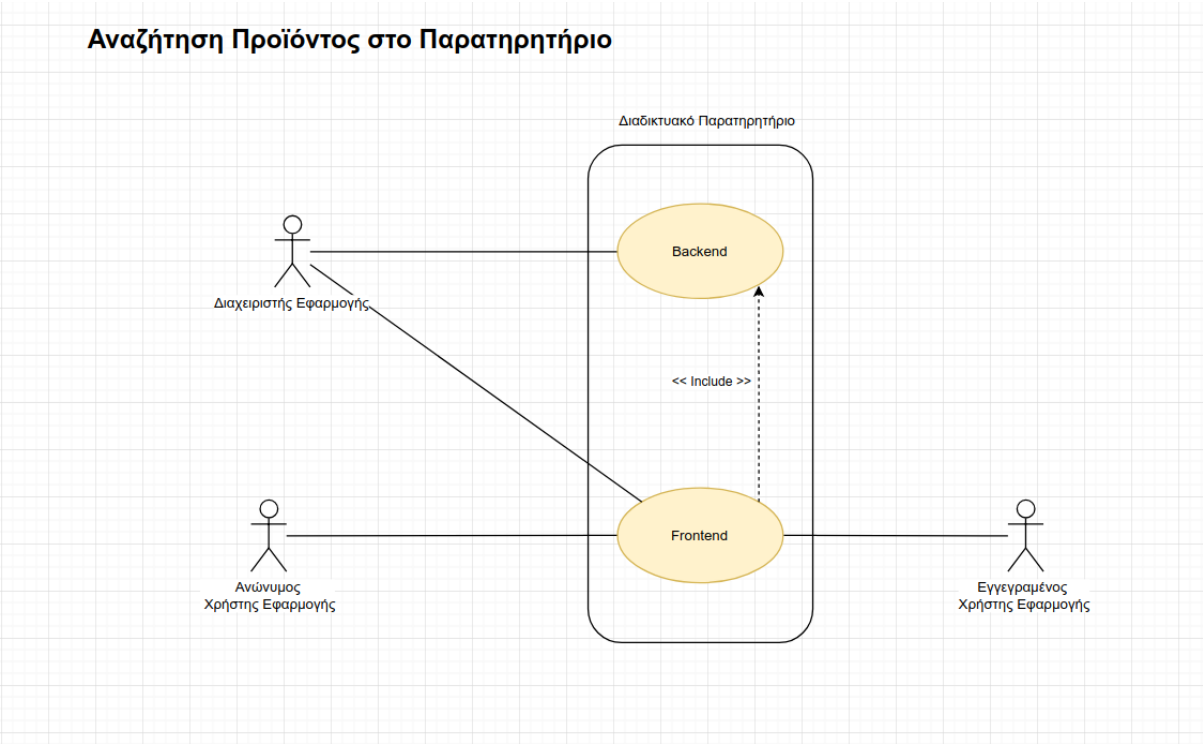


1.3.1 Διεπαφές με εξωτερικά συστήματα και εφαρμογές λογισμικού



1.3.2 Διεπαφές με το χρήστη





1.3.3 Διεπαφές με υλικό

N/A

1.3.4 Διεπαφές επικοινωνιών

N/A

2. Αναφορές - πηγές πληροφοριών

N/A

3. Προδιαγραφές απαιτήσεων λογισμικού

3.1 Εξωτερικές διεπαφές

- *POST {baseURL}/login*

Σύνδεση – διαπίστευση ενός χρήστη. Επιστρέφεται ένα authentication token, το οποίο θα περιλαμβάνεται σε όλες τις επόμενες κλήσεις κωδικοποιημένο στο custom HTTP header με όνομα X-OBSERVATORY-AUTH.

- *POST {baseURL}/logout*

Αποσύνδεση ενός χρήστη. Ο χρήστης θα πρέπει να έχει συνδεθεί επιτυχώς σε προηγούμενο χρόνο και το αίτημα θα πρέπει να περιλαμβάνει το αντίστοιχο authentication token στο X-OBSERVATORY-AUTH custom HTTP header. Το authentication token ακυρώνεται.

- *GET {baseURL}/products*

Επιστρέφεται η λίστα των προϊόντων του παρατηρητηρίου. Οι υποστηριζόμενες παράμετροι ως μέρος του URL είναι: start(Integer, default 0) , count (Integer, default 20) , status (String, με επιτρεπτές τιμές: ALL | WITHDRAWN | ACTIVE, default ACTIVE), sort (String, με επιτρεπτές τιμές: id|ASC, id|DESC, name|ASC, name|DESC, default id|DESC). Τα αποτελέσματα επιστρέφονται ως: start (Integer), count (Integer), total (Long), products (List<Product>), ενώ το Product περιέχει: id (Long), name (Long), description (String), category (String), tags (List<String>), withdrawn (Boolean)

- *POST {baseURL}/products*

Δημιουργία νέου προϊόντος. Η αίτηση θα πρέπει να περιέχει στο body (και όχι στο query) τις τιμές για τα υποχρεωτικά πεδία του προς δημιουργία προϊόντος (name, description, category, tags). Το αποτέλεσμα της αίτησης, σε περίπτωση επιτυχούς δημιουργίας νέου προϊόντος, θα είναι η πλήρης κωδικοποίηση των δεδομένων του, όπως περιγράφηκε προηγουμένως.

- *GET {baseURL}/products/{id}*

Επιστρέφονται τα δεδομένα του προϊόντος με το συγκεκριμένο id.

- *PUT {baseURL}/products/{id}*

Ενημέρωση των στοιχείων του προϊόντος με το συγκεκριμένο id. Τα πεδία θα πρέπει να κωδικοποιηθούν στο body και όχι στο query της αίτησης. Το αποτέλεσμα θα είναι η πλήρης κωδικοποίηση των δεδομένων του προϊόντος.

- *PATCH {baseURL}/products/{id}*

Μερική ενημέρωση του προϊόντος με το συγκεκριμένο id.

- *DELETE {baseURL}/products/{id}*

Διαγραφή του προϊόντος με το συγκεκριμένο id. Αν ο χρήστης που εκτελεί την αίτηση έχει το ρόλο του Εθελοντή, το σύστημα θα πρέπει να θέσει την τιμή withdrawn=true στο προϊόν. Αν ο χρήστης έχει το ρόλο του Διαχειριστή, το σύστημα θα προχωρά στη

διαγραφή του προϊόντος και όλων των σχετικών του πληροφοριών. Το αποτέλεσμα θα είναι απλό μήνυμα επιβεβαίωσης “message” = “OK”.

- *GET {baseURL}/shops*

Επιστρέφεται η λίστα των καταστημάτων του παρατηρητηρίου. Οι υποστηριζόμενες παράμετροι (ως μέρος του URL query), είναι ίδιες με την περίπτωση των προϊόντων. Ωστόσο, αντί για products, επιστρέφεται shops (List<Shop>), με κάθε Shop να περιέχει: id(Long), name (String), address (String), lng (Double), lat (Double), tags (List<String>), withdrawn (Boolean).

- *POST {baseURL}/shops*

Δημιουργία νέου καταστήματος. Η αίτηση θα πρέπει να περιέχει στο body (και όχι στο query) τις τιμές για όλα τα πεδία του προς δημιουργία καταστήματος (όλα είναι υποχρεωτικά). Αποτέλεσμα η πλήρης κωδικοποίηση του καταστήματος.

- *GET {baseURL}/shops/{id}*

Επιστρέφονται τα δεδομένα του καταστήματος με το συγκεκριμένο id. Η κωδικοποίηση των δεδομένων θα γίνεται όπως έχει ήδη περιγραφεί προηγουμένως,

- *PUT {baseURL}/shops/{id}*

Ενημέρωση των στοιχείων του καταστήματος με το συγκεκριμένο id. Οι προδιαγραφές είναι ίδιες με την PUT {baseURL}/products/{id}.

- *PATCH {baseURL}/shops/{id}*

Μερική ενημέρωση του καταστήματος με το συγκεκριμένο id. Οι προδιαγραφές είναι ίδιες με την PATCH {baseURL}/products/{id}.

- *DELETE {baseURL}/shops/{id}*

Διαγραφή του καταστήματος με το συγκεκριμένο id. Οι προδιαγραφές είναι ίδιες με την DELETE {baseURL}/products/{id}

- *GET {baseURL}/prices*

Αναζήτηση τιμών προϊόντων σε καταστήματα. Οι υποστηριζόμενες παράμετροι είναι: start (Integer, default 0), count (Integer, default 20), geoDist(Integer), geoLng (Double), geoLat (Double), dateFrom (Date, με τη μορφή EEEE-MM-HH), dateTo (Date, με τη μορφή EEEE-MM-HH), shops (List<String>), products (List<String>), tags (List<String>), sort (List<String>). Τα πεδία geoDist, geoLng, geoLat θα πρέπει είτε όλα είτε κανένα να έχουν τιμές. Αν δοθούν τιμές σε κάποια από τα πεδία αυτά (αλλά όχι σε όλα), το σύστημα απαντάει με το κατάλληλο μήνυμα λάθους. Το ίδιο ισχύει και για τα πεδία dateFrom, dateTo. Τα πεδία shops, products, tags είναι προαιρετικά και λαμβάνουν μηδέν ή περισσότερες τιμές. Τα αποτελέσματα επιστρέφονται με την κωδικοποίηση: start (Integer), count (Integer), total (Long), prices (List<Price>), όπου Price κωδικοποιείται ως: price (Double), date (Date, με τη μορφή EEEE-MM-HH), productName (String), productId (Long), productTags (List<String>), shopId (String), shopName (String), shopTags (List<String>), shopAddress (String), shopDist (Integer).

- *POST {baseURL}/prices*

Η αίτηση θα πρέπει να περιέχει στο body (και όχι στο query) τις εξής τιμές υποχρεωτικά: price (Double), dateFrom (Date, με τη μορφή EEEE-MM-HH), dateTo (Date, με τη μορφή EEEE-MM-HH), productId (Long), shopId (String).

3.2 Λειτουργίες: περιπτώσεις χρήσης

3.2.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ (Εισαγωγή τιμής στο παρατηρητήριο)

3.2.1.1Χρήστες (ρόλοι) που εμπλέκονται

Οι χρήστες είναι οι Εγγεγραμμένοι Χρήστες και οι Διαχειριστές. Δεν χρειάζεται αλληλεπίδραση μεταξύ χρηστών.

3.2.1.2Προϋποθέσεις εκτέλεσης

Οι χρήστες πρέπει να έχουν καταχωρήσει τα στοιχεία τους στο Παρατηρητήριο. Το Κατάστημα πρέπει να υπάρχει αν πρόκειται για απλό Εγγεγραμμένο χρήστη.

3.2.1.3Περιβάλλον εκτέλεσης

Αποτελεί διαδικτυακή διεπαφή χρήστη που αλληλεπιδρά με τη Βάση Δεδομένων.

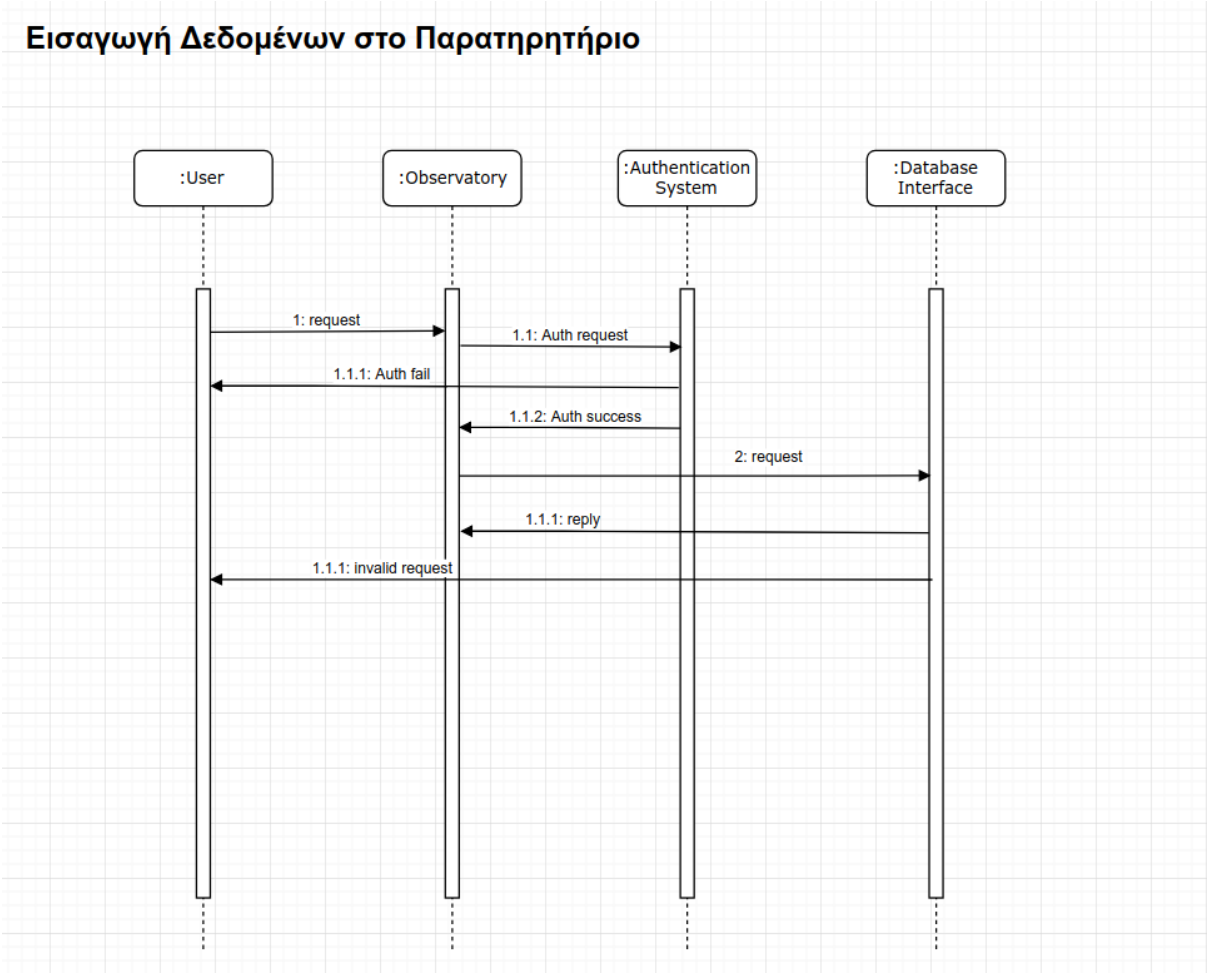
3.2.1.4Δεδομένα εισόδου

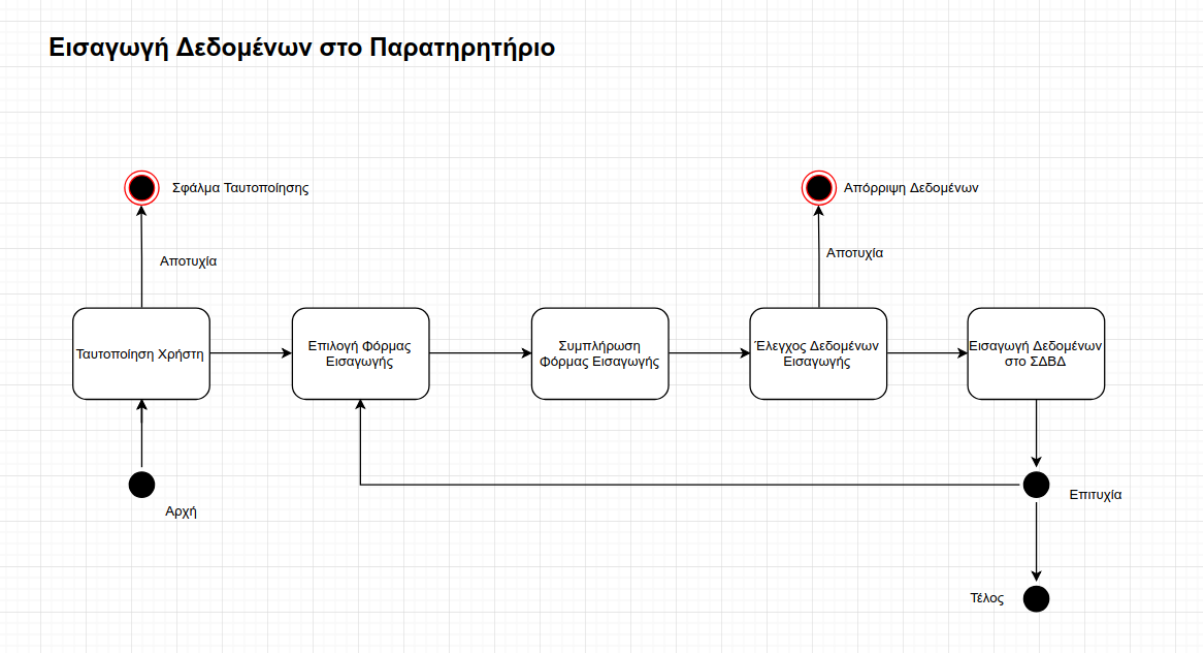
Τα δεδομένα εισόδου δεν ελέγχονται πέραν των βασικών ελέγχων, καθώς ο σημασιολογικός έλεγχος τους είναι πρακτικά αδύνατος.

3.2.1.5Παράμετροι

Οι παράμετροι που επηρεάζουν την ορθότητα των δεδομένων παράγονται αυτόματα από το σύστημα και άρα δεν τίθεται θέμα εγκυρότητας δεδομένων, με δεδομένη την ορθότητα του συστήματος.

3.2.1.6Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά





- *Βήμα 1: Ταυτοποίηση. Επιστροφή σφάλματος σε περίπτωση αποτυχίας*
- *Βήμα 2: Πλοήγηση στην Φόρμα Εισαγωγής*
- *Βήμα 3: Συμπλήρωση Φόρμας Εισαγωγής*
- *Βήμα 4: Έλεγχος δεδομένων από το σύστημα και απόρριψη σε περίπτωση λανθασμένης εισαγωγής*
- *Βήμα 5: Εισαγωγή δεδομένων στη Βάση Δεδομένων του συστήματος και επιστροφή κατάλληλα μορφοποιημένου μηνύματος στο χρήστη*

3.2.1.7Δεδομένα εξόδου

3.2.2 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ (Αναζήτηση προϊόντος στο παρατηρητήριο)

3.2.2.1Χρήστες (ρόλοι) που εμπλέκονται

Οι χρήστες είναι οι Ανώνυμοι, οι Εγγεγραμμένοι Χρήστες και οι Διαχειριστές. Δεν χρειάζεται αλληλεπίδραση μεταξύ χρηστών.

3.2.2.2Προϋποθέσεις εκτέλεσης

Η μόνη συνθήκη που πρέπει να ισχύει είναι να είναι πλήρως χρηστική η εφαρμογή.

3.2.2.3Περιβάλλον εκτέλεσης

Αποτελεί διαδικτυακή διεπαφή χρήστη που αλληλεπιδρά με τη Βάση Δεδομένων αναζητώντας σε αυτή αυτό που χρήστης έχει υποβάλει.

3.2.2.4Δεδομένα εισόδου

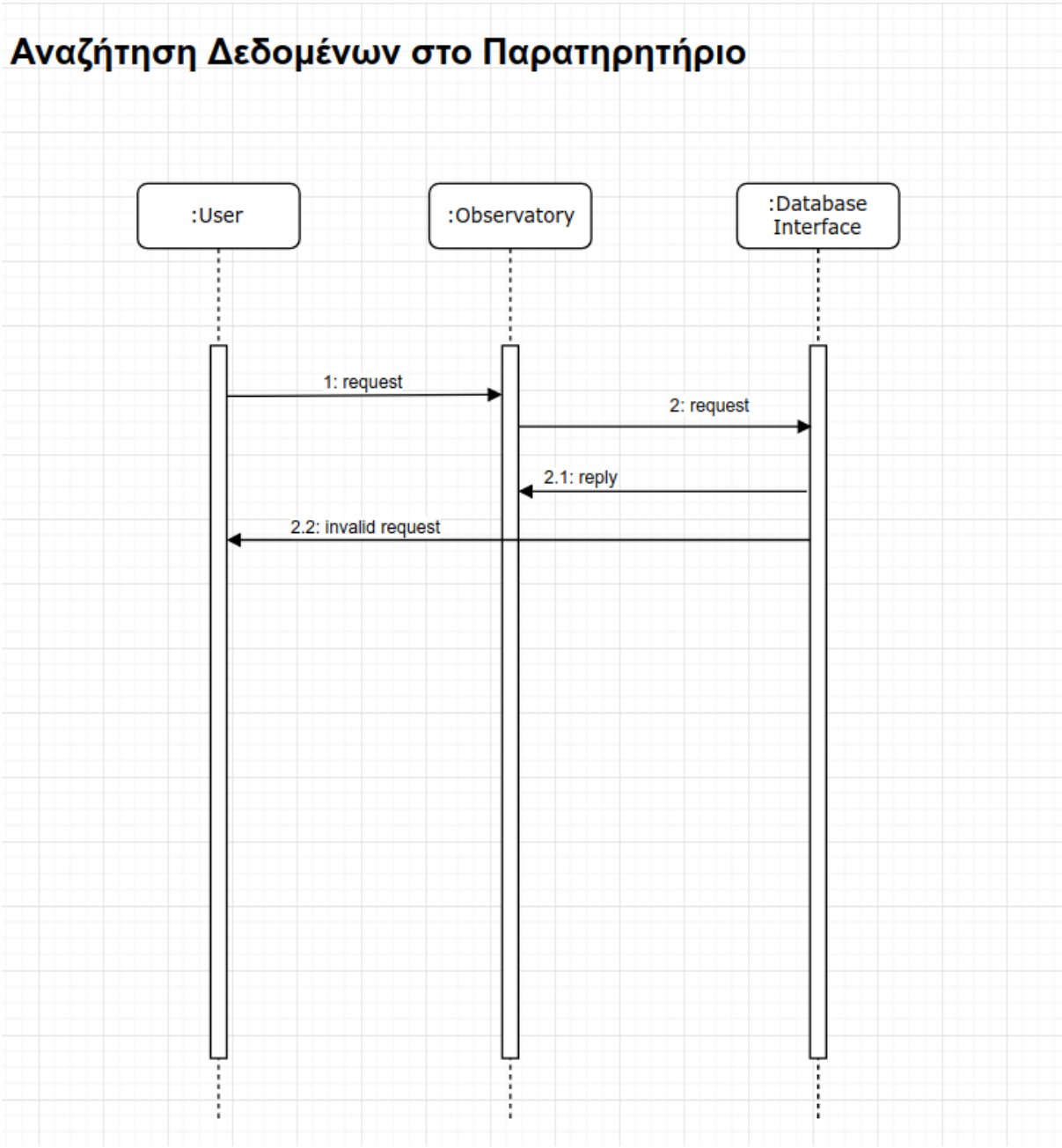
Τα δεδομένα εισόδου δεν ελέγχονται, καθώς ο έλεγχος τους είναι πρακτικά αδύνατος. Σε περίπτωση μη εγκυρότητας τα αποτελέσματα που θα επιστραφούν θα είναι απλά μη σχετικά με αυτό που ο χρήστης επιθυμούσε.

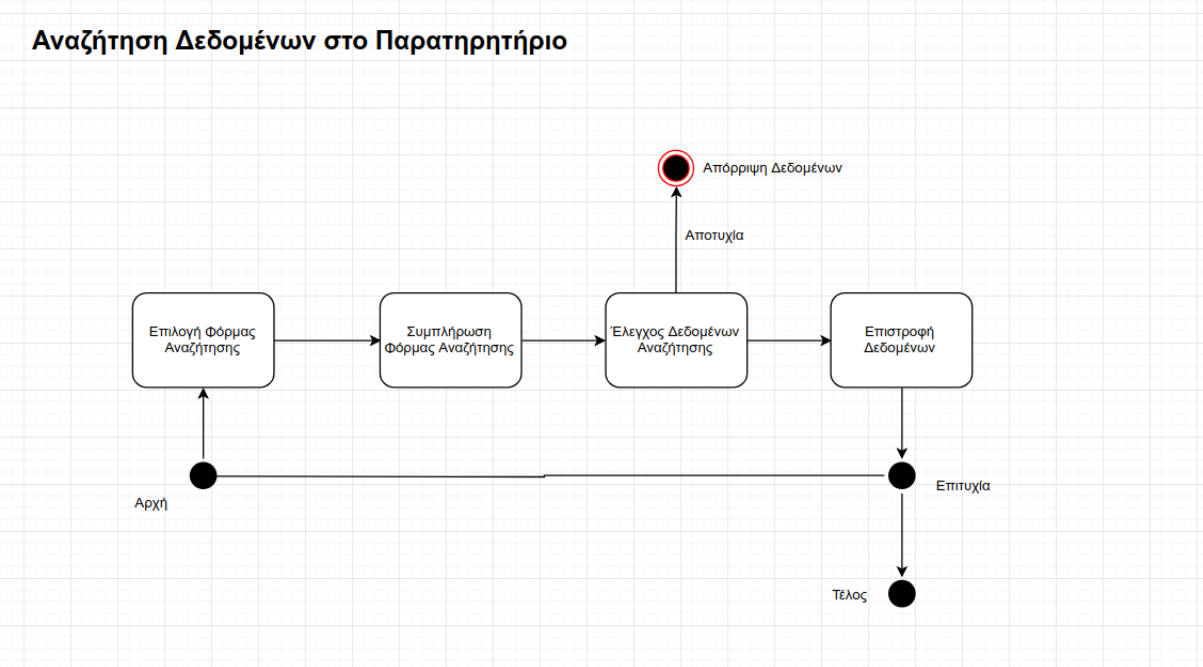
3.2.2.5Παράμετροι

Οι παράμετροι που επηρεάζουν την ορθότητα των δεδομένων παράγονται αυτόματα από το σύστημα και άρα δεν τίθεται θέμα εγκυρότητας δεδομένων, με δεδομένη την

ορθότητα του συστήματος.

3.2.2.6Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά





- *Βήμα 1: Πλοήγηση στην Φόρμα Αναζήτησης*
- *Βήμα 2: Συμπλήρωση Φόρμας Αναζήτησης*
- *Βήμα 3: Έλεγχος δεδομένων από το σύστημα και απόρριψη σε περίπτωση λανθασμένης εισαγωγής*
- *Βήμα 4: Εισαγωγή δεδομένων στη Βάση Δεδομένων του συστήματος και επιστροφή κατάλληλα μορφοποιημένου μηνύματος στο χρήστη*

3.2.2.7Δεδομένα 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)

3.3 Απαιτήσεις επιδόσεων

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

3.4 Απαιτήσεις οργάνωσης δεδομένων

3.4.1 Τεχνική περιγραφή των δεδομένων που διαχειρίζεται το λογισμικό και των σχετικών μετρικών φορτίου δεδομένων εισόδου, επεξεργασίας κ.λπ.

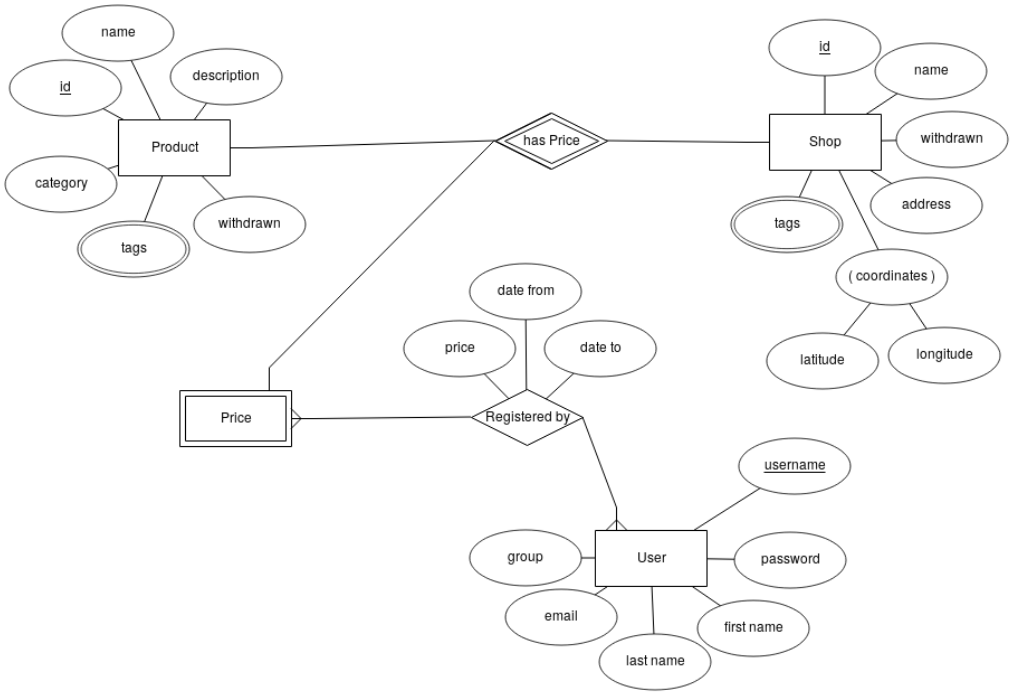
Η εφαρμογή μας αρχικά για κάθε πεδίο τιμών της δέχεται είσοδο url query μέσω των παραμέτρων start, count, status, sort. Στη συνέχεια μπορεί να δεχτεί query από αίτημα με υποστηριζόμενες παραμέτρους τις start, count, geoDist, geoLng, geoLat, dateFrom, dateTo, shops, products, tags και sort.

Τα σχετικά πρότυπα δεδομένων που δέχεται αλλά και επιστρέφει το παρατηρητήριο είναι JSON. Κάθε στιγμή η εφαρμογή μπορεί να εξυπηρετεί πολλούς χρήστες ταυτόχρονα και για αυτό πρέπει πάντα να προνοεί απο άποψη μνήμης ώστε να μπορεί να διαχειριστεί κάποια μαζική είσοδο δεδομένων στην βάση της.

3.4.2 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

Απαιτήσεις πρόσβασης και περιορισμοί.

3.4.3 Μοντέλο δεδομένων



3.4.4 Προδιαγραφές ακεραιότητας δεδομένων

Κανόνες ακεραιότητας και εγκυρότητας δεδομένων

Τα δεδομένα ελέγχονται ενδελεχώς για την εξακρίβωση της ορθής μορφοποίησης τους από τους χρήστες, όπου είναι δυνατόν, και είναι εξασφαλισμένη, στο μέγιστο δυνατό, η εισαγωγή τους στην Βάση Δεδομένων του συστήματος. Η κακόβουλη, αλλά παραταύτα ορθή καταχώρηση δεδομένων μπορεί μόνο να εντοπιστεί χειρονακτικά και η διασφάλιση της εγκυρότητας των δεδομένων αυτών δεν είναι εξασφαλισμένη και η επιχείρηση δε φέρει καμία ευθύνη επ' αυτού.

3.4.5 Προδιαγραφές διατήρησης δεδομένων

Η διατήρηση των δεδομένων διασφαλίζεται κατά το μέγιστο δυνατό και επαφίεται σε προϋπάρχοντα τεκμηριωμένα και καθιερωμένα εργαλεία.

3.5 Περιορισμοί σχεδίασης

Με δεδομένο πως χρησιμοποιείται το framework "Django", οι περιορισμοί σχεδίασης είναι οι εξής:

- *Η δομή του συστήματος βελτιστοποιείται απομονώνοντας τις διαφορετικές συνιστώσες*
- *Τα μοντέλα-οντότητες μας ξεκινούν από κεφαλαίο γράμμα*
- *Τα αυτοματοποιημένα test πρέπει να συμπεριλαμβάνονται σε ανάλογη κλάση και να ξεκινούν με την λέξη "test"*
- *Οι ρυθμίσεις αποθηκεύονται σε JSON μορφή*

3.6 Λοιπές απαιτήσεις

3.6.1 Απαιτήσεις διαθεσιμότητας λογισμικού

Οι εθελοντές πρέπει ανα πάσα στιγμή να μπορούν να καταχωρούν τιμές προϊόντων στο παρατηρητήριο, καθώς και να διορθώνουν τις ήδη υπάρχουσες. Το σύστημα πρέπει να έχει κατάλληλους χρόνους απόκρισης και επεξεργασίας δεδομένων κάτω από τις προβλεπόμενες συνθήκες λειτουργίας. Δυνατότητα του συστήματος να λειτουργήσει κάτω από διαφορετικά περιβάλλοντα χωρίς να απαιτούνται αλλαγές από τον χρήστη. Το σύστημα πρέπει να είναι ικανό να προσαρμοστεί στην αύξηση του αριθμού χρηστών. Το σύστημα οφείλει να διατηρεί ένα επαρκές επίπεδο χρηστικότητας ακόμα και σε περίπτωση τεχνικών λαθών.

3.6.2 Απαιτήσεις ασφάλειας

Τα προσωπικά δεδομένα των εθελοντών πρέπει να διαφυλάσσονται σύμφωνα με την κείμενη νομοθεσία. Δεν πρέπει να είναι δυνατή η αντιστοίχιση χρήστη-καταχώρησης από απλούς χρήστες.

3.6.3 Απαιτήσεις συντήρησης

Οποιεσδήποτε αλλαγές στον κώδικα της εφαρμογής και του συστήματος θα πρέπει να επιτυγχάνουν στα προϋπάρχοντα test ορθής λειτουργικότητας.

4. Παράρτημα

4.1 Παραδοχές και εξαρτήσεις

N/A

4.2 Ακρωνύμια και συντομογραφίες

N/A

4.3 Υποστηρικτικά έγγραφα, πρότυπα κ.λπ.

N/A