

# Έγγραφο απαιτήσεων λογισμικού (SRS)

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

## Cheapset

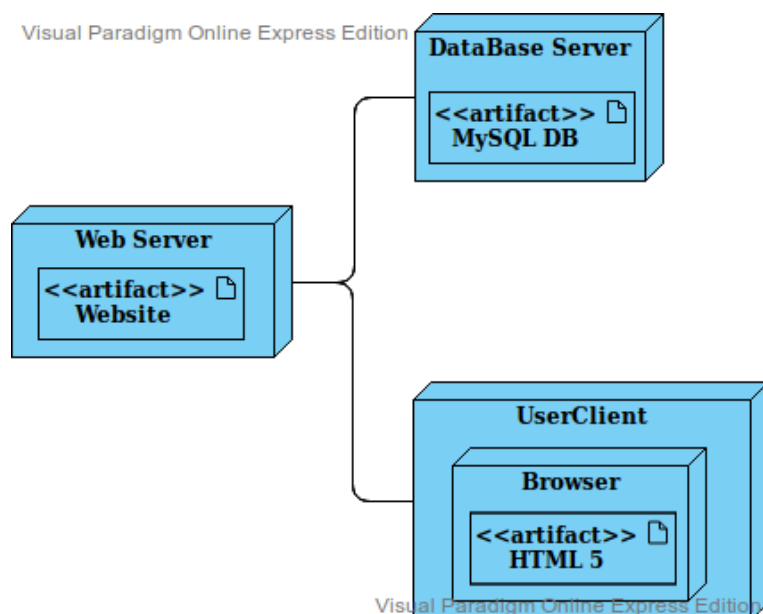
### 1. Εισαγωγή

#### 1.1 Εισαγωγή: σκοπός του λογισμικού

Η Cheapset είναι μια online εφαρμογή που έχει ως στόχο να δίνει τη δυνατότητα σε όλους να ανακαλύπτουν τον κόσμο της τεχνολογίας και να συνεισφέρουν ενεργά στη διάδοσή της. Μέσω της Cheapset αλλάζει ριζικά η οπτική του καταναλωτή για την αγορά. Ικανοποιείται στο έπακρο η ανάγκη του για εύρεση του επιθυμητού συνδυασμού ποιότητας-τιμής στο προϊόν τεχνολογίας της επιλογής του, με τον πλέον άμεσο και εύκολο τρόπο. Στόχος της Cheapset είναι η συμβολή της στην εξισορρόπηση των τιμών της αγοράς, η προώθηση ενός υγιούς συναγωνισμού στην ραγδαία εξελισσόμενη αγορά των προϊόντων τεχνολογίας, καθώς και η παροχή υψηλού επιπέδου υπηρεσιών στο χρήστη, έχοντας ως γνώμονα την εξυπηρέτηση των συμφερόντων και αναγκών τόσο του απαιτητικού καταναλωτή, όσο και της σύγχρονης επιχείρησης.

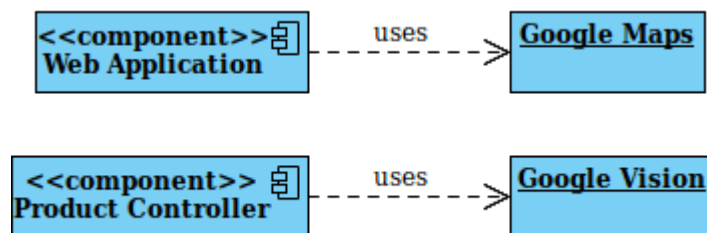
#### 1.2 Επισκόπηση του λογισμικού

Η Cheapset είναι μια διαδικτυακή εφαρμογή, οργανωμένη κατά το μοντέλο MVC. Βασίζει τη λειτουργία της σε ένα RESTful API, μέσω του οποίου γίνεται η επικοινωνία των frontend και backend τμημάτων της εφαρμογής. Το backend διαχειρίζεται με τη χρήση controller τη βάση δεδομένων και υλοποιεί το API. Το frontend τμήμα υλοποιεί την λογική της εφαρμογής, παίρνοντας τα απαραίτητα δεδομένα μέσω του RESTful API.



### 1.3.1 Διεπαφές με εξωτερικά συστήματα και εφαρμογές λογισμικού

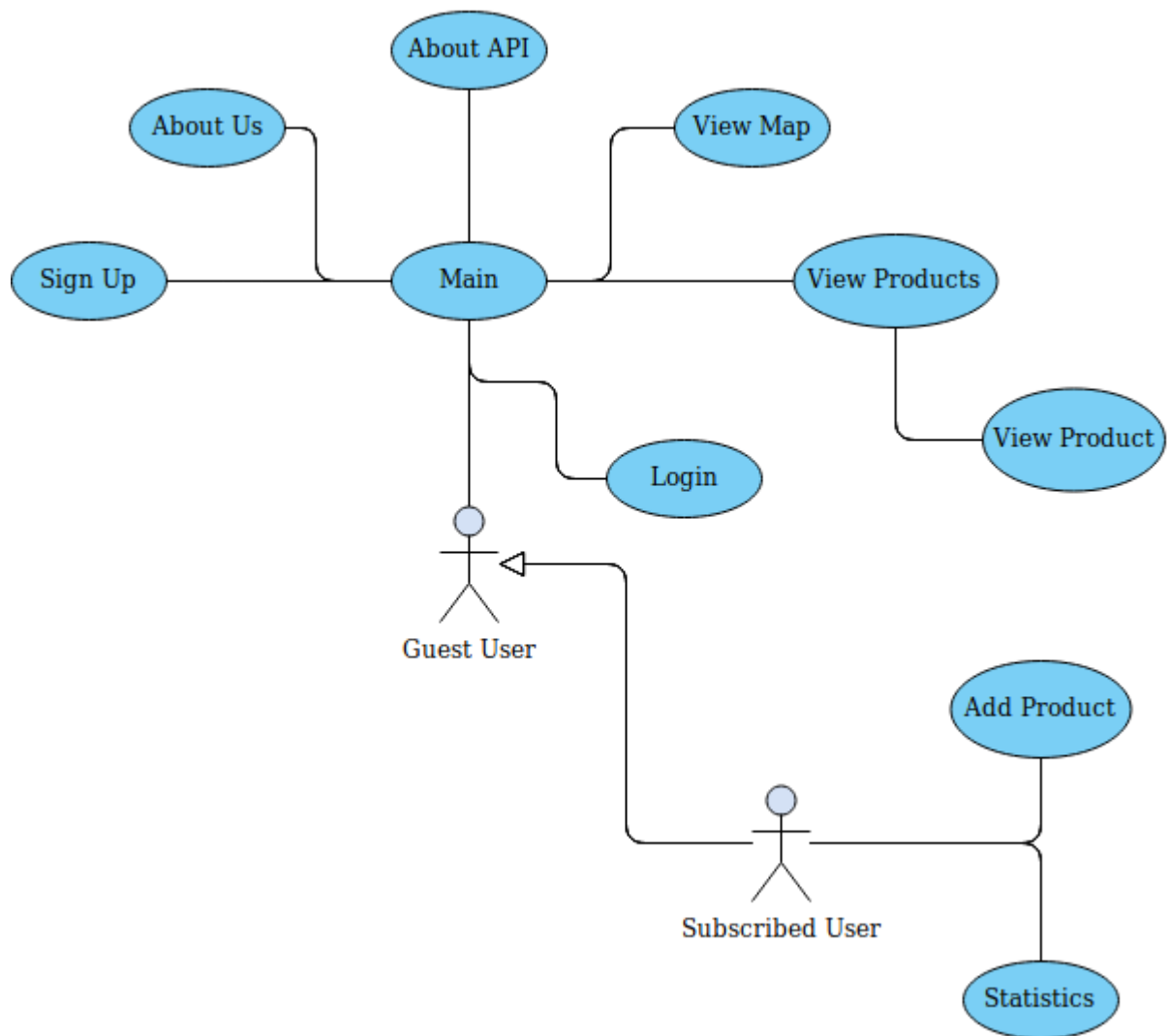
Τα εξωτερικά συστήματα που χρησιμοποιούνται στην εφαρμογή είναι το σύστημα google maps και το google vision. Το πρώτο χρησιμοποιείται ώστε να αξιοποιείται η τοποθεσία του χρήστη ως δεδομένο στην αναζήτηση προϊόντων, καθώς και για προβολή σχετικών χαρτών για την πλοήγησή του. Το δεύτερο χρησιμοποιείται ώστε να διευκολυνθεί η διαδικασία της εισαγωγής των στοιχείων ενός προϊόντος στην βάση δεδομένων, αξιοποιώντας φωτογραφία του προϊόντος από την οποία εξάγονται ορισμένα χαρακτηριστικά. Και στις δύο περιπτώσεις η εφαρμογή επικοινωνεί με τα αντίστοιχα συστήματα μέσω του API που προσφέρει η google με το οποίο παράγει request και του επιστρέφονται responses με τα επιθυμητά αποτελέσματα. Παρατίθενται τα component diagrams που περιγράφουν τις διεπαφές αυτές.



### 1.3.2 Διεπαφές με το χρήστη

Οι χρήστες της εφαρμογής διακρίνονται στον επισκέπτη και στον εγγεγραμμένο χρήστη. Ο επισκέπτης έχει τη δυνατότητα να επισκεφτεί τις σελίδες Main, About Us, About API, Sign Up, View Map, View Products, View Product, Login, ενώ ο εγγεγραμμένος χρήστης μπορεί να χρησιμοποιήσει επιπλέον τις σελίδες Add Product και Statistics. Αναλυτικά, για κάθε σελίδα ισχύουν τα παρακάτω:

- Η Main αποτελεί σελίδα-πλοηγό προς τις άλλες σελίδες.
- Η About Us περιέχει πληροφορίες για την ομάδα ανάπτυξης της εφαρμογής.
- Η About API περιέχει πληροφορίες για το API που παρέχεται από την εφαρμογή.
- Με τη Sign Up γίνεται η εγγραφή του χρήστη
- Η View Map απεικονίζει σε χάρτη τα υποστηριζόμενα από την εφαρμογή καταστήματα.
- Η View Products απεικονίζει όλα τα προϊόντα της βάσης δεδομένων, που ανήκουν σε μια συγκεκριμένη κατηγορία.
- Η View Product περιλαμβάνει πληροφορίες για ένα συγκεκριμένο προϊόν.
- Με τη Login γίνεται η είσοδος του χρήστη στην εφαρμογή .
- Με την Add Product εισάγει ο εγγεγραμμένος χρήστης νέο προϊόν στη βάση, μέσω φωτογραφίας.
- Η Statistics απεικονίζει στατιστικά στοιχεία που αφορούν στα προϊόντα της βάσης.



### 1.3.3 Διεπαφές με υλικό

Για την λήψη φωτογραφίας από την οποία θα εξάγονται τα βασικά χαρακτηριστικά των προϊόντων χρησιμοποιείται η κάμερα του κινητού. Συγκεκριμένα, η εφαρμογή ζητά έγκριση για χρήση της κάμερας για του κινητού και ύστερα την ανοίγει. Όταν ο χρήστης τραβήξει την κατάλληλη φωτογραφία, η εφαρμογή την αποθηκεύει και την χρησιμοποιεί η ίδια για την αναγνώριση του κειμένου σε αυτήν και τελικά την συμπλήρωση όσο το δυνατόν περισσότερων στοιχείων στην φόρμα υποβολής νέου προϊόντος.

## 3. Προδιαγραφές απαιτήσεων λογισμικού

### 3.1 Εξωτερικές διεπαφές

Όπως αναφέρεται και στη ενότητα 1.3.1 για κάποιες λειτουργίες του συστήματος χρησιμοποιούνται εξωτερικές υπηρεσίες της google. Συγκεκριμένα η λειτουργίες αυτές και η συνεργασία τους με τις υπηρεσίες αυτές περιγράφονται παρακάτω:

### 1. Εισαγωγή νέων προϊόντων/τιμών από τον χρήστη:

Για την εισαγωγή των απαραίτητων στοιχείων του νέου προϊόντος γίνεται χρήση του google vision api. Κατά την διαδικασία εισαγωγής νέου προϊόντος ζητείται από τον χρήστη να δώσει άδεια χρήσης της κάμερας της συσκευής του στην εφαρμογή. Μετά την έγκριση του χρήστη αναμένεται η λήψη φωτογραφίας της καρτέλας χαρακτηριστικών του προϊόντος που υπάρχει στο κατάστημα. Έπειτα η φωτογραφία αυτή αποστέλλεται στον Web-Server μέσω του RESTfull API. Όταν η φωτογραφία ληφθεί από το Web-Server, γίνεται μια κλήση στο vision API με δεδομένα την ίδια την εικόνα και αίτημα την Image-to-Text ανάλυση. Όταν φτάσει η απάντηση από το vision API στέλνεται ως απάντηση στον client για την επιβεβαίωση των δεδομένων από τον χρήστη. Με την επιβεβαίωση του χρήστη το client στέλνει τα δεδομένα με αίτηση POST στο RESTful API.

Παράλληλα με τα παραπάνω για την καταγραφή του καταστήματος που βρίσκεται ο χρήστης χρησιμοποιείται το google maps API στέλνοντας ένα request με δεδομένα τις συντεταγμένες που λαμβάνονται από το GPS του κινητού και αίτημα την υπηρεσία What is Here.

### 2. Εύρεση Καταστημάτων:

Ο χρήστης έχει την δυνατότητα να πλοηγηθεί στην σελίδα .../mapsearch στην οποία του παρέχονται εργαλεία για την εύρεση καταστημάτων που πιθανόν να τον ενδιαφέρει να επισκεφτεί. Στην σελίδα αυτή κεντρικό στοιχείο είναι ένας χάρτη μέσω του google map API στο οποίο φαίνονται σημειωμένα τα καταστήματα για τα οποία η εφαρμογή διαθέτει προϊόντα. Μετά την αναζήτηση/φιλτράρισμα των καταστημάτων βάση κριτηρίων ορισμένων από τον χρήστη τα καταστήματα “δημιουργούνται” πάνω στον χάρτη ως ένα mark του google map.

### 3. Σελίδα Πληροφοριών Προϊόντος:

Στην σελίδα αυτή πέρα από την περιγραφή του προϊόντος διατίθεται και ένας χάρτης μέσω του google map API στον οποίο φαίνονται όλα τα καταστήματα στα οποία είναι διαθέσιμο αυτό το προϊόν.

## 3.2 Λειτουργίες: περιπτώσεις χρήσης

### 3.2.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 1: Εισαγωγή Προϊόντος

#### 3.2.1.1 Χρήστες (ρόλοι) που εμπλέκονται

Για την εισαγωγή προϊόντος στην βάση, ο χρήστης που αιτείται την εισαγωγή πρέπει να είναι εγγεγραμμένος.

#### 3.2.1.2 Προϋποθέσεις εκτέλεσης

Ο εγγεγραμμένος χρήστης καλείται να δώσει ορισμένα στοιχεία για το προϊόν, ανάλογα με την κατηγορία του, καθώς και το κατάστημα στο οποίο βρίσκεται. Αν τα στοιχεία που δόθηκαν είναι ορθά, τότε το προϊόν εισάγεται στην βάση δεδομένων. Πρέπει επίσης η εισαγωγή να γίνεται από κινητή συσκευή ώστε να είναι δυνατή η αναγνώριση της τοποθεσίας του χρήστη και συνεπώς να καταγραφεί το κατάστημα στο οποίο βρίσκεται.

#### 3.2.1.3 Περιβάλλον εκτέλεσης

Η συμπλήρωση των κατάλληλων δεδομένων για την εισαγωγή γίνεται στην σελίδα Add Product. Η αξιολόγηση των δεδομένων αυτών γίνεται από τον controller, ενώ η αποθήκευση του προϊόντος γίνεται από το DBMS.

#### 3.2.1.4 Δεδομένα εισόδου

Ο χρήστης το όνομά του, την κατηγορία στην οποία ανήκει, tags και μία περιγραφή. Προαιρετικά μπορεί να δώσει και άλλες πληροφορίες σχετικά με το προϊόν ανάλογα με την κατηγορία του προϊόντος. Πιο συγκεκριμένα υποστηρίζονται οι παρακάτω κατηγορίες:

Φορητός υπολογιστής

Τα δεδομένα που μπορεί να προσθέσει ο χρήστης είναι τα εξής:

- CPU (τύπος)
- Storage (μέγεθος δίσκου)
- Graphics (κάρτα γραφικών)
- Μάρκα
- RAM (μέγεθος μνήμης RAM)

Κινητά τηλέφωνα

Τα δεδομένα που μπορεί να προσθέσει ο χρήστης είναι τα εξής:

- Μέγεθος οθόνης
- Τεχνολογία οθόνης
- CPU
- RAM
- Μάρκα

Tablet

Τα δεδομένα που μπορεί να προσθέσει ο χρήστης είναι τα εξής:

- Μέγεθος οθόνης
- Τεχνολογία οθόνης
- Συνδεσιμότητα (WiFi, 4G)
- CPU
- RAM
- Μάρκα

Κονσόλες

Τα δεδομένα που μπορεί να προσθέσει ο χρήστης είναι τα εξής:

- Χώρος αποθήκευσης
- Τρόπος σύνδεσης οθόνης (HDMI)
- Μάρκα

Τηλεοράσεις

Τα δεδομένα που μπορεί να προσθέσει ο χρήστης είναι τα εξής:

- Τύπος (LED, LCD...)
- Μέγεθος
- Ανάλυση
- Συχνότητα
- Μάρκα
- Smart (ναι ή όχι)

#### 3.2.1.5 Παράμετροι

Για την εισαγωγή του προϊόντος στην βάση είναι αναγκαίο να είναι γνωστή η τοποθεσία του χρήστη καθώς και η ώρα που κάνει την εισαγωγή. Η τοποθεσία είναι πάντα γνωστή αφού η εισαγωγή

προϊόντος γίνεται μέσω κινητού και ζητείται επιβεβαίωση της εκτίμησης από τον χρήστη. Είναι απαραίτητη ώστε να αποθηκευτεί το κατάστημα στο οποίο ανήκει το προϊόν προς εισαγωγή.

#### *3.2.1.6 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά*

##### **Βήμα 1**

Έλεγχος του κατά πόσον ο χρήστης είναι εγγεγραμμένος ώστε να έχει τα δικαιώματα να εισαγάγει προϊόν. Στην περίπτωση που ο χρήστης δεν έχει το δικαίωμα εισαγωγής προϊόντος επιστρέφεται μήνυμα λάθους.

##### **Βήμα 2**

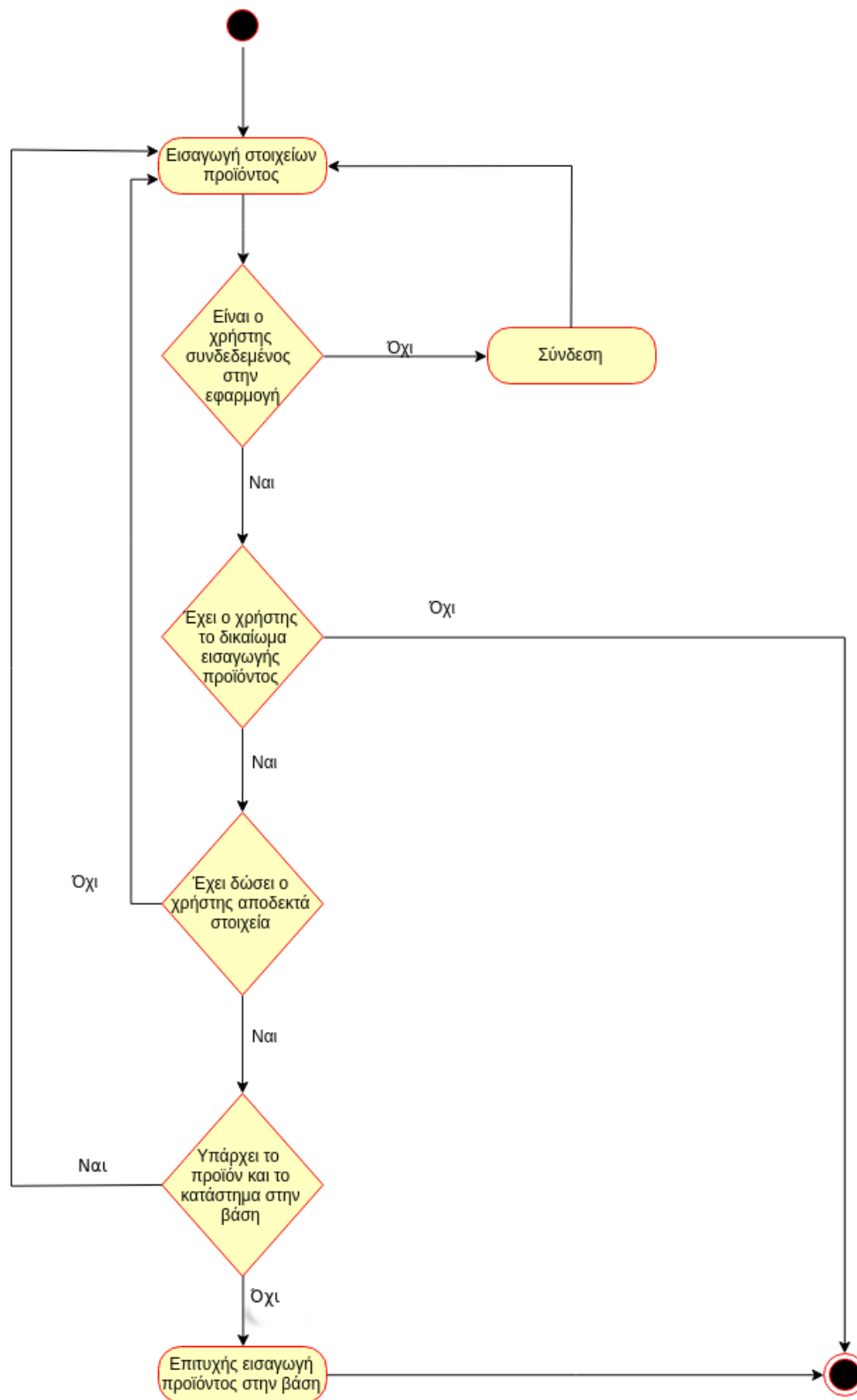
Έλεγχος ύπαρξης των υποχρεωτικών πεδίων. Στην περίπτωση που δεν υπάρχουν όλα, δεν γίνεται εισαγωγή προϊόντος και επιστρέφεται μήνυμα λάθους.

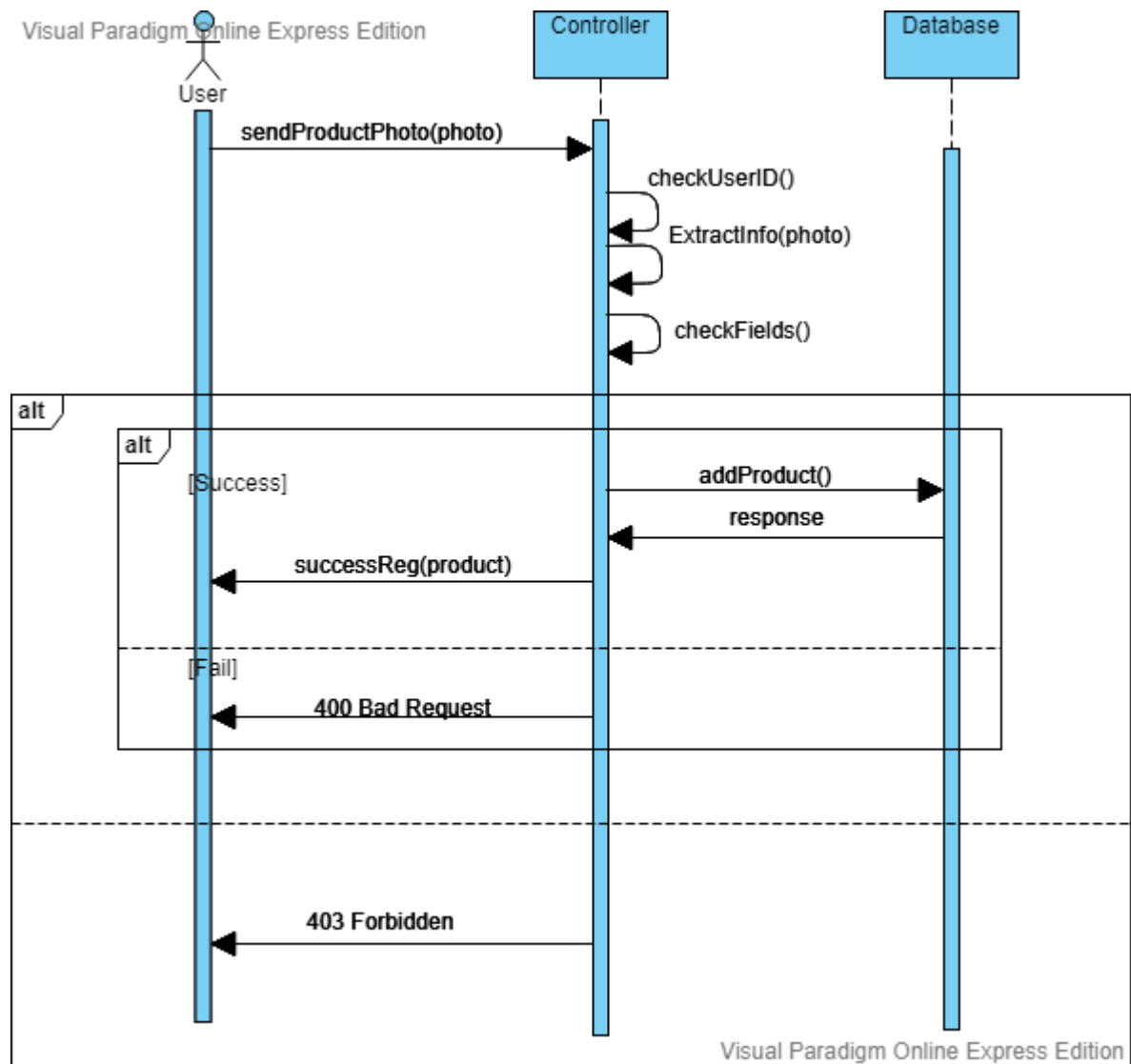
##### **Βήμα 3**

Έλεγχος ύπαρξης του συγκεκριμένου προϊόντος στην βάση. Αν υπάρχει το ίδιο προϊόν στην βάση, δεν είναι δυνατή η εισαγωγή του και επιστρέφεται μήνυμα λάθους.

##### **Βήμα 4**

Εισαγωγή του προϊόντος στην βάση δεδομένων.





### 3.2.1.7 Δεδομένα εξόδου

Στην συγκεκριμένη περίπτωση χρήσης τα δεδομένα εξόδου αποτελούνται μόνο από μηνύματα σφάλματος σε περίπτωση που ο χρήστης δεν είναι εγγεγραμμένος ή προσφέρει λανθασμένα ή ελλιπή στοιχεία για την εισαγωγή του προϊόντος, καθώς και από το μήνυμα επιβεβαίωσης στην περίπτωση επιτυχούς εισαγωγής του στην βάση. Το διάγραμμα αλληλουχίας που παρατέθηκε παραπάνω καλύπτει και τα δεδομένα εξόδου.

## 3.2.2 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 2: Εύρεση Προϊόντος

### 3.2.2.1 Χρήστες (ρόλοι) που εμπλέκονται

Κάθε χρήστης της εφαρμογής έχει τη δυνατότητα αναζήτησης προϊόντος.

### 3.2.2.2 Προϋποθέσεις εκτέλεσης

Δεν απαιτείται να πληρούνται κάποιες συγκεκριμένες συνθήκες.



### 3.2.2.3 Περιβάλλον εκτέλεσης

Ο χρήστης είναι σε θέση να αναζητήσει το προϊόν είτε στην διαδικτυακή εφαρμογή, είτε στην εφαρμογή για κινητό. Η εύρεση προϊόντος μπορεί να γίνει είτε στη Main Page, κάνοντας χρήση του search bar, είτε με απλή πλοήγηση στις σελίδες View Products, με προϊόντα ανά κατηγορία.

### 3.2.2.4 Δεδομένα εισόδου

Τα δεδομένα εισόδου, στην περίπτωση που ο χρήστης χρησιμοποιήσει το search bar της Main Page, είναι το όνομα του προϊόντος που αναζητεί. Σε άλλη περίπτωση, δεδομένα εισόδου μπορούν να θεωρηθούν οι επιλογές φιλτραρίσματος της λίστας των προβαλλόμενων προϊόντων.

### 3.2.2.5 Παράμετροι

Παράμετρο για την προβολή του επιλεγμένου προϊόντος και της τιμής του αποτελεί η ημερομηνία αναζήτησής του, μιας και η τιμή του προϊόντος εξαρτάται από την τρέχουσα ημερομηνία.

### 3.2.2.6 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Βήμα 1

Λήψη ονόματος προϊόντος από χρήστη

Βήμα 2

Εκτέλεση SELECT query στη βάση, για την εύρεση ονόματος προϊόντος στο μοντέλο.

Βήμα 3

Αν η βάση επιστρέψει το προϊόν, τότε εκτελείται SELECT query στην βάση για την εύρεση των πληροφοριών του προϊόντος, επιστρέφονται οι πληροφορίες και εμφανίζονται στον χρήστη, αλλιώς εμφανίζεται μήνυμα λάθους.

Βήμα 4

Αν ο χρήστης αναζητήσει προϊόν ανά κατηγορία, τότε κάνει κλικ σε κάποια από τις καρτέλες της Main Page

Βήμα 5

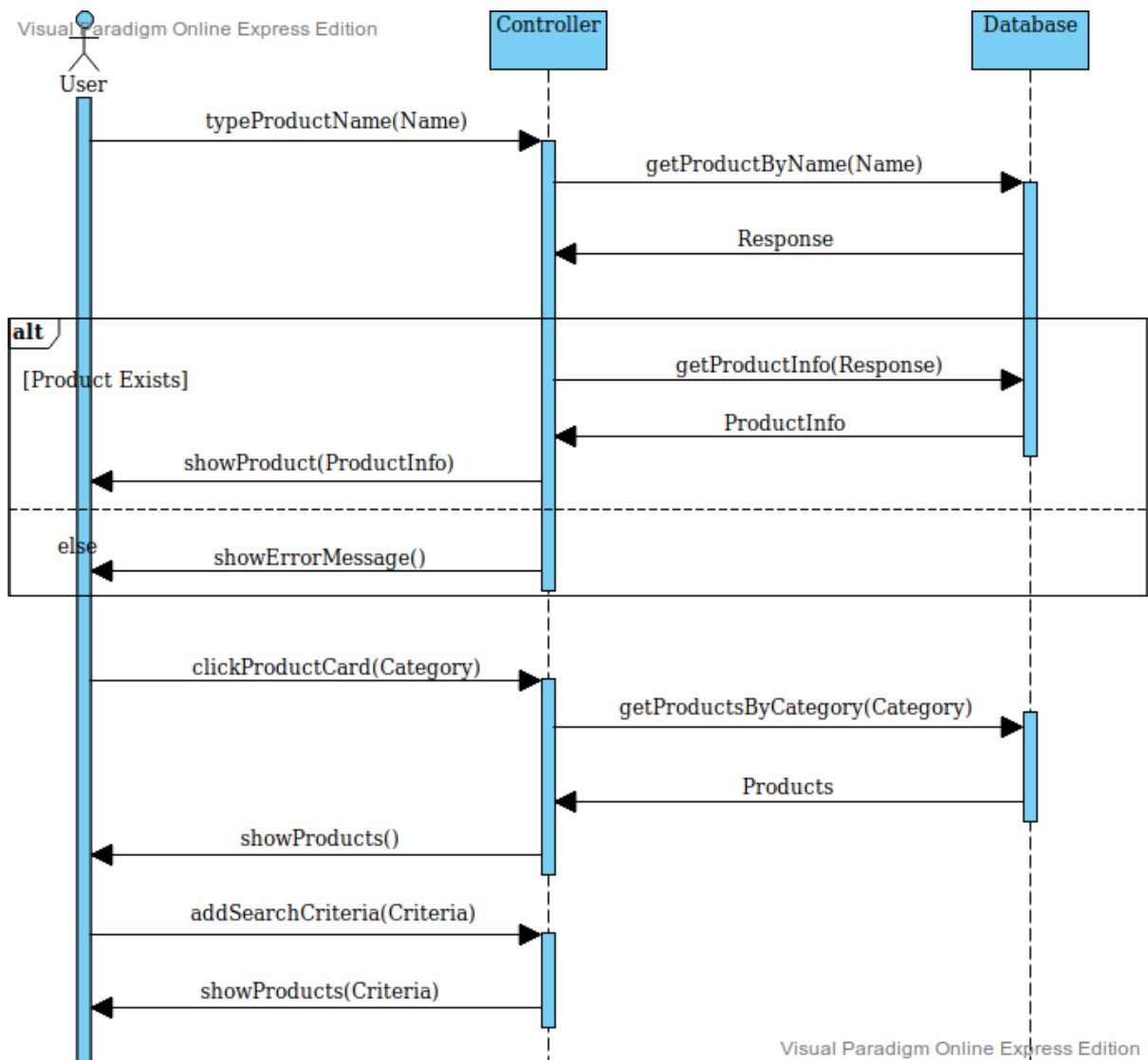
Ο Controller στέλνει SELECT query προς την βάση για να λάβει τα προϊόντα της συγκεκριμένης κατηγορίας.

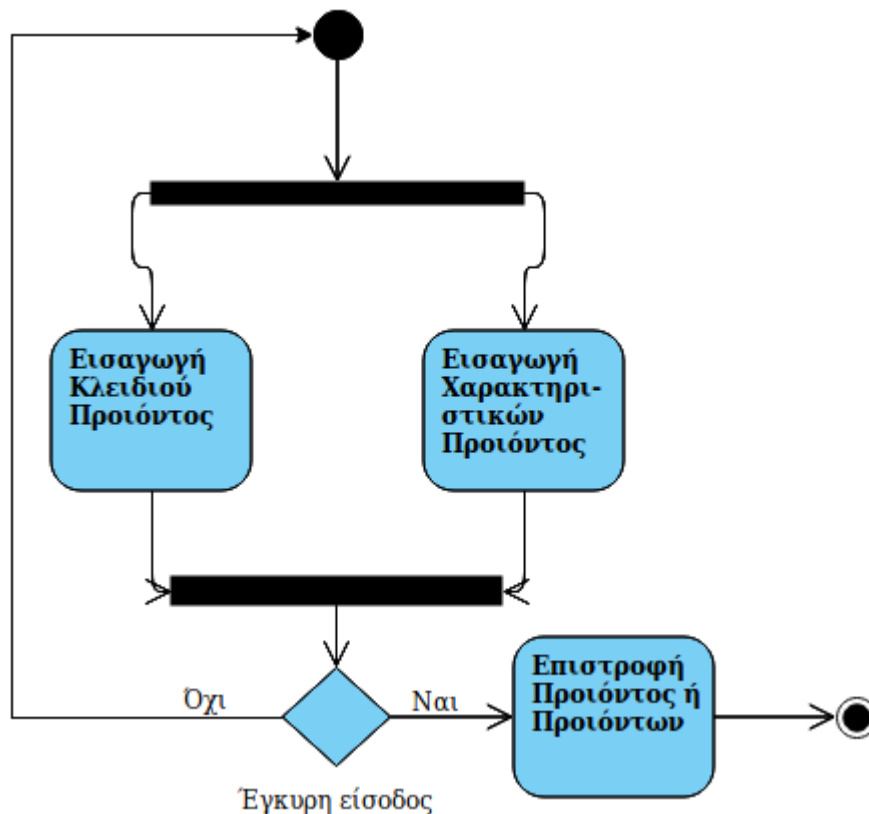
Βήμα 6

Η βάση επιστρέφει τα προϊόντα που ανήκουν στην κατηγορία και εμφανίζονται αυτά στον χρήστη

Βήμα 7

Αν ο χρήστης το επιθυμεί, εισάγει παραμέτρους για το φιλτράρισμα των εμφανιζόμενων προϊόντων και ο controller αναλαμβάνει επιτόπου την διαλογή τους.





### 3.2.2.7 Δεδομένα εξόδου

Τα δεδομένα εξόδου αυτής της περίπτωσης χρήσης είναι τα ευρισκόμενα προϊόντα, καθώς και το μήνυμα λάθους, σε περίπτωση που δεν βρεθεί το αναζητούμενο προϊόν. Το διάγραμμα αλληλουχίας που παρατέθηκε παραπάνω καλύπτει και τα δεδομένα εξόδου.

## 3.2.3 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 3: Εύρεση καταστήματος στον χάρτη

### 3.2.3.1 Χρήστες (ρόλοι) που εμπλέκονται

Κάθε χρήστης της εφαρμογής έχει τη δυνατότητα αναζήτησης καταστημάτων στο χάρτη.

### 3.2.3.2 Προϋποθέσεις εκτέλεσης

Δεν απαιτείται να πληρούνται κάποιες συγκεκριμένες συνθήκες για την εύρεση του καταστήματος.

### 3.2.3.3 Περιβάλλον εκτέλεσης

Στην Chearset και συγκεκριμένα στην σελίδα View Map απεικονίζονται σε χάρτη τα υποστηριζόμενα από την εφαρμογή καταστήματα, ενώ ο χρήστης έχει την δυνατότητα να πλοηγηθεί στην σελίδα `.../mapsearch` στην οποία εμφανίζονται τα καταστήματα της επιλογής του. Συγκεκριμένα, μέσω του google map API, που χρησιμοποιήθηκε ως το βασικό εργαλείο αναζήτησης των καταστημάτων, απεικονίζονται στον χάρτη ως ένα mark του google map, τα καταστήματα για τα οποία η εφαρμογή διαθέτει προϊόντα. Τέλος, η αξιολόγηση των δεδομένων αυτών γίνεται από τον controller, ενώ η αποθήκευση των συντεταγμένων των καταστημάτων γίνεται από το DBMS.

### 3.2.3.4 Δεδομένα εισόδου

Τα δεδομένα εισόδου, στην περίπτωση που ο χρήστης χρησιμοποιήσει το mapsearch, είναι το κατάστημα το οποίο επιθυμεί να επισκεφθεί. Επιπλέον, ως δεδομένο εισόδου ορίζεται και η κατηγορία προϊόντος με έξοδο την τοποθεσία του καταστήματος στο χάρτη (mark του google map).

### 3.2.3.5 Παράμετροι

Η τοποθεσία του χρήστη αποτελεί παράμετρο για την εύρεση των καταστημάτων και τη χρήση φίλτρων για την αναζήτηση αυτών.

### 3.2.3.6 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Βήμα 1

Λήψη ονόματος καταστήματος από χρήστη

Βήμα 2

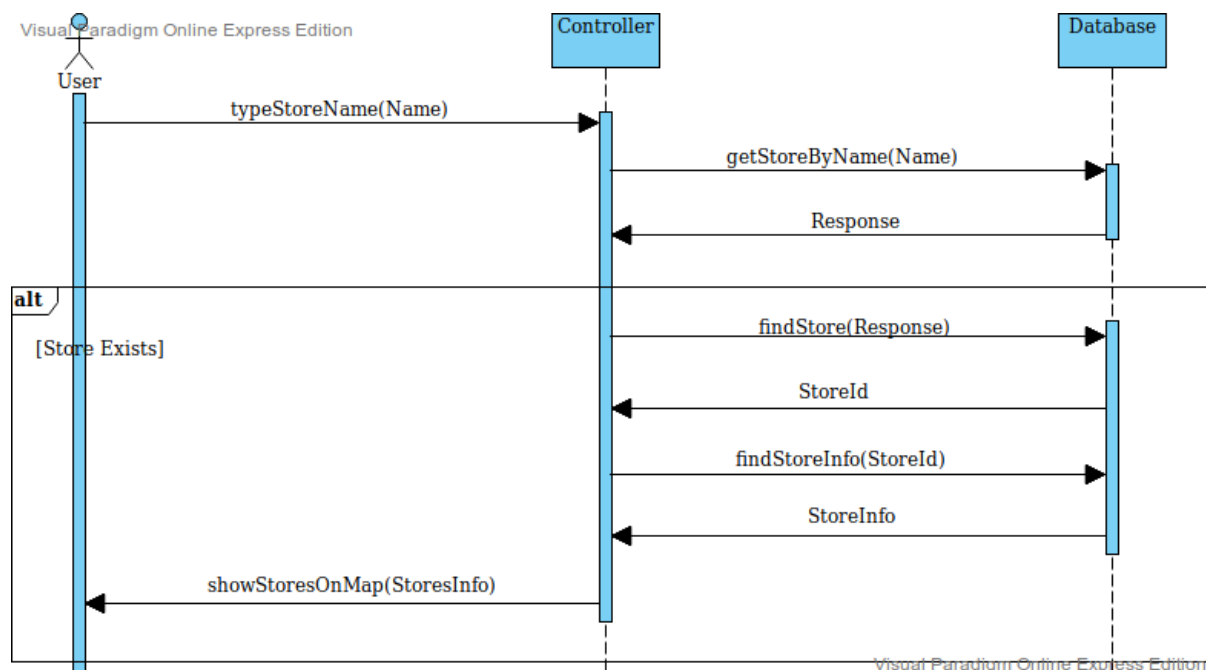
Ο Controller στέλνει SELECT query στη βάση, για την εύρεση ονόματος καταστήματος στο μοντέλο.

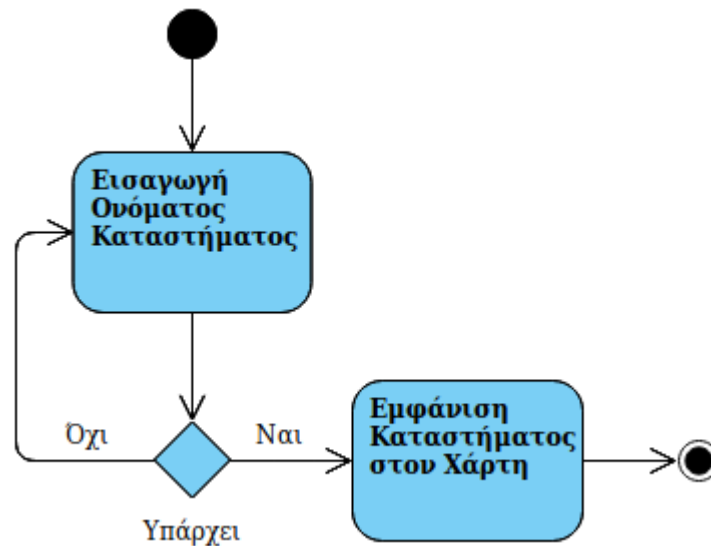
Βήμα 3

Αν η βάση επιστρέψει το κατάστημα, τότε εκτελείται SELECT query στην βάση για την εύρεση του καταστήματος, επιστρέφονται οι πληροφορίες και εμφανίζονται στον χρήστη, αλλιώς εμφανίζεται μήνυμα λάθους.

Βήμα 4

Η βάση επιστρέφει το κατάστημα με τις αντίστοιχες πληροφορίες (τοποθεσία, κτλ) και το εμφανίζει στον χάρτη .





### 3.2.3.7 Δεδομένα εξόδου

Τα δεδομένα εξόδου είναι τα καταστήματα, καθώς και το μήνυμα λάθους, σε περίπτωση που δεν βρεθεί το αναζητούμενο κατάστημα. Το διάγραμμα αλληλουχίας που παρατέθηκε παραπάνω καλύπτει και τα δεδομένα εξόδου.

## 3.3 Απαιτήσεις επιδόσεων

Τίτλος	Περιγραφή	Ποσοτική Μέτρηση Επιδόσεων
Χρόνος Απόκρισης του Συστήματος	Ο μέσος χρόνος απόκρισης του συστήματος για κάθε πιθανή αλληλεπίδραση χρήστη με αυτό	Ο μέσος χρόνος απόκρισης στα αιτήματα των χρηστών της εφαρμογής (εισαγωγή προϊόντος, αναζήτηση προϊόντος, εύρεση καταστήματος, στατιστικά κλπ) δεν θα πρέπει να ξεπερνάει τα 2 sec, έτσι ώστε να δίνεται η αίσθηση της άμεσης αλληλεπίδρασης του χρήστη με το σύστημα.

Φόρτος Εργασίας/Απόδοση	Ο συνολικός αριθμός χρηστών και μη που μπορεί να υποστηρίξει η εφαρμογή ταυτόχρονα	Θα πρέπει να λάβουμε υπόψιν πόσες φορές τη μέρα ένας χρήστης συνδέεται στην εφαρμογή και αποσυνδέεται από αυτήν κατά μέσο όρο, τον μέσο αριθμό αιτημάτων ανα χρήστη καθώς και τον μέσο χρόνο παραμονής ενός χρήστη στην εφαρμογή συμπεριλαμβανομένου και του “χρόνου σκέψης” του χρήστη, δηλαδή τον χρόνο που μεσολαβεί μεταξύ δύο διαδοχικών ενεργειών του χρήστη.
Επεκτασιμότητα	Η εφαρμογή θα πρέπει όσο αυξάνεται ο αριθμός των χρηστών να μπορεί να ανταποκριθεί στα αιτήματά τους, ενώ ταυτόχρονα να διατηρήσει τον χρόνο απόκρισης σε αυτά χαμηλό	

### 3.4 Απαιτήσεις οργάνωσης δεδομένων

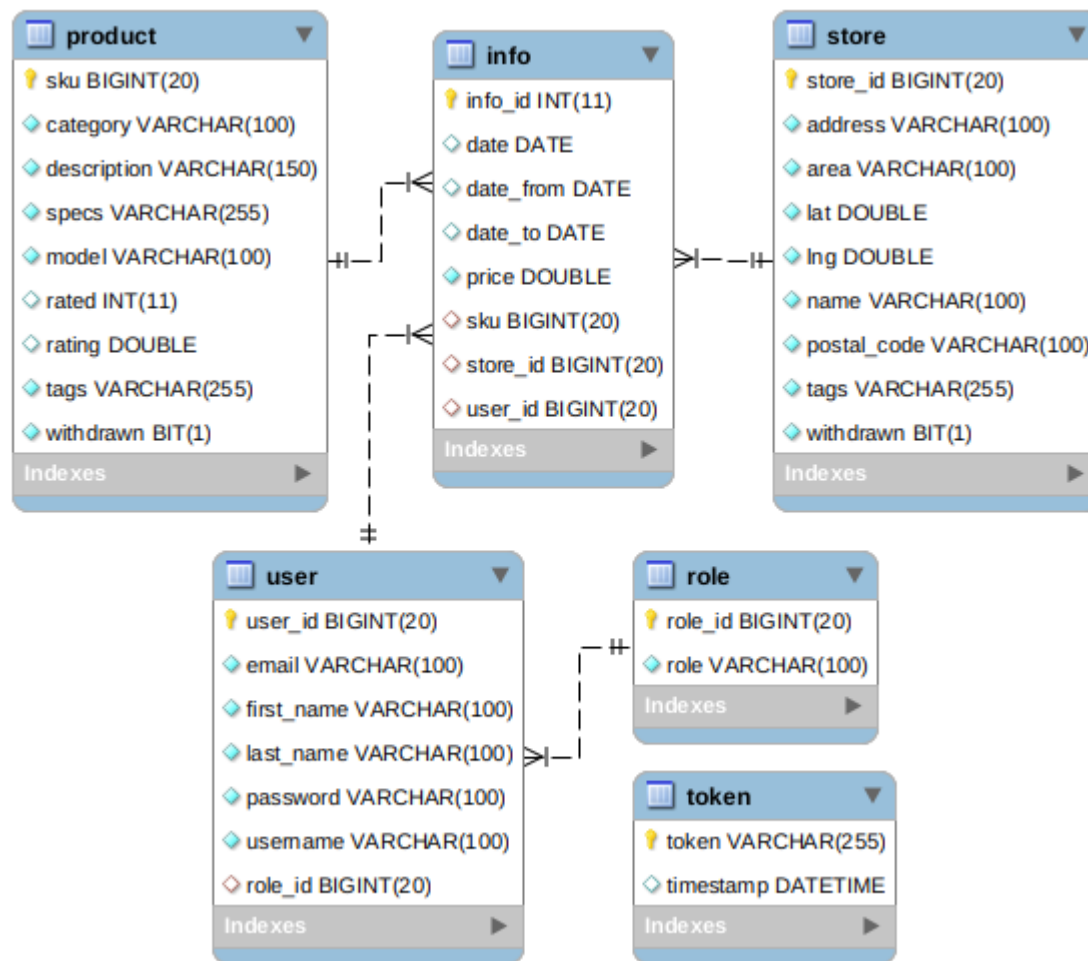
#### 3.4.1 Τεχνική περιγραφή των δεδομένων που διαχειρίζεται το λογισμικό και των σχετικών μετρικών φορτίου δεδομένων εισόδου, επεξεργασίας κ.λπ.

Τα δεδομένα εισόδου που λαμβάνουμε από τον χρήστη είναι μια φωτογραφία στην οποία αναγράφονται βασικές πληροφορίες για το προϊόν . Κάθε είδος προϊόντος έχει διαφορετικές απαιτήσεις ως προς τις πληροφορίες που πρέπει να παρέχονται. Το σύστημα μας είναι crowdsourcing και συνεπώς έχουμε αυξανόμενο μέγεθος δεδομένων. Έτσι είναι αδύνατο να προβλέψουμε το μέγεθος που θα φτάσει η εφαρμογή και οι πόροι που θα χρειαστούν για την υποστήριξη της (disk space) .

#### 3.4.2 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

Όλοι οι χρήστες ανεξάρτητα από το αν είναι εγγεγραμμένοι έχουν πρόσβαση στα δεδομένα των προϊόντων και στα στατιστικά που παρέχονται. Ωστόσο , οι χρήστες δεν μπορούν να επεξεργαστούν τα στοιχεία που έχουν καταχωρηθεί στη βάση δεδομένων της ιστοσελίδας (πληροφορίες για προϊόντα ,καταστήματα και διαγράμματα) παρά μόνο να τα διαβάσουν.

### 3.4.3 Μοντέλο δεδομένων (μοντέλο κλάσεων UML ή/και μοντέλο ER)



### 3.4.4 Προδιαγραφές ακεραιότητας δεδομένων

Εισαγωγή στοιχείων στην βάση δεδομένων μας έχουν μόνο οι εγγεγραμμένοι χρήστες οι οποίοι καλούνται να δώσουν ορισμένα στοιχεία για το προϊόν ανάλογα με την κατηγορία του καθώς και το κατάστημα στο οποίο βρίσκεται. Ο χρήστης είναι υποχρεωμένος να κάνει την εισαγωγή του προϊόντος από το κινητό του και για την εύρεση του καταστήματος που βρίσκεται χρησιμοποιείται η τοποθεσία του. Τα στοιχεία που καταχωρούνται από τον χρήστη ελέγχονται από τον controller και στην περίπτωση που είναι ελλιπή ή λανθασμένα ή το προϊόν υπάρχει ήδη στη βάση δεδομένων απορρίπτονται και δεν γίνεται η εισαγωγή στη βάση. Στη συνέχεια εμφανίζεται ενημερωτικό μήνυμα για την αποτυχία της καταχώρησης του προϊόντος.

### 3.4.5 Προδιαγραφές διατήρησης δεδομένων

Τα δεδομένα διατηρούνται στην βάση δεδομένων χωρίς χρονικούς περιορισμούς.

## 3.5 Περιορισμοί σχεδίασης

Στην περιγραφόμενη εφαρμογή ακολουθούνται τα Java community standards και τα Spring Boot best practices. Παρατίθενται οι ισότοποι που τα περιγράφουν.

- <https://www.oracle.com/technetwork/java/javase/overview/codeconvtoc-136057.html>

## 3.6 Λοιπές απαιτήσεις

### 3.6.1 Απαιτήσεις διαθεσιμότητας λογισμικού

Προκειμένου να επιτευχθεί υψηλή διαθεσιμότητα, τουλάχιστον 95%, της εφαρμογής (24/7/365) θα πρέπει όλα τα συστατικά της εφαρμογής (server, database) να είναι ανθεκτικά σε φθορές ή να έχουμε φροντίσει, σε περίπτωση που κάποιο από αυτά δυσλειτουργεί, για την γρήγορη αντικατάσταση ή επαναφορά του. Αυτό μπορεί να επιτευχθεί είτε έχοντας backup εξυπηρετητές είτε πολλαπλούς ενεργούς εξυπηρετητές. Επίσης, προφανώς η εφαρμογή θα πρέπει να είναι συνδεδεμένη στο Internet έτσι ώστε να καθίσταται δυνατή τόσο η χρησιμοποίηση της από κάθε χρήστη αλλά και η επικοινωνία μεταξύ των διαφόρων συστατικών του συστήματος. Έτσι μάλιστα η εφαρμογή θα μπορεί να πάρει αυτόματα τις συντεταγμένες της τοποθεσίας του χρήστη προκειμένου να πραγματοποιηθεί αναζήτηση με κριτήριο την απόσταση. Τέλος, απαιτείται στενή παρακολούθηση του συστατικών της εφαρμογής καθώς και συχνή συντήρηση αυτών, με όσο μικρότερο χρόνο απαιτείται για τη σωστή ολοκλήρωση της διαδικασίας.

### 3.6.2 Απαιτήσεις ασφάλειας

Προκειμένου να αντιμετωπιστούν τα διάφορα θέματα σχετικά με την ασφάλεια έχουν παρθεί τα κατάλληλα μέτρα. Αρχικά, η εφαρμογή χτίζεται πάνω στο πρωτόκολλο HTTPS, το οποίο φροντίζει για την ασφαλή μετάδοση των δεδομένων από την μεριά του client στο server και αντιστρόφως, μέσω διαφόρων αλγορίθμων κρυπτογράφησης. Στη συνέχεια, θα πρέπει να αναφέρουμε ότι η σύνδεση στην εφαρμογή γίνεται μόνο με έγκυρο συνδυασμό username-password ενώ η πιστοποίηση ενός χρήστη της εφαρμογής επιτυγχάνεται με τη χρήση token το οποίο έχει ημερομηνία λήξης. Έτσι σε περίπτωση υποκλοπής του, θα ζητηθεί ανανέωση αυτού μετά από σύντομο χρονικό διάστημα προκειμένου να συνεχιστεί η σύνδεση στην εφαρμογή. Ακόμη όλα οι κωδικοί πρόσβασης των χρηστών αποθηκεύονται στην βάση υφιστάμενοι κωδικοποίηση (Base64 Encoding), γεγονός που τους καθιστά μη ορατούς ακόμη και από τον διαχειριστή της εφαρμογής (το ότι μπορείς να κάνεις decode το Base64 από χιλιάδες site κλαιν). Τέλος, έχουμε φροντίσει να γράψουμε τον κώδικα έτσι ώστε να είναι αδύνατη η ανάκτηση οποιαδήποτε πληροφορίας από την βάση μέσω sql injections.

### 3.6.3 Απαιτήσεις συντήρησης

Όσον αφορά την διατηρησιμότητα της εφαρμογής ο κώδικας είναι χωρισμένος σε κομμάτια προκειμένου να διευκολύνεται η εποπτεία και η επέκτασή του. Για το λόγο αυτό χρησιμοποιήσαμε το MVC μοντέλο, έτσι ώστε να διακρίνουμε το σύνολο της εφαρμογής σε τρία μεγάλα μέρη : αυτό που βλέπει ο χρήστης (view), ότι έχει σχέση με δεδομένα (model) και την διαχείριση των αιτημάτων του χρήστη (controller). Τέλος, έχουν χρησιμοποιηθεί διάφορα tests για την αξιολόγηση της ορθής συμπεριφοράς του κώδικα.