

# Έγγραφο απαιτήσεων λογισμικού (SRS)

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

## Online Price Tracker (OPT)

### Εισαγωγή

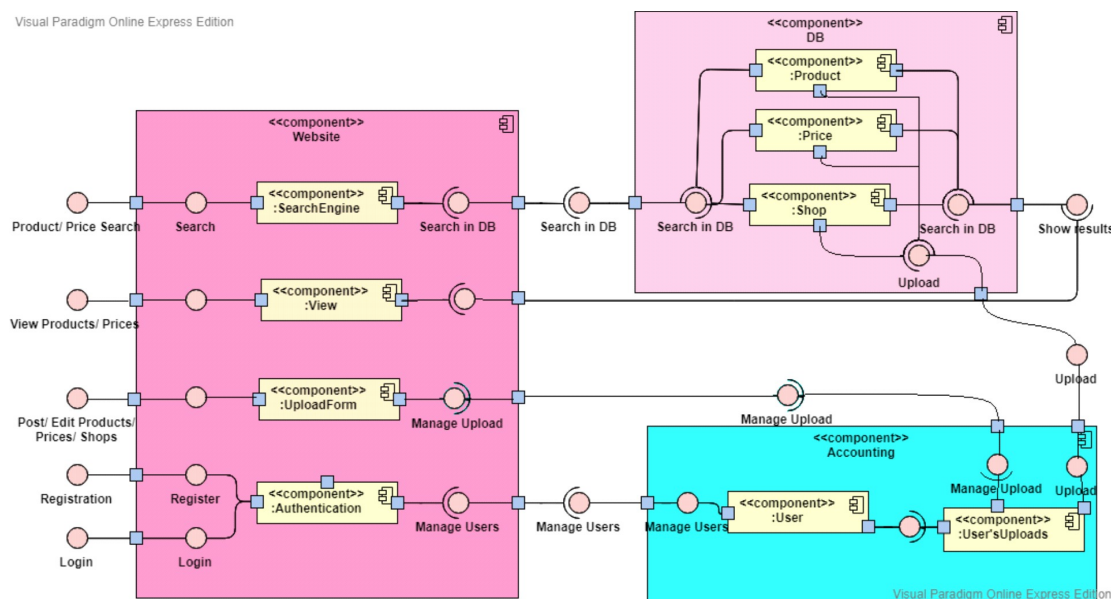
#### 1.1 Εισαγωγή: σκοπός του λογισμικού

##### Οριοθέτηση του σκοπού του συστήματος

Το σύστημα που συνιστά το παρατηρητήριο τιμών έχει ως σκοπό τη διευκόλυνση των καταναλωτών στην προσπάθειά τους να λαμβάνουν ορθές, οικονομικά συμφέρουσες, αγοραστικές αποφάσεις. Πιο συγκεκριμένα, το λογισμικό μας αποβλέπει στην έγκυρη ενημέρωση των χρηστών του για τις τιμές διαφόρων προϊόντων σε καταστήματα του τόπου διαμονής τους προσφέροντάς τους παράλληλα κατάλληλα εργαλεία για τη σύγκριση των τιμών αυτών. Τα προϊόντα και τα καταστήματα που θα καλύπτονται από το λογισμικό μας οφείλουν να αποτελούν σημαντικό τμήμα των αγοραστικών επιλογών της περιοχής δραστηριοποίησής μας. Ακόμη, η εφαρμογή έχει σαν στόχο την όσο το δυνατόν πιο γρήγορη ενημέρωση των δεδομένων της, μέσω των εθελοντών χρηστών της, ώστε αυτά να προσαρμόζονται ταχύτατα σε τυχόν τιμολογιακές αλλαγές ή μεταβολές των προσφορών των διαφόρων καταστημάτων. Έτσι η εφαρμογή μας θα επιτυγχάνει το στόχο της, θα βοηθάει δηλαδή ουσιαστικά τους καταναλωτές να αποφασίζουν ποια προϊόντα πρέπει να προμηθευτούν από ποια καταστήματα, ώστε να εξασφαλίζουν το ελάχιστο δυνατό κόστος, επιτρέποντάς τους έτσι να σχεδιάζουν αποτελεσματικά τις αγορές τους.

#### 1.2 Επισκόπηση του λογισμικού

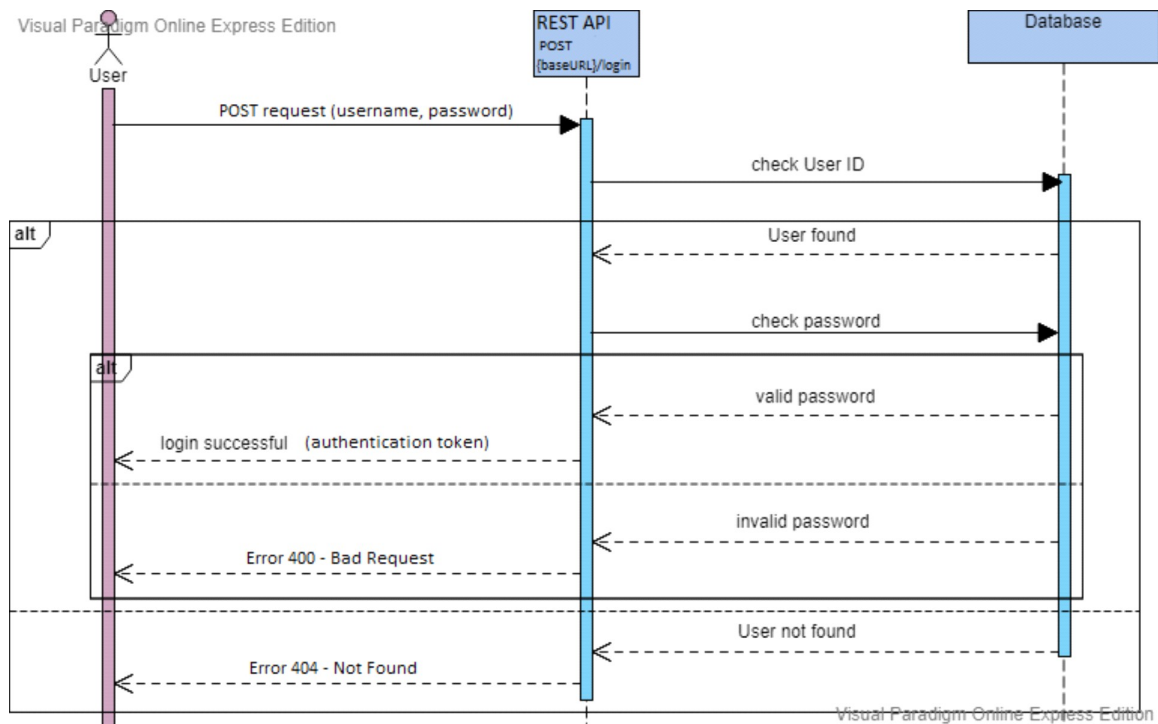
Γενική περιγραφή με χρήση διαγράμματος UML . Οι διεπαφές αναφέρονται μόνο ως τίτλοι ή/και σε διάγραμμα.



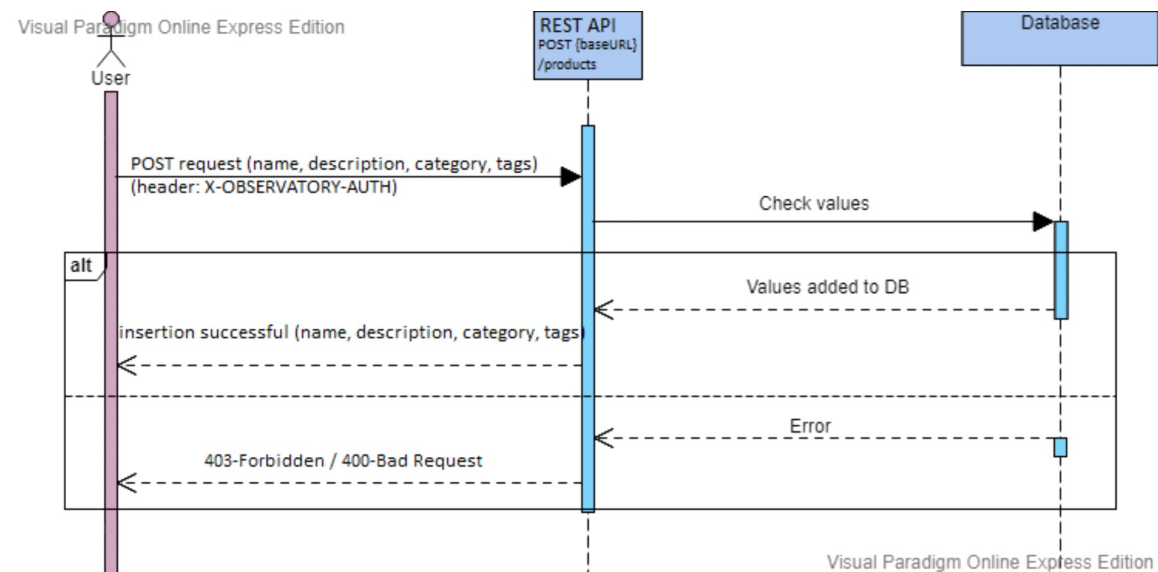
#### 1.3.1 Διεπαφές με εξωτερικά συστήματα και εφαρμογές λογισμικού

Προδιαγραφή διεπαφών με εξωτερικά συστήματα και λογισμικό, με αναφορά σε πρότυπα ανταλλαγής δεδομένων και κλήσης υπηρεσιών. Χρήση διαγραμμάτων UML.

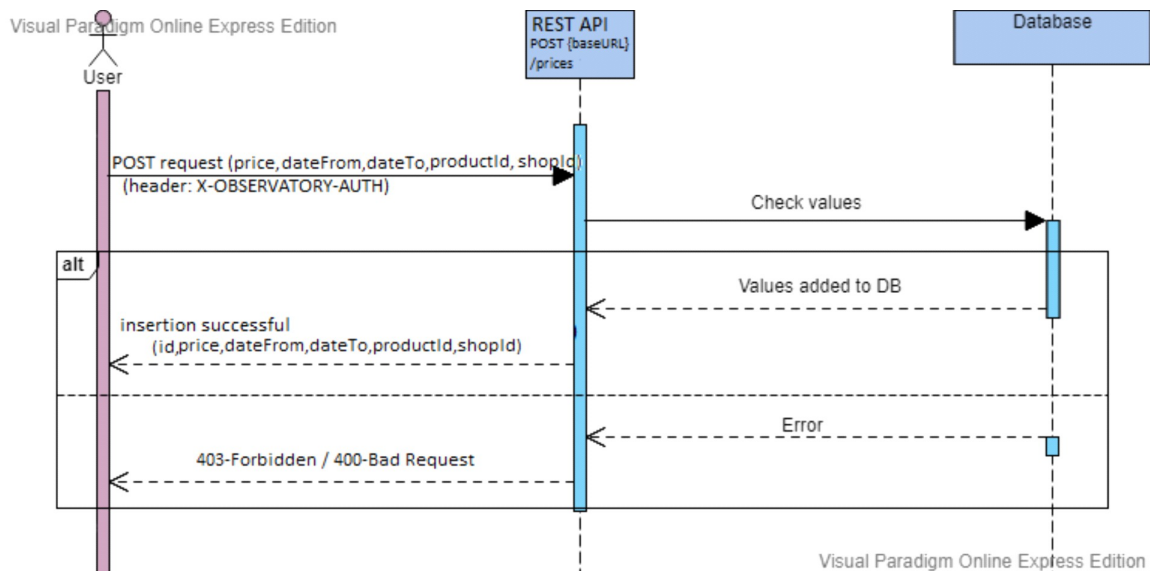
### POST {baseUrl}/login



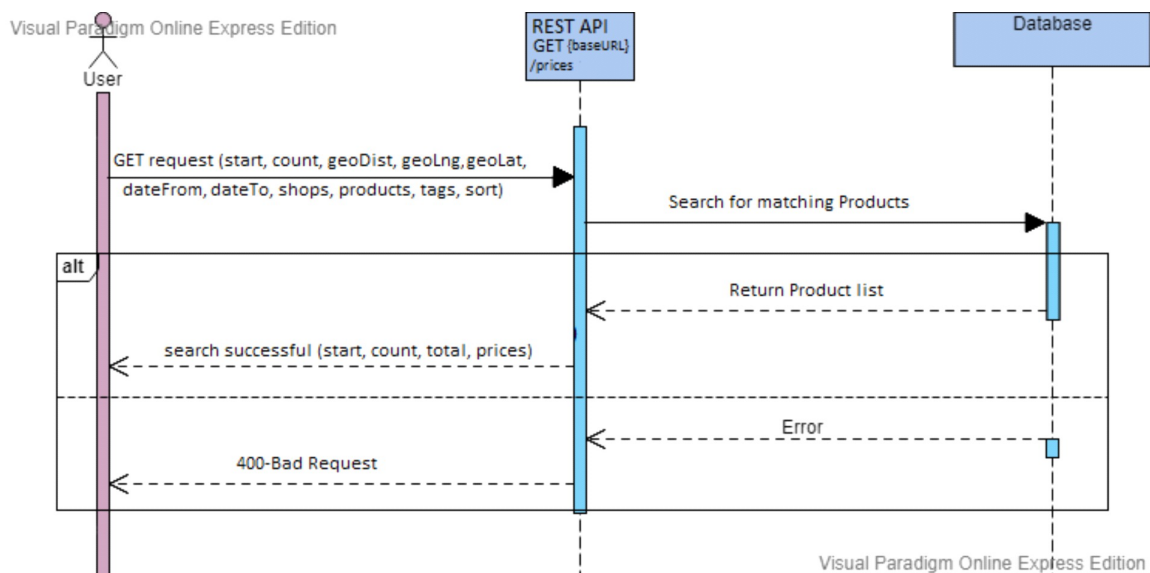
### POST {baseUrl}/products



### POST {baseUrl}/prices

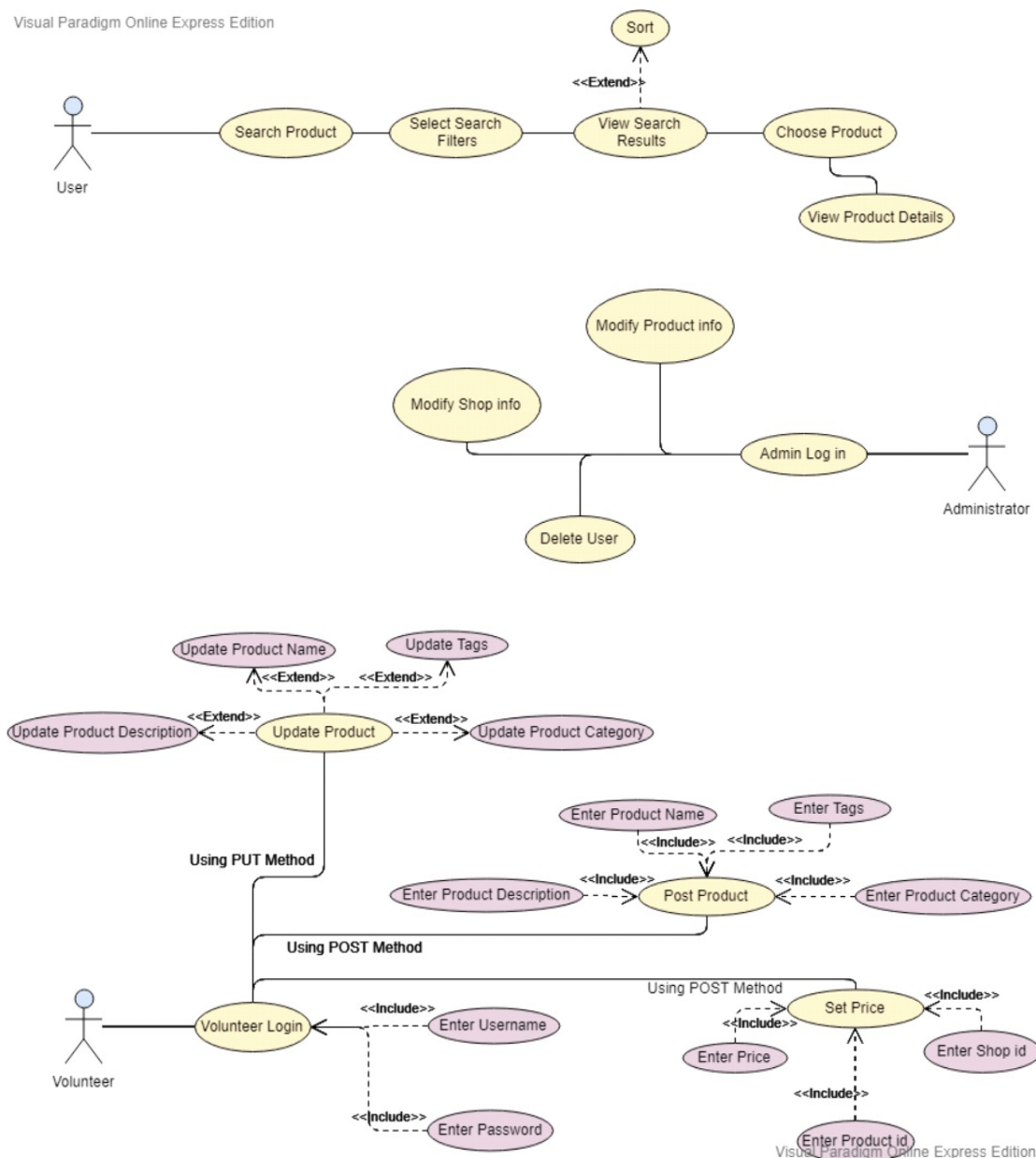


### GET {baseUrl}/prices



## 1.3.2 Διεπαφές με το χρήστη

Προδιαγραφή διεπαφών με το χρήστη. Μοντέλο Use Case (UML).



### 1.3.3 Διεπαφές με υλικό

Προδιαγραφή διεπαφών με υλικό (εφόσον απαιτείται, πχ αναγνώστες κ.ά.)

ΝΑ ΜΗΝ ΣΥΜΠΛΗΡΩΘΕΙ ΑΝ ΔΕΝ ΑΠΑΙΤΕΙΤΑΙ

Θα μπορούσε να γίνει χρήση της διεπαφής με το υλικό των αναγνωστών ώστε να παίρνουμε την τοποθεσία τους μέσω συστήματος GPS. Ωστόσο, επιλέξαμε να υλοποιήσουμε την εφαρμογή με τρόπον που οι αναγνώστες ανεξαρτήτως συσκευής να εισάγουν κατά την αναζήτηση την περιοχή που τους ενδιαφέρει για την αναζήτηση. Συνεπώς δεν απαιτείται κάποια διεπαφή υλικού.

### 1.3.4 Διεπαφές επικοινωνιών

Προδιαγραφή διεπαφών επικοινωνιών (αφορά στοιχεία λογισμικού που υλοποιούν τέτοιες διεπαφές, εφόσον υπάρχουν)

ΝΑ ΜΗΝ ΣΥΜΠΛΗΡΩΘΕΙ ΑΝ ΔΕΝ ΑΠΑΙΤΕΙΤΑΙ

Οι χρήστες μπορούν να επικοινωνούν μαζί μας (π.χ. μέσω email) προκειμένου να καταχωρούν feedback για την ποιότητα των παρεχόμενων υπηρεσιών ή για διαμαρτυρία για τα περιεχόμενα της σελίδας μας.

# Αναφορές - πηγές πληροφοριών

*Αναφορά σε πηγές πληροφοριών στο μέτρο της αναγκαιότητας για την κατανόηση του συστήματος*

*ΝΑ ΜΗΝ ΣΥΜΠΛΗΡΩΘΕΙ ΑΝ ΔΕΝ ΑΠΑΙΤΕΙΤΑΙ*

Δεν υπήρξαν εξωτερικές πηγές πληροφοριών.

## Προδιαγραφές απαιτήσεων λογισμικού

### 3.1 Εξωτερικές διεπαφές

*Λεπτομερής τεχνική προδιαγραφή των διεπαφών που αναφέρονται στην ενότητα 1.3.1.*

*Προαιρετική χρήση λογισμικού προτυποποίησης διεπαφών χρήστη (mock-up).*

POST {baseUrl}/login

Διαπιστεύει το χρήστη.

Λαμβάνει στο body τμήμα της αίτησης τις ακόλουθες παραμέτρους:

username

password

Επιστρέφει ένα authentication token.

POST {baseUrl}/products

Δημιουργεί νέο προϊόν και το εισάγει στη Βάση Δεδομένων.

Ο χρήστης πρέπει να είναι διαπιστευμένος.

Η αίτηση περιέχει στο body τμήμα της τα στοιχεία για το προϊόν που θα προστεθεί (name, description, category, tags).

POST {baseUrl}/prices

Προσθέτει μία τιμή για ένα προϊόν.

Η αίτηση περιέχει τις ακόλουθες τιμές στο body τμήμα της:

price: Double

dateFrom: Date (yyyy-mm-dd)

dateTo: Date (yyyy-mm-dd)

productId: Long

shopId: Long

Ως αποτέλεσμα επιστρέφονται τα δεδομένα του προϊόντος:

id: Long

price: Double

dateFrom: Date (yyyy-mm-dd)

dateTo: Date (yyyy-mm-dd)

productId: Long

shopId: Long

GET {baseUrl}/prices

Αναζητεί προϊόντα σε διάφορα καταστήματα στη Βάση Δεδομένων σύμφωνα με ορισμένα κριτήρια.

Στο query του αιτήματος πρέπει να βρίσκονται παράμετροι από τις ακόλουθες:

start: Integer (default=0)

count: Integer (default=20)

geoDist: Integer (distance in km from interest point)

geoLng: Double

geoLat: Double

dateFrom: Date (yyyy-mm-dd)

dateTo: Date (yyyy-mm-dd)

shops: List<String> (list of shopIds)

products: List<String> (list of productIds)

tags: List<String>

sort: List<String> (combination of geo.dist, price and date; default:price|ASC)

Σημείωση: Τα geoDist, geoLng, geoLat πρέπει να λαμβάνουν και τα τρία τιμή ή κανένα τους, αλλιώς έχουμε σφάλμα.

Επιστρέφονται τα αποτελέσματα στην ακόλουθη μορφή:

start: Integer

count: Integer

total: Long

prices: List<Price>

Κάθε Price είναι διαμορφωμένο ως εξής:

price: Double

date: Date (yyyy-mm-dd)

productName: String

productId: Long

productTags: List<String>

shopId: String

shopName: String

shopTags: List<String>

shopAddress: String

shopDist: Integer

Γενικές σημειώσεις:

Οι υπόλοιπες λειτουργίες του REST API έχουν παρόμοια λειτουργικότητα.

Στις περιπτώσεις που αναφέρεται ότι ο χρήστης πρέπει να είναι διαπιστευμένος εννοείται ότι πρέπει στο custom HTTP header να περιλαμβάνει το authentication token που έλαβε κατά τη διαπίστευση με όνομα X-OBSERVATORY-AUTH.

Υποστηρίζεται και XML και JSON για τα αποτελέσματα. Το τι επιστρέφεται εξαρτάται από την τιμή του format στο url (json|xml, default:json).

Τα σφάλματα που μπορεί να επιστραφούν είναι τα εξής: 404-Not Found (μη ύπαρξη ζητούμενου πόρου), 401-Not Authorized (η ενέργεια που προσπαθεί να εκτελέσει ο χρήστης απαιτεί επιπλέον δικαιοδοσίες), 403-Forbidden (πρόβλημα απαιτήσεων διαπίστευσης), 400-Bad Request (μη έγκυρες παράμετροι).

## 3.2 Λειτουργίες: περιπτώσεις χρήσης

Λεπτομερής προδιαγραφή των λειτουργιών του λογισμικού σε επίπεδο περιπτώσεων χρήσης.

Για κάθε μία λειτουργία δίνονται τα ακόλουθα.

ΟΜΑΔΕΣ 5 ΑΤΟΜΩΝ: 2-3 περιπτώσεις χρήσης

ΟΜΑΔΕΣ 6 ΑΤΟΜΩΝ: 3-4 περιπτώσεις χρήσης

ΟΜΑΔΕΣ 7 ΑΤΟΜΩΝ: 4-5 περιπτώσεις χρήσης

### 3.2.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 1: Σύνδεση χρήστη

#### 3.2.1.1 Χρήστες (ρόλοι) που εμπλέκονται

Αναφορά στους ρόλους που αφορά η περίπτωση χρήσης

Οι ρόλοι που σχετίζονται με αυτήν την περίπτωση χρήσης είναι οι χρήστες (αναγνώστες ή συμμετέχοντες στο crowdsourcing).

#### 3.2.1.2 Προϋποθέσεις εκτέλεσης

Καταγραφή των συνθηκών που πρέπει να ισχύουν ώστε να μπορεί να εκτελεστεί η περίπτωση χρήσης

Προϋπόθεση για την εκτέλεση αυτής της περίπτωσης χρήσης είναι ο χρήστης που επιθυμεί να συνδεθεί να έχει επισκεφτεί τη σελίδα login του λογισμικού μας.

#### 3.2.1.3 Περιβάλλον εκτέλεσης

Αναφορά στο περιβάλλον στο οποίο εκτελείται η περίπτωση χρήσης. Πχ "διαδικτυακή διεπαφή χρήστη", "DBMS" κλπ

Το περιβάλλον εκτέλεσης αυτής της περίπτωσης χρήσης είναι η διαδικτυακή διεπαφή χρήστη.

#### 3.2.1.4 Δεδομένα εισόδου

Καταγραφή δεδομένων εισόδου και εξόδου και συνθηκών εγκυρότητας αυτών.

Δεδομένα εισόδου αποτελούν το User ID καθώς και το password που εισάγει ο χρήστης προς σύνδεση μέσω της διεπαφής της ιστοσελίδας στο σύστημα μας. Συνθήκη εγκυρότητας αυτών αποτελεί η ύπαρξη στη Βάση Δεδομένων μας μίας εγγραφής με τιμές στα αντίστοιχα πεδία ίσες με αυτές που εισήγαγε ο χρήστης. Δεν έχουμε δεδομένα εξόδου, παρά μόνο ένα μήνυμα ανακοίνωσης επιτυχίας/αποτυχίας.

#### 3.2.1.5 Παράμετροι

Καταγραφή παραμέτρων και συνθηκών εγκυρότητας αυτών

**Δεν υπάρχουν καθορισμένες παράμετροι για τη συγκεκριμένη περίπτωση χρήσης.**

#### 3.2.1.6 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).

Βήμα 1: Ο χρήστης εισάγει το User ID και το password μέσω της κατάλληλης φόρμας στη σελίδα σύνδεσης.

Βήμα 2: Ο χρήστης υποβάλει τα παραπάνω στοιχεία προς εξέταση στη σελίδα σύνδεσης.

Βήμα 3: Η σελίδα σύνδεσης μεταβιβάζει το User ID στη Βάση Δεδομένων.

Βήμα 4: Η Βάση Δεδομένων ελέγχει αν υπάρχει εγγραφή χρήστη με User ID ίδιο με αυτό που της έστειλε η σελίδα σύνδεσης. Αν είναι συνεχίζουμε στην περίπτωση (α), αλλιώς στην περίπτωση (β).

Βήμα 5α: Η Βάση Δεδομένων αποστέλει στη σελίδα σύνδεσης μήνυμα που την ενημερώνει ότι

βρέθηκε χρήστης με το User ID που της δόθηκε.

Βήμα 6α: Η σελίδα σύνδεσης μεταβιβάζει το password στη Βάση Δεδομένων.

Βήμα 7α: Η Βάση Δεδομένων εξετάζει αν το password που έλαβε είναι ίδιο με αυτό που βρίσκεται στην εγγραφή που εντόπισε προηγουμένως (μέσω του User ID). Αν είναι συνεχίζουμε στην περίπτωση (γ), αλλιώς στην περίπτωση (δ).

Βήμα 8γ: Η Βάση Δεδομένων αποστέλει στη σελίδα σύνδεσης μήνυμα που την ενημερώνει ότι το password που της δόθηκε είναι έγκυρο.

Βήμα 9γ: Η σελίδα σύνδεσης ενημερώνει το χρήστη ότι η σύνδεση στο σύστημα πραγματοποιήθηκε με επιτυχία.

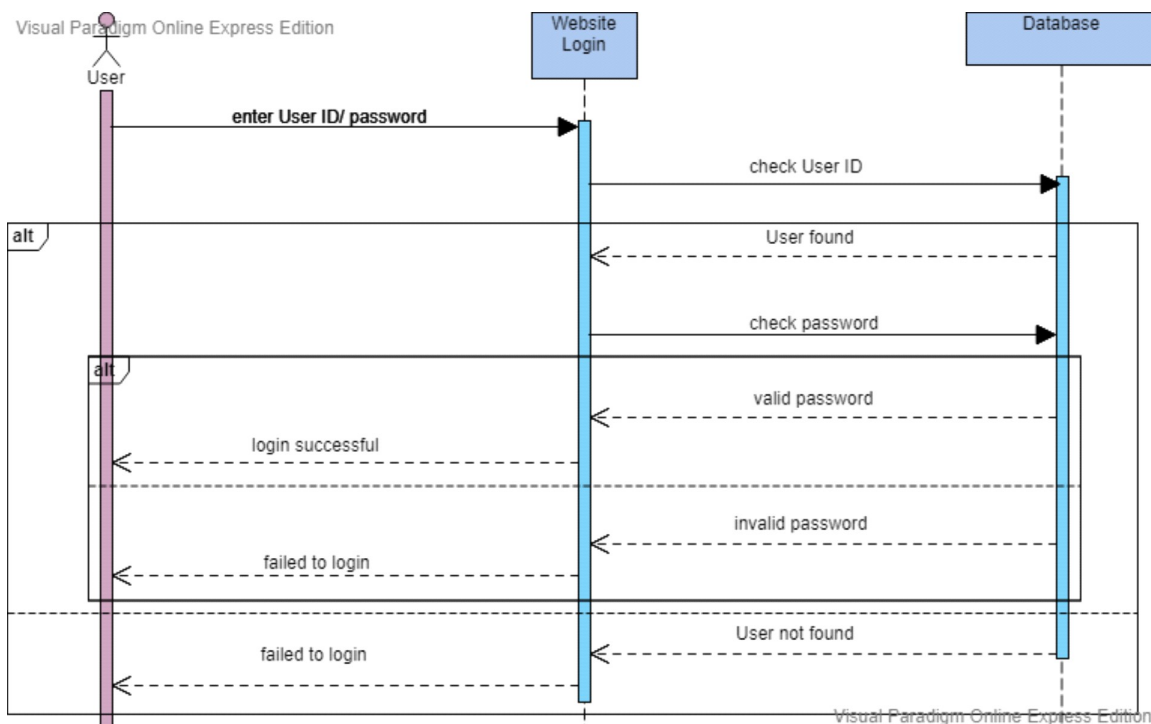
Βήμα 5β: Η Βάση Δεδομένων αποστέλει στη σελίδα σύνδεσης μήνυμα που την ενημερώνει ότι δε βρέθηκε χρήστης με το User ID που της δόθηκε.

Βήμα6β: Η σελίδα σύνδεσης ενημερώνει το χρήστη ότι η σύνδεση στο σύστημα απέτυχε.

Βήμα 8δ: Η Βάση Δεδομένων αποστέλει στη σελίδα σύνδεσης μήνυμα που την ενημερώνει το password που της δόθηκε δεν είναι έγκυρο.

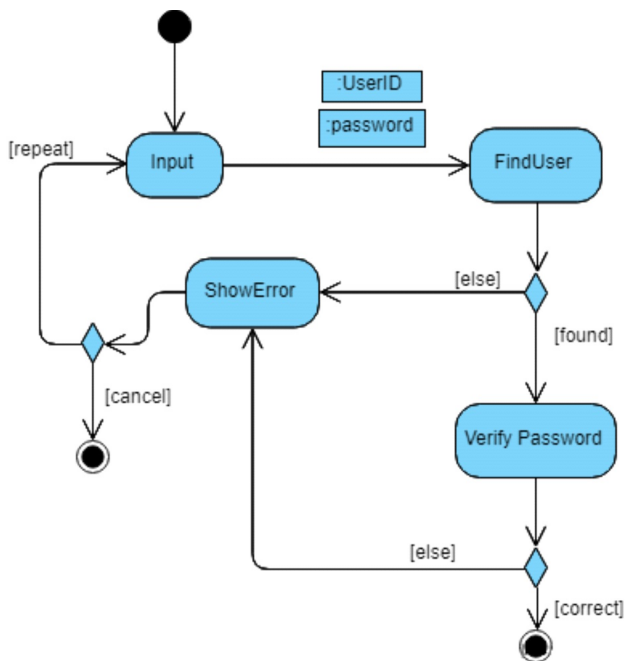
Βήμα9δ: Η σελίδα σύνδεσης ενημερώνει το χρήστη ότι η σύνδεση στο σύστημα απέτυχε.

#### UML διάγραμμα αλληλουχίας (sequence diagram)



#### UML διάγραμμα δραστηριοτήτων (activity diagram)





OnlinePriceTracker/routes/user.js

### 3.2.1.7 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)

Δεν υπάρχουν δεδομένα εξόδου. Ο χρήστης απλά ενημερώνεται για την επιτυχία ή την αποτυχία της σύνδεσής του στο σύστημα, όπως περιγράφεται στην προηγούμενη υποενότητα.

### 3.2.1.8 Παρατηρήσεις

Ο,τι δεν εντάσσεται στα προηγούμενα, εφόσον υπάρχει

Μία παρατήρηση είναι ότι στον πίνακα της Βάσης Δεδομένων με τα στοιχεία χρηστών, το πεδίο User ID αποτελεί κλειδί. Αυτό έχει ως συνέπεια οποιοδήποτε User ID και να εισάγει ένας χρήστης, αυτό να εμφανίζεται το πολύ σε μία εγγραφή στον πίνακα αυτόν.

Μία δεύτερη παρατήρηση είναι ότι τα password μπορεί να επιλεγεί να είναι αποθηκευμένα στη Βάση Δεδομένων σε κρυπτογραφημένη μορφή για λόγους ασφαλείας (π.χ. αποφυγή κινδύνων διαρροής δεδομένων από τη Βάση Δεδομένων). Στην περίπτωση αυτή η σελίδα σύνδεσης οφείλει να κρυπτογραφήσει το password που της δόθηκε από το χρήστη με βάση το πρωτόκολλο κρυπτογράφησης που ενδεομένως έχει επιλεγεί, προτού το αποστείλει στη Βάση Δεδομένων για τον έλεγχο της εγκυρότητάς του.

## 3.2.2 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 2: Ενημέρωση τιμής προϊόντος

### 3.2.2.1 Χρήστες (ρόλοι) που εμπλέκονται

Αναφορά στους ρόλους που αφορά η περίπτωση χρήσης

Αυτή η περίπτωση χρήσης αφορά το ρόλο του χρήστη που συμμετέχει στο crowdsourcing (εθελοντής).

### 3.2.2.2 Προϋποθέσεις εκτέλεσης

*Καταγραφή των συνθηκών που πρέπει να ισχύουν ώστε να μπορεί να εκτελεστεί η περίπτωση χρήσης*

Για να μπορεί να εκτελεστεί με επιτυχία αυτή η περίπτωση χρήσης με επιτυχία πρέπει ο χρήστης να έχει λογαριασμό στην εφαρμογή μας και να επιθυμεί να τροποποιήσει την τιμή ενός προϊόντος σε ένα κατάστημα.

### **3.2.2.3 Περιβάλλον εκτέλεσης**

*Αναφορά στο περιβάλλον στο οποίο εκτελείται η περίπτωση χρήσης. Πχ "διαδικτυακή διεπαφή χρήστη", "DBMS" κλπ*

Περιβάλλον εκτέλεσης αυτής της περίπτωσης χρήσης αποτελεί η διαδικτυακή διεπαφή χρήστη του λογισμικού μας.

### **3.2.2.4 Δεδομένα εισόδου**

*Καταγραφή δεδομένων εισόδου και εξόδου και συνθηκών εγκυρότητας αυτών.*

Δεδομένα εισόδου αποτελεί το User ID καθώς και το password του συμμετέχοντα στ crowdsourcing, καθώς επίσης (και κυρίως) το Shop ID (αναγνωριστικό καταστήματος), το Product ID (αναγνωριστικό προϊόντος) και η νέα τιμή του προϊόντος. Η συνθήκη εγκυρότητας των δύο πρώτων αναλύθηκε στην προηγούμενη περίπτωση χρήσης. Το Shop ID για να είναι έγκυρο πρέπει να έχει καταχωρηθεί ως αναγνωριστικό κάποιου καταστήματος στη Βάση Δεδομένων του συστήματός μας. Ομοίως, το Product ID πρέπει να έχει συσχετιστεί στη Βάση Δεδομένων μας με κάποιο προϊόν του καταστήματος αυτού. Η νέα τιμή μπορεί γενικά να θεωρηθεί έγκυρη, όντας προφανώς κάποιος θετικός πραγματικός αριθμός, ωστόσο διασταύρησής της με ιστορικές τιμές του συγκεκριμένου προϊόντος ίσως ήταν χρήσιμη για την αύξηση της αξιοπιστίας του crowdsourcing μέσω φιλτραρίσματος αλλαγών. Δεδομένο εξόδου αποτελεί η νέα τιμή του προϊόντος μετά από την ενημέρωση.

### **3.2.2.5 Παράμετροι**

*Καταγραφή παραμέτρων και συνθηκών εγκυρότητας αυτών, εφόσον υπάρχουν παράμετροι.*

Παράμετρος για τη συγκεκριμένη περίπτωση χρήσης θα μπορούσε να θεωρηθεί το ιστορικό τιμών του συγκεκριμένου προϊόντος, το οποίο θα μπορούσε να χρησιμοποιηθεί για την απόφαση του αν η νέα προτεινόμενη τιμή είναι έγκυρη ή όχι.

### **3.2.2.6 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).*

Βήμα 1: Ο χρήστης-εθελοντής εισάγει το User ID και το password του στη φόρμα της σελίδας σύνδεσής μας και τα υποβάλει στην ιστοσελίδα.

Βήμα 2: Η σελίδα μεταβιβάζει τα στοιχεία αυτά στη Βάση Δεδομένων, ζητώντας της να επιχειρήσει ταυτοποίηση του χρήστη.

Βήμα 3: Σε περίπτωση που η Βάση Δεδομένων βρει εγγραφή που έχει τα στοιχεία που της δόθηκαν ως τιμές στα αντίστοιχα πεδία συνεχίζουμε με την περίπτωση (α), αλλιώς με την περίπτωση (β).

Βήμα 4α: Η Βάση Δεδομένων ενημερώνει την ιστοσελίδα ότι ο χρήστης πιστοποιήθηκε με επιτυχία.

Βήμα 5α: Η ιστοσελίδα ενημερώνει το χρήστη ότι η σύνδεσή του στο σύστημα πραγματοποιήθηκε με επιτυχία.

Βήμα 6α: Ο χρήστης στέλνει αίτημα δημοσίευσης (POST) τιμής στην ιστοσελίδα.

Βήμα 7α: Η ιστοσελίδα ζητάει από το χρήστη να καθορίσει τα στοιχεία εισόδου για την ενημέρωση.

Βήμα 8α: Ο χρήστης αποστέλει στην ιστοσελίδα το Product ID, το Shop ID και τη νέα τιμή του προϊόντος.

Βήμα 9α: Η ιστοσελίδα αποστέλει τα παραπάνω στοιχεία στη Βάση Δεδομένων για έλεγχο.

Βήμα 10α: Η Βάση Δεδομένων ελέγχει αν ο συνδυασμός Product ID και Shop ID ανταποκρίνεται σε κάποια εγγραφή της. Αν αυτό αληθεύει συνεχίζουμε με την περίπτωση (γ), αλλιώς με την περίπτωση (δ).

Βήμα 11γ: Η Βάση Δεδομένων ενημερώνει την ιστοσελίδα ότι τα δεδομένα που της δόθηκαν είναι έγκυρα.

Βήμα 12γ: Η ιστοσελίδα ενημερώνει το χρήστη ότι η τιμή του προϊόντος ενημερώθηκε με επιτυχία.

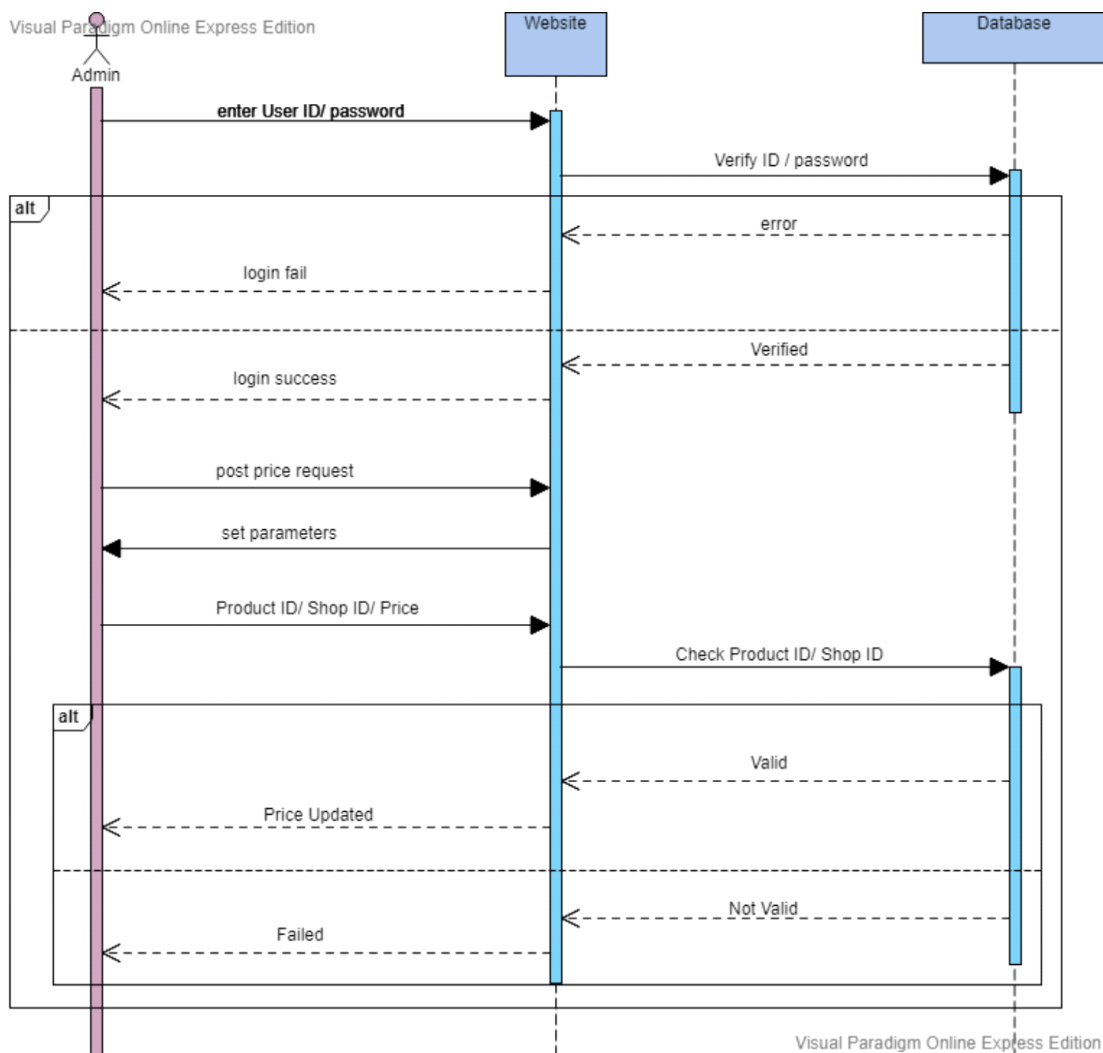
Βήμα 11δ: Η Βάση Δεδομένων ενημερώνει την ιστοσελίδα ότι τα δεδομένα που της δόθηκαν δεν ήταν έγκυρα.

Βήμα 12δ: Η ιστοσελίδα ανακοινώνει στο χρήστη την αποτυχία της ενημέρωσης της τιμής που ζητήθηκε.

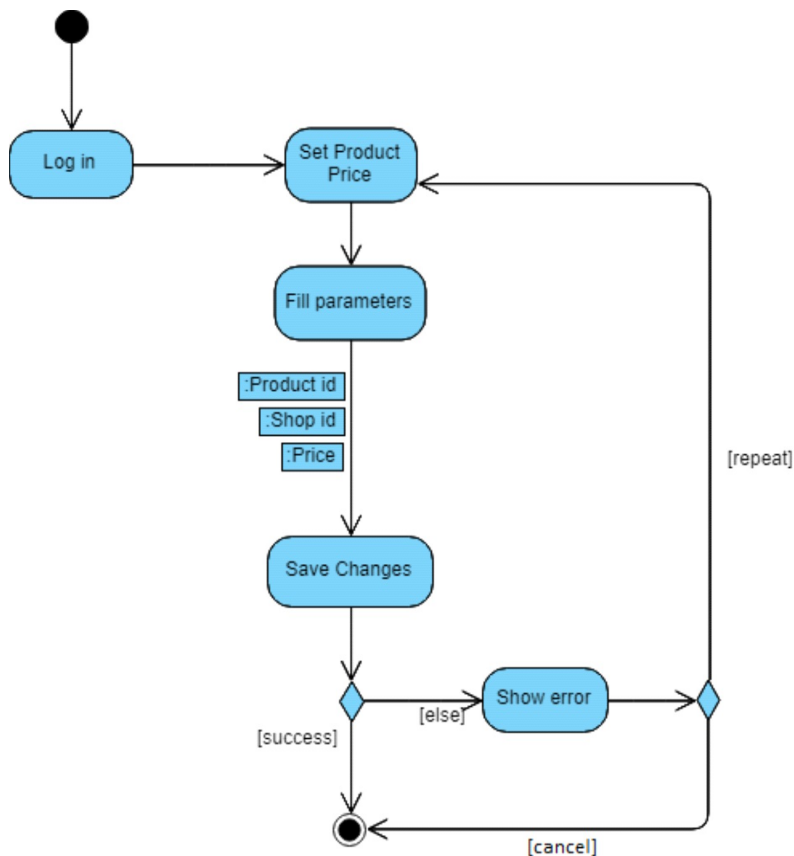
Βήμα 4β: Η Βάση Δεδομένων ενημερώνει την ιστοσελίδα ότι υπήρξε σφάλμα κατά την προσπάθεια σύνδεσης του χρήστη.

Βήμα 5β: Η ιστοσελίδα ανακοινώνει στο χρήστη ότι η σύνδεση απέτυχε.

UML διάγραμμα αλληλουχίας (sequence diagram):



UML διάγραμμα δραστηριοτήτων (activity diagram):



OnlinePriceTracker/api/controllers/prices\_controller.js

### 3.2.2.7 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)

Δεδομένο εξόδου αποτελεί η τιμή του προϊόντος σε περίπτωση που η ενημέρωσή της έγινε με επιτυχία. Η νέα τιμή είναι αυτή που δόθηκε από το χρήστη και η διαδικασία εισαγωγής της στη Βάση Δεδομένων του συστήματος είναι αυτή που φαίνεται στο διάγραμμα της προηγούμενης υποενότητας.

### 3.2.2.8 Παρατηρήσεις

Ο,τι δεν εντάσσεται στα προηγούμενα, εφόσον υπάρχει

Ενδεχομένως προηγούμενες τιμές του συγκεκριμένου προϊόντος (ιστορικό τιμών προϊόντος) θα μπορούσαν να ληφθούν υπόψη για την εξακρίβωση του αν η προτεινόμενη αλλαγή είναι λογικοφανής σύμφωνα με κάποιο ευριστικό κριτήριο. Για παράδειγμα το σύστημα θα μπορούσε να αντιλαμβάνεται απότομες διακυμάνσεις στην τιμή ενός προϊόντος και να απορρίπτει τις σχετικές αλλαγές ή να τις αντιμετωπίζει με κάποια επιφύλαξη, ώστε να εξασφαλίσει την αξιοπιστία των δεδομένων που παρέχει στους καταναλωτές.

## 3.2.3 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 3: Προσθήκη/Ενημέρωση προϊόντος

### 3.2.3.1 Χρήστες (ρόλοι) που εμπλέκονται

*Αναφορά στους ρόλους που αφορά η περίπτωση χρήσης*

Ο ρόλος που σχετίζεται με αυτήν την περίπτωση χρήσης είναι αυτός του χρήστη εθελοντή, δηλαδή του συμμετέχοντα στο crowdsourcing.

### **3.2.3.2 Προϋποθέσεις εκτέλεσης**

*Καταγραφή των συνθηκών που πρέπει να ισχύουν ώστε να μπορεί να εκτελεστεί η περίπτωση χρήσης*

Για να εκτελεστεί αυτή η περίπτωση χρήσης πρέπει ένας χρήστης εθελοντής να επιθυμεί να συνδεθεί στην ιστοσελίδα μας και να διαθέτει πληροφορίες για ένα συγκεκριμένο προϊόν τις οποίες επιθυμεί να εισάγει στο σύστημα ή να τις χρησιμοποιήσει για να ενημερώσει (μερικά ή ολικά) τις υπάρχουσες.

### **3.2.3.3 Περιβάλλον εκτέλεσης**

*Αναφορά στο περιβάλλον στο οποίο εκτελείται η περίπτωση χρήσης. Πχ "διαδικτυακή διεπαφή χρήστη", "DBMS" κλπ*

Η περίπτωση χρήσης εκτελείτε στο περιβάλλον της διαδικτυακής διεπαφής χρήστη της εφαρμογής μας.

### **3.2.3.4 Δεδομένα εισόδου**

*Καταγραφή δεδομένων εισόδου και εξόδου και συνθηκών εγκυρότητας αυτών.*

Δεδομένα εισόδου αποτελούν σε πρώτο πλάνο τα στοιχεία σύνδεσης χρήστη (User ID, password) και σε δεύτερο πλάνο τα στοιχεία του προϊόντος (όνομα, περιγραφή, ετικέτες, κατηγορία). Παράμετροι εξόδου είναι τα νέα/τροποποιημένα στοιχεία του προϊόντος σε περίπτωση επιτυχίας.

### **3.2.3.5 Παράμετροι**

*Καταγραφή παραμέτρων και συνθηκών εγκυρότητας αυτών, εφόσον υπάρχουν παράμετροι.*

Παράμετροι της περίπτωσης χρήσης αυτής αποτελούν τα προϊόντα που είναι καταχωρημένα στη Βάση Δεδομένων καθώς τα στοιχεία αυτά θα καθορίσουν αν η ενέργεια του χρήστη αφορά προσθήκη ή ενημέρωση. Η εγκυρότητα των δεδομένων αυτών μέχρι πρότινος θεωρείται δεδομένη καθώς ήταν ήδη εισαγμένα στη Βάση, ενώ σε περίπτωση ενημέρωσής τους θεωρούνται πλέον άκυρα.

### **3.2.3.6 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).*

Βήμα 1: Ο χρήστης-εθελοντής εισάγει το User ID και το password του στη φόρμα της σελίδας σύνδεσής μας και τα υποβάλει στην ιστοσελίδα.

Βήμα 2: Η ιστοσελίδα μεταβιβάζει τα στοιχεία αυτά στη Βάση Δεδομένων, ζητώντας της να επιχειρήσει ταυτοποίηση του χρήστη.

Βήμα 3: Σε περίπτωση που η Βάση Δεδομένων βρει εγγραφή που έχει τα στοιχεία που της δόθηκαν ως τιμές στα αντίστοιχα πεδία συνεχίζουμε με την περίπτωση (α), αλλιώς με την περίπτωση (β).

Βήμα 4α: Η Βάση Δεδομένων ενημερώνει την ιστοσελίδα ότι ο χρήστης πιστοποιήθηκε με επιτυχία.

Βήμα 5α: Η ιστοσελίδα ενημερώνει το χρήστη ότι η σύνδεσή του στο σύστημα πραγματοποιήθηκε με επιτυχία.

Βήμα 6α: Ο χρήστης στέλνει αίτημα ενημέρωσης ή προσθήκης προϊόντος στην ιστοσελίδα.

Βήμα 7α: Η ιστοσελίδα ζητάει από το χρήστη να καθορίσει τα στοιχεία εισόδου για την ενημέρωση ή την εισαγωγή.

Βήμα 8α: Ο χρήστης αποστέλει στην ιστοσελίδα το Product name, description, category καθώς

και τα αναλογα tags.

Βήμα 9α: Η ιστοσελίδα αποστέλει τα παραπάνω στοιχεία στη Βάση Δεδομένων για έλεγχο.

Βήμα 10α: Η Βάση Δεδομένων ελέγχει αν τα δεδομένα που της δόθηκαν είναι έγκυρα. Αν αυτό αληθεύει συνεχίζουμε με την περίπτωση (γ), αλλιώς με την περίπτωση (δ).

Βήμα 11γ: Η Βάση Δεδομένων ενημερώνει την ιστοσελίδα ότι τα δεδομένα που της δόθηκαν είναι έγκυρα.

Βήμα 12γ: Η ιστοσελίδα ενημερώνει το χρήστη ότι η εισαγωγή/ενημέρωση των πληροφοριών του προϊόντος έγινε με επιτυχία.

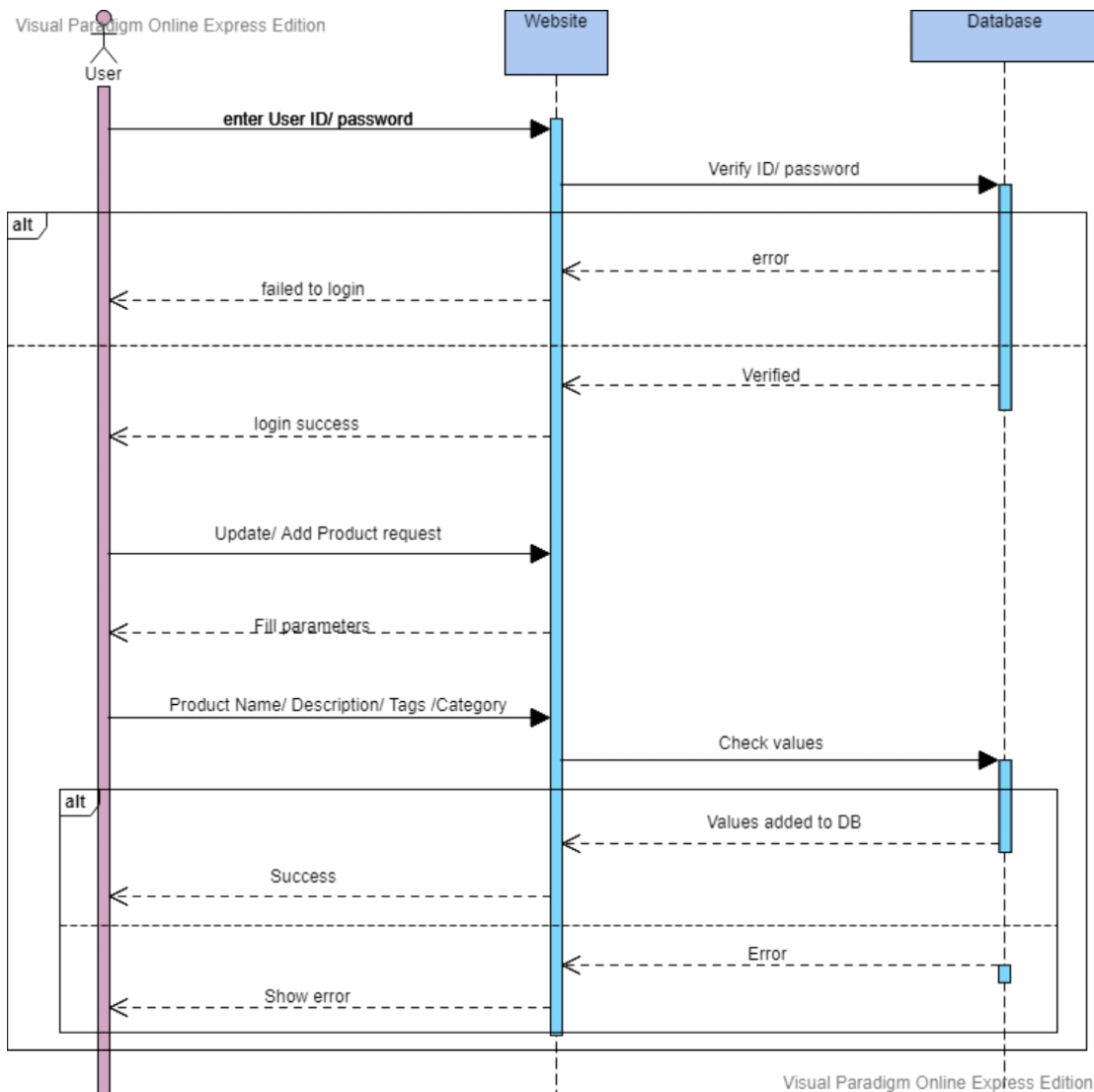
Βήμα 11δ: Η Βάση Δεδομένων ενημερώνει την ιστοσελίδα ότι τα δεδομένα που της δόθηκαν δεν ήταν έγκυρα.

Βήμα 12δ: Η ιστοσελίδα ανακοινώνει στο χρήστη την αποτυχία της εισαγωγής/ενημέρωσης του προϊόντος ενδιαφέροντος.

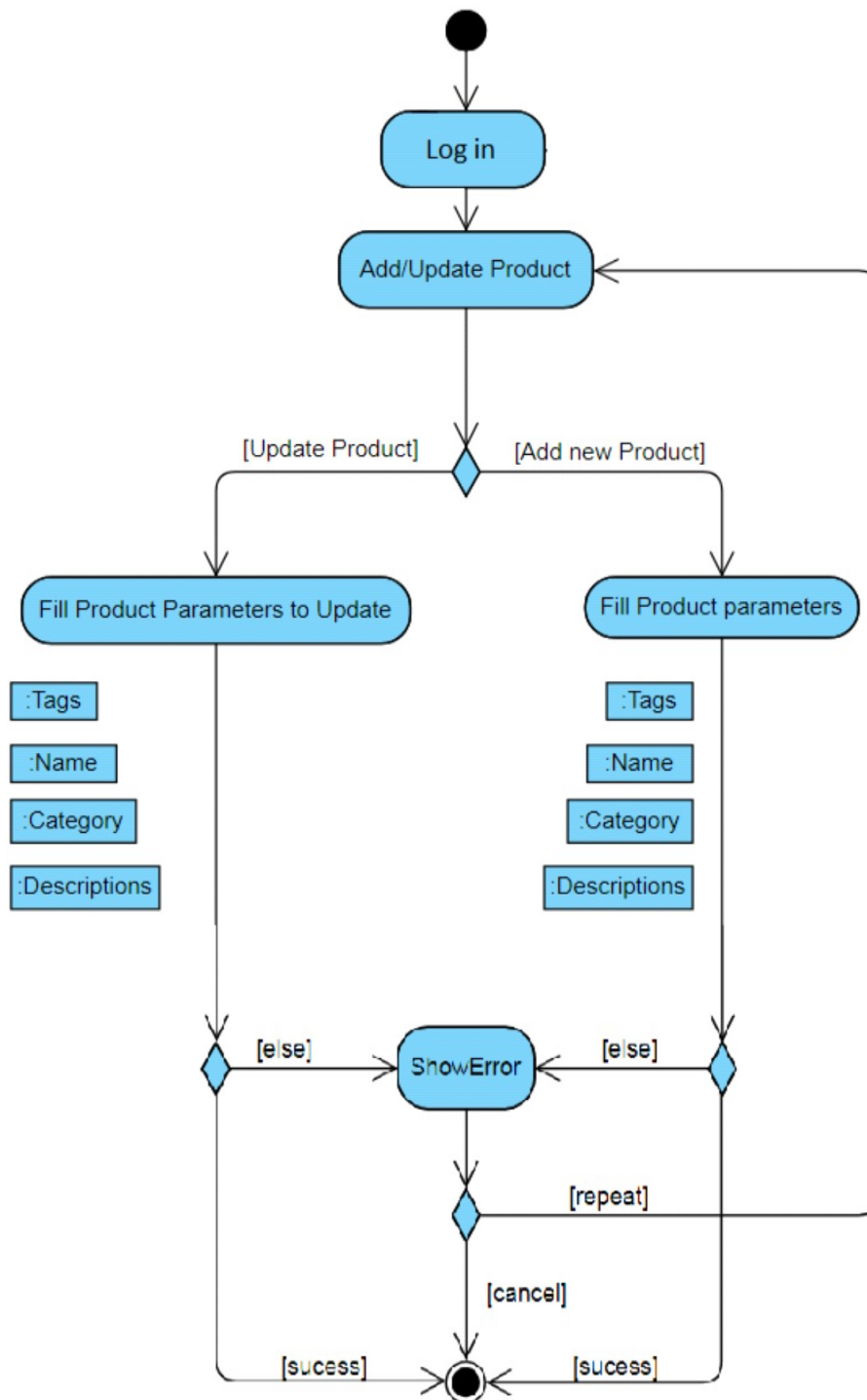
Βήμα 4β: Η Βάση Δεδομένων ενημερώνει την ιστοσελίδα ότι υπήρξε σφάλμα κατά την προσπάθεια σύνδεσης του χρήστη.

Βήμα 5β: Η ιστοσελίδα ανακοινώνει στο χρήστη ότι η σύνδεση απέτυχε.

UML διάγραμμα αλληλουχίας (sequence diagram):



UML διάγραμμα δραστηριοτήτων (activity diagram):



OnlinePriceTracker/api/controllers/products\_controller.js

### 3.2.3.7 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)

Τα δεδομένα εξόδου αποτελούν τα όλα τα στοιχεία του προϊόντος σε περίπτωση εισαγωγής ή τα τροποποιημένα σε περίπτωση ενημέρωσης του προϊόντος. Ο τρόπος εισαγωγής τους από το χρήστη και η αποθηκευσή τους στη Βάση Δεδομένων απεικονίζονται στο διάγραμμα UML της



προηγούμενης υποενότητας.

### 3.2.3.8 Παρατηρήσεις

*Ο,τι δεν εντάσσεται στα προηγούμενα, εφόσον υπάρχει*

Χρήσιμο θα ήταν να διευκρινιστούν δύο πράγματα στην υποενότητα αυτή. Το πρώτο αφορά τη φύση των πεδίων που αφορούν τα διάφορα προϊόντα και το δεύτερο κάποιες τεχνικές λεπτομέρειες της διαδικασίας ενημέρωσης/εισαγωγής προϊόντος.

Σε ό,τι αφορά τα σχετικά με το προϊόν πεδία που μας ενδιαφέρουν σε αυτήν την περίπτωση χρήσης ισχύουν τα παρακάτω: Το πεδίο name αποτελεί το όνομα του προϊόντος με βάση το οποίο το αναγνωρίζουν οι καταναλωτές (σε αντιδιαστολή με το Product ID που αποτελεί αναγνωριστικό για το σύστημά μας). Το πεδίο description αποτελεί μία σύντομη περιγραφή του προϊόντος στο οποίο αναφέρονται χαρακτηριστικά του, ποσοτικά και ποιοτικά, συστατικά του στοιχεία (βλέπε τρόφιμα), εμπειρίες χρήσεις και γενικά πληροφορίες οι οποίες θα μπορούσαν να θεωρηθούν χρήσιμες για το καταναλωτικό κοινό. Το πεδίο category αναφέρεται στη γενική ομάδα προϊόντων στην οποία εντάσσεται το συγκεκριμένο προϊόν (π.χ. γαλακτοκομικά, τηλεοράσεις, είδη προσωπικής υγιεινής). Το πεδίο tags είναι πολύτιμο (παίρνει πολλές τιμές) και αναφέρεται σε ειδικότερες ομάδες στις οποίες μπορεί να ενταχθεί το προϊόν (π.χ. αγελαδινό γάλα, χαμηλά λιπαρά, ελληνική προέλευση).

Τώρα σχετικά με τις τεχνικές λεπτομέρειες για να επιτραπεί η εισαγωγή του προϊόντος στη Βάση Δεδομένων μας πρέπει να εισαχθούν τιμές και στα τέσσερα σχετικά πεδία, τις οποίες καλείται να δώσει ο χρήστης. Δεν ισχύει όμως το ίδιο και για την ενημέρωση, για την οποία απαραίτητα πεδία είναι μόνο αυτά που πρόκειται να τροποποιηθούν. Ο χρήστης λοιπόν μπορεί να εισάγει μόνο τις τιμές των πεδίων αυτών κατά την ενημέρωση.

Τέλος, αξίζει να σημειωθεί ότι θα ήταν χρήσιμη μία λίστα από επιτρεπτά tags η οποία θα μπορεί να επεκτείνεται είτε από έμπιστους χρήστες (με κάποιο κριτήριο), είτε από τους διαχειριστές του συστήματος, ενώ θα είναι ευπρόσδεκτες προτάσεις και από άλλους χρήστες. Σκοπός αυτής της τακτικής είναι να υπάρχει συγκεκριμένο tag για κάθε ειδική ομάδα προϊόντων (π.χ. μόνο ελληνική προέλευση και όχι ελληνική προέλευση και προϊόν από Ελλάδα) ώστε να αποφεύγεται η σύγχυση των χρηστών και να διευκολύνεται η αναζήτηση προϊόντων στη Βάση Δεδομένων.

## 3.2.4 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 4: Αναζήτηση προϊόντος

### 3.2.4.1 Χρήστες (ρόλοι) που εμπλέκονται

*Αναφορά στους ρόλους που αφορά η περίπτωση χρήσης*

Αυτή η περίπτωση χρήσης αφορά όλους τους επισκέπτες της ιστοσελίδας μας. Ωστόσο μπορούμε να πούμε ότι ο ρόλος που σχετίζεται πιο πολύ με αυτήν είναι αυτός του αναγνώστη.

### 3.2.4.2 Προϋποθέσεις εκτέλεσης

*Καταγραφή των συνθηκών που πρέπει να ισχύουν ώστε να μπορεί να εκτελεστεί η περίπτωση χρήσης*

Για την εκτέλεση αυτής της περίπτωσης χρήσης πρέπει ο αναγνώστης να έχει υπόψη του κάποια στοιχεία για το προϊόν που επιθυμεί να αναζητήσει.

### 3.2.4.3 Περιβάλλον εκτέλεσης

*Αναφορά στο περιβάλλον στο οποίο εκτελείται η περίπτωση χρήσης. Πχ "διαδικτυακή διεπαφή χρήστη", "DBMS" κλπ*

Αυτή η περίπτωση χρήσης εκτελείται στο περιβάλλον της διαδικτυακής διεπαφής χρήστη της ιστοσελίδας μας.

### 3.2.4.4 Δεδομένα εισόδου



*Καταγραφή δεδομένων εισόδου και εξόδου και συνθηκών εγκυρότητας αυτών.*

Δεδομένα εισόδου αποτελούν διάφορα στοιχεία για το προϊόν που αναζητάται και για την περιοχή ενδιαφέροντος. Συγκεκριμένα τα στοιχεία αυτά είναι: Χρονική περίοδος ενδιαφέροντος, Product ID, Shop ID, Tags και Location. Δεδομένα εξόδου αποτελούν οι πληροφορίες των διαφόρων προϊόντων προς εμφάνιση.

#### 3.2.4.5 Παράμετροι

*Καταγραφή παραμέτρων και συνθηκών εγκυρότητας αυτών, εφόσον υπάρχουν παράμετροι.*

Παράμετροι για αυτήν την περίπτωση χρήσεις αποτελούν η περιοχή του χρήστη και το ιστορικό τιμών των διαφορών προϊόντων.

#### 3.2.4.6 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).*

Βήμα 1: Ο αναγνώστης στέλνει αίτημα αναζήτησης προϊόντος στην ιστοσελίδα.

Βήμα 2: Η ιστοσελίδα ζητάει από το χρήστη να επιλέξει τα φίλτρα που θα χρησιμοποιηθούν για την αναζήτησή του.

Βήμα 3: Ο αναγνώστης αποστέλει στην ιστοσελίδα τη χρονική περίοδο ενδιαφέροντος προς αναζήτηση, το Product ID του προϊόντος, το Shop ID του καταστήματος, tags που αφορούν το προϊόν και την τοποθεσία που βρίσκεται/τον ενδιαφέρει.

Βήμα 4: Η ιστοσελίδα αποστέλει τα παραπάνω στοιχεία στη Βάση Δεδομένων για αναζήτηση σχετικών προϊόντων.

Βήμα 5: Η Βάση Δεδομένων επιστρέφει μία λίστα με προϊόντα που ικανοποιούν τα φίλτρα αναζήτησης του αναγνώστη (συνοπτικές πληροφορίες και λεπτομέρειες σχετικά με αυτά).

Βήμα 6: Η ιστοσελίδα εμφανίζει στον αναγνώστη τα προϊόντα που βρέθηκαν.

Βήμα 7: Ο αναγνώστης επιλέγει ενδεχομένως ένα κριτήριο ταξινόμησης και συνεχίζουμε με την περίπτωση (α), διαφορετικά συνεχίζουμε με την περίπτωση (β).

Βήμα 8α: Ο αναγνώστης ενημερώνει την ιστοσελίδα για το κριτήριο ταξινόμησης που επέλεξε.

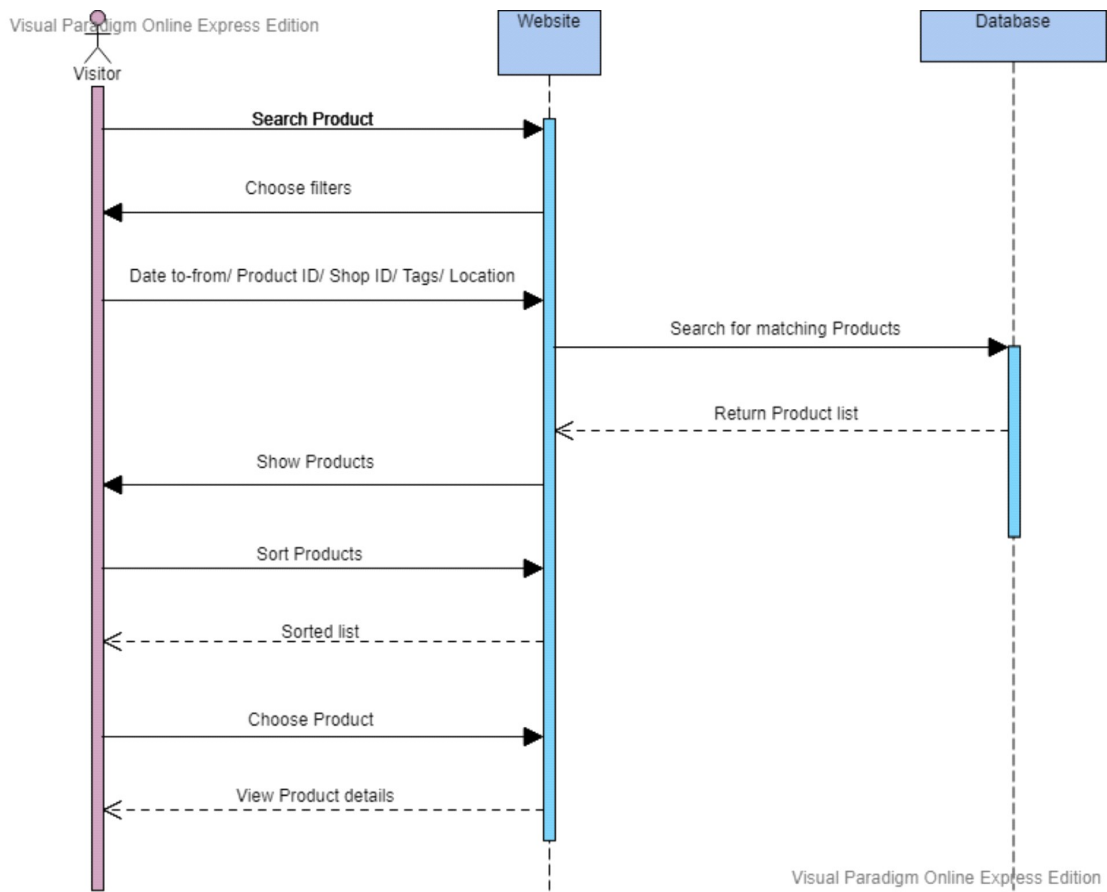
Βήμα 9α: Η ιστοσελίδα επιστρέφει στον αναγνώστη μια ταξινομημένη λίστα με τα προϊόντα που τον ενδιαφέρουν.

Βήμα 10α: Επιστρέφουμε στο Βήμα 7.

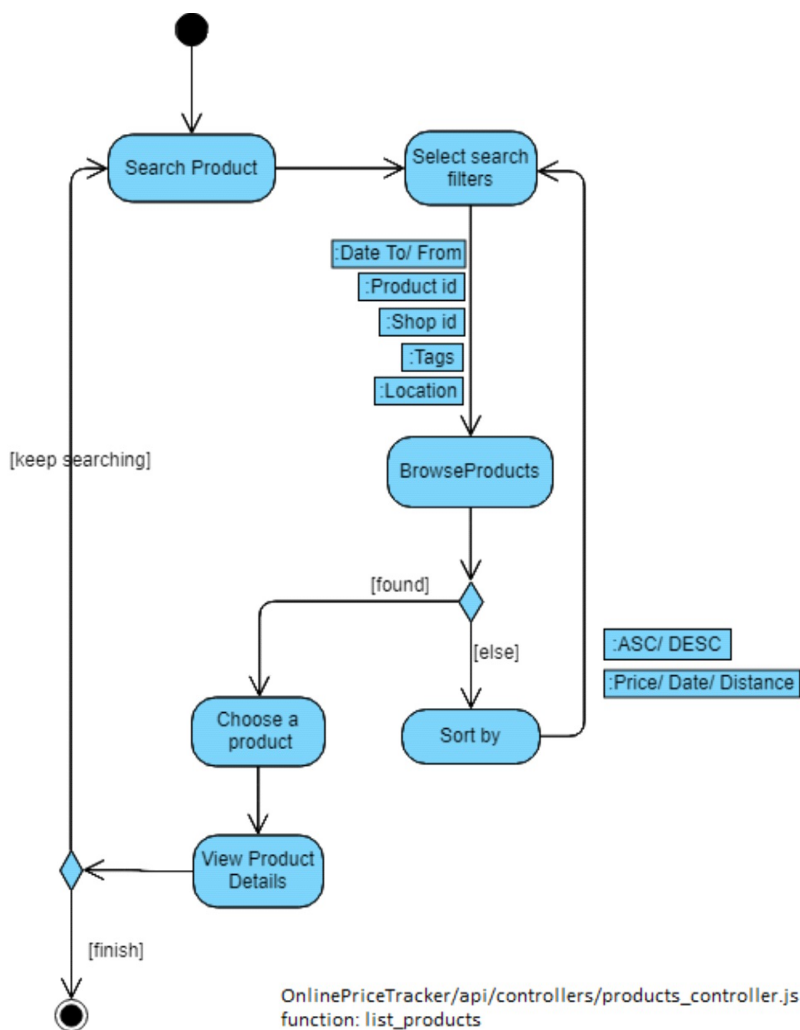
Βήμα 8β: Ο αναγνώστης επιλέγει ένα προϊόν και ενημερώνει σχετικά την ιστοσελίδα.

Βήμα 9β: Η ιστοσελίδα εμφανίζει στον αναγνώστη λεπτομέρειες σχετικά με το προϊόν που επέλεξε.

UML διάγραμμα αλληλουχίας (sequence diagram):



UML διάγραμμα δραστηριοτήτων (activity diagram):



### 3.2.4.7 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)

Δεν υφίστανται τροποποιήσεις δεδομένων. Τα δεδομένα που εξάγονται προς την ιστοσελίδα και στη συνέχεια προς το χρήστη αφορούν ένα υποσύνολο των προϊόντων που είναι αποθηκευμένα στη Βάση Δεδομένων και τις σχετικές τους λεπτομέρειες, ενώ η διαδικασία εξαγωγής τους απεικονίζεται στα διαγράμματα της προηγούμενης ενότητας.

### 3.2.4.8 Παρατηρήσεις

Ο,τι δεν εντάσσεται στα προηγούμενα, εφόσον υπάρχει

Μπορούμε να σημειώσουμε ορισμένα πράγματα. Καταρχήν ο αναγνώστης μπορεί να μην καθορίσει όλα τα φίλτρα αφήνοντας ελεύθερες ορισμένες παραμέτρους αναζήτησης. Στην περίπτωση αυτή η αναζήτηση γίνεται με βάση τα πεδία που έχουν συμπληρωθεί. Ακόμη ορισμένα πεδία μπορεί να προκύπτουν φυσικά από κάποια άλλα. Για παράδειγμα αν ο αναγνώστης επιλέξει ένα συγκεκριμένο Product ID τότε το προϊόν που αναζητάται είναι συγκεκριμένο. Στην περίπτωση αυτή τα tags μπορούν να χρησιμοποιηθούν για την εμφάνιση επιπλέον σχετικών προϊόντων. Επίσης, η τοποθεσία μπορεί να μην εισάγεται από τον ίδιο τον αναγνώστη αλλά να συμπληρώνεται αυτόματα από την εφαρμογή αν ο αναγνώστης έχει

ενεργοποιήσει αυτήν τη δυνατότητα. Επιπλέον, το ιστορικό αναζητήσεων, αγορών και γενικά δράσης του αναγνώστη μπορούν να λαμβάνονται υπόψη για την εμφάνιση κατάλληλων προτάσεων. Τέλος, ορισμένα ενδεικτικά κριτήρια ταξινόμησης προϊόντων είναι το ελάχιστο κόστος, η εγγύτητα του καταστήματος και η ημερομηνία (ως έξτρα θα μπορούσε να θεωρηθεί η δημοτικότητα του προϊόντος). Η ταξινόμηση μπορεί να γίνει σε αύξουσα ή σε φθίνουσα σειρά. Ως default ταξινόμηση μπορεί να θεωρηθεί αυτή ως προς την τιμή σε αύξουσα σειρά κόστους.

### 3.2.5 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 5: Διαγραφή χρήστη

#### 3.2.5.1 Χρήστες (ρόλοι) που εμπλέκονται

*Αναφορά στους ρόλους που αφορά η περίπτωση χρήσης*

Σε αυτήν την περίπτωση χρήσης εμπλέκεται ο ρόλος του admin δηλαδή του διαχειριστή του συστήματος.

#### 3.2.5.2 Προϋποθέσεις εκτέλεσης

*Καταγραφή των συνθηκών που πρέπει να ισχύουν ώστε να μπορεί να εκτελεστεί η περίπτωση χρήσης*

Για την εκτέλεση αυτής της περίπτωσης χρήσης πρέπει ο admin να είναι επιτυχώς συνδεδεμένος στο σύστημα και να έχει υπόψην του το User ID του χρήστη που θα επιθυμούσε να διαγράψει.

#### 3.2.5.3 Περιβάλλον εκτέλεσης

*Αναφορά στο περιβάλλον στο οποίο εκτελείται η περίπτωση χρήσης. Πχ "διαδικτυακή διεπαφή χρήστη", "DBMS" κλπ*

Η περίπτωση χρήσης εκτελείται μέσω της ιστοσελίδας (διαδικτυακή διεπαφή χρήστη) ωστόσο σημαντικό ρόλο παίζει και το DBMS.

#### 3.2.5.4 Δεδομένα εισόδου

*Καταγραφή δεδομένων εισόδου και εξόδου και συνθηκών εγκυρότητας αυτών.*

Μοναδικό δεδομένο εισόδου αποτελεί το User ID του χρήστη προς διαγραφή. Το User ID αυτό είναι έγκυρο αν υπάρχει εγγραφή στον πίνακα της Βάσης Δεδομένων με τους χρήστες με τιμή που ταιριάζει σε αυτό. Δεν υπάρχουν δεδομένα εξόδου, παρά μόνο ένα μήνυμα επιτυχίας/αποτυχίας της διαδικασίας διαγραφής χρήστη.

#### 3.2.5.5 Παράμετροι

*Καταγραφή παραμέτρων και συνθηκών εγκυρότητας αυτών, εφόσον υπάρχουν παράμετροι.*

Παραμέτρους αποτελούν τα πιστοποιητικά ταυτοποίησης του admin, ενώ η εγκυρότητα αυτών καθορίζεται από το σύστημα.

#### 3.2.5.6 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).*

Βήμα 1: Ο διαχειριστής συμπληρώνει την ανάλογη φόρμα και στέλνει στην ιστοσελίδα το User ID του χρήστη που επιθυμεί να διαγράψει.

Βήμα 2: Η ιστοσελίδα μεταβιβάζει το User ID αυτό στη Βάση Δεδομένων ώστε αυτή να προχωρήσει σε έλεγχο.

Βήμα 3: Η Βάση Δεδομένων ελέγχει αν υπάρχει κάποια εγγραφή στον πίνακα με τους χρήστες με το User ID που της δόθηκε. Αν υπάρχει συνεχίζουμε με την περίπτωση (α), διαφορετικά συνεχίζουμε με την περίπτωση (β).

Βήμα 4α: Η Βάση Δεδομένων ενημερώνει την ιστοσελίδα ότι βρέθηκε ο ζητούμενος χρήστης.

Βήμα 5α: Η ιστοσελίδα στέλνει εντολή στη Βάση Δεδομένων να διαγράψει τον παραπάνω χρήστη από το σύστημα.

Βήμα 6α: Η Βάση Δεδομένων προσπαθεί να διαγράψει το χρήστη από το σύστημα. Σε περίπτωση επιτυχίας συνεχίζουμε με την περίπτωση (γ), διαφορετικά με την περίπτωση (δ).

Βήμα 7γ: Η Βάση Δεδομένων ενημερώνει το διαχειριστή ότι η διαγραφή του χρήστη πραγματοποιήθηκε με επιτυχία.

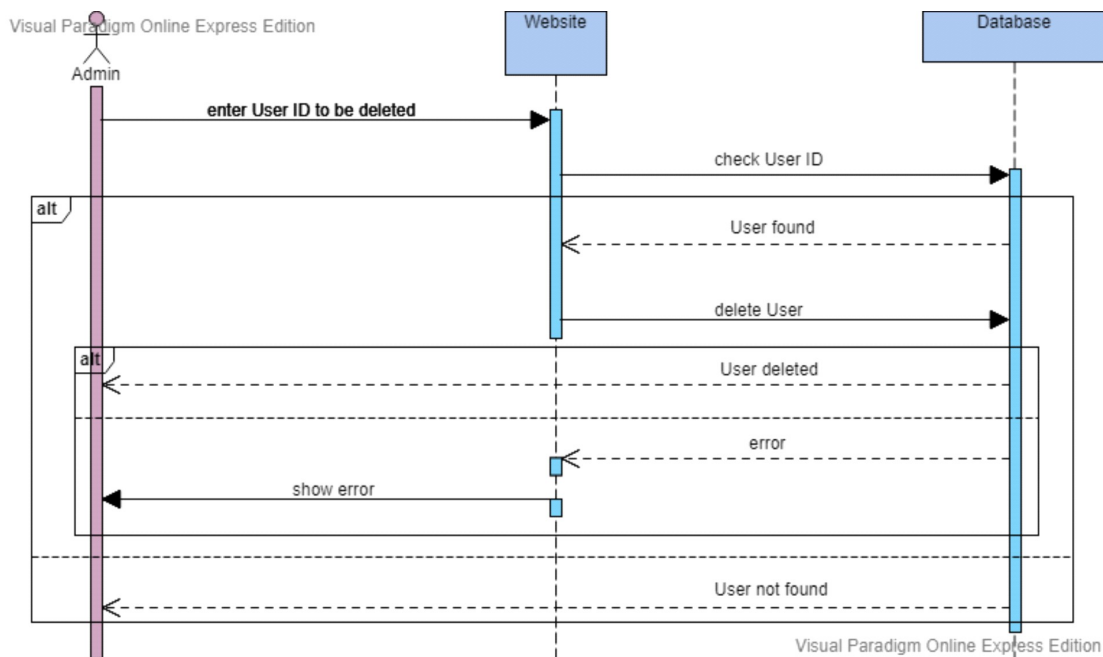
Βήμα 7δ: Η Βάση Δεδομένων ενημερώνει την ιστοσελίδα ότι παρουσιάστηκε κάποιο σφάλμα κατά την προσπάθεια διαγραφής του χρήστη, αποστέλοντάς της παράλληλα ορισμένες αναγνωριστικές πληροφορίες σχετικές με το σφάλμα αυτό.

Βήμα 8δ: Η ιστοσελίδα εμφανίζει τις πληροφορίες του σφάλματος διαγραφής στο διαχειριστή του συστήματος.

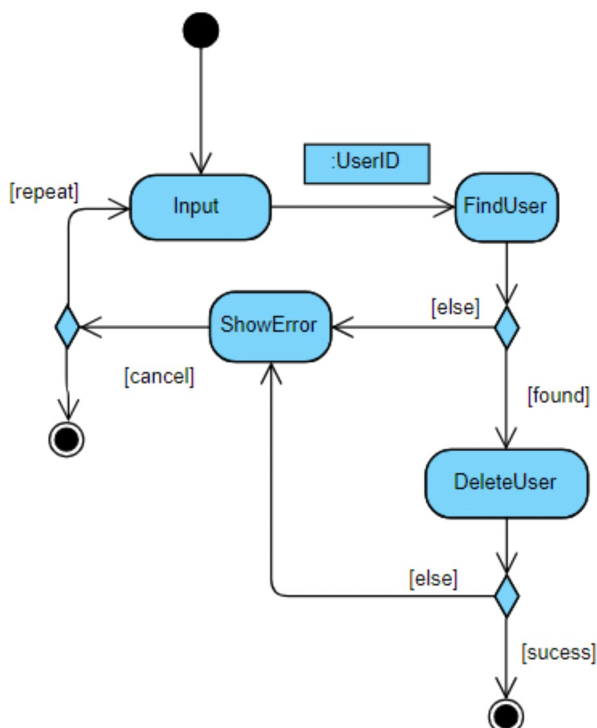
Βήμα 9δ: Ο διαχειριστής επιλέγει αν θέλει να επαναλάβει τη διαδικασία επιστρέφοντας στο Βήμα 1.

Βήμα 4β: Η Βάση Δεδομένων ενημερώνει το διαχειριστή ότι δε βρέθηκε χρήστης με το δοθέν User ID καταχωρημένος στο σύστημα.

UML διάγραμμα αλληλουχίας (sequence diagram):



UML διάγραμμα δραστηριοτήτων (activity diagram):



### 3.2.5.7 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)

Η μεταβολή που υφίσταται σε περίπτωση επιτυχίας της περίπτωσης χρήσης είναι η διαγραφή μίας εγγραφής με τα στοιχεία χρήστη, που έχει User ID ίδιο με το δοθέν από το διαχειριστή, από τον πίνακα χρηστών στη Βάση Δεδομένων του συστήματός μας. Η διαδικασία που οδηγεί στη διαγραφή αυτή (μεταβολή της Βάσης Δεδομένων) απεικονίζεται στα UML διαγράμματα της προηγούμενης υποενότητας.

### 3.2.5.8 Παρατηρήσεις

Ο,τι δεν εντάσσεται στα προηγούμενα, εφόσον υπάρχει

Αξιοσημείωτοι είναι οι λόγοι που μπορούν να οδηγήσουν ένα διαχειριστή (admin) στη διαγραφή ενός χρήστη από το σύστημα, όπως περιγράφηκε προηγουμένως. Ο ένας λόγος είναι σχετικό αίτημα από το χρήστη αυτό, διότι δεν επιθυμεί να διατηρεί πλέον λογαριασμό στην ιστοσελίδα μας. Ο δεύτερος λόγος είναι η ενδεχόμενη επιβλαβής για το σύστημα συμπεριφορά του χρήστη (εδώ γίνεται αναφορά σε χρήστες που συμμετέχουν στο crowdsourcing). Αυτός ο λόγος αφορά την πιστοποιημένη αποστολή αιτημάτων από το χρήστη αυτό για αντικατάσταση στοιχείων των διαφόρων προϊόντων (περιγραφή, τιμές) με νέα λανθασμένα στοιχεία. Συμπεριφορές όπως οι παραπάνω διαταράσσουν την ακεραιότητα των δεδομένων του συστήματος και είναι αναγκαίο να αντιμετωπίζονται με διαγραφή των σχετικών χρηστών μόλις διαπιστώνονται.

...

(η ενότητα 3.2.X.1 - 3.2.X.8 επαναλαμβάνεται για όλες τις λειτουργίες που συμπεριλαμβάνονται στο έγγραφο, όπως απαιτούνται από τις ομάδες ανάλογα με τον αριθμό των ατόμων)

### 3.3 Απαιτήσεις επιδόσεων

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

Η επιδόσεις του συστήματός μας μπορούν να μετρώνται και να αξιολογούνται ως το πλήθος των τροποποιήσεων δεδομένων (ενημέρωση/προσθήκη προϊόντος, ενημέρωση τιμής) και των προτάσεων προϊόντων (ταξινόμηση με βάση κριτήριο επιλεγόμενο από το χρήστη και εμφάνιση προτάσεων) που το σύστημα μπορεί να ικανοποιήσει στη μονάδα του χρόνου. Προφανώς οι επιδόσεις που απαιτούνται με βάση τον παραπάνω ορισμό σε μία δεδομένη και σύντομη χρονική περίοδο είναι ανάλογες με το πλήθος των χρηστών που χρησιμοποιούν ενεργά την εφαρμογή κατά τη διάρκεια της περιόδου αυτής (ενεργοί χρήστες). Αυτό προκύπτει άμεσα αν θεωρήσουμε ότι ο κάθε ενεργός χρήστης καταβάλλει αιτήσεις προς το λογισμικό με κάποιο σταθερό ρυθμό και ότι οι αποκλίσεις που παρουσιάζονται από το ρυθμό αυτό είναι σχετικά μικρές. Σε περιόδους με πολλούς ενεργούς χρήστες λοιπόν οι επιδόσεις πρέπει να είναι αυξημένες ανάλογα.

Οι παραπάνω δύο μετρικές που ορίσαμε μπορούν να χρησιμοποιηθούν κατάλληλα για τη μέτρηση της επίδοσης του λογισμικού μας συστήματος. Ωστόσο η πολλαπλότητά τους υπαγορεύει την ανάγκη για την καθιέρωση ενός κριτηρίου προαιρερότητας επί αυτών. Το κριτήριο αυτό είναι η έμφαση στις αιτήσεις εμφάνισης προτάσεων έναντι των αιτήσεων ενημέρωσης θέτοντάς τους την κατάλληλη προτεραιότητα. Αυτό επειδή είναι πιο σημαντικό για την εκόνα του OPT να μπορούν οι καταναλωτές (αναγνώστες) να αντλούν γνώση για να λαμβάνουν καταναλωτικές αποφάσεις, από το να είναι τα δεδομένα απόλυτα ακριβή. Συγκεκριμένα, μικρές αποκλίσεις στις τιμές οι οποίες σύντομα πρόκειται να διορθωθούν μπορούν να γίνουν ανεκτές. Συνεπώς έχουμε σαν επικρατές κριτήριο επίδοσης το throughput (και το latency) σε εμφανίσεις προτάσεων προϊόντων, και από τη στιγμή που η επίδοσης αυτές κρίνονται ικανοποιητικές τίθενται ως επιπλέον κριτήρια τα αντίστοιχα μεγέθη στις ενημερώσεις της Βάσης Δεδομένων.

### 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.

*Αναλυτική αναφορά στα δεδομένα εισόδου, τα σχετικά πρότυπα δεδομένων και υπηρεσιών, καθώς και σε μετρικές που σχετίζονται με τα δεδομένα (storage capacity planning).*

Τα δεδομένα εισόδου στο σύστημα μπορούν να διαχωριστούν στα δεδομένα χρηστών, στα δεδομένα προϊόντων και στα δεδομένα καταστημάτων.

Τα δεδομένα χρηστών δημιουργούνται κατά την εγγραφή και έχουν ως κύριο σκοπό τον έλεγχο της πρόσβασης και την εξατομίκευση των προτάσεων προϊόντων προς τους χρήστες. Τα δεδομένα αυτά είναι το username και το password τα οποία τίθενται μία φορά και αποθηκεύονται στη Βάση Δεδομένων, ενώ υπάρχει δυνατότητα τροποποίησης, καθώς και ένα μη υποχρεωτικό ιστορικό αναζητήσεων προϊόντων και ενεργειών που έχουν σαν στόχο τη δημιουργία ενός προφίλ συμπεριφοράς χρήστη για την αναγνώριση μελλοντικών αναγκών και προτιμήσεων σε προϊόντα, με αποτέλεσμα τη βελτίωση των υπηρεσιών αναζήτησης.

Τα δεδομένα που αφορούν τα προϊόντα είναι το όνομά τους, η τιμή τους ανά κατάσταση (συνοδευόμενη από διάρκεια ισχύος), λεκτική περιγραφή ποσοτικών και ποιοτικών χαρακτηριστικών από τους χρήστες, ετικέτες (tags), η κατηγορία στην οποία ανήκουν και η διαθεσιμότητά τους. Τα δεδομένα αυτά είναι αποθηκευμένα στη Βάση Δεδομένων, και ενώ γενικά αφορούν όλα τα ταυτόσημα προϊόντα, η τιμή αποτελεί συνάρτηση του καταστήματος (βλέπε και ER διάγραμμα). Η εισαγωγή τους μπορεί να γίνει μία φορά, ωστόσο πιο ρεαλιστική είναι η θεώρηση της ενημέρωσής τους και μάλιστα συχνής.

Τα δεδομένα που αφορούν τα καταστήματα είναι το όνομά τους, η διεύθυνσή τους που μπορεί να χρησιμοποιηθεί για την τοποθέτησή τους στο χάρτη, ετικέτες ομαδοποίησης, το γεωγραφικό μήκος και πλάτος όπου βρίσκεται το κατάστημα και οι τιμές των προϊόντων που διαθέτει. Τα δεδομένα αυτά εισάγονται μία φορά κατά την εισαγωγή του καταστήματος στο σύστημα και είναι σχετικά σταθερά, με εξαίρεση βέβαια τις σχετικές με το κατάστημα τιμές προϊόντων.



Ως μετρική φορτίου εισόδου θα μπορούσε να θεωρηθεί το μέγεθος των δεδομένων που η εισαγωγή τους στο σύστημα προκαλεί αύξηση του απαιτούμενου αποθηκευτικού χώρου. Όπως είπαμε, έχουμε δεδομένα που αφορούν χρήστες, καταστήματα, προϊόντα και γενικά (στις κατηγορίες προϊόν και κατάσταση) τιμές. Το πλήθος των καταστημάτων και των χρηστών που εισάγονται στο σύστημα μέχρις ότου η δημοτικότητα της εφαρμογής να σταθεροποιηθεί είναι ίσως σημαντικό ποσοστό επί των συνολικών, ωστόσο μετά από αυτήν τη φάση μετάβασης θα είναι αμελητέα οπότε μπορούν με ασφάλεια να αφαιρεθούν από τη μετρική. Μένουν λοιπόν τα δεδομένα προϊόντων και τιμών. Εδώ πρέπει να παρατηρήσουμε ότι αν και γενικά θα εισάγονται/ενημερώνονται προϊόντα στο σύστημα οι εισαγωγές θα είναι φραγμένες από το πλήθος των διαφορετικών προϊόντων ενώ οι ενημερώσεις δεν επηρεάζουν το απαιτούμενο μέγεθος αποθήκευσης και άρα δεν επηρεάζουν τη μετρική. Ωστόσο οι τιμές θα εισάγονται με πολύ μεγαλύτερο ρυθμό στη Βάση Δεδομένων, καθώς έχουμε τιμές για όλα τα προϊόντα σε όλα τα καταστήματα και οι ιστορικές τιμές διατηρούνται. Μία μετρική λοιπόν φορτίου δεδομένων εισόδου αποτελεί με καλή προσέγγιση το πλήθος αλλαγών τιμής που εισάγονται στο σύστημα ανά δεδομένο χρονικό διάστημα.

Επεξεργασία έχουμε κυρίως για να δώσουμε απάντηση σε αναζητήσεις προϊόντων (εύρεση και ταξινόμηση), οπότε η σχετική μετρική για το φορτίο επεξεργασίας είναι το πλήθος αναζητήσεων προϊόντων που γίνονται από τους αναγνώστες ανά δεδομένο χρονικό διάστημα.

### 3.4.2 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

*Απαιτήσεις πρόσβασης και περιορισμοί.*

Η πρόσβαση στα δεδομένα της ιστοσελίδας πρέπει να μπορεί να γίνεται στον κατάλληλο κάθε φορά βαθμό με βάση τα δικαιώματα που υπαγορεύονται από το είδος του χρήστη (επισκέπτης, αναγνώστης, εθελοντής, admin). Οι επισκέπτες και οι αναγνώστες έχουν ίδια δικαιώματα πρόσβασης σε δεδομένα μέσω αναζητήσεων προϊόντων με τη διαφορά ότι οι αναγνώστες διατηρούν ιστορικό αναζητήσεων. Οι εθελοντές έχουν επιπλέον το δικαίωμα εισαγωγής και τροποποίησης δεδομένων σχετικών με καταστήματα και προϊόντα. Οι διαχειριστές τέλος έχουν και το δικαίωμα διαγραφής χρηστών με βάση τα πρωτόκολλα που αναφέραμε σε προηγούμενη ενότητα.

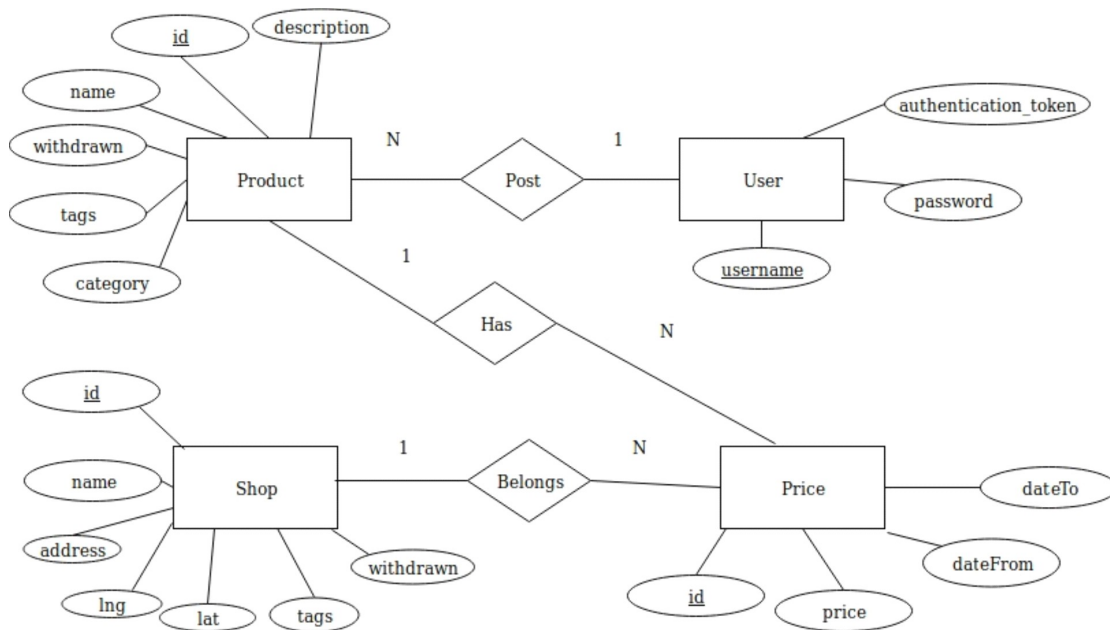
Η πρόσβαση, δηλαδή τα αντίστοιχα δικαιώματα, σε δεδομένα παρέχεται στους χρήστες από τη στιγμή που συνδέονται στο σύστημά μας με επιτυχία. Εκτός από τα δικαιώματα που αναφέραμε, η πρόσβαση σε δεδομένα έχει και ανάλογους περιορισμούς. Τα έξτρα δικαιώματα που αναφέραμε σε κάθε επίπεδο στερούνται από τα κατώτερα. Γενικά οι διαχειριστές έχουν πρόσβαση σε όλα τα δεδομένα με εξαίρεση ευαίσθητα δεδομένα χρηστών (passwords, ιστορικό αναζήτησης).

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

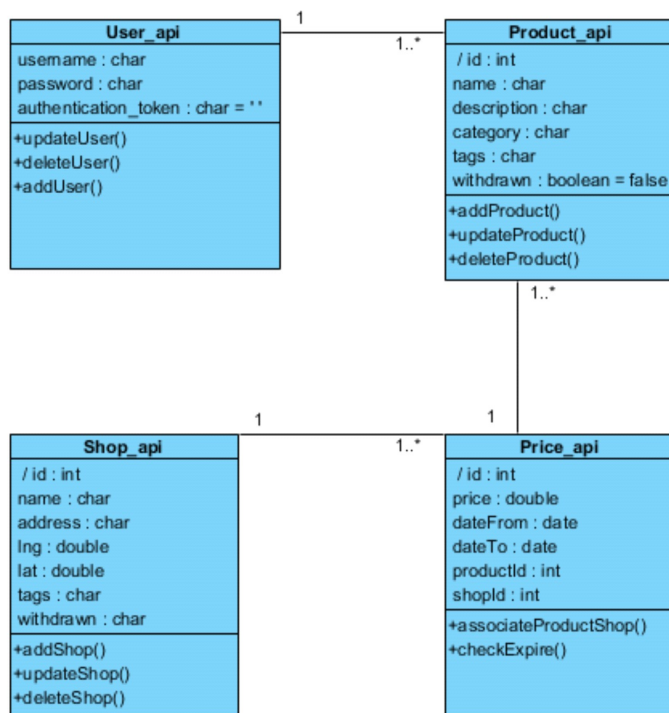
*Μοντέλα δεδομένων UML ή/και ER*

Στο παρακάτω διάγραμμα ER φαίνονται οι διάφορες οντότητες, τα σχετικά τους δεδομένα, καθώς και η συσχέτιση μεταξύ τους, όπως αυτά αποθηκεύονται στο σύστημά μας (DBMS).





Στο ακόλουθο UML διάγραμμα κλάσεων (class diagram) βλέπουμε την αντίστοιχη UML απεικόνιση:



### 3.4.4 Προδιαγραφές ακεραιότητας δεδομένων

#### Κανόνες ακεραιότητας και εγκυρότητας δεδομένων

Στο σημείο αυτό πρέπει να επικεντρωθούμε στο γεγονός ότι οι αναγνώστες χρησιμοποιούν τα δεδομένα που τους παρέχουμε για να λάβουν σημαντικές καταναλωτικές αποφάσεις. Για το λόγο αυτόν η ακεραιότητα και η εγκυρότητα των δεδομένων αυτών είναι υψίστης σημασίας. Σφάλματα σε τιμές πέρα από μικρές-ανεκτές αποκλίσεις από τις πραγματικές τιμές είναι ανεπιθύμητα καθώς μπορούν να δημιουργήσουν δυσφήμιση καταστημάτων/προϊόντων και να υποβαθμίσουν την εικόνα του λογισμικού μας στους χρήστες και σε τρίτα πρόσωπα. Για το λόγο αυτό πρέπει να καθοριστεί μία ανώτατη επιτρεπτή ποσοστιαία απόκλιση της αποθηκευμένης τιμής κάθε προϊόντος από την πραγματική και σε περίπτωση αναφοράς

τακτικής υπέρβασής της οι χρήστες να ενημερώνονται σχετικά κατά την αναζήτηση ώστε να μην παραπλανούνται. Προϊόντα σε καταστήματα των οποίων οι τιμές έχουν καιρό να ενημερωθούν πρέπει επίσης να επισημαίνονται με παρόμοιο τρόπο.

### 3.4.5 Προδιαγραφές διατήρησης δεδομένων

*Απαιτήσεις διατήρησης δεδομένων σε βάθος χρόνου.*

Τα δεδομένα που βρίσκονται αποθηκευμένα στη Βάση Δεδομένων μας πρέπει να διατηρούνται ως έχουν, χωρίς απώλειες, σε απεριόριστο βάθος χρόνου, καθ' όλο το χρόνο ζωής του συστήματός μας, με εξαίρεση την περίπτωση ενημέρωσής τους ή διαγραφής από τους χρήστες. Όταν έχουμε ενημέρωση δεδομένων, τα παλαιότερα δεδομένα μπορούν να πάσουν να διατηρούνται από το σύστημά μας. Ειδική περίπτωση αποτελεί το ιστορικό των τιμών των διαφόρων προϊόντων, του οποίου η διατήρηση ίσως είναι σκόπιμη για λόγους χρησιμότητας και συγκρίσεων. Επίσης, κατά τη διαγραφή δεδομένων από τους χρήστες του συστήματος (π.χ. διαγραφή χρήστη) τα σχετικά με τη διαγραφή δεδομένα μπορούν να μη διατηρούνται πλέον, ώστε να απελευθερώνονται οι αντίστοιχοι πόροι αποθήκευσης από το σύστημα.

## 3.5 Περιορισμοί σχεδίασης

*Λεπτομερής τεχνική τεκμηρίωση των περιορισμών σχεδίασης οι οποίοι επιβάλλονται από απαιτήσεις συμμόρφωσης σε πρότυπα, κανονισμούς, ή άλλους περιορισμούς του έργου. Περιλαμβάνεται η πολιτική ονοματολογίας οντοτήτων δεδομένων και πεδίων. Τέτοιοι περιορισμοί μπορεί να επιβάλλονται από τη χρήση βιβλιοθηκών, frameworks, περιβαλλόντων ανάπτυξης κλπ*

Οι περιορισμοί που τίθενται στην ονοματολογία των πεδίων και των οντοτήτων υπαγορεύονται από το DBMS που έχει επιλεγεί προς χρήση. Η ονοματολογία που παρουσιάζεται πλήρως στο ER διάγραμμα ικανοποιεί τους περιορισμούς αυτούς. Επίσης οι περιορισμοί στους τύπους που μπορούν να λάβουν τα δεδομένα των διαφόρων πεδίων ικανοποιούνται με την επιλογή των τύπων όπως αυτοί απεικονίζονται στο class diagram (βλέπε ενότητα 3.4.3).

## 3.6 Λοιπές απαιτήσεις

### 3.6.1 Απαιτήσεις διαθεσιμότητας λογισμικού

*Τεκμηρίωση απαιτήσεων διαθεσιμότητας*

Το λογισμικό πρέπει να είναι συνεχώς διαθέσιμο στους χρήστες του και ιδίως στους αναγνώστες, οι οποίοι το χρησιμοποιούν για τη λήψη σημαντικών καταναλωτικών αποφάσεων. Τυχόν περίοδοι υποχρεωτικής μη διαθεσιμότητας (π.χ. λόγω συντήρησης του λογισμικού) είναι σημαντικό στο βαθμό του δυνατού να εμφανίζονται κατά τις ώρες που η χρήση της εφαρμογής είναι ιδιαίτερα χαμηλή (π.χ. νυκτερινές ώρες). Ακόμη σε περίπτωση που υπάρχει υψηλός φόρτος στο σύστημα εμποδίζοντας την ταχεία αποκρισιμότητά του, είναι προτιμότερο να εμφανίζονται προσεγγιστικά δεδομένα στους αναγνώστες, ενημερώνοντάς τους βέβαια για το γεγονός αυτό.

### 3.6.2 Απαιτήσεις ασφάλειας

*Τεκμηρίωση απαιτήσεων ασφαλείας*

Η ασφάλεια των δεδομένων των χρηστών είναι υψηλής σημασίας στην περίπτωση του λογισμικού μας, καθώς τυχόν διαρροές μπορούν να εκθέσουν ευαίσθητα προσωπικά τους δεδομένα, τα οποία σχετίζονται με τις αγοραστικές τους συνήθειες. Αυτό είναι προφανώς ανεπιθύμητο και πρέπει να αποφεύγεται με την παρότρυνση χρήσης ισχυρών passwords και τη διασφάλιση του απορρήτου των στοιχείων σύνδεσης που είναι αποθηκευμένα στη Βάση Δεδομένων.

Ακόμη πρέπει να διασφαλίζεται η ασφάλεια των τιμών των διαφόρων προϊόντος, δηλαδή η παρεμπόδιση της επιτηδευμένης διαστρέυλώς τους, καθώς αυτό μπορεί να οδηγήσει σε αποπροσανατολισμό των καταναλωτών, σε δυσφήμιση καταστημάτων και γενικά να έχει αρνητικές για την εικόνα μας συνέπειες. Το φιλτράρισμα των εισαγόμενων στο σύστημα δεδομένων, η αξιολόγηση της αξιοπιστίας των διαφόρων χρηστών που συμμετέχουν στο crowdsourcing και η τακτική διατήρηση backups των δεδομένων της Βάσης Δεδομένων,

αποτελούν λίγους τρόπους που μπορούν να οδηγήσουν σε αυτήν την κατεύθυνση.

Τέλος, εφαρμογές τρίτων που χρησιμοποιούνται από το σύστημά μας πρέπει να διατηρούνται ενημερωμένες για την αποφυγή τυχόν κενών ασφαλείας.

### 3.6.3 Απαιτήσεις συντήρησης

#### *Τεκμηρίωση απαιτήσεων συντήρησης*

Η συντήρηση του λογισμικού πρέπει να αποτελεί βασικό μέλημά μας. Μέσω αυτής εξασφαλίζεται η ταχύτητα απόκρισής του για την άμεση εξυπηρέτηση των χρηστών καθώς και τήρηση των βασικών κανόνων ασφαλείας. Η συντήρηση του λογισμικού θα αποτελείται από τρία βασικά μέρη: Την ενημέρωση των χρησιμοποιούμενων εφαρμογών τρίτων, τη δημιουργία backup δεδομένων και τον εκσυγχρονισμό της διεπαφής με τους χρήστες και του συστήματος γενικότερα. Παρακάτω θα αναλύσουμε τους τρόπους χρήσης και τις συνέπειες του κάθε ενός από τα προηγούμενα μέρη.

Η ενημέρωση των χρησιμοποιούμενων εφαρμογών τρίτων επιτυγχάνεται με τον έλεγχο του αν υπάρχει πιο πρόσφατη έκδοση των εφαρμογών αυτών αν τακτά χρονικά διαστήματα και την ενημέρωσή τους σε περίπτωση που κάτι τέτοιο ισχύει. Η ενημέρωση πρέπει να γίνεται όσο το δυνατόν συντομότερα από τη διαπίστωση της ύπαρξης νέας έκδοσης, αλλά σε ώρες χαμηλής σχετικά χρήσης τις εφαρμογής (π.χ. νύχτα Σαββάτου/Κυριακής), ώστε να μην αντιμετωπίζουν προβλήματα αποκρισιμότητας οι χρήστες της. Με τον τρόπο αυτόν η εφαρμογή μας καλύπτει κενά ασφαλείας λόγω τρίτων έγκαιρα, αυξάνει τις δυνατότητες που μπορεί να αξιοποιήσει και να παρέχει και βελτιώνει τους χρόνους απόκρισής της.

Η δημιουργία backup αφορά την αποθήκευση σε δίσκους και εκτός του ενεργού συστήματος παλαιότερων δεδομένων τον οποίων η διατήρηση θεωρείται σκόπιμη, ωστόσο η ανάκτησή τους γίνεται με πολύ μικρή συχνότητα, ή και γενικότερα των δεδομένων που είναι αποθηκευμένα στη Βάση Δεδομένων μας. Μέσω του backup εξασφαλίζεται ότι σε περίπτωση ακραίας κατάστασης που θα οδηγήσει σε απώλεια των δεδομένων ή μέρους αυτών θα είναι δυνατή η ανάκτησή τους (βλέπε ασφάλεια). Ακόμη επιτρέπει την απομάκρυνση ορισμένων δεδομένων από το σύστημα μειώνοντας τον απαιτούμενο "γρήγορο" αποθηκευτικό χώρο και αυξάνοντας παράλληλα την ταχύτητα απόκρισης του συστήματος.

Τέλος, ο εκσυγχρονισμός της διεπαφής της διαδικτυακής εφαρμογής και γενικά του συστήματος, αφορά τις αλλαγές που έχουν ως στόχο τις βελτιώσεις και γενικά την προσαρμογή στις μεταβαλλόμενες και αυξανόμενες απαιτήσεις των χρηστών. Έτσι το OPT θα δείχνει συνεχώς ανανεωμένο, μοντέρνο και ενεργό, ενώ οι χρήστες του θα αισθάνονται ικανοποιημένοι από τις υπηρεσίες που θα τους παρέχονται και ενεργά μέλη μίας κοινότητας καταναλωτών που θα διαχειριζόμαστε, καθώς το feedback τους θα εξετάζεται και θα λαμβάνεται ουσιαστικά υπόψη στις τροποποιήσεις του λογισμικού.

## Παράρτημα

### 4.1 Παραδοχές και εξαρτήσεις

Είναι πιθανό ορισμένα καταστήματα να επιθυμούν τη μη τήρηση στοιχείων (διαθέσιμα προϊόντα και τιμές) σχετικά με αυτά από το σύστημά μας. Στην περίπτωση αυτή θα ήταν συνετή η συμμόρφωσή μας με την επιθυμία τους έπειτα από σχετική αναφορά με απομάκρυνση των σχετικών δεδομένων από το σύστημά μας. Ακόμη τυχόν ψευδή ή ασύμβατα με τη σελίδα μας στοιχεία μπορούν να αναφέρονται σε εμάς ώστε να απομακρύνονται.

### 4.2 Ακρωνύμια και συντομογραφίες

Ως συντομογραφία για το παρατηρητήριο τιμών μας (Online Price Tracker) χρησιμοποιείται το OPT.

#### 4.3 Υποστηρικτικά έγγραφα, πρότυπα κ.λπ.

Δεν έγινε χρήση κάποιου υποστηρικτικού εγγράφου ή προτύπου.