

**Class-diagram-v1.0**

**Project name: “Κολοκυθάκι”**

### **Σύνθεση Ομάδας**

Νίκος Καπετανίδης, 1054290, 5ο έτος

Νίκος Παπαλεξανδρής, 1054427, 5ο έτος

Βασίλης Τσαρουχάς 1059553, 4ο έτος

Αριστείδης Αλεξόπουλος, 1046285, 4ο έτος

**Συγγραφή:** Αριστείδης Αλεξόπουλος, Νίκος Καπετανίδης,  
Τσαρουχάς Βασίλειος, Νίκος Παπαλεξανδρής

**Επιμέλεια:** Αριστείδης Αλεξόπουλος, Νίκος Καπετανίδης,  
Τσαρουχάς Βασίλειος, Νίκος Παπαλεξανδρής

## Περιγραφή:

Στο παρόν έγγραφο θα αναφερθούμε στις αλλαγές που θα γίνουν στο domain model v0.3 μετά την συγγραφή του κώδικα του project. Έτσι καταλήγουμε στην πρώτη έκδοση του class diagram. Οι αλλαγές αυτές είναι ως επί το πλείστο αλλαγές στα ονόματα των attributes και των συναρτήσεων, καθώς έχει γίνει και προσθήκη μερικών επιπλέον συναρτήσεων στις κλάσεις μας. Οι αλλαγές είναι χρωματισμένες με μπλε χρώμα έτσι ώστε να είναι εύκολα διακριτές.

## Κλάσεις:

Το domain της ιστοσελίδας μας “Κολοκυθάκι” αποτελείται από κάποιες κλάσεις. Αυτές αναγράφονται παρακάτω, ακολουθούμενες από μια σύντομη περιγραφή σχετικά με το τι αντιπροσωπεύουν.

**User:** Τον άτομο που έχει δημιουργήσει λογαριασμό στην ιστοσελίδα. Αποτελεί την root κλάση της οντολογίας. Έγινε αλλαγή του ονόματος από Basic User σε user.

Αλλαγές v0.3: Προσδέθηκαν τα attributes που φαίνονται στο σχήμα καθώς και η συναρτήσεις διαθέσιμες στον basic User.

Attributes:

+Username:String : Το όνομα χρήστη που έχει επιλέξει ο χρήστης κατά την εγγραφή του. Είναι public (+) επειδή θέλουμε όλοι να έχουν πρόσβαση σε αυτό. Επίσης είναι τύπου string.

-pwd:String : Ο κωδικός που έχει επιλέξει ο χρήστης κατά την εγγραφή του. Είναι private (-) επειδή δεν θέλουμε όλοι να έχουν πρόσβαση σε

αυτό. Επίσης είναι τύπου string.

-email:String : Το email εγγραφής του χρήστη. Είναι private (-) επειδή θέλουμε όλοι να έχουν πρόσβαση σε αυτό. Επίσης είναι τύπου string.

-Birth\_Day:Date : Η ημερομηνία γέννησης του χρήστη. Είναι private (-) επειδή δεν θέλουμε όλοι να έχουν πρόσβαση σε αυτό. Επίσης είναι τύπου date.

-Credit Card:String : Ο αριθμός της κάρτας του χρήστη που χρησιμοποιεί για τις αγορές του. Είναι private (-) επειδή δεν θέλουμε όλοι να έχουν πρόσβαση σε αυτό. Επίσης είναι τύπου string.

-City:String : Η πόλη στην οποία κατοικεί ο χρήστης. Είναι private (-) επειδή δεν θέλουμε όλοι να έχουν πρόσβαση σε αυτό. Επίσης είναι τύπου string.

-Address:String : Η διεύθυνση του χρήστη. Είναι private (-) επειδή δεν θέλουμε όλοι να έχουν πρόσβαση σε αυτό. Επίσης είναι τύπου string.

-Name:String : Το όνομα του χρήστη. Είναι private (-) επειδή δεν θέλουμε όλοι να έχουν πρόσβαση σε αυτό. Επίσης είναι τύπου string.

-Iname:String : Το επώνυμο του χρήστη. Είναι private (-) επειδή δεν θέλουμε όλοι να έχουν πρόσβαση σε αυτό. Επίσης είναι τύπου string.

-Phone:String : Το τηλέφωνο του χρήστη. Είναι private (-) επειδή δεν θέλουμε όλοι να έχουν πρόσβαση σε αυτό. Επίσης είναι τύπου string.

-Ban\_Type:enum : Ο τύπος του ban που μπορεί να έχει ένας χρήστης. Αν δεν έχει υποστεί ban είναι null. Είναι private (-) επειδή δεν θέλουμε όλοι να έχουν πρόσβαση σε αυτό. Επίσης είναι τύπου enum.

-Ban\_Time:Date : Η ημερομηνία μέχρι την οποία είναι αποκλεισμένος ένας χρήστης. Αν δεν έχει υποστεί ban είναι null. Είναι private (-) επειδή δεν θέλουμε όλοι να έχουν πρόσβαση σε αυτό. Επίσης είναι τύπου date.

Functions:

+createUser(Username:String, Pwd:String, Email:String, Birth\_Day:Date): : Μέσω αυτής γίνεται η εγγραφή του χρήστη στο site. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Log\_In(Username:String, Pwd:String): Μέσω αυτής ο χρήστης κάνει σύνδεση. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Create\_Recipe(Username:String, Name:String, Description:String, Category:enum, ExecutionTime:time, Difficulty:enum): : Μέσω αυτής ο χρήστης δημιουργεί μια νέα συνταγή. Είναι public (+) επειδή θέλουμε η χρήση

της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Edit\_Recipe(Rec\_Name:String,Description:String, Category:enum,ExecutionTime:time, Difficulty:enum): : Μέσω αυτής ο χρήστης επεξεργάζεται μια συνταγή που έχει ήδη δημιουργήσει. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void. Έγινε διόρθωση από Name σε Rec\_Name, διότι υπήρχε λάθος.

+Create\_List(Listname:String): : Μέσω αυτής ο χρήστης φτιάχνει μια νέα λίστα. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Search():Recipe : Μέσω αυτής ο χρήστης κάνει αναζήτηση στο site. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης επιστρέφει ως αποτέλεσμα συνταγές, οπότε ο τύπος επιστροφής είναι αντικείμενα τύπου Recipe.

+Add\_to\_Basket(Username:String, ItemID:int): : Μέσω αυτής ο χρήστης προσθέτει αντικείμενα στο καλάθι του. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Remove\_From\_Basket(Username:String, ItemID:int): : Μέσω αυτής ο χρήστης αφαιρεί προϊόντα από το καλάθι του. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Place\_Order(Username:String, OrderID:int): : Μέσω αυτής ο χρήστης τοποθετεί μια νέα παραγγελία. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Payment(Username:String, OrderID:int): : Μέσω αυτής ο χρήστης εξοφλεί το ποσό μιας παραγγελίας. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Remove\_Comment(Username:String, CommentID:int): : Μέσω αυτής ο χρήστης αφαιρεί ένα σχόλιο του. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Remove\_Recipe(Username:String, RecID:int): : Μέσω αυτής ο χρήστης διαγράφει μια συνταγή που έχει συγγράψει. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Add\_Rating(Username:String, RecID:int): : Μέσω αυτής ο χρήστης προσθέτει βαθμολογία σε κάποια συνταγή. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Add\_Comment(Username:String, CommentID:int): : Μέσω αυτής ο χρήστης προσθέτει ένα σχόλιο σε κάποια συνταγή. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Add\_Ingredient(IngredientName:String, amount:double, RecID:int): : Μέσω αυτής ο χρήστης προσθέτει ένα συστατικό στην λίστα των συστατικών μια συνταγής. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+emptySignup(name:String, email:String, username:String, pwd:String, pwdRepeat:String):Boolean: Μέσω αυτής γίνεται ο έλεγχος για το αν ο χρήστης έχει συμπληρώσει όλα τα απαιτούμενα πεδία πριν πατήσει το κουμπί για την εγγραφή του. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης επιστρέφει τιμή τύπου Boolean.

+invalidUid(username:String):Boolean: Μέσω αυτής γίνεται ο έλεγχος για το αν ο χρήστης έχει πληκτρολογήσει username το οποίο να πληροί όλα τα κριτήρια. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης επιστρέφει τιμή τύπου Boolean.

+invalidEmail(email:String):Boolean: Μέσω αυτής γίνεται ο έλεγχος για το αν ο χρήστης έχει πληκτρολογήσει το email του σωστά, δηλαδή πρέπει να πληροί όλα τα κριτήρια. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης επιστρέφει τιμή τύπου Boolean.

+pwdMatch(pwd:String, pwdRepeat:String): Boolean: Μέσω αυτής γίνεται ο έλεγχος αν ταυτίζονται τα περιεχόμενα των πεδίων με τον κωδικό και την επαλήθευση του κωδικού. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης επιστρέφει τιμή τύπου Boolean.

+uidExists(conn, username:String, email:String):Boolean: Μέσω αυτής γίνεται ο έλεγχος για το αν ο χρήστης έχει ήδη λογαριασμό στην ιστοσελίδα. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης επιστρέφει τιμή τύπου Boolean.

**Admin:** Τον χρήστη τύπου Διαχειριστή που έχει πρόσβαση στην ιστοσελίδα και διαθέτει κάποιες επιπρόσθετες ικανότητες.

Αλλαγές v0.3: Προσδέθηκαν τα attributes που φαίνονται στο σχήμα καθώς και η συναρτήσεις διαθέσιμες στον Admin.

Attributes:

+isAdmin:boolean : Το αναγνωριστικό που δηλώνει ότι ένας χρήστης είναι διαχειριστής. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου boolean.

Functions:

+Ban\_User(Username:String, BanType:enum, Ban\_Time:Date): : Μέσω αυτής ο διαχειριστής αποκλείει κάποιον χρήστη. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή άρα είναι void.

+Add\_to\_Featurred(RecID:int, Name:String): : Μέσω αυτής ο διαχειριστής προσθέτει κάποια συνταγή στις προτεινόμενες. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή άρα είναι void.

+Remove\_Comment\_Admin(Username:String, CommentID:int): : Μέσω αυτής ο διαχειριστής μπορεί να διαγράψει ένα σχόλιο άλλου χρήστη. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή άρα είναι void.

+Remove\_Recipe\_Admin(RecID:int, Name:String): : Μέσω αυτής ο διαχειριστής μπορεί να διαγράψει συνταγή κάποιου χρήστη. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή άρα είναι void.

**Recipe:** Τις διαφορετικές συνταγές που είναι καταχωρημένες στο Συνταγοβιβλίο.

Αλλαγές v0.3: Προστέθηκαν τα attributes που φαίνονται στο σχήμα καθώς και οι συναρτήσεις διαθέσιμες στην κλάση Recipe. Επίσης έγινε αλλαγή του ονόματος σε Recipe από Συνταγή.

Attributes:

+Username:String : Το όνομα χρήστη του χρήστη που θα καταχωρίσει την συνταγή. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου string.

+Rec\_Name:String : Το όνομα της συνταγής που καταχωρεί ο χρήστης. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου string.

-RecID:int Το αναγνωριστικό id της συνταγής που καταχωρίζεται. Είναι private (-) επειδή δεν θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου int.

+Description:String : Η περιγραφή της συνταγής που καταχωρείται. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου string.

+Category:enum : Η κατηγορία στην οποία ανήκει η συνταγή. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου enum.

+ExecutionTime:time : Ο χρόνος εκτέλεσης που δηλώνει ο συγγραφέας της συνταγής. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου time.

+Difficulty:enum : Ο βαθμός δυσκολίας που δηλώνει ο συγγραφέας της συνταγής. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου string.

+Season:enum : Η εποχή στην οποία ταιριάζει η συνταγή που δηλώνει ο συγγραφέας της συνταγής, εάν το επιθυμεί. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου string.

+Ingredient\_List:Struct list : Η λίστα των υλικών που απαιτούνται που δηλώνει ο συγγραφέας της εργασίας. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου string.

Functions: Δεν διαθέτει συναρτήσεις.

Το Ingredient\_List είναι μια λίστα τύπου struct η οποία περιέχει το όνομα του υλικού, το είδος του και επίσης τη ποσότητα του.

```
Struct ingredient(){
```

```
    String name;
```

```
    double amount;
```



```
Enum Type;  
}
```

**Ingredients:** Τα διάφορα υλικά που δίνονται ως περιεχόμενο σε κάθε συνταγή. Έχει μετατραπεί σε attribute της κλάσης Συνταγή, αφού διορθώσαμε πρώτα την σχέση που είχε με αυτήν από Generalization σε Association.

**Σκεύος:** Τα σκεύη που απαιτούνται για την υλοποίηση της συνταγής. Έχει μετατραπεί σε attribute της κλάσης Συνταγή, αφού διορθώσαμε πρώτα την σχέση που είχε με αυτήν από Generalization σε Association.

Αλλαγές v0.3: Αφαιρέθηκε λόγω αλλαγής στη λειτουργία του site.

**Υποκατηγορία Συνταγής:** Τις διαφορετικές κατηγορίες στις οποίες ανήκουν οι συνταγές.

Αλλαγές v0.3: Προστέθηκε σαν attribute στη κλάση Recipe.

**Review:** Τις κριτικές τις οποίες καταχωρούν οι χρήστες για τις συνταγές.

Αλλαγές v0.3: Αλλαγή του ονόματος σε Review από Κριτική που ήταν προηγουμένως. Προστέθηκαν τα attributes που φαίνονται στο σχήμα καθώς και οι συναρτήσεις στην κλάση Review.

Attributes:

+ CommentID:int : Το αναγνωριστικό id των σχολίων που αφήνουν οι χρήστες. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου int.

+Rating:enum : Η κριτική που θα αφήσει κάθε χρήστης. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου enum.

+Average\_Rating:double : Ο μέσος όρος των βαθμολογιών των χρηστών. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου double.

-RecID:int : Το αναγνωριστικό id της κάθε συνταγής. Είναι private (-)

επειδή δεν θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου int.

Functions:

+Calcutate\_Average\_Rating(RecID:int, Rating:enum):double : Μέσω αυτής γίνεται ο υπολογισμός του μέσου όρου των βαθμολογιών. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης επιστρέφει τιμή τύπου double.

**Comment:** Τα σχόλια τα οποία καταχωρούν οι άλλοι χρήστες κάτω από τις συνταγές.

Αλλαγές v0.3: Αλλαγή του ονόματος σε Comment από Σχόλιο. Προστέθηκαν τα attributes που φαίνονται στο σχήμα καθώς και οι συναρτήσεις στην κλάση Comment.

Attributes:

+ Username:String : Το όνομα χρήστη του χρήστη που κάνει το σχόλιο. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου String.

+Text:String : Το περιεχόμενο του σχολίου του χρήστη. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου String.

+Date:date : Την ημερομηνία την οποία ο χρήστης θα καταχωρίσει το σχόλιο. Είναι public (+) επειδή θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου date.

Functions: Δεν διαθέτει συναρτήσεις.

~~**Rating:** Την βαθμολογία που καταχωρούν οι άλλοι χρήστες για κάποια συνταγή.~~

Αλλαγές v0.3: Προστέθηκε σαν attribute στην κλάση Review.

**Difficulty:** Τον βαθμό δυσκολίας μια συνταγής τον οποίο δηλώνει ο συντάκτης της κατά την καταχώρηση. Έχει μετατραπεί σε attribute της κλάσης Συνταγή, αφού διορθώσαμε πρώτα την σχέση που είχε. Ήταν

ενωμένη με την κλάση Κριτική μέσω Generalization. Έγινε αλλαγή σε σχέση Association με την κλάση Συνταγή.

Αλλαγές v0.3: Αλλαγή του ονόματος σε Difficulty από Δυσκολία.  
Προστέθηκε σαν attribute στην κλάση Recipe.

~~**Average Rating:** Τον μέσο όρο των βαθμολογιών που έχει λάβει μια συνταγή από όλους όσους την υλοποίησαν.~~

Αλλαγές v0.3: Προστέθηκε σαν συνάρτηση στην κλάση Recipe.

**List:** Την λίστα που περιλαμβάνει τις αποθηκευμένες/ αγαπημένες συνταγές του κάθε χρήστη.

Αλλαγές v0.3: Μετατράπηκε στη κλάση List από Favorites Book.  
Προσδέθηκαν τα attributes που φαίνονται στο σχήμα.

Attributes:

+Listname:String : Το όνομα της λίστας που δημιουργεί ο χρήστης για να αποθηκεύει συνταγές. Είναι public (+) γιατί θέλουμε να έχουν όλοι πρόσβαση σε αυτό. Επίσης επιστρέφει αντικείμενα Recipe.

-isPrivate:Boolean : Ο τύπος της λίστας. Μπορεί να είναι private ή public. Είναι private (-) επειδή δεν θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου Boolean.

-RecID:int : Το αναγνωριστικό id της συνταγής που έχει προσθέσει ο χρήστης στη λίστα του. Είναι private (-) επειδή δεν θέλουμε να έχουν όλοι οι χρήστες πρόσβαση σε αυτό. Επίσης είναι τύπου int.

Functions: Δεν διαθέτει συναρτήσεις.

**Feed:** Την λίστα που εμφανίζει συνταγές στην αρχική σελίδα.

Αλλαγές v0.3: Προσδέθηκαν τα attributes που φαίνονται στο σχήμα καθώς και η συναρτήσεις διαθέσιμες στον Feed.

Attributes:

+FeedID:int : Το αναγνωριστικό id του του feed. Είναι public (+) γιατί θέλουμε να έχουν όλοι πρόσβαση σε αυτό. Επίσης είναι τύπου int.

+RecID:int : Το αναγνωριστικό id της κάθε συνταγής. Χρησιμοποιείται για να εμφανίζεται κάθε συνταγή στο feed. Είναι τύπου public (+) γιατί θέλουμε να έχουν όλοι πρόσβαση σε αυτή. Επίσης είναι τύπου int.

Functions:

+Find\_Recommended(RecID:int, Rec\_Name:String):Recipe : Μέσω αυτής επιστρέφονται οι προτεινόμενες συνταγές. Είναι public (+) γιατί θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης επιστρέφει αντικείμενα Recipe.

+Find\_Most\_Viewed(RecID:int, Rec\_Name:String):Recipe : Μέσω αυτής επιστρέφονται οι πιο δημοφιλείς συνταγές. Είναι public (+) γιατί θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης επιστρέφει αντικείμενα Recipe.

+Find\_Best\_Rated(RecID:int, Rec\_Name:String):Recipe : Μέσω αυτής επιστρέφονται οι συνταγές με τις καλύτερες κριτικές. Είναι public (+) γιατί θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης επιστρέφει αντικείμενα Recipe.

+Find\_Seasonal(RecID:int, Rec\_Name:String, ItemID:int, Season:enum):Recipe : Μέσω αυτής επιστρέφονται συνταγές που ταιριάζουν στην τρέχουσα εποχή. Είναι public (+) γιατί θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης επιστρέφει αντικείμενα Recipe.

~~**Προτεινόμενη Συνταγή:** Την λίστα που εμφανίζει τις προτεινόμενες συνταγές. Έγινε αλλαγή στη σχέση με την κλάση Feed. Από Generalization αλλάξαμε τη σχέση σε Aggregation.~~

Αλλαγές v0.3: Προστέθηκε σαν συνάρτηση στην κλάση Feed. Αλλαγή Ονόματος σε Find\_Recommended.

~~**Καλύτερη Συνταγή:** Την συνταγή η οποία έχει τις καλύτερες κριτικές. Έγινε αλλαγή στη σχέση με την κλάση Feed. Από Generalization αλλάξαμε την σχέση σε Aggregation.~~

Αλλαγές v0.3: Προστέθηκε σαν συνάρτηση στην κλάση Feed. Αλλαγή Ονόματος σε Find\_Best\_Rated.

~~**Most Viewed:** Την συνταγή η οποία έχει τις περισσότερες προβολές. Έγινε αλλαγή στη σχέση με την κλάση Feed. Από Generalization αλλάξαμε τη σχέση σε Aggregation.~~

Αλλαγές v0.3: Προστέθηκε σαν συνάρτηση στην κλάση Feed. Αλλαγή Ονόματος σε Find\_Most\_Viewed.

~~**Εποχιακό Υλικό:** Την λίστα που εμφανίζει συνταγές των οποίων τα υλικά είναι εντός εποχής. Έγινε αλλαγή στην σχέση με την κλάση Feed. Από Generalization αλλάξαμε την σχέση σε Aggregation.~~

Αλλαγές v0.3: Προστέθηκε σαν συνάρτηση στην κλάση Feed. Αλλαγή Ονόματος σε Find\_Seasonal.

~~**Account Settings:** Τις πληροφορίες του κάθε εγγεγραμμένου χρήστη.~~

Αλλαγές v0.3: Τα attributes του προστέθηκαν σαν attributes στην κλάση Basic User.

~~**Username:** Το όνομα χρήστη που έχει δηλώσει ο κάθε χρήστης κατά την εγγραφή του. Έχει μετατραπεί σε attribute της κλάσης Account Settings, αφού διορθώσαμε πρώτα την σχέση που είχε. Ήταν ενωμένη με την αυτήν μέσω Generalization. Έγινε αλλαγή σε σχέση Association με την κλάση Account Settings.~~

~~**Password:** Τον κωδικό πρόσβασης που έχει δηλώσει ο κάθε χρήστης κατά την εγγραφή του. Έχει μετατραπεί σε attribute της κλάσης Account Settings, αφού διορθώσαμε πρώτα την σχέση που είχε. Ήταν ενωμένη με την αυτήν μέσω Generalization. Έγινε αλλαγή σε σχέση Association με την κλάση Account Settings.~~

~~**Email Address:** Το email που έχει δηλώσει ο κάθε χρήστης κατά την εγγραφή του. Έχει μετατραπεί σε attribute της κλάσης Account Settings, αφού διορθώσαμε πρώτα την σχέση που είχε. Ήταν ενωμένη με την αυτήν μέσω Generalization. Έγινε αλλαγή σε σχέση Association με την κλάση Account Settings.~~

**Ημερομηνία Γέννησης:** Την ημερομηνία γέννησης που έχει δηλώσει ο χρήστης στο προφίλ του εάν το επιθυμεί. Έχει μετατραπεί σε attribute της κλάσης Account Settings, αφού διορθώσαμε πρώτα την σχέση που είχε. Ήταν ενωμένη με την αυτήν μέσω Generalization. Έγινε αλλαγή σε σχέση Association με την κλάση Account Settings.

**Credit Card:** Τον αριθμό πιστωτικής κάρτας που έχει δηλώσει ο χρήστης για αγορές από το Marketplace εάν το επιθυμεί. Έχει μετατραπεί σε attribute της κλάσης Account Settings, αφού διορθώσαμε πρώτα την σχέση που είχε. Ήταν ενωμένη με την αυτήν μέσω Generalization. Έγινε αλλαγή σε σχέση Association με την κλάση Account Settings.

**Πόλη:** Την πόλη που μένει ο χρήστης και έχει δηλώσει στο προφίλ του εάν το επιθυμεί. Έχει μετατραπεί σε attribute της κλάσης Account Settings, αφού διορθώσαμε πρώτα την σχέση που είχε. Ήταν ενωμένη με αυτήν μέσω Generalization. Έγινε αλλαγή σε σχέση Association με την κλάση Account Settings.

**Διεύθυνση:** Την διεύθυνση της κατοικίας του χρήστη την οποία έχει δηλώσει στο προφίλ του εάν το επιθυμεί. Έχει μετατραπεί σε attribute της κλάσης Account Settings, αφού διορθώσαμε πρώτα την σχέση που είχε. Ήταν ενωμένη με την κλάση Πόλη μέσω Generalization. Έγινε αλλαγή σε σχέση Association με την κλάση Account Settings.

**Ονοματεπώνυμο:** Το ονοματεπώνυμο του χρήστη, το οποίο έχει δηλώσει στο προφίλ του εάν το επιθυμεί. Διαχωρίστηκε στις ακόλουθες δύο κλάσεις.

**Όνομα:** Το όνομα του χρήστη, το οποίο έχει δηλώσει στο προφίλ του εάν το επιθυμεί. Έχει μετατραπεί σε attribute της κλάσης Account Settings, αφού διορθώσαμε πρώτα την σχέση που είχε. Ήταν ενωμένη με αυτήν μέσω Generalization. Έγινε αλλαγή σε σχέση Association με την κλάση Account Settings.

**Επώνυμο:** Το επώνυμο του χρήστη, το οποίο έχει δηλώσει στο προφίλ του εάν το επιθυμεί. Έχει μετατραπεί σε attribute της κλάσης Account Settings, αφού διορθώσαμε πρώτα την σχέση που είχε. Ήταν ενωμένη με αυτήν μέσω Generalization. Έγινε αλλαγή σε σχέση Association με την κλάση Account Settings.

**Marketplace:** ~~Τη σελίδα (Χώρος) στην οποία εμφανίζονται όλα τα καταστήματα από τα οποία μπορεί ένας Χρήστης να αγοράσει προϊόντα.~~

**Store:** Τα καταστήματα από τα οποία μπορούν οι χρήστες να αγοράσουν προϊόντα.

Αλλαγές v0.3: Μετατράπηκε στην κλάση Store από Κατάστημα που ήταν προηγουμένως. Προστέθηκαν τα attributes που φαίνονται στο σχήμα καθώς και οι συναρτήσεις διαθέσιμες στο Store.

Attributes:

-StoreID:int: Το αναγνωριστικό id του καταχωρημένου καταστήματος. Είναι private (-) επειδή δεν θέλουμε να έχουν πρόσβαση όλοι σε αυτό. Επίσης είναι τύπου int.

-Store\_pwd:String: Ο κωδικός με το οποίο έκανε εγγραφή στο site το κατάστημα. Είναι private (-) επειδή δεν θέλουμε να έχουν πρόσβαση όλοι σε αυτό. Επίσης είναι τύπου string.

-Store\_Email:String: Το email με το οποίο έκανε εγγραφή στο site το κατάστημα. Είναι private (-) επειδή δεν θέλουμε να έχουν πρόσβαση όλοι σε αυτό. Επίσης είναι τύπου string.

+Store\_Name:String: Το όνομα του καταστήματος. Είναι public (+) επειδή θέλουμε να έχουν πρόσβαση όλοι σε αυτό. Επίσης είναι τύπου string.

+Store\_City:String: Η πόλη που έχει δηλώσει το κατάστημα κατά την εγγραφή του. Είναι public (+) επειδή θέλουμε να έχουν πρόσβαση όλοι σε αυτό. Επίσης είναι τύπου string.

+Store\_Address:String: Η διεύθυνση που έχει δηλώσει το κατάστημα κατά την εγγραφή του. Είναι public (+) επειδή θέλουμε να έχουν πρόσβαση όλοι σε αυτό. Επίσης είναι τύπου string.

+Store\_Phone:String: Το τηλέφωνο που έχει δηλώσει το κατάστημα στο οποίο μπορούν οι χρήστες να καλέσουν. Είναι public (+) επειδή θέλουμε να έχουν πρόσβαση όλοι σε αυτό. Επίσης είναι τύπου string.

-Store\_Bank\_Account:String: Ο τραπεζικός λογαριασμός του καταστήματος. Είναι private (-) επειδή δεν θέλουμε να έχουν πρόσβαση όλοι σε αυτό. Επίσης είναι τύπου string.

+Store\_Rating:enum: Η κριτική που έχουν δώσει οι χρήστες στο κάθε κατάστημα. Είναι public (+) επειδή θέλουμε να έχουν πρόσβαση όλοι σε αυτό. Επίσης είναι τύπου enum.

+Store\_Distance:double: Η εμβέλεια εντός της οποίας εξυπηρετεί το κάθε κατάστημα. Είναι public (+) επειδή θέλουμε να έχουν πρόσβαση όλοι σε αυτό. Επίσης είναι τύπου double.

+Store\_Item\_List:Struct list: Η λίστα που περιέχει τα προϊόντα που διαθέτει το κάθε κατάστημα. Είναι public (+) επειδή θέλουμε να έχουν πρόσβαση όλοι σε αυτό. Επίσης είναι τύπου struct list.

Functions:

+Calculate\_Average\_Rating(StoreID:int, Store\_Rating:enum):double: Μέσω αυτής γίνεται υπολογισμός του μέσου όρου των κριτικών των χρηστών. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης είναι τύπου double.

+Edit\_Store(StoreID:int, Store\_Pwd:String, Store\_Email:String, Store\_Name:String, Store\_Phone:String, Store\_Bank\_Account:String): : Μέσω αυτής μπορεί να γίνει επεξεργασία των στοιχείων του καταστήματος. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Add\_Item(ItemID:int, Item\_Stock:int): : Μέσω αυτής γίνεται προσθήκη νέων προϊόντων στο απόθεμα ή αύξηση αυτού, στο απόθεμα του καταστήματος. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Remove\_Item(ItemID:int, Item\_Stock:int): : Μέσω αυτής γίνεται αφαίρεση στο απόθεμα του καταστήματος. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Enter\_Address(Store\_City:String, Store\_Address:String, Store\_Distance:double): : Μέσω αυτής γίνεται η ενημέρωση της διεύθυνσης του καταστήματος. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης δεν επιστρέφει τιμή οπότε είναι void.

+Find\_Stock(ItemID:int, Item\_Stock:int):Item\_Stock : Μέσω αυτής επιστρέφεται το απόθεμα ενός προϊόντος του καταστήματος. Είναι public (+) επειδή θέλουμε η χρήση της να είναι κοινόχρηστη. Επίσης επιστρέφει το Item\_Stock, δηλαδή το απόθεμα που είναι τύπου int.

Το Store\_Item\_List είναι μια λίστα τύπου struct το οποίο περιέχει το όνομα του υλικού, το είδος του, το ID του αντικειμένου, το απόθεμα



του, τη μάρκα του, την τιμή του. Το ItemID στη συνάρτηση Find\_Stock() είναι attribute του store item list που αναφέρθηκε παραπάνω.

```
Struct store_item(){  
    String Item_name;  
    Int Item_id;  
    Enum Item-type;  
    String Item_brand;  
    double Item_price;  
    int item_Stock;  
}
```

~~**Store Item:** Τα διάφορα υλικά που μπορούν να αγοράσουν οι χρήστες. Έγινε αλλαγή στην σχέση που είχε με την κλάση κατάστημα από Generalization σε Composition~~

Αλλαγές v0.3: Έχει προστεθεί σαν attribute στο Store.

~~**Τιμή:** Την τιμή του κάθε υλικού. Έχει μετατραπεί σε attribute της κλάσης Υλικό καθώς έγινε αλλαγή στην μεταξύ τους σχέση από Composition σε Association.~~

~~**Πόλη:** Την πόλη την οποία εδρεύει το κάθε κατάστημα. Έχει μετατραπεί σε attribute της κλάσης Κατάστημα καθώς έγινε αλλαγή στην μεταξύ τους σχέση από Generalization σε Association.~~

~~**Διεύθυνση:** Την διεύθυνση στην οποία βρίσκεται το κάθε κατάστημα. Έχει μετατραπεί σε attribute της κλάσης Κατάστημα, καθώς έγινε σύνδεση με αυτήν με Association. Πριν ήταν ενωμένη με την κλάση Πόλη μέσω Generalization.~~

~~**Ταμείο:** Το ταμείο του κάθε καταστήματος, στο οποίο γίνεται η αγορά των υλικών από τους χρήστες. Έγινε αλλαγή στην σχέση που είχε με την κλάση κατάστημα από Composition σε Aggregation.~~

Αλλαγές v0.3: Αφαιρέθηκε γιατί η λειτουργικότητα της κλάσης αυτής υλοποιείται από τη κλάση Order.

**Basket:** Το καλάθι αγορών του κάθε χρήστη που περιέχει τα προϊόντα επιλογής του από το Marketplace.

Αλλαγές v0.3: Προστέθηκαν τα attributes που φαίνονται στο σχήμα καθώς και η συναρτήσεις διαθέσιμες στον Basket. Το ItemID είναι attribute του Store\_Item\_List που βρίσκεται στο Store.

Attributes:

-StoreID:int: Το αναγνωριστικό id του καταστήματος από το οποίο ο χρήστης θα κάνει την παραγγελία. Είναι private (-) επειδή δεν θέλουμε να έχουν πρόσβαση όλοι σε αυτό.

-ItemID:int: Το αναγνωριστικό id του προϊόντος το οποίο θα αγοράσει ο χρήστης με την παραγγελία του. Είναι private (-) επειδή δεν θέλουμε να έχουν πρόσβαση όλοι σε αυτό.

+Username:String: Το αναγνωριστικό id του χρήστη που θα κάνει την παραγγελία. Χρησιμοποιούμε το όνομα χρήστη με το οποίο έχει εγγραφεί.

-BasketID:int Το αναγνωριστικό id του καλάθιού του χρήστη. Είναι private (-) επειδή δεν θέλουμε να έχουν πρόσβαση όλοι σε αυτό.

Functions:

+Refresh\_Basket(BasketID:int, Username:String, ItemID:int, Amount:int, StoreID:int): Χρησιμοποιείται για να γίνεται ανανέωση του καλάθιού κάθε φορά που ο χρήστης αλλάζει την ποσότητα των προϊόντων που έχει προσθέσει. Είναι public (+) επειδή η χρήση της θέλουμε να είναι κοινόχρηστη. Επίσης δεν επιστρέφει κάποια τιμή για αυτό έχουμε επιλέξει να είναι void.

+Calculate\_Total(StoreID:int, Username:String, ItemID:int, Item\_Price:double, BasketID):double: Επιστρέφει την συνολική αξία των προϊόντων που έχει προσθέσει ο χρήστης στο καλάθι του. Είναι public (+) επειδή η τιμή που επιστρέφει θέλουμε να είναι κοινόχρηστη. Επίσης η τιμή που επιστρέφεται θα είναι τύπου double, όπως δηλαδή και οι τιμές των προϊόντων.

**Order:** Την παραγγελία που υποβάλλει ο κάθε χρήστης για προϊόντα από το Marketplace.

Αλλαγές v0.3: Προστέθηκαν τα attributes που φαίνονται στο σχήμα. Το Item Id είναι attribute του store item list που βρίσκεται στο Store.

Attributes:

- StoreID:int: Το αναγνωριστικό id του καταστήματος από το οποίο ο χρήστης θα κάνει την παραγγελία. Είναι private (-) επειδή δεν θέλουμε να έχουν πρόσβαση όλοι σε αυτό.
- ItemID:int: Το αναγνωριστικό id του προϊόντος το οποίο θα αγοράσει ο χρήστης με την παραγγελία του. Είναι private (-) επειδή δεν θέλουμε να έχουν πρόσβαση όλοι σε αυτό.
- +Username:String: Το αναγνωριστικό id του χρήστη που θα κάνει την παραγγελία. Χρησιμοποιούμε το όνομα χρήστη με το οποίο έχει εγγραφεί. Είναι public (+) επειδή το όνομα χρήστη είναι κοινόχρηστο.
- OrderID:int: Το αναγνωριστικό id της παραγγελίας που θα καταχωρηθεί. Είναι private (-) επειδή δεν θέλουμε να έχουν πρόσβαση όλοι σε αυτό.
- BasketID:int Το αναγνωριστικό id του καλαθιού του χρήστη. Είναι private (-) επειδή δεν θέλουμε να έχουν πρόσβαση όλοι σε αυτό.

Functions: Δεν διαθέτει συναρτήσεις.

Νέες κλάσεις:

~~Δημόσια Λίστα: Την λίστα που συμπεριλαμβάνει τις συνταγές που έχει αποθηκεύσει ο χρήστης και μπορούν να την προβάλουν όλοι οι άλλοι χρήστες.~~

~~Προσωπική Λίστα: Την λίστα που συμπεριλαμβάνει τις συνταγές που έχει αποθηκεύσει ο χρήστης και μπορεί να την προβάλει μόνο ο ίδιος, δηλαδή όλοι οι άλλοι χρήστες δεν μπορούν να έχουν πρόσβαση.~~

~~Create Recipe Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας δημιουργίας μια νέας συνταγής. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο χρήστης δημιουργήσει μια καινούρια συνταγή.~~

- ~~Υλικό: Τα υλικά που δηλώνει ο χρήστης ότι θα χρειαστούν για την συνταγή.~~
- ~~Σκεύος: Τα σκεύη που δηλώνει ο χρήστης ότι θα χρειαστούν για την συνταγή.~~

- ~~Δυσκολία: Την δυσκολία της συνταγής που δημιουργεί ο χρήστης.~~

~~Edit Recipe Handler: Είναι κλάση ειδικού τύπου της κλάσης Create Recipe Handler με την διαφορά ότι, συμπεριλαμβάνει τις πληροφορίες που ήδη έχουν καταχωρηθεί σε μια συνταγή από τον χρήστη, οι οποίες θα επεξεργαστούν για αλλαγή των περιεχομένων της συνταγής.~~

~~Create List Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας δημιουργίας μιας νέας λίστας. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο χρήστης δημιουργήσει μια νέα λίστα.~~

- ~~Όνομα: Το όνομα της λίστας που θα δημιουργηθεί από τον χρήστη.~~
- ~~Τύπος: Ο τύπος της λίστας που θα δημιουργηθεί από τον χρήστη. Μπορεί να είναι Προσωπική ή Δημόσια.~~

~~Login Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας της σύνδεσης ενός χρήστη. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο χρήστης θα κάνει απόπειρα σύνδεσης στο λογαριασμό του.~~

- ~~Username: Το όνομα χρήστη με το οποίο έχει εγγραφεί ο χρήστης.~~
- ~~Password: Ο κωδικός τον οποίο έχει διαλέξει ο χρήστης.~~

~~Signup Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας εγγραφής ενός χρήστη. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο χρήστης κάνει εγγραφή νέου λογαριασμού.~~

- ~~Username: Το όνομα που θα επιλέξει ο χρήστης για αποκτήσει πρόσβαση στην ιστοσελίδα.~~
- ~~Password: Τον κωδικό που θα εισάγει ο χρήστης για να αποκτήσει πρόσβαση στην ιστοσελίδα.~~
- ~~Email Address: Το email το οποίο θα καταχωρίσει ο χρήστης για να αποκτήσει πρόσβαση στην ιστοσελίδα.~~

~~Search Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας αναζήτησης. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο χρήστης κάνει κάποια αναζήτηση.~~

- ~~Keyword: Τις λέξεις κλειδιά που μπορεί να εισάγει ο χρήστης για να περιορίσει τα αποτελέσματα της αναζήτησης.~~
- ~~Rating: Την επιλογή ορίων αξιολόγησης συνταγών από άλλους χρήστες που μπορεί να διαλέξει ο χρήστης για να περιορίσει τα αποτελέσματα της αναζήτησης.~~
- ~~Tag: Τα tags ή αλλιώς ετικέτες τις οποίες έχουν καταχωρήσει οι άλλοι χρήστες, τα οποία μπορεί να επιλέξει ο χρήστης για να περιορίσει τα αποτελέσματα της αναζήτησης.~~

~~Add To Basket Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας προσθήκης προϊόντος στο καλάθι. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο χρήστης προσθέσει ένα προϊόν ενός καταστήματος στο καλάθι του.~~

- ~~Υλικό: Το προϊόν ή υλικό που θα επιλέξει ο χρήστης να αγοράσει.~~
- ~~Ποσότητα: Την ποσότητα του υλικού που θα επιλέξει ο χρήστης να αγοράσει.~~

~~Remove From Basket Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας αφαίρεσης προϊόντος από το καλάθι. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο χρήστης αφαιρέσει ένα προϊόν από το καλάθι του.~~

- ~~Υλικό: Το προϊόν ή υλικό που θα επιλέξει ο χρήστης να αγοράσει.~~

~~Place Order Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας ολοκλήρωσης παραγγελίας. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο χρήστης προχωρήσει στην ολοκλήρωση μιας παραγγελίας.~~

~~Payment Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας πληρωμής παραγγελίας. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο χρήστης προχωρήσει στην πληρωμή μιας παραγγελίας.~~

- ~~Order: Η παραγγελία που θα καταχωρίσει ο χρήστης.~~
- ~~Τρόπος πληρωμής: Ο τρόπος πληρωμής που θα επιλέξει ο χρήστης για να ολοκληρώσει την παραγγελία του.~~

~~Ban User Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας αποκλεισμού χρήστη. Δηλαδή συμπεριλαμβάνει τις~~

πληροφορίες που θα καταχωρηθούν όταν ο διαχειριστής (Admin) επιλέξει να αποκλείσει κάποιον χρήστη.

- ~~Username: Το όνομα χρήστη αυτού που θα αποκλείσει ο διαχειριστής.~~
- ~~Τύπος Ban: Ο τύπος του Ban που θα επιλέξει ο διαχειριστής. Μπορεί να είναι για κάποιο χρονικό διάστημα ή μόνιμο.~~
- ~~Ημερομηνία Λήξης Ban: Την ημερομηνία που ορίζει ο διαχειριστής ότι θα γίνει άρση του αποκλεισμού του λογαριασμού κάποιου χρήστη που έχει γίνει Ban.~~

~~Remove Comment Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας αφαίρεσης σχολίου. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο απλός χρήστης ή ο διαχειριστής επιλέξει να διαγράψει κάποιο σχόλιο.~~

~~Remove Recipe Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας αφαίρεσης συνταγής. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο απλός χρήστης ή ο διαχειριστής επιλέξει να διαγράψει κάποια συνταγή.~~

~~Add To Featured Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας προσθήκης συνταγής στο feed. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο διαχειριστής επιλέξει να προσθέσει μια συνταγή στις προωθούμενες (Feed).~~

~~Edit Store Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας επεξεργασίας του καταστήματος του πωλητή. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο πωλητής επιλέξει να αλλάξει κάποια από τα προϊόντα τα οποία έχει καταχωρίσει στο κατάστημα του.~~

- ~~Υλικό: Τα προϊόντα τα οποία έχει καταχωρίσει ο πωλητής στο κατάστημά του.~~

~~Set Delivery Distance Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας ορισμού εμβέλειας εξυπηρέτησης. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο πωλητής επιλέξει να ορίσει την εμβέλεια μέχρι την οποία εξυπηρετεί το κατάστημα του με διανομή προϊόντων.~~

- ~~Εμβέλεια: Την απόσταση εντός της οποίας εξυπηρετεί κάποιο κατάστημα.~~

~~Add Item Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας προσθήκης προϊόντος στο κατάστημα. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο πωλητής επιλέξει να προσθέσει κάποιο προϊόν στο κατάστημά του.~~

- ~~Υλικό: Τα προϊόντα τα οποία θέλει να προσθέσει ο πωλητής στο κατάστημά του.~~

~~Remove Item Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας αφαίρεσης προϊόντος από το κατάστημα. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο πωλητής επιλέξει να αφαιρέσει κάποιο προϊόν από το κατάστημά του.~~

- ~~Υλικό: Τα προϊόντα τα οποία θέλει να αφαιρέσει ο πωλητής από το κατάστημά του.~~

~~Enter Address Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας προσθήκης διεύθυνσης καταστήματος. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο πωλητής επιλέξει να εισάγει την διεύθυνση του καταστήματός του.~~

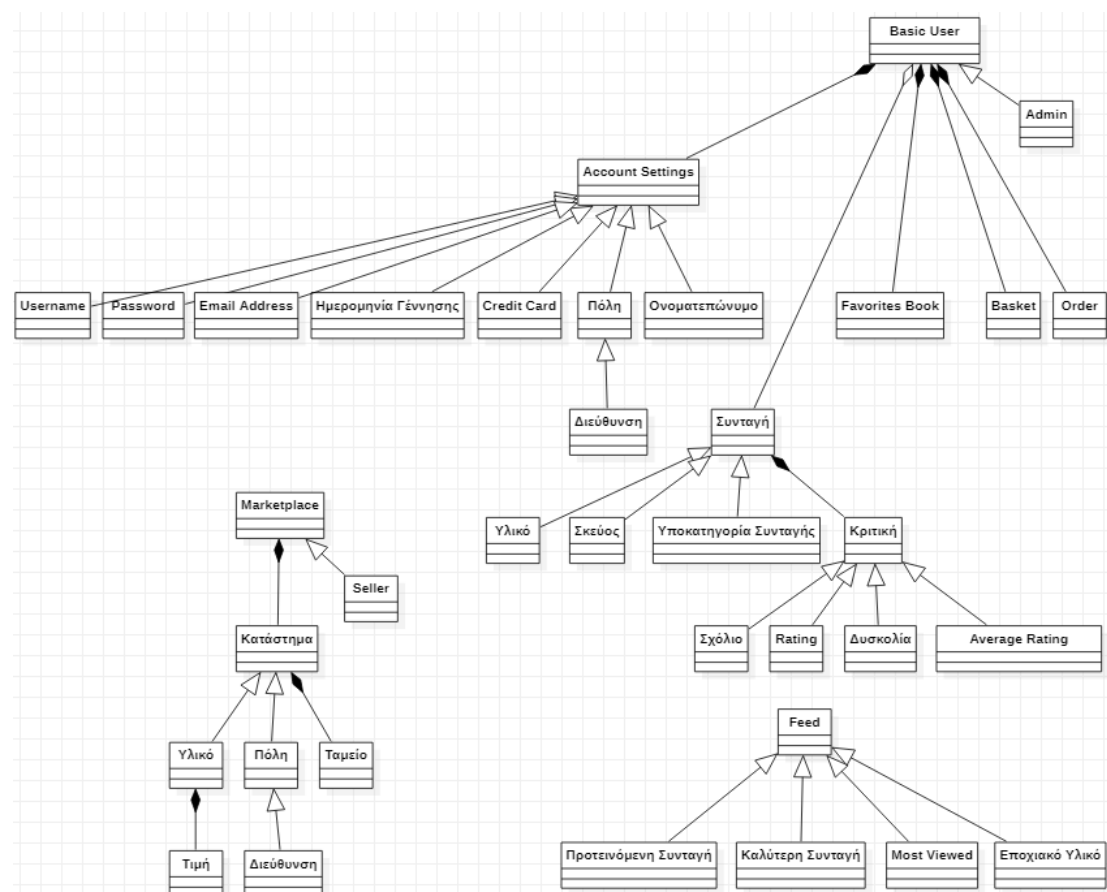
- ~~Διεύθυνση: Η διεύθυνση του καταστήματος του πωλητή, την οποία θα καταχωρίσει για πρώτη φορά.~~

~~Change Address Handler: Η κλάση που είναι υπεύθυνη για την υλοποίηση της διαδικασίας αλλαγής της διεύθυνσης του καταστήματος. Δηλαδή συμπεριλαμβάνει τις πληροφορίες που θα καταχωρηθούν όταν ο πωλητής επιλέξει να εισάγει την διεύθυνση του καταστήματός του.~~

- ~~Διεύθυνση: Η διεύθυνση του καταστήματος του πωλητή, της οποίας το περιεχόμενο θα αλλάξει.~~

Αλλαγές v0.3: Οι παραπάνω κλάσεις προστέθηκαν σαν συναρτήσεις στις κλάσεις Basic User, Admin, Store.

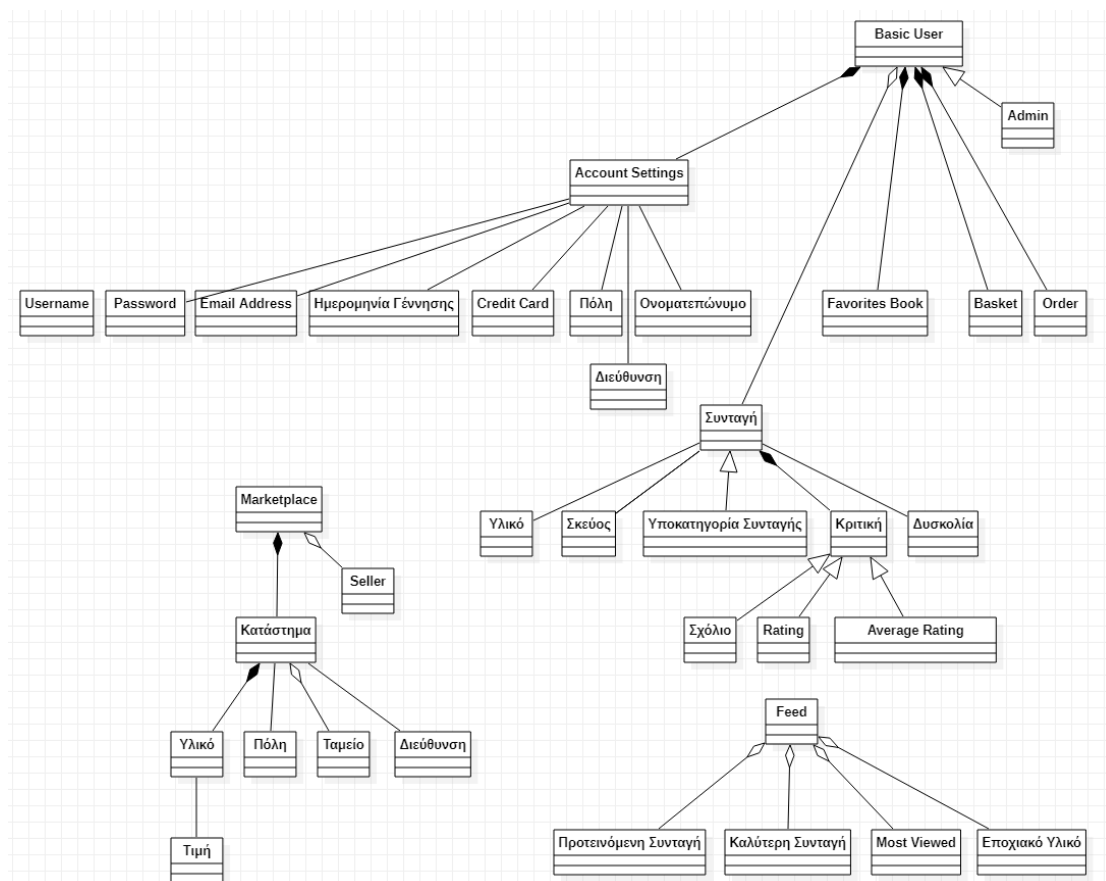
Παρακάτω ακολουθεί το διάγραμμα του Domain Model:



Σχήμα 1.



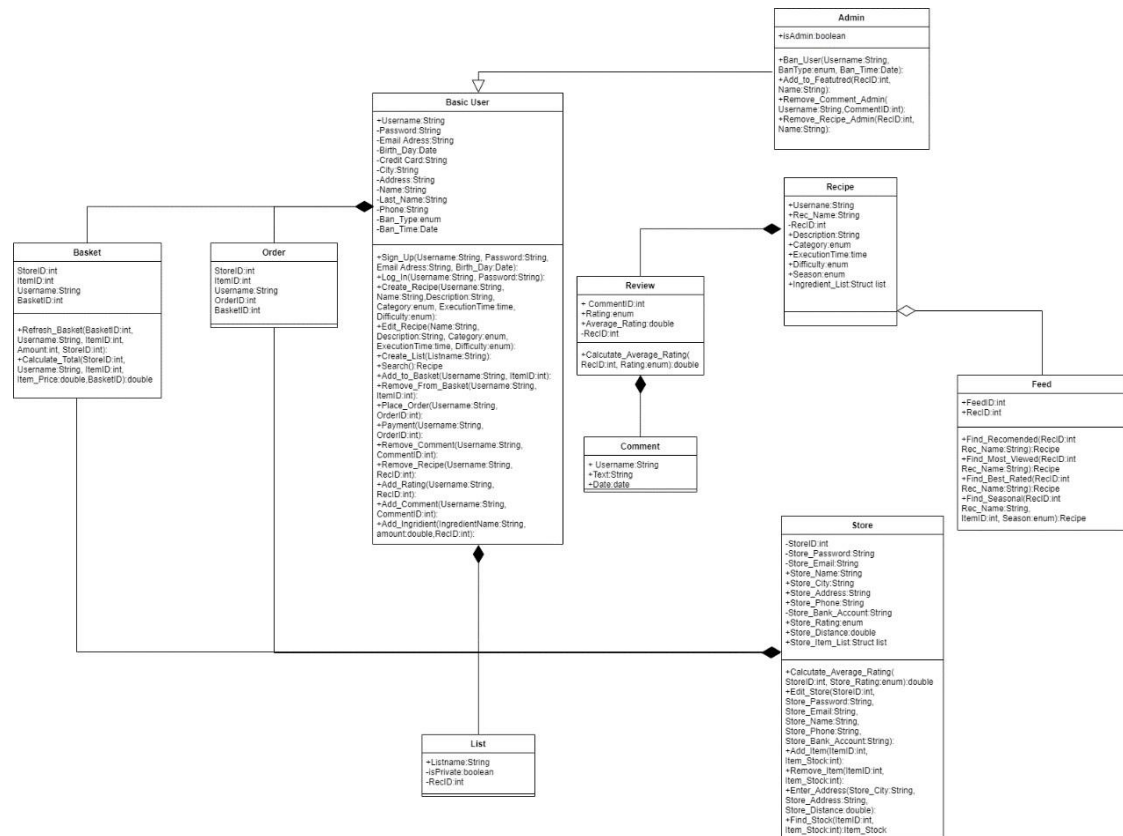
Παρακάτω ακολουθεί το διάγραμμα του Domain Model, μετά τις πρώτες αλλαγές:



Σχήμα 2.

The UML class diagram illustrates the structure of a food delivery system. It features several classes and their relationships:

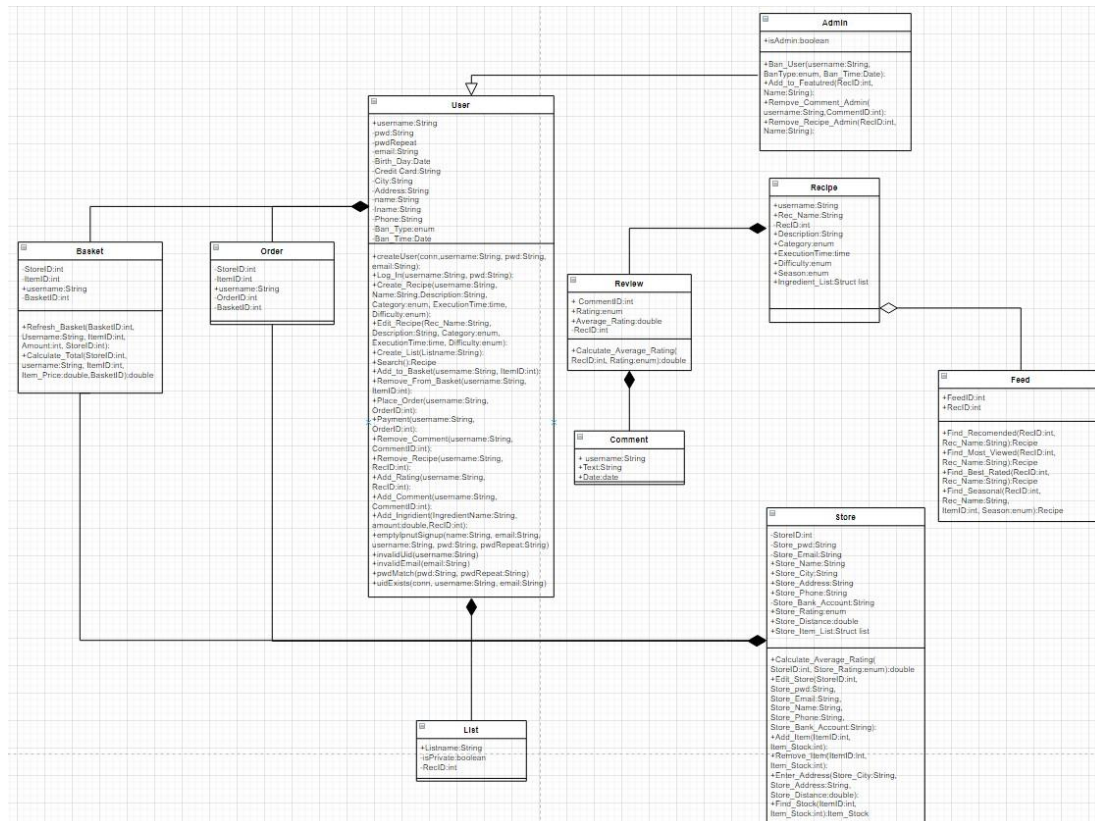
- Handlers:** A series of handler classes at the top, each with specific attributes:
  - Create Recipe Handler:** Yav, Ziliac, Zucuklu
  - Create List Handler:** Okey, Yimec
  - Login Handler:** Username, Password
  - Signup Handler:** Username, Password, Email Address
  - Search Handler:** Keyword, Rating, Tag
  - Add to Basket Handler:** Yav, Yachars
  - Remove from Basket Handler:** Yav
  - Place Order Handler:** (Empty)
  - Payment Handler:** Order, Tiplenik, Miktarlar
- User Roles:**
  - Basic User:** Inherits from **User** (attributes: Username, Password, Email Address, Hurupunlu, Tiplenik). It has a many-to-many relationship with **Menu Item** and a one-to-many relationship with **Order**.
  - Admin:** Inherits from **User**. It has a one-to-many relationship with **Order**.
  - Seller:** Has a one-to-many relationship with **Dish** and a one-to-many relationship with **Food**.
  - Restaurant:** Has a composition relationship with **Dish** and a one-to-many relationship with **Food**.
- Food Items:**
  - Menu Item:** Has attributes Yav, Yach, and Turl. It is linked to **Basic User** and **Seller**.
  - Dish:** Has attributes Yav, Zucuk, and Zucuklu. It is linked to **Seller** and **Restaurant**.
  - Food:** Has a composition relationship with **Dish** and a one-to-many relationship with **Drink**.
  - Drink:** Has attributes Yav, Zucuk, and Zucuklu. It is linked to **Food** and **Restaurant**.
- Other Classes:**
  - Account Settings:** Username, Password, Email Address, Hurupunlu, Tiplenik, Credit Card, Pilm, Zucuklu, Okey, Zucuklu, Emilevce.
  - Order:** Has attributes Yav, Zucuk, and Zucuklu.
  - Payment:** Has attributes Yav, Zucuk, and Zucuklu.
  - Recipe:** Has attributes Yav, Zucuk, and Zucuklu.
  - Recipe Handler:** Yav, Zucuk, Zucuklu.
  - Recipe List Handler:** Yav, Zucuk, Zucuklu.
  - Recipe Search Handler:** Yav, Zucuk, Zucuklu.
  - Recipe Add to Basket Handler:** Yav, Zucuk, Zucuklu.
  - Recipe Remove from Basket Handler:** Yav, Zucuk, Zucuklu.
  - Recipe Place Order Handler:** Yav, Zucuk, Zucuklu.
  - Recipe Payment Handler:** Yav, Zucuk, Zucuklu.



Σχήμα 4.

Παρακάτω ακολουθεί το διάγραμμα του Class Diagram v0.1:

Παραδοχή: Το Class Diagram του Σχήματος 5 το έχουμε συμπεριλάβει και ως εικόνα σε περίπτωση που δεν εμφανίζεται σωστά στο παρόν έγγραφο.



Σχήμα 5.

## Εργαλεία που χρησιμοποιήθηκαν:

Για την δημιουργία του διαγράμματος του Domain Model (Σχήμα 1., Σχήμα 2.) χρησιμοποιήθηκε το πρόγραμμα StarUML.

Για την δημιουργία των διαγραμμάτων (Σχήμα 3. , Σχήμα 4. , [Σχήμα 5.](#)) του Domain Model χρησιμοποιήθηκε το <https://app.diagrams.net>.