

Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Μεταπτυχιακή Διατριβή **στα Πληροφοριακά και Επικοινωνιακά Συστήματα**



Διατροφικό – Διαιτολογικό Λογισμικό
για μη Διατροφολόγους Επαγγελματίες Υγείας

Νικητίδης Νικόλαος

Επιβλέπων Καθηγητής
Ρεφανίδης Ιωάννης

Αύγουστος, 2014

Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

**Διατροφικό – Διαιτολογικό Λογισμικό
για μη Διατροφολόγους Επαγγελματίες Υγείας**

Νικητίδης Νικόλαος

**Επιβλέπων Καθηγητής
Ρεφανίδης Ιωάννης**

Η παρούσα μεταπτυχιακή διατριβή υποβλήθηκε
προς μερική εκπλήρωση των απαιτήσεων για απόκτηση

μεταπτυχιακού τίτλου σπουδών
στα Πληροφοριακά Συστήματα

από τη Σχολή Θετικών και Εφαρμοσμένων Επιστημών
του Ανοικτού Πανεπιστημίου Κύπρου

Αύγουστος, 2014

Περίληψη

Είναι γνωστό ότι τα διατροφικά προβλήματα απασχολούν σε μεγάλο βαθμό τον σύγχρονο άνθρωπο. Από αυτά, το σημαντικότερο είναι η παχυσαρκία, που, με τον έναν ή τον άλλον τρόπο, απασχολεί σχεδόν έναν στους τέσσερις ανθρώπους. Επίσης σχετίζεται με αυξημένη νοσηρότητα και θνησιμότητα.

Οι απαντήσεις που δόθηκαν από την επιστήμη (αλλαγή τρόπου ζωής, μείωση ενεργειακής πρόσληψης, φαρμακοθεραπεία, χειρουργικές παρεμβάσεις) δε φαίνεται να έλυσαν το πρόβλημα. Επίσης, στην πράξη, φαίνεται πως μόνο ένα μικρό ποσοστό ασθενών είναι εφικτό, για διάφορους λόγους, να παρακολουθείται από εξειδικευμένο διαιτολόγο διατροφολόγο.

Με τη μεταπτυχιακή διατριβή αυτή, προτείνεται η δημιουργία ενός διαιτητικού λογισμικού για μη εξειδικευμένους επιστήμονες. Αυτό θα είναι απλό, εύχρηστο και θα μπορεί σε σύντομο χρονικό διάστημα να παρέχει διαιτολογία.

Στο παρόν πόνημα αναφέρεται όλη η διαδικασία που ακολουθείται, ώστε να παραχθεί το εν λόγω λογισμικό. Αρχικά παρουσιάζονται οι υφιστάμενες απαιτήσεις και το μοντέλο πεδίου προβλήματος.

Στη συνέχεια παρουσιάζονται και σχολιάζονται οι επιθυμητές οθόνες συστήματος. Ακολούθως αναφέρονται οι περιπτώσεις χρήσης, δηλαδή σκιαγραφούνται οι λειτουργίες τις οποίες θα μπορεί να επιτελέσει ο χρήστης.

Έπειτα παρέχεται η ανάλυση ευρωστίας, η οποία θα αποτελέσει τη βάση για τη σχεδίαση που υλοποιείται μέσω των διαγραμμάτων ακολουθίας. Τέλος αναφέρονται ορισμένα θέματα σχετικά με την υλοποίηση του προγράμματος.

Summary

It is well known that nutritional problems largely concern the modern man. Among these, the most important is obesity, which concerns, in different ways, nearly one in four people. It is also associated with increased morbidity and mortality.

The answers given by scientists (such as lifestyle changes, reduced energy intake, pharmacotherapy, surgical procedures) do not seem to resolve the problem. It also appears that, practically, only a small percentage of patients, it is feasible, for various reasons, to be monitored by specialized dietitian nutritionist.

In this thesis, the creation of a dietary software for non-specialist scientists is proposed. This will be simple, easy to use and could quickly provide diets.

This essay presents the entire process followed, in order to produce such software. Initially, the existing requirements are presented, as well as the domain model.

Subsequently are presented and discussed the desired system displays. Following this, the situations of use are indicated, I. e. the functions that can be performed by the user are outlined.

Afterwards, the robustness analysis and diagrams are given, which will form the basis for the design, implemented through the sequence diagrams. Finally, certain issues are reported relating to the implementation of the program.

Ευχαριστίες

Επιθυμώ να ευχαριστήσω το Ανοικτό Πανεπιστήμιο Κύπρου για τη δυνατότητα, που μου παρείχε να συμμετάσχω στο παρόν μεταπτυχιακό πρόγραμμα.

Επίσης, τον επιβλέποντα της πτυχιακής διατριβής κ. Ρεφανίδη, για την ουσιαστική επικοινωνία και τα εύστοχα σχόλια.

Τέλος, την οικογένεια μου για την υπομονή και την κατανόηση, που επέδειξε κατά το χρονικό διάστημα εκπόνησης αυτής της διατριβής.

Περιεχόμενα

1	Εισαγωγή	1
1.1	Το Διατροφικό θέμα στην εποχή μας	1
1.2	Παχυσαρκία	2
1.3	Αντιμετώπιση παχυσαρκίας.....	2
1.4	Υφιστάμενα εμπορικά λογισμικά	3
1.5	Σκοπός του προγράμματος.....	4
2	Οδηγίες Χρήσεις	5
2.1	Γενικά.....	5
3	Μοντέλο Πεδίου Προβλήματος	21
3.1	Απαιτήσεις συστήματος.....	21
3.2	Απαιτήσεις.....	22
3.3	Μοντέλο Πεδίου Προβλήματος	24
4	Οθόνες συστήματος	26
4.1	Εισαγωγικά.....	26
4.2	Αρχική οθόνη.....	27
4.3	Οθόνη Μετρήσεις.....	28
4.4	Οθόνη Στόχοι.....	29
4.5	Οθόνη Διατολόγιο.....	30
4.6	Οθόνη Αναζήτηση.....	31
4.7	Οθόνη Στατιστικά.....	32
4.8	Οθόνη Αρχεία.....	33
4.9	Οθόνη Βοήθεια.....	34
4.10	Οθόνη Περί.....	35

5	Περιπτώσεις χρήσης	36
5.1	Εισαγωγικά.....	36
5.2	Καθορισμός ιδανικού βάρους ΠΧ01.....	37
5.3	Καθορισμός κατάστασης βάρους ατόμου ΠΧ02.....	37
5.4	Καθορισμός ποσότητας λίπους σώματός ΠΧ03.....	38
5.5	Καθορισμός ύπαρξης κοιλιακής παχυσαρκίας ΠΧ04.....	38
5.6	Καθορισμός ενέργειας (θερμίδες) ενός τροφίμου ΠΧ05.....	39
5.7	Καθορισμός μακροθρεπτικών χαρακτηριστικών ενός τροφίμου ΠΧ06.....	39
5.8	Καθορισμός θερμίδων - στόχο μιας δίαιτας ΠΧ07.....	40
5.9	Δημιουργία ενός διαιτολογίου ΠΧ08.....	40
5.10	Επίδειξη στατιστικών της δίαιτας ΠΧ09.....	43
5.11	Φόρτωση βάσης δεδομένων ΠΧ10.....	43
5.12	Φόρτωση Διαιτολογίου ΠΧ11.....	44
5.13	Αποθήκευση διαιτολογίου ΠΧ12.....	44
5.14	Αποθήκευση διαιτολογίου σε εκτυπώσιμη μορφή ΠΧ13.....	45
5.15	Λήψη Βοήθειας ΠΧ14.....	45
5.14	Επίδειξη «ταυτότητας» του προγράμματος ΠΧ15.....	46
6	Ανάλυση ευρωστίας	47
6.1	Εισαγωγικά.....	47
6.2	Καθορισμός ιδανικού βάρους ΔΕ01.....	48
6.3	Καθορισμός κατάστασης βάρους ατόμου ΔΕ02.....	49
6.4	Καθορισμός ποσότητας λίπους σώματος ΔΕ03.....	50
6.5	Καθορισμός ύπαρξης κοιλιακής παχυσαρκίας ΔΕ04.....	51
6.6	Καθορισμός ενέργειας (θερμίδες) ενός τροφίμου ΔΕ05.....	52
6.7	Καθορισμός μακροθρεπτικών χαρακτηριστικών ενός τροφίμου ΔΕ06.....	53
6.8	Καθορισμός θερμίδων - στόχο μιας δίαιτας ΔΕ07.....	54
6.9	Δημιουργία ενός διαιτολογίου ΔΕ08.....	55
6.10	Επίδειξη στατιστικών της δίαιτας ΔΕ09.....	62
6.11	Φόρτωση βάσης δεδομένων ΔΕ10.....	63
6.12	Φόρτωση Διαιτολογίου ΔΕ11.....	64
6.13	Αποθήκευση διαιτολογίου ΔΕ12.....	65
6.14	Αποθήκευση διαιτολογίου σε εκτυπώσιμη μορφή ΔΕ13.....	66
6.15	Λήψη Βοήθειας ΔΕ14.....	67
6.16	Επίδειξη «ταυτότητας» του προγράμματος ΔΕ15.....	68

7	Σχεδίαση	69
7.1	Διαγράμματα Ακολουθίας (Sequence Diagrams).....	69
7.2	Καθορισμός ιδανικού βάρους ΔΑ01.....	70
7.3	Καθορισμός κατάστασης βάρους ατόμου ΔΑ02.....	70
7.4	Καθορισμός ποσότητας λίπους σώματος ΔΑ03.....	71
7.5	Καθορισμός ύπαρξης κοιλιακής παχυσαρκίας ΔΑ04.....	71
7.6	Καθορισμός ενέργειας (θερμίδες) ενός τροφίμου ΔΑ05.....	72
7.7	Καθορισμός μακροθρεπτικών χαρακτηριστικών ενός τροφίμου ΔΑ06.....	72
7.8	Καθορισμός θερμίδων - στόχο μιας δίαιτας ΔΑ07.....	73
7.9	Δημιουργία ενός διαιτολογίου ΔΑ08.....	73
7.10	Επίδειξη στατιστικών της δίαιτας ΔΑ09.....	77
7.11	Φόρτωση βάσης δεδομένων ΔΑ10.....	77
7.12	Φόρτωση Διαιτολογίου ΔΑ11.....	78
7.13	Αποθήκευση διαιτολογίου ΔΑ12.....	78
7.14	Αποθήκευση διαιτολογίου σε εκτυπώσιμη μορφή ΔΑ13.....	79
7.15	Λήψη Βοήθειας ΔΑ14.....	79
7.16	Επίδειξη «ταυτότητας» του προγράμματος ΔΑ15.....	80
8	Ανάπτυξη	81
8.1	Γλώσσα προγραμματισμού.....	81
8.2	Υλοποίηση.....	83
8.3	Δημιουργία διαγραμμάτων.....	84
8.4	Εφαρμογή στην πράξη.....	85
9	Επίλογος	86
9.1	Επισημάνσεις.....	86
9.2	Μελλοντικές προεκτάσεις.....	87
	Βιβλιογραφία	88
A	Παράρτημα	A-1
A.1	Ο κώδικας του προγράμματος.....	A-1

Κεφάλαιο 1

Εισαγωγή

1.1 Το Διατροφικό θέμα στην εποχή μας

Στην εποχή μας και στον δυτικό κόσμο, θα αναμέναμε το θέμα της διατροφής να έχει λυθεί τόσο ποσοτικά όσο και ποιοτικά . Δυστυχώς διαπιστώνουμε ότι αυτό δε συμβαίνει.

Στην καθημερινότητα συναντούμε ολοένα και περισσότερο διατροφικές διαταραχές. Αρκετά συχνά απαντώνται περιπτώσεις ελαττωμένης θερμιδικής πρόσληψης, για παράδειγμα σε περιπτώσεις ψυχογενούς ανορεξίας.

Σε άλλες περιπτώσεις συναντούμε αυξημένη ενεργειακή πρόσληψη (ψυχογενής βουλιμία, ιδιοπαθής παχυσαρκία κτλ), ενώ υπάρχουν και περιπτώσεις στις οποίες υπάρχει μεν ορθή ενεργειακή κάλυψη, αλλά το διαιτολόγιο δεν είναι ισορροπημένο όσον αφορά την αναλογία των μακροθρεπτικών συστατικών (πρωτεΐνες,, υδατάνθρακες, λίπη).

1.2 Παχυσαρκία

Ανάμεσα στα διατροφικά προβλήματα δεσπόζουσα θέση κατέχει το πρόβλημα της παχυσαρκίας. Ως παχυσαρκία ορίζεται η κατάσταση κατά την οποία υπάρχει υπερβολική συγκέντρωση λίπους, που μπορεί να επηρεάσει αρνητικά την υγεία WHO 2009 [01].

Η παχυσαρκία σχετίζεται με πολλές παθήσεις, αλλά και με μειωμένο προσδόκιμο επιβίωσης [02].

Η παχυσαρκία είναι ένα από τα σημαντικότερα προβλήματα δημόσια υγείας. και μάλιστα, σύμφωνα με τον Παγκόσμιο Οργανισμό Υγείας (ΠΟΥ), επηρεάζει παγκοσμίως πάνω από 300 εκατομμύρια ανθρώπους [03].

Στις ΗΠΑ αναφέρεται ότι επηρεάζει έναν στους τέσσερις ενήλικες [04].

Στην Ελλάδα, σε μια μελέτη του 2003 σε 4299 παιδιά, βρέθηκε ότι τα αγόρια είναι υπέρβαρα κατά 21,7 % και παχύσαρκα 2,5% Τα κορίτσια υπέρβαρα κατά 6,5% και παχύσαρκα κατά 1,2 % [05].

1.3 Αντιμετώπιση παχυσαρκίας

Για τις διατροφικές διαταραχές έχουν προταθεί διάφορα μέτρα, (αλλαγή συνηθειών ζωής, υποθερμική δίαιτα, φαρμακευτική αγωγή, χειρουργική παρέμβαση) [06].

Φαίνεται ότι οι παρεμβάσεις αυτές, αν και χρήσιμες, δεν είναι ικανές να περιορίσουν το φαινόμενο.

Από τις παραπάνω, η υποθερμική δίαιτα συνδυάζει αρκετά πλεονεκτήματα. Είναι λιγότερο παρεμβατική και, όταν γίνεται με επιμέλεια, έχει αποτελέσματα.

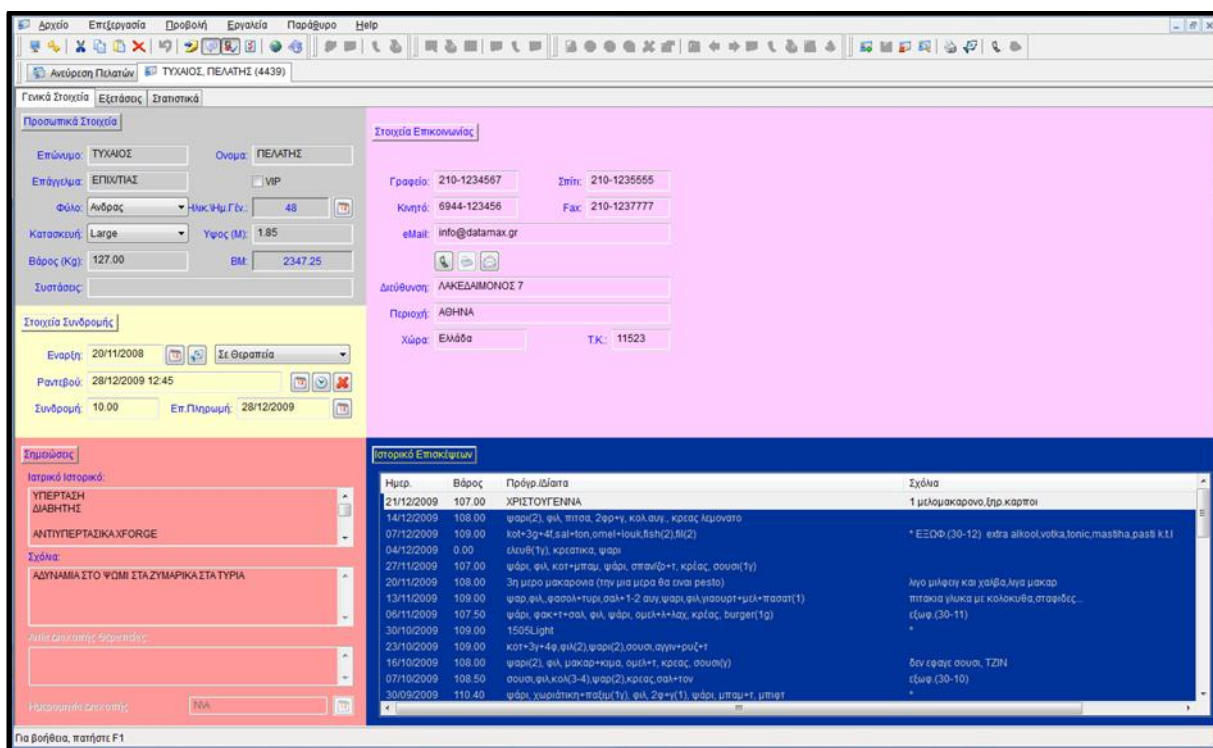
Ωστόσο, η πράξη έχει δείξει ότι ο ελεγχόμενος πληθυσμός από επαγγελματίες διαιτολόγους είναι μικρός. Η σκέψη στην παρούσα εργασία είναι να δημιουργηθεί ένα λογισμικό, που να απευθύνεται σε επαγγελματίες υγείας που έρχονται σε επαφή με μεγάλο όγκο ασθενών, ώστε εύκολα και γρήγορα να παρέχουν διαιτητικές κατευθύνσεις.

1.4 Υφιστάμενα εμπορικά λογισμικά

Στην ελληνική αγορά υπάρχουν λογισμικά πακέτα που στοχεύουν σε επαγγελματίες διαιτολόγους, όπως Dietspeak [07], Dietician [08], e-dietician [09] κτλ.

Ενδεικτικά, ο κατασκευαστής ενός από αυτά τα προγράμματα, περιγράφει το προϊόν του ως εξής: «Το e-dietitian είναι ένα λογισμικό πρόγραμμα (software) διαχείρισης διαιτολογικού γραφείου, προϊόν μιας ομάδας προγραμματιστών / αναλυτών που το σχεδίασε ειδικά για τον επαγγελματία διαιτολόγο / διατροφολόγο».

Αποτιμώντας τα προγράμματα αυτά, μπορούμε να αναφέρουμε ότι είναι πολύπλοκα, με πολλές λειτουργίες, απευθύνονται στον επαγγελματία διαιτολόγο με εξειδικευμένες γνώσεις και ως εμπορικές εφαρμογές κοστίζουν.



Εικόνα 1.1: Θθνή εμπορικού λογισμικού

1.5 Σκοπός του προγράμματος

Επομένως καθίσταται εμφανές ότι τα παραπάνω προγράμματα δεν είναι κατάλληλα για τον μη επαγγελματία διαιτολόγο - διατροφολόγο. Είναι περίπλοκα, δύσχρηστα, έχουν πολλές εξεζητημένες λειτουργίες και απαιτούν πολλές θεωρητικές γνώσεις για να χρησιμοποιηθούν. Επιπλέον κοστίζουν αρκετά.

Υφίσταται, λοιπόν, η ανάγκη δημιουργίας ενός προγράμματος με τις στοιχειώδεις λειτουργίες, χωρίς ιδιαίτερες απαιτήσεις σε υλικό (που θα μπορούσε να λειτουργήσει, μεταξύ άλλων, και στους απαρχειωμένους υπολογιστές, που ακόμη συναντούμε στα νοσοκομεία και στα κέντρα υγείας) και να χρησιμοποιηθεί από άτομα, που δε διαθέτουν την ειδική κατάρτιση του διατροφολόγου διαιτολόγου.

Με τον τρόπο αυτό θα μπορούσε να υποστηριχτεί διατροφικά ο τεράστιος όγκος των περιστατικών, που χρήζουν υποστήριξης, αλλά δεν παρακολουθούνται από διαιτολόγο για διάφορους λόγους (οικονομικούς, προσβασιμότητας κτλ).

Εκτιμάται ότι μαζική προσφορά διατροφικής φροντίδας θα έχει σημαντικά οφέλη στα άτομα, αλλά και στη δημόσια υγεία εν γένει: μείωση της νοσηρότητας και θνησιμότητας, ευεξία, ψυχολογική ευρωστία και αύξηση του προσδόκιμου επιβίωσης, είναι μερικά από αυτά.

Κεφάλαιο 2

Οδηγίες χρήσης

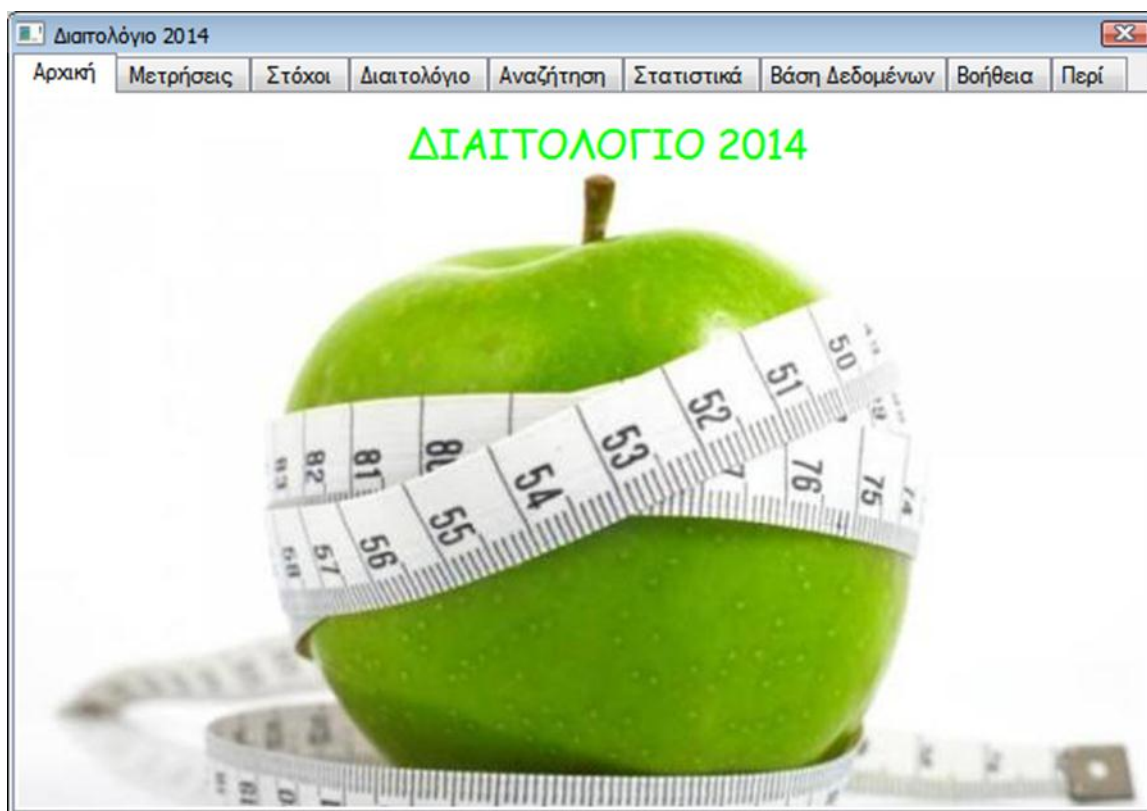
2.1 Γενικά

Παρακάτω παρατίθενται οι οδηγίες χρήσης του προγράμματος. Ακόμη και για τα απλά προγράμματα, είναι χρήσιμο να παρέχονται οι σχετικές οδηγίες, ώστε ο χρήστης να εισάγεται γρηγορότερα στη λογική του προγράμματος, να μπορεί ευκολότερα και γρηγορότερα να επιτελέσει τις διάφορες λειτουργίες και να καθίσταται γνώστης όλων των δυνατοτήτων που παρέχονται.

Δόθηκε ιδιαίτερη μέριμνα στις οδηγίες αυτές, ώστε να αναδεικνύεται το πώς λειτουργεί το σύστημα (ροή του προγράμματος), τι δεδομένα απαιτούνται στην είσοδο, τι αποτελέσματα παρέχονται και πώς αυτά υπολογίζονται.

Έτσι καθίσταται δυνατό, αν κάποιος επιθυμεί να ξαναφτιάξει εκ νέου το λογισμικό, να διαθέτει όλες τις απαραίτητες πληροφορίες. Για την αποφυγή επαναλήψεων τα στοιχεία που παρέχονται στις οδηγίες χρήσης δεν περιγράφονται σε άλλα σημεία της μεταπτυχιακής διατριβής.

ΟΔΗΓΙΕΣ ΧΡΗΣΗΣ



του προγράμματος

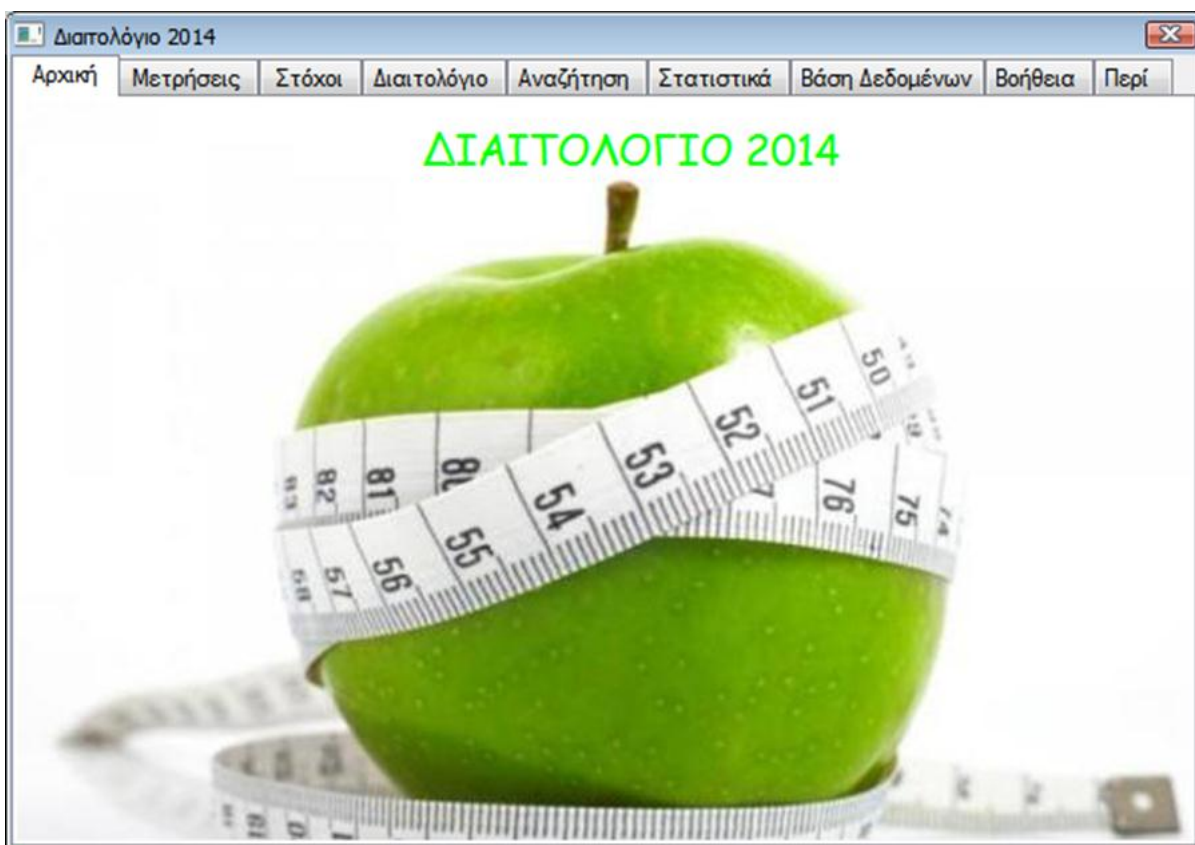
«ΔΙΑΙΤΟΛΟΓΙΟ 2014»

Θεσσαλονίκη 2014

Εισαγωγή

Σκοπός

Το πρόγραμμα φιλοδοξεί να αποτελέσει εργαλείο εκτίμησης και παρέμβασης σε διατροφικά προβλήματα από άτομα που δεν είναι εξειδικευμένοι επιστήμονες στο χώρο της διατροφής. Επομένως, οφείλει να είναι απλό και εύχρηστο, χωρίς να απαιτεί εξειδικευμένες γνώσεις.



Φάσεις – Βήματα υλοποίησης διαιτολογίου

Προκειμένου να υλοποιηθεί μια διαβητολογική παρέμβαση, απαιτούνται οι εξής φάσεις – βήματα.

- Εισαγωγή δεδομένων (Βάρος, ύψος, ηλικία κτλ)
- Υπολογισμοί
- Καθορισμός στόχων
- Υλοποίηση διαιτολογίου
- Έλεγχοι

Παρακάτω εξηγούνται εκτενώς οι αναφερθείσες φάσεις.

Εισαγωγή δεδομένων

Πρόκειται για τη φάση κατά την οποία εισάγει ο χρήστης τα στοιχεία εκείνα που απαιτούνται, ώστε να σχηματιστεί η εικόνα, όσον αφορά την κατάσταση βάρους του ατόμου, που επιθυμούμε να παρέμβουμε διαιτητικά.

Ατομικά Στοιχεία	
Φύλο	<input type="text"/>
Ηλικία [έτη]	<input type="text"/>
Βάρος [Kg]	<input type="text"/>
Ύψος [cm]	<input type="text"/>
Δραστηριότητα	<input type="text"/>

Περιφέρειες	
Ιδανικό Βάρος [Kg]	Ισοζύγιο [Kg]
-	-
ΔΜΣ	Περιγραφή
-	-
Ολικό λίπος [Kg]	Ολικό λίπος [%]
-	-
Άλιπη μάζα [Kg]	Άλιπη μάζα [%]
-	-
Μέση/Ισχία	
-	
Βασικός Μετ.	BM + Φυσ. Δραστ.
-	-

Οθόνη Μετρήσεις

Στη φάση αυτή εισάγονται τα εξής στοιχεία:

- Φύλο (Άντρας ή Γυναίκα)
- Ηλικία (Ετη)
- Βάρος (Κιλά)
- Ύψος (cm)
- Φυσική Δραστηριότητα (Καθιστική ζωή, Μικρή, Μέτρια, Μεγάλη, Πολύ μεγάλη)

Τα στοιχεία αυτά είναι κεφαλαιώδους σημασίας, γιατί χρησιμεύουν στον υπολογισμό δεικτών και παραμέτρων, που θα αποτελέσουν «πυξίδα» για την κατάστρωση του σχεδίου απώλειας βάρους.

Επίσης εισάγονται οι περιφέρειες. Έχουν σημασία οι εξής:

- Περιφέρεια Μέσης
- Περιφέρεια Ισχίων

Οι περιφέρειες παρουσιάζουν ιδιαίτερο ενδιαφέρον, διότι σχετίζονται με την κοιλιακή παχυσαρκία, που συναρτάται με νοσήματα, όπως καρδιοπάθειες και σακχαρώδης διαβήτης.

Πρέπει να σημειωθεί ότι στην οθόνη «Μετρήσεις» εμφανίζονται και τα αποτελέσματα της φάσης των υπολογισμών.

Υπολογισμοί

Με την εισαγωγή των απαραίτητων στοιχείων υπολογίζονται αυτόματα οι ακόλουθες παράμετροι:

The screenshot shows a software window titled "Διαπολόγιο 2014" with a menu bar containing "Αρχική", "Μετρήσεις", "Στόχοι", "Διαιτολόγιο", "Αναζήτηση", "Στατιστικά", "Αρχεία", "Βοήθεια", and "Περί". The main area is divided into several sections:

- Ατομικά Στοιχεία:** Includes input fields for "Φύλο" (set to "Άντρας"), "Ηλικία [έτη]", "Βάρος [Kg]" (96), "Ύψος [cm]" (185), and "Δραστηριότητα".
- Περιφέρειες:** Includes input fields for "Μέσης (cm)" and "Ισχίων (cm)".
- Calculated Metrics:**
 - Ιδανικό Βάρος [Kg]: 83
 - Ισοζύγιο [Kg]: +13
 - ΔΜΣ: 28
 - Περιγραφή: Υπέρβαρος
 - Ολικό λίπος [Kg]: 26
 - Ολικό λίπος [%]: 27
 - Άλιπη μάζα [Kg]: 70
 - Άλιπη μάζα [%]: 73
 - Μέση/Ισχία: -
 - Βασικός Μετ.: -
 - BM + Φυσ. Δραστ.: -

Ιδανικό Βάρος

Είναι το βάρος, που θα έπρεπε να έχει το αξιολογούμενο άτομο. Υπάρχουν διάφορες μέθοδοι που έχουν προταθεί για τον υπολογισμό του. Στο συγκεκριμένο πρόγραμμα χρησιμοποιείται η μέθοδος Hamwi.

Μέθοδος Hamwi

Για τις γυναίκες:

45.2 kg για ύψος μέχρι 1.52 m και 0.89 kg για κάθε cm πάνω από το 1.52 m

Για τους άνδρες:

48 kg για ύψος μέχρι 1.52 m και 1.06 kg για κάθε cm πάνω από το 1.52 m

Ισοζύγιο

Πρόκειται τη διαφορά του τρέχοντος βάρους από το ιδανικό και μετριέται σε κιλά.

$$\text{Ισοζύγιο} = \text{Βάρος} - \text{Ιδανικό Βάρος}$$

Θετικές τιμές σημαίνουν ότι το άτομο έχει παραπάνω κιλά από το ιδανικό, ενώ αρνητικές ότι έχει λιγότερα κιλά από το κανονικό.

Π.χ. ισοζύγιο +13 σημαίνει ότι το άτομο έχει περισσότερα κιλά από το ιδανικό και μάλιστα 13.

Δείκτης μάζας σώματος

Ο δείκτης μάζας σώματος (Body Mass Index, BMI) αποτελεί σημαντικό δείκτη της κατάστασης του ατόμου και υπολογίζεται από τον τύπο:

$$\text{Δείκτης μάζας σώματος} = \text{Βάρος} / \text{Ύψος}^2$$

Βάρος σε Kg, Ύψος σε m

Περιγραφή

Στο σημείο αυτό παρέχεται περιγραφικά η κατάσταση όσον αφορά το βάρος του ατόμου. Οι χαρακτηρισμοί και οι αντίστοιχες τιμές του ΔΜΣ απεικονίζονται παρακάτω.

«Λιποβαρής» ΔΜΣ < 18.5

«Φυσιολογικός» 18.5 >= ΔΜΣ < 25.0

«Υπέρβαρος» 25 <= ΔΜΣ < 30.0

«Ελαφριά παχύσαρκος» 30.0 >= ΔΜΣ < 35.0

«Μέτρια παχύσαρκος» 35.0 >= ΔΜΣ < 40.0

«Βαριά παχύσαρκος» ΔΜΣ >= 40

Ολικό Λίπος

Το ολικό λίπος σώματος υπολογίζεται με διάφορες μεθόδους. Στο συγκεκριμένο πρόγραμμα χρησιμοποιείται η μέθοδος Kvist.

Μέθοδος Kvist

Για Άνδρα: Ολικό Λίπος = (1,36 * (Βάρος / Ύψος) – 42) * 0,923

Για Γυναίκα: Ολικό Λίπος = (1,61 * (Βάρος / Ύψος) – 38.3) * 0,923

Ολικό Λίπος σε Kg, Βάρος σε Kg, Ύψος σε m

Ολικό Λίπος Ποσοστό

Εκφράζει το ποσοστό του βάρους του ολικού λίπους στο συνολικό βάρος του σώματος.

Υπολογίζεται ως εξής:

Ολικό Λίπος Ποσοστό = (Ολικό Λίπος / Βάρος Σώματος) * 100%

Ολικό Λίπος σε Kg, Βάρος Σώματος σε Kg

Άλιπη Μάζα

Η άλιπη μάζα είναι το υπόλοιπο βάρος του σώματος αν αφαιρέσουμε το λίπος.

Υπολογίζεται ως εξής:

Άλιπη Μάζα = Βάρος Σώματος – Ολικό Λίπος

Βάρος σώματος σε Kg, Ολικό Λίπος σε Kg

Άλιπη Μάζα Ποσοστό

Εκφράζει το ποσοστό του βάρους της άλιπης μάζας στο συνολικό βάρος του σώματος.

Υπολογίζεται ως εξής:

Άλιπη Μάζα Ποσοστό = 100 - Ολικό Λίπος Ποσοστό

Περιφέρεια μέσης

Η περιφέρεια μέσης σχετίζεται με σημαντικά νοσήματα, όπως τα καρδιαγγειακά και ο σακχαρώδης διαβήτης. Τα όρια κάτω από τα οποία ο κίνδυνος για εμφάνιση καρδιαγγειακών και διαβήτη τύπου 2 θεωρείται μειωμένος, είναι τα ακόλουθα:

Άνδρας < 102 cm

Γυναίκα < 88 cm

Λόγος Μέσης προς Ισχία

Έχει σημασία να γνωρίζουμε αν υπάρχει κεντρική παχυσαρκία. Αυτή σχετίζεται με καρδιαγγειακά και αυξημένη αντίσταση στην ινσουλίνη, που οδηγεί σε διαβήτη τύπου 2. Τα όρια της κεντρικής παχυσαρκίας φαίνονται παρακάτω:

Όρια κεντρικής παχυσαρκίας

Άνδρες Λόγος Μέσης προς Ισχία > 1

Γυναίκες Λόγος Μέσης προς Ισχία > 0,85

Βασικός Μεταβολισμός

Βασικός μεταβολισμός είναι η ενέργεια που απαιτείται για να εκτελέσει ο οργανισμός τις βασικές του ανάγκες, όπως καρδιακή λειτουργία, κυτταρικές διεργασίες κ.τ.λ.. Στο συγκεκριμένο πρόγραμμα υπολογίζεται από τον τύπο των Harris and Benedict.

Harris and Benedict

Άνδρες $BM = 66 + (13,7 * \text{Βάρος}) + (5 * \text{Ύψος}) - (6,8 * \text{Ηλικία})$

Γυναίκες $BM = 655 + (9,6 * \text{Βάρος}) + (1,85 * \text{Ύψος}) - (4,7 * \text{Ηλικία})$

Βάρος σε Kg, Ύψος σε m, Ηλικία σε έτη

Βασικός Μεταβολισμός Φυσική Δραστηριότητα

Στο Βασικό Μεταβολισμό προστίθεται η ενέργεια που απαιτείται για τη φυσική δραστηριότητα του ατόμου.

Καθορισμός στόχων

Στη φάση αυτή κάνουμε επιλογές όσον αφορά την παρέμβαση που επιθυμούμε (π.χ. καθορισμός της απώλειας κιλών, επιλογή ρυθμού απώλειας κ.α.).

Οθόνη Στόχοι

Σε αυτήν την οθόνη εισάγονται οι επιλογές του χρήστη, όσον αφορά τους στόχους της διαβητολογικής παρέμβασης.

Βάρος [Kgr]	-
Προτεινόμενο Βάρος	-
Επιθυμητό Βάρος	<input type="text"/>
Μεταβολή βάρους Σώματος [g/m]	<input type="text" value="-4000"/>
Μεταβολή πρόσληψης [cal/d]	<input type="text" value="-1000"/>
Διάρκεια [d]	-
BM + ΦΔ	-
Πρόταση [Kcal/d]	-
Ζητούμενο [Kcal/d]	<input type="text"/>

Προτεινόμενο βάρος

Το πρόγραμμα υπολογίζει το βάρος, που αποτελεί τον επιδιωκόμενο στόχο. Υπολογίζεται με βάση $\Delta M\Sigma = 25$, που είναι η τιμή, η οποία παρέχει το υψηλότερο βάρος, που θα μπορούσε να χαρακτηριστεί φυσιολογικό.

Επιθυμητό βάρος

Δίνεται η δυνατότητα στο χρήστη να αποδεχτεί το προτεινόμενο ή να επιλέξει κάποιο άλλο βάρος στόχο.

Μεταβολή βάρους

Δίνεται στον χρήστη η δυνατότητα να επιλέξει πόσο γρήγορα θα επιτευχθεί η μεταβολή βάρους. Π.χ. μια τιμή – 2000 σημαίνει ελάττωση του βάρους (λόγω του «-») και μάλιστα με ρυθμό 2000 gr το μήνα.

ΠΡΟΣΟΧΗ: απώλεια βάρους με ρυθμό μεγαλύτερο του 4000gr ανά μήνα δε συνιστάται.

Διάρκεια διαιτολογικής παρέμβασης

Το πρόγραμμα υπολογίζει αυτόματα τις ημέρες, που θα απαιτηθούν, ώστε να επιτευχθεί η μεταβολή του βάρους με τον ζητούμενο ρυθμό.

Πρόταση προγράμματος για ενέργεια στόχο

Δίνεται η τελική πρόταση σε θερμίδες, με τα δεδομένα που έχουν δοθεί και τα ζητούμενα.

Υπολογίζεται ως εξής:

Πρόταση = Βασικός Μεταβολισμός + Φυσική Δραστηριότητα + Μεταβολή πρόσληψης

Όλες οι τιμές σε Kcal/d

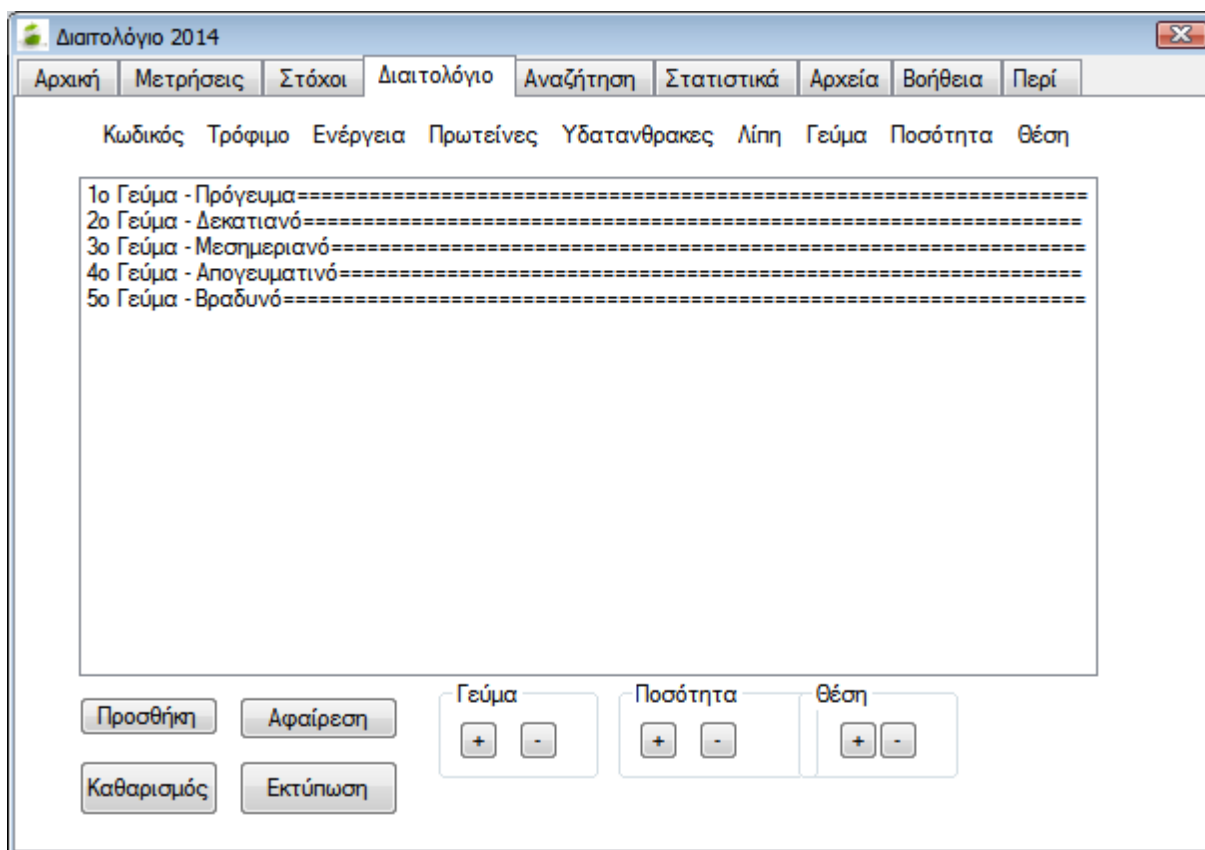
Ζητούμενη ενέργεια

Δίνεται η δυνατότητα στον χρήστη να αποδεχτεί την πρόταση ή να ορίσει δικό του ενεργειακό στόχο.

Υλοποίηση διαιτολογίου

Οθόνη Διαιτολόγιο

Για την υλοποίηση του διαιτολογίου χρησιμοποιείται το παράθυρο ΔΙΑΙΤΟΛΟΓΙΟ.



Κάθε τροφή μπορεί να ανήκει σε ένα από τα 5 γεύματα (πρωινό, δεκατιανό, μεσημεριανό, απογευματινό, προ του ύπνου). Η τροφή μπορεί να έχει διαφορετική ποσότητα από 1, δηλαδή μπορεί να έχει ποσότητα 2 (x2), 3 (x3) κτλ. Η θέση της τροφής εντός του γεύματος μπορεί να αλλάξει. Υποστηρίζονται ο ακόλουθες λειτουργίες:

Προσθήκη: Για την προσθήκη μιας τροφής στο το διαιτολόγιο.

Αφαίρεση: Για την αφαίρεση μιας τροφής από το διαιτολόγιο.

Γεύμα + : αυξάνει το γεύμα στο οποίο ανήκει η τροφή.

Γεύμα - : μειώνει το γεύμα στο οποίο ανήκει η τροφή.

Ποσότητα +: αυξάνει την ποσότητα της τροφής.

Ποσότητα -: μειώνει την ποσότητα της τροφής.

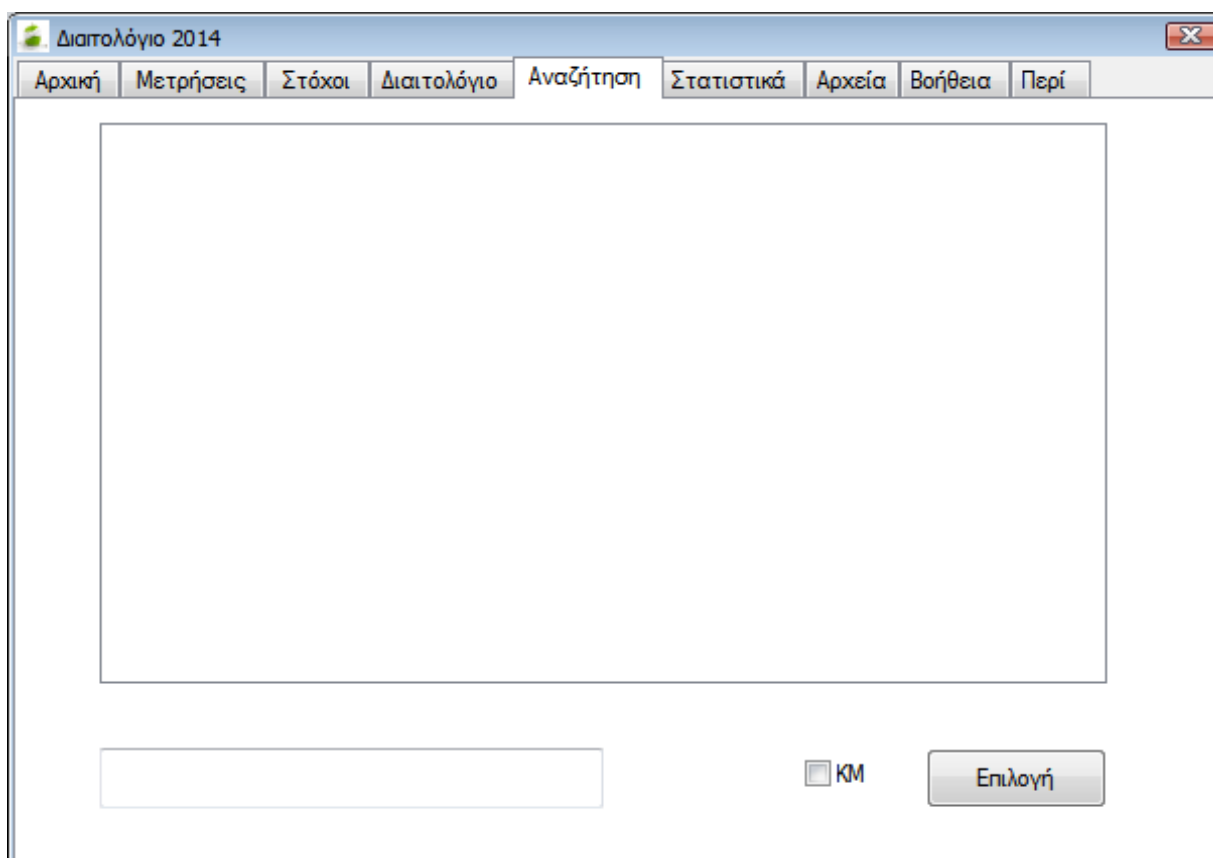
Θέση +: αυξάνει τη θέση στο γεύμα στο οποίο ανήκει η τροφή.

Θέση -: μειώνει τη θέση στο γεύμα στο οποίο ανήκει η τροφή.

ΠΑΡΑΤΗΡΗΣΗ: Προκειμένου να είναι δυνατή η μεταβολή των στοιχείων ενός τροφίμου είναι αυτονόητο ότι πρώτα θα πρέπει να επιλεγεί.

Οθόνη Αναζήτηση

Στο παράθυρο υπάρχει θέση όπου εισάγεται η λέξη κλειδί, για να γίνει αναζήτηση στη βάση δεδομένων τροφών.



Υπάρχει η δυνατότητα να είναι ευαίσθητη η αναζήτηση σε κεφαλαία, μικρά ή να είναι ανεξάρτητη (case sensitive).

Όταν υπάρχουν ευρήματα, αναφέρεται ο αριθμός των ευρημάτων για την πληροφόρηση του χρήστη.

Τέλος, υπάρχει η δυνατότητα επιλογής του συγκεκριμένου τροφίμου και εισαγωγής του στο διαιτολόγιο. Η διαδικασία προσθήκης επαναλαμβάνεται όσες φορές χρειαστεί.

Έλεγχοι

Οι έλεγχοι αποτελούν το τελευταίο βήμα στη δημιουργία του διαιτολογίου και γίνονται μέσω των στατιστικών του διαιτολογίου

Οθόνη Στατιστικά

Στην οθόνη αυτή αναφέρονται τα στατιστικά του διαιτολογίου που έχει διαμορφωθεί. Χρησιμοποιείται για τον έλεγχο και την τροποποίηση του διαιτολογίου, ώστε να πληροί τις προδιαγραφές που έχουν τεθεί.

Κατηγορία	Τροφές	Ενέργεια	%Ενέργεια
Πρόγευμα	2	90.00	4
Δεκατιανό	1	56.00	3
Μεσημεριανό	3	1059.00	50
Απογευματινό	1	274.00	13
Βραδυνό	3	627.00	30
ΣΥΝΟΛΟ	10	2106.00	100

Κατηγορία	Πρωτεΐνη	Υδατ/κες	Λίπος	%Πρωτεΐνη	%Υδατ/κες	%Λίπος
Πρόγευμα	32.00	44.00	0.00	42	58	0
Δεκατιανό	20.00	28.00	0.00	42	58	0
Μεσημεριανό	268.00	388.00	360.00	26	38	35
Απογευματινό	28.00	208.00	36.00	10	76	13
Βραδυνό	64.00	440.00	72.00	11	76	13
ΣΥΝΟΛΟ	412.00	1108.00	468.00	21	56	24

Κάλυψη ζητούμενων ενεργειακών αναγκών: 95.73%

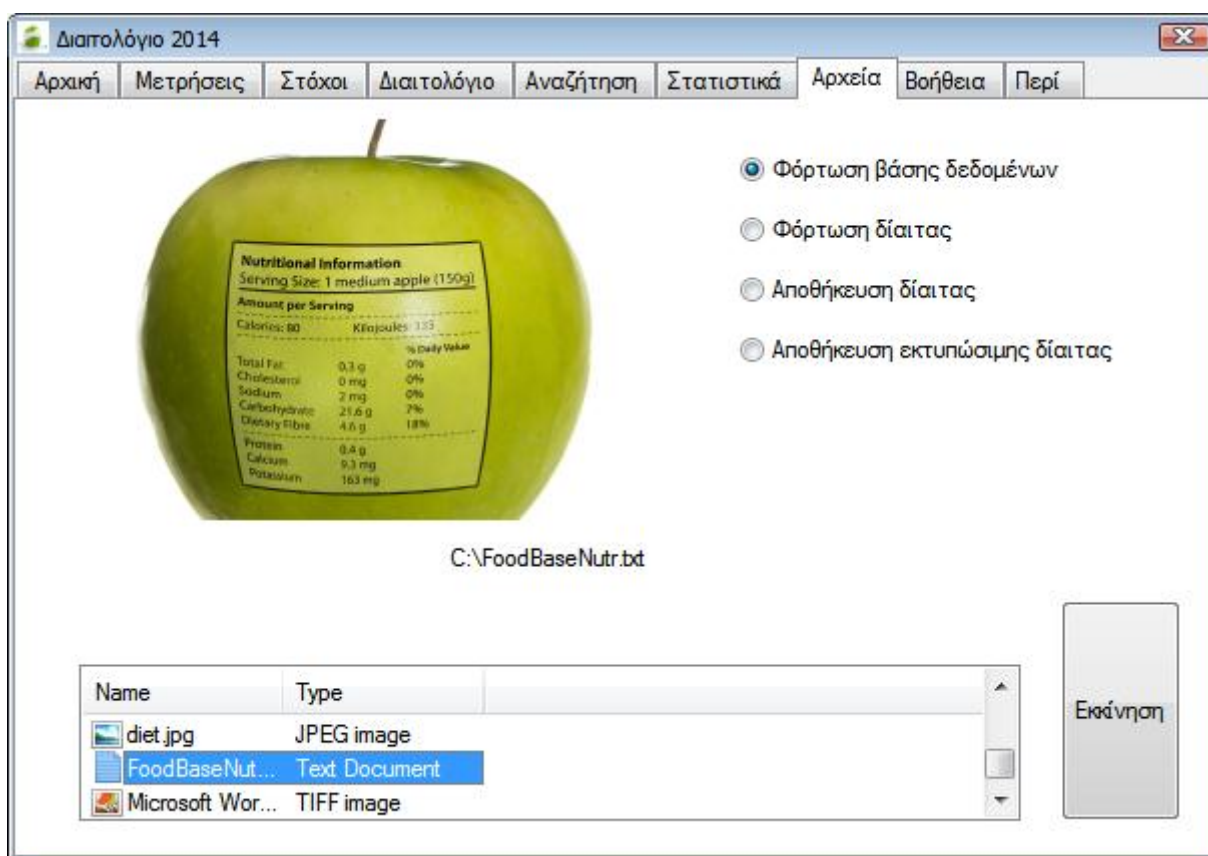
Είναι αυτονόητο ότι, εάν από τα στατιστικά προκύπτει σημαντική απόκλιση από τον στόχο, θα πρέπει να αφαιρεθούν ή / και να προστεθούν τρόφιμα στο διαιτολόγιο, ώστε να αμβλυθούν οι διαφορές.

Άλλες λειτουργίες

Εκτός από τις ανωτέρω λειτουργίες, κεφαλιώδους σημασίας για την υλοποίηση του διαιτολογίου, υπάρχουν και άλλες, που διευκολύνουν τη λειτουργία του προγράμματος. Αυτές παρουσιάζονται με βάση τις οθόνες που τις φιλοξενούν.

Οθόνη Αποθήκευση

Σε αυτή την οθόνη δίδεται η δυνατότητα ανάκτησης ή αποθήκευσης.



Πιο συγκεκριμένα δίνεται η δυνατότητα των παρακάτω ενεργειών:

- Φόρτωση βάσης δεδομένων
- Φόρτωση διαιτολογίου
- Αποθήκευση διαιτολογίου
- Αποθήκευση διαιτολογίου σε εκτυπώσιμη μορφή

Σε όλες τις περιπτώσεις ανάκτησης, επιλέγεται το αρχείο και έπειτα το πλήκτρο «Εκκίνηση», ενώ για τις περιπτώσεις αποθήκευσης, επιλέγεται το όνομα του αρχείου, η θέση στον δίσκο και τέλος το πλήκτρο «Εκκίνηση».

Οθόνη Βοήθεια

Στο παράθυρο Βοήθεια αναφέρονται βασικές πληροφορίες, που θα βοηθήσουν τον χρήστη να χρησιμοποιήσει το πρόγραμμα.



Οθόνη Περί

Στο παράθυρο αυτό παρέχεται η «ταυτότητα» του προγράμματος.



Δηλαδή παρέχονται πληροφορίες για τον σκοπό υλοποίησης του προγράμματος, τον φορέα για τον οποίο υλοποιήθηκε, καθώς και για τον χρόνο ολοκλήρωσής του.

Κεφάλαιο 3

Μοντέλο Πεδίου Προβλήματος

3.1 Απαιτήσεις συστήματος

Το πρώτο στάδιο για την ανάπτυξη ενός πληροφοριακού συστήματος, είναι να οριστούν οι προδιαγραφές του, δηλαδή οι απαιτήσεις του. Πρόκειται για την ανάγκη της περιγραφής του τι πρέπει να κάνει το σύστημα, ανεξάρτητα από την τεχνολογία υλοποίησης [10].

Αφού ορίσουμε τις απαιτήσεις του συστήματος, κατασκευάζουμε το μοντέλο του πεδίου προβλήματος (Domain Modeling), δηλαδή κάνουμε μια γραφική απεικόνιση των οντοτήτων/εννοιών (κλάσεις πεδίου) του συστήματος και των συσχετίσεών τους.

3.2 Απαιτήσεις

Το πληροφοριακό σύστημα που καλούμαστε να αναπτύξουμε πρέπει να μπορεί να παράγει απλά και γρήγορα διαιτολόγια, από χρήστες επιστήμονες υγείας μη διαιτολόγους.

Για τη λειτουργία του, αρχικά χρειάζεται η εισαγωγή των προσωπικών στοιχείων του ατόμου, για το οποίο θα δημιουργηθεί το διαιτολόγιο. Σε αυτά ανήκουν το φύλο, το ύψος, το βάρος, η φυσική δραστηριότητα κτλ. Το πρόγραμμα, βάσει αυτών των στοιχείων, κάνει διάφορους υπολογισμούς παραμέτρων απαραίτητων για το προσδιορισμό της κατάστασης, π.χ. ιδανικό βάρος, δείκτης μάζας σώματος κ.τ.λ., και του ενεργειακού στόχου, που θα έχει το διαιτολόγιο.

Για τη διαμόρφωση του διαιτολογίου, αναζητείται η τροφή ενδιαφέροντος με χρήση λέξεων κλειδιών, επιλέγεται από κατάλογο παρεμφερών περιγραφών η πιο κατάλληλη. Τροποποιείται η ποσότητα και το γεύμα στο οποίο ανήκει, αν απαιτείται. Η διαδικασία επαναλαμβάνεται για κάθε τρόφιμο που θέλουμε να προστεθεί στο διαιτολόγιο.

Για τον έλεγχο του διαιτολογίου το πρόγραμμα παρουσιάζει στο τέλος της διαμόρφωσης του διαιτολογίου, αλλά και οποιαδήποτε άλλη στιγμή επιθυμούμε, τα τρέχοντα στατιστικά του διαιτολογίου. Έτσι οποιαδήποτε παρέμβαση και διόρθωση είναι δυνατή.

Το πρόγραμμα δίνει τη δυνατότητα να χρησιμοποιηθεί και άλλη βάση δεδομένων, πέραν αυτής που αρχικά παρέχεται, οπότε υφίσταται η δυνατότητα φόρτωσης βάσης δεδομένων. Επίσης, υπάρχει η δυνατότητα φόρτωσης δίαιτας, που έχει δημιουργηθεί στο παρελθόν, αλλά και αποθήκευσης δίαιτας που έχει νόημα να διατηρηθεί.

Επιπρόσθετα,, σημαντική θεωρείται και η δυνατότητα εκτύπωσης, οπότε παρέχεται η δυνατότητα αποθήκευσης αρχείου άμεσα εκτυπώσιμου. Επίσης παρέχεται η δυνατότητα εκτύπωσης απευθείας από το πρόγραμμα.

Επιθυμητή θεωρείται η παροχή στοιχειώδους βοήθειας, όπως και η παροχή πληροφοριών για το ίδιο το πρόγραμμα.

- A1** Να επιτρέπει την εισαγωγή ατομικών στοιχείων (φύλο, ηλικία, βάρος, ύψος, περιφέρεια μέσης, περιφέρεια ισχίων κ.τ.λ.)
- A2** Να υπολογίζει χρήσιμες παραμέτρους (ιδανικό βάρος, δείκτης μάζας σώματος, Ολικό λίπος, βασικός μεταβολισμός κ.τ.λ.).
- A3** Να επιτρέπει την εισαγωγή κρίσιμων επιλογών (επιθυμητό βάρος, μεταβολή βάρους σώματος, ενεργειακό ζητούμενο κ.τ.λ.)
- A4** Να επιτρέπει την αναζήτηση ενός τροφίμου μέσα σε βάση δεδομένων με τη χρήση λέξεων κλειδιών.
- A5** Να επιτρέπει την προσθήκη ενός τροφίμου στο διαιτολόγιο
- A6** Να επιτρέπει τη διαγραφή ενός τροφίμου από το διαιτολόγιο
- A7** Να επιτρέπει τον «καθαρισμό» του διαιτολογίου, δηλαδή την αφαίρεση όλων των τροφίμων από αυτό.
- A8** Να επιτρέπει την εκτύπωση του διαιτολογίου.
- A9** Να επιτρέπει την αλλαγή γεύματος του τροφίμου.
- A10** Να επιτρέπει την αλλαγή ποσότητας του τροφίμου.
- A11** Να επιτρέπει την αλλαγή θέσεως του τροφίμου εντός του γεύματος.
- A12** Να παρουσιάζει τα στατιστικά του διαιτολογίου.
- A13** Να επιτρέπει τη φόρτωση βάσης δεδομένων τροφών.
- A14** Να επιτρέπει τη φόρτωση διαιτολογίου.
- A15** Να επιτρέπει την αποθήκευση διαιτολογίου.
- A16** Να επιτρέπει την αποθήκευση διαιτολογίου σε άμεσα εκτυπώσιμη μορφή.
- A17** Να εμφανίζει βοήθεια, όταν ζητείται.

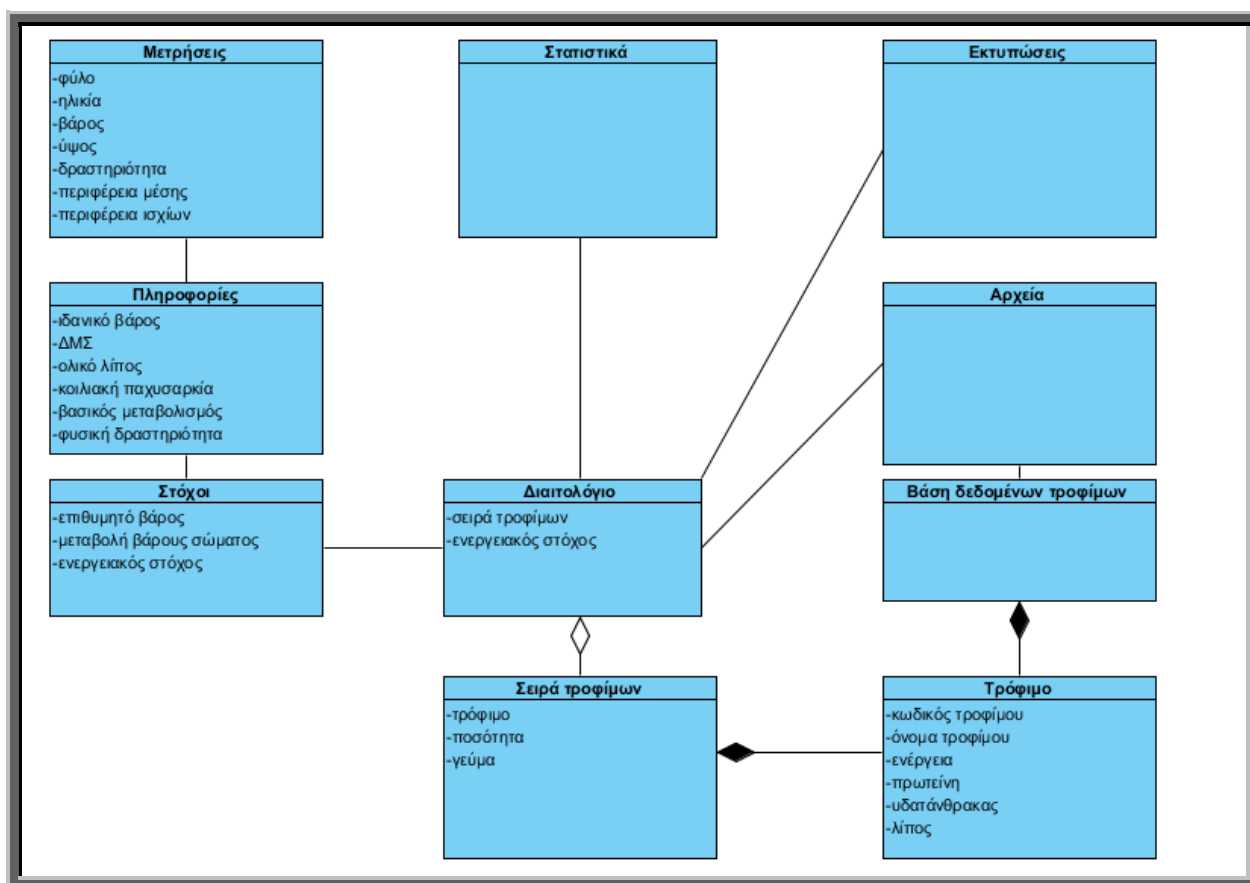
Πίνακας 2.1: Απαιτήσεις

3.3 Μοντέλο Πεδίου Προβλήματος (Domain Modeling)

Το μοντέλο πεδίου προβλήματος είναι δημιούργημα, το οποίο συνεχώς ανανεώνεται κατά την πορεία προς την υλοποίηση, και αντανακλά την τρέχουσα κατανόηση, που έχουμε για το πρόβλημα [11]. Το μοντέλο πεδίου προβλήματος, όπως θα διαφανεί ακολούθως, υλοποιείται με διάγραμμα κλάσεων [12].

Παρακάτω ακολουθεί η λεκτική περιγραφή του Μοντέλου πεδίου προβλήματος

- Ο χρήστης εισάγει στοιχεία και μετρήσεις.
- Από αυτά προκύπτουν πληροφορίες.
- Βάση αυτών επιλέγονται οι στόχοι.
- Με βάση τους στόχους προκύπτει το διαιτολόγιο.
- Το διαιτολόγιο περιλαμβάνει ενεργειακό στόχο και σειρά τροφίμων.
- Η σειρά τροφίμων αποτελείται από τρόφιμα.
- Η βάση δεδομένων αποτελείται από τρόφιμα.
- Η βάση δεδομένων ανακτάται από τα αρχεία.
- Το διαιτολόγιο αποθηκεύεται στα αρχεία.
- Επίσης, το διαιτολόγιο ανακτάται από αυτά.
- Το διαιτολόγιο εκτυπώνεται και προκύπτουν οι εκτυπώσεις.
- Το διαιτολόγιο έχει τα στατιστικά του.



Εικόνα 2.1: Μοντέλο Πεδίου Προβλήματος

Κεφάλαιο 4

Οθόνες συστήματος

4.1 Εισαγωγικά

Με τον όρο «Περιπτώσεις χρήσης», χαρακτηρίζονται όλα τα πιθανά σενάρια, που μπορεί να εκτελέσουν οι χρήστες του συστήματος για να πετύχουν έναν συγκεκριμένο σκοπό.

Στο συγκεκριμένο πρόγραμμα, όπως έχει αναφερθεί, υπάρχει μόνο ένας χρήστης που είναι ο μη εξειδικευμένος στη διατροφή επαγγελματίας υγείας.

4.2 Αρχική οθόνη

Η αρχική οθόνη είναι η οθόνη που πρωτοαντικρίζει ο χρήστης με την εκκίνηση του προγράμματος.



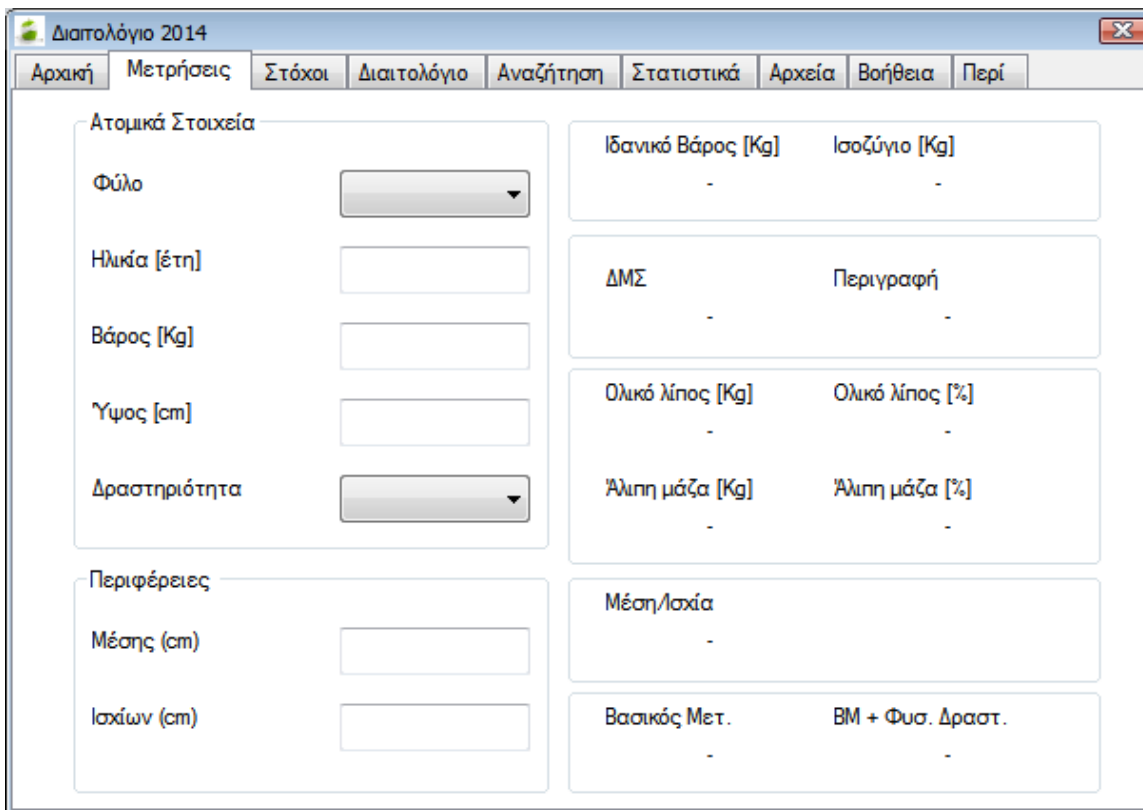
Εικόνα 3.1: Αρχική Οθόνη

Αναφέρει το όνομα του προγράμματος και συνοδεύεται από το κατάλληλο γραφικό.

Δεν έχει κάποια λειτουργικότητα, αλλά φιλοδοξεί να προδιαθέσει ευχάριστα το χρήστη.

4.3 Οθόνη Μετρήσεις

Η οθόνη Μετρήσεις περιλαμβάνει τρία τμήματα.



Εικόνα 3.2: Οθόνη Μετρήσεις

Στο *πρώτο* υπάρχουν τα προσωπικά στοιχεία του ατόμου, για το οποίο θα δημιουργηθεί το διατολόγιο. Συγκεκριμένα περιλαμβάνονται το φύλο (άντρας ή γυναίκα), η ηλικία σε έτη, το βάρος σε κιλά, το ύψος σε εκατοστόμετρα και η φυσική δραστηριότητα.

Στο *δεύτερο* τμήμα υπάρχουν οι περιφέρειες. Σε αυτό το τμήμα περιλαμβάνεται η περιφέρεια μέσης και η περιφέρεια ισχίων.

Στην *τρίτο* τμήμα υπολογίζονται οι ακόλουθες τιμές: Ιδανικό βάρος, Ισοζύγιο, Δείκτης μάζας σώματος και παρέχεται η αντίστοιχη περιγραφή, Ολικό λίπος και Άλιπη μάζα με τα αντίστοιχα ποσοστά, ο λόγος Περιμέτρου Μέσης προς Περίμετρο Ισχίων, ο Βασικός Μεταβολισμός και το άθροισμά του με τη Φυσική Δραστηριότητα.

Οι τιμές αυτές αποτελούν σημαντικές πληροφορίες, που θα βοηθήσουν στη λήψη σημαντικών αποφάσεων.

4.4 Οθόνη Στόχοι

Η οθόνη στόχοι αποτελεί το σημείο που εισάγονται στοιχεία, σχετικά με αποφάσεις που έχουν ληφθεί και αφορούν τη διαιτητική παρέμβαση.

Βάρος [Kgr]	-
Προτεινόμενο Βάρος	-
Επιθυμητό Βάρος	<input type="text"/>
Μεταβολή βάρους Σώματος [g/m]	<input type="text" value="-4000"/>
Μεταβολή πρόσληψης [cal/d]	<input type="text" value="-1000"/>
Διάρκεια [d]	-
BM + ΦΔ	-
Πρόταση [Kcal/d]	-
Ζητούμενο [Kcal/d]	<input type="text"/>

Εικόνα 3.3: Οθόνη Στόχοι

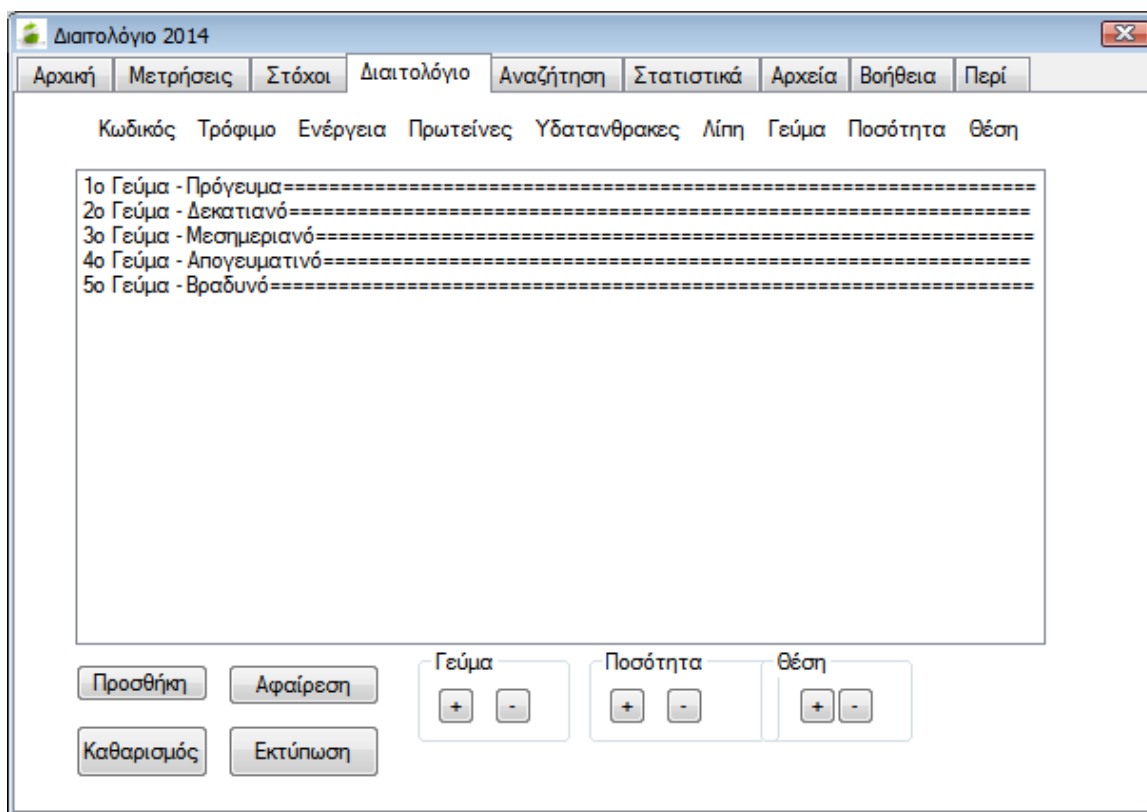
Το *Βάρος* αποτελεί υπόμνηση του βάρους σώματος που έχει εισαχθεί, όπως και το πεδίο *BM+ΦΔ* όπου εικονίζεται το άθροισμα του βασικού μεταβολισμού και της φυσικής δραστηριότητας. Το *Προτεινόμενο Βάρος* είναι το μέγιστο βάρος που μπορεί να έχει το άτομο εντός «φυσιολογικών» ορίων (να μην είναι δηλαδή «υπέρβαρος»).

Το *Επιθυμητό Βάρος* είναι το βάρος - στόχος που επιλέγει ο χρήστης (μπορεί να αποδεχτεί ή όχι το προτεινόμενο). Η *Μεταβολή Βάρους Σώματος* αποτελεί το ρυθμό με τον οποίο επιλέγει ο χρήστης να μεταβληθεί το βάρος. Η προεπιλεγμένη τιμή είναι απώλεια 4000gr (4Kg) το μήνα.

Αυτόματα υπολογίζεται η *Μεταβολή Διαιτητικής Πρόσληψης* ανά ημέρα και η τελική *Πρόταση* του προγράμματος για τον ημερήσιο ενεργειακό στόχο. Αυτόν μπορεί να αποδεχτεί ή να απορρίψει ο χρήστης, εισάγοντας την επιθυμητή τιμή στο πεδίο *Ζητούμενο*. Το πρόγραμμα υπολογίζει και τις ημέρες που απαιτούνται με τα δεδομένα στοιχεία έτσι ώστε να επιτευχθεί ο στόχος.

4.5 Οθόνη Διαιτολόγιο

Στην οθόνη Διαιτολόγιο δημιουργείται το διαιτολόγιο, με την επιλογή των τροφών που το αποτελούν.



Εικόνα 3.4: Οθόνη Διαιτολόγιο

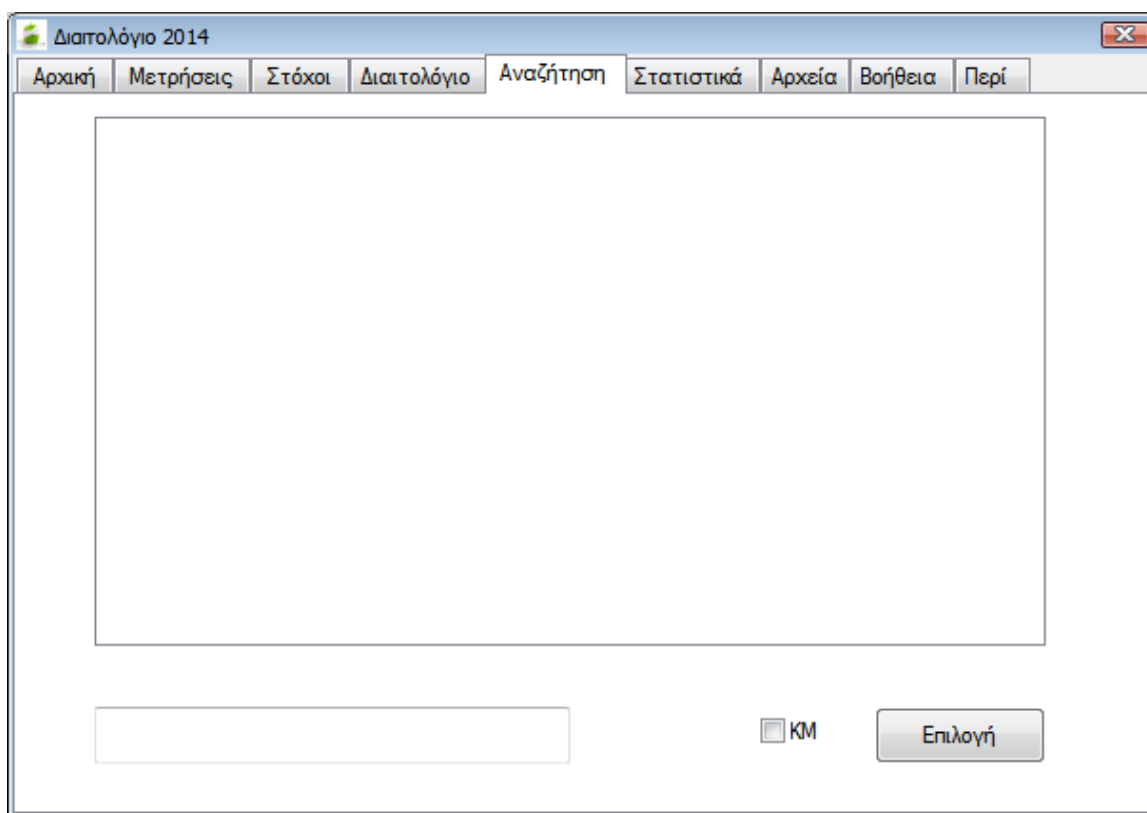
Στη οθόνη αυτή κυριαρχεί ένα πεδίο, όπου τοποθετούνται οι εισηγμένες τροφές στο κατάλληλο γεύμα. Οι απαιτούμενες επεξηγήσεις των τιμών υπάρχουν άνωθεν.

Σε αυτήν την οθόνη υπάρχουν τα εξής κουμπιά με τις ακόλουθες λειτουργίες.

- **ΠΡΟΘΗΚΗ:** Προσθήκη τροφίμου στο διαιτολόγιο.
- **ΑΦΑΙΡΕΣΗ:** Αφαίρεση τροφίμου από το διαιτολόγιο.
- **ΔΙΑΓΡΑΦΗ:** Διαγραφή όλων των τροφίμων από το διαιτολόγιο.
- **ΕΚΤΥΠΩΣΗ:** Εκτύπωση του διαιτολογίου.
- **ΓΕΥΜΑ:** Μεταβολή του τροφίμου στα γεύματα
- **ΠΟΣΟΤΗΤΑ:** Μεταβολή της ποσότητας του τροφίμου.
- **ΘΕΣΗ:** Μεταβολή της θέσης του τροφίμου εντός του γεύματος

4.6 Οθόνη Αναζήτηση

Στη οθόνη «Αναζήτηση» επιτυγχάνεται η αναζήτηση και η ανεύρεση του επιθυμητού τροφίμου εντός της βάσης δεδομένων τροφίμων που διαθέτουμε.



Εικόνα 3.5: Οθόνη Αναζήτηση

Περιλαμβάνει ένα πεδίο, όπου εισάγεται η λέξη κλειδί του τροφίμου που αναζητούμε.

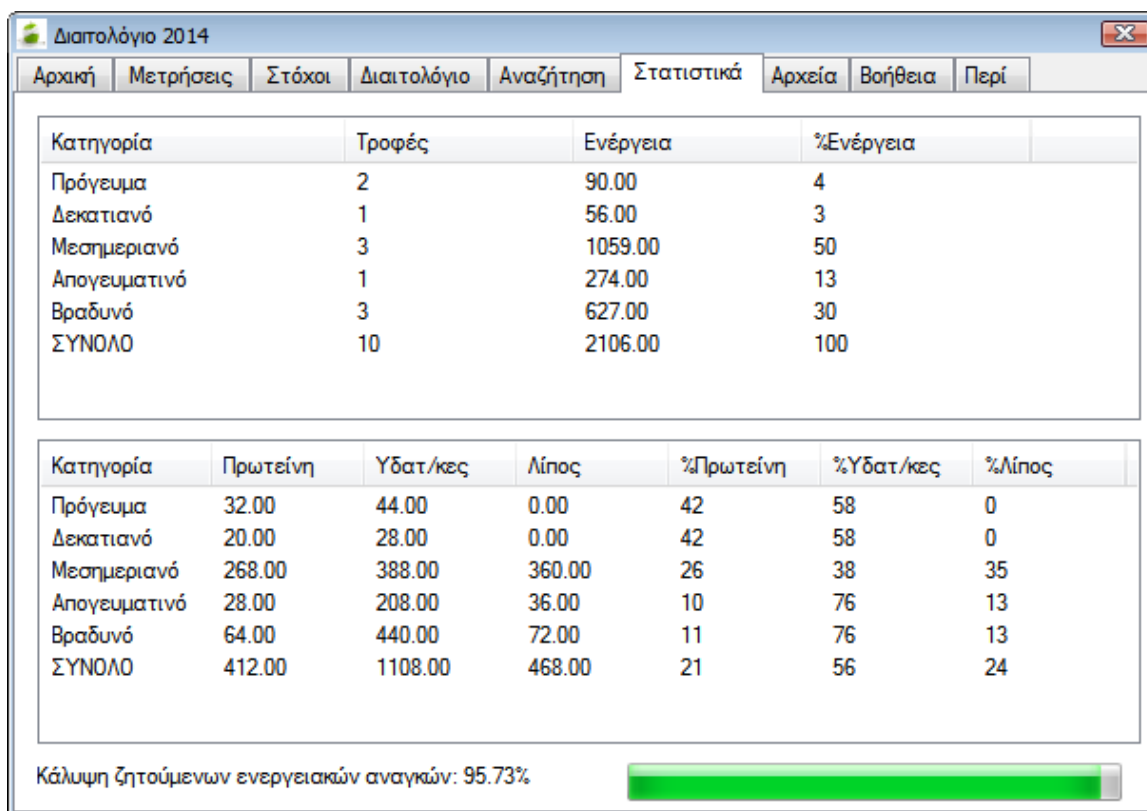
Υπάρχει ένα πεδίο όπου εμφανίζονται οι τροφές οι σχετικές με τη λέξη κλειδί.

Υπάρχει το κουμπί «Επιλογή», που χρησιμοποιείται όταν έχει εντοπιστεί το τρόφιμο που επιθυμούμε να εισάγουμε στο διαιτολόγιο.

Τέλος υφίσταται η επιλογή «KM», που δηλώνει ότι η αναζήτηση θα λαμβάνει υπόψη την κατάσταση των γραμμάτων (Κεφαλαία –Μικρά).

4.7 Οθόνη Στατιστικά

Στην οθόνη «Στατιστικά» εμφανίζονται οι στατιστικές παράμετροι του διαιτολογίου που έχει διαμορφωθεί, ώστε να αξιολογηθεί και να εντοπιστούν αλλαγές, που είναι απαραίτητες.



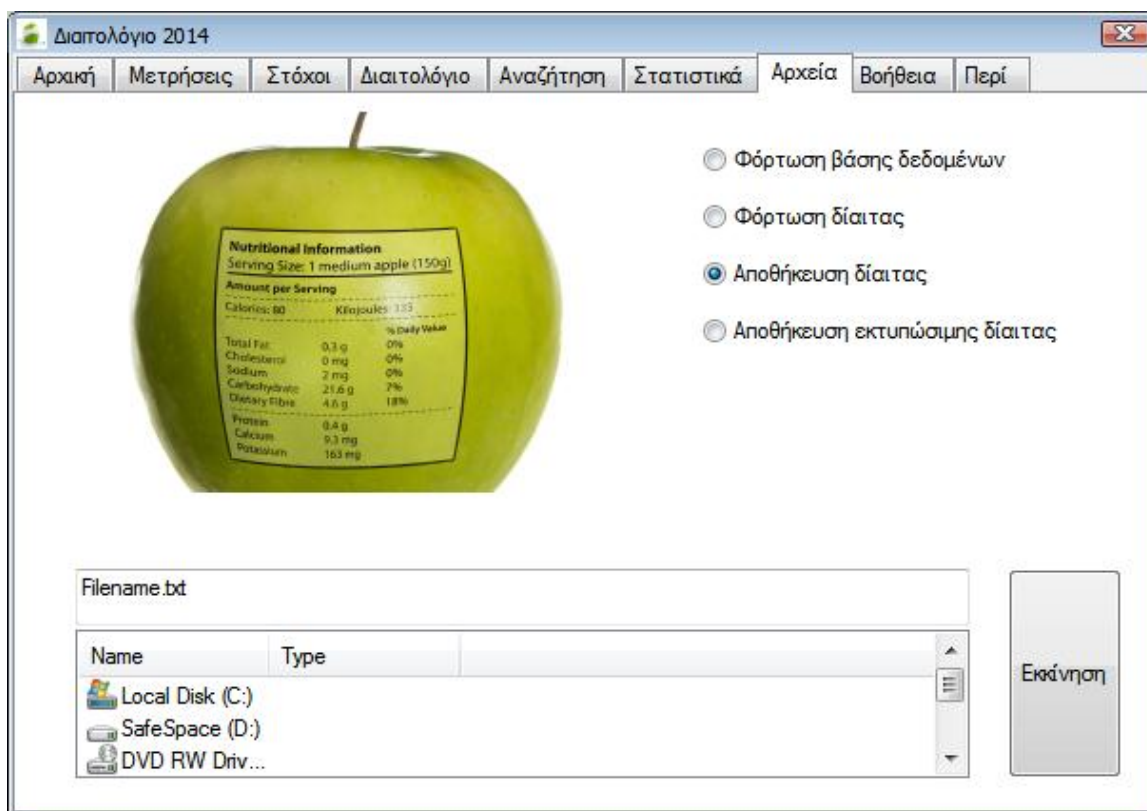
Εικόνα 3.6: Οθόνη Στατιστικά

Διαθέτει δύο πεδία στα οποία εικονίζονται ποικίλες στατιστικές πληροφορίες, όπως αριθμός τροφών ανά γεύμα, ενέργεια και ποσοστό ενέργειας ανά γεύμα, μακροθρεπτικά συστατικά και ποσοστά αυτών ανά γεύμα κ.τ.λ.

Επίσης αναφέρεται το ποσοστό κάλυψης των ζητούμενων ενεργειακών αναγκών. Το ποσοστό συνοδεύεται από την κατάλληλη απεικόνιση.

4.8 Οθόνη Αρχεία

Σε αυτήν την οθόνη εκτελούνται οι λειτουργίες αποθήκευσης - ανάκτησης του προγράμματος, που είναι απαραίτητες για την απρόσκοπτη λειτουργία του.



Εικόνα 3.7: Οθόνη Αρχεία

Η οθόνη διαθέτει ευχάριστο σχετικό γραφικό και επιλογή μεταξύ τεσσάρων λειτουργιών.

Υπάρχει πεδίο επιλογής ονόματος αρχείου για τις λειτουργίες αποθήκευσης και πεδίο επιλογής θέσης για την αποθήκευση ή αρχείου για την ανάκτηση.

Τέλος υπάρχει κουμπί που επιλέγεται για να ξεκινήσει η διαδικασία.

4.9 Οθόνη Βοήθεια

Σκοπός της οθόνης είναι να παρέχει βασική βοήθεια στον χρήστη της εφαρμογής, που αντιμετωπίζει δυσχέρειες στη χρήση του προγράμματος.

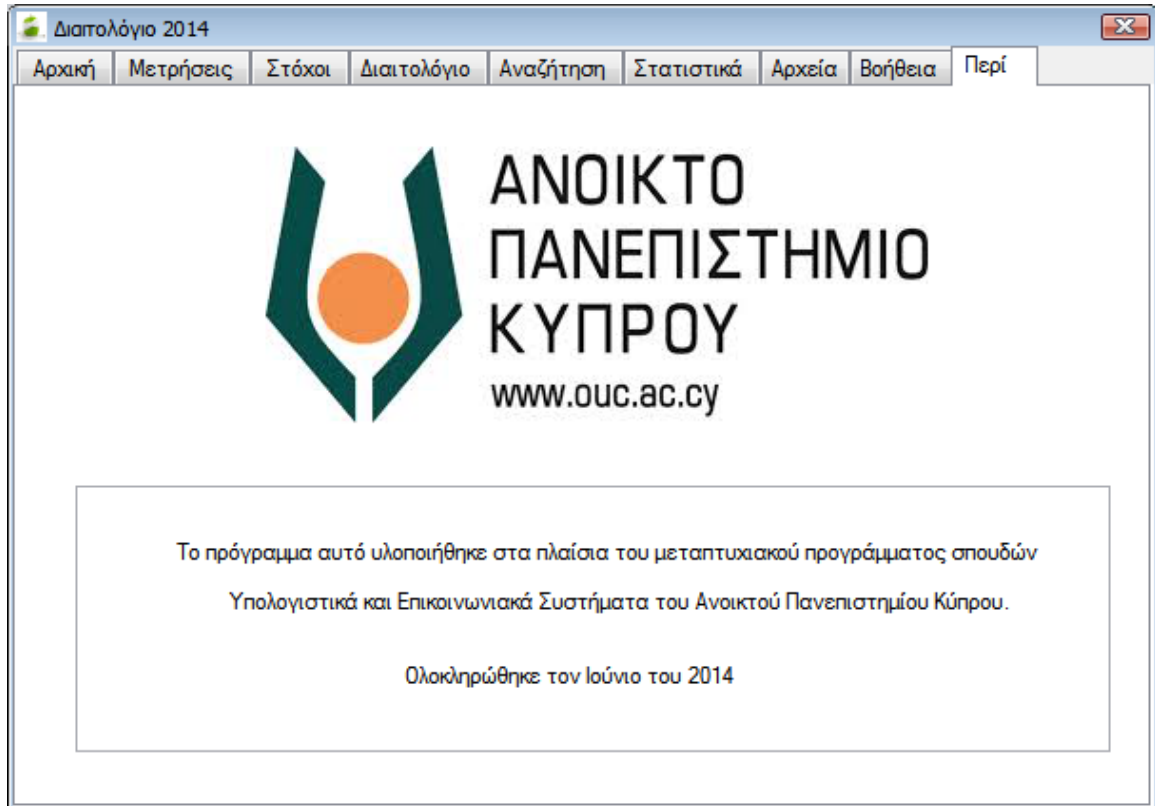


Εικόνα 3.8: Οθόνη Βοήθεια

Η οθόνη διαθέτει ένα γραφικό στοιχείο και ένα πεδίο στο οποίο αναφέρονται χρήσιμες πληροφορίες για τον χειρισμό του προγράμματος.

4.10 Οθόνη Περί

Σκοπός της οθόνης είναι να παρέχει την «ταυτότητα» της εφαρμογής, ώστε να ενημερωθεί κατάλληλα ο χρήστης.



Εικόνα 3.9: Οθόνη Περί

Η οθόνη, όπως και η οθόνη «Βοήθεια», διαθέτει ένα γραφικό στοιχείο και ένα πεδίο στο οποίο αναφέρονται πληροφορίες, όπως ο λόγος υλοποίησης, ο φορέας για τον οποίο δημιουργήθηκε, καθώς και ο χρόνος ολοκλήρωσης του προγράμματος.

Κεφάλαιο 5

Περιπτώσεις χρήσης

5.1 Εισαγωγικά

Μια περίπτωση χρήσης (Use Case) είναι μια αλληλουχία ενεργειών που εκτελεί το λογισμικό αλληλεπιδρώντας με το χρήστη ή με εξωτερικά συστήματα, προκειμένου να υλοποιήσει μια λειτουργική απαίτηση [13].

Οι περιπτώσεις χρήσης κατέχουν κεντρική θέση στη μοντελοποίηση της συμπεριφοράς του συστήματος, του υποσυστήματος ή μιας κλάσης [14].

Με το όρο «Περιπτώσεις χρήσης» χαρακτηρίζονται όλα τα πιθανά σενάρια που μπορεί να εκτελέσουν οι χρήστες του συστήματος για να πετύχουν ένα συγκεκριμένο σκοπό.

Στο συγκεκριμένο πρόγραμμα, όπως έχει αναφερθεί, διατίθεται μόνο ένας χρήστης που είναι ο μη εξειδικευμένος στη διατροφή επαγγελματίας υγείας.

5.2 Καθορισμός ιδανικού βάρους ΠΧ01

Περιγραφή: Ο χρήστης επιθυμεί να δει το ιδανικό βάρος για άτομο συγκεκριμένου ύψους και βάρους. Συμπληρώνει τα αντίστοιχα πεδία και το σύστημα επιστρέφει το ιδανικό βάρος.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Δεν υπάρχουν

Βασική ροή:

1. Ο χρήστης επιλέγει «Μετρήσεις».
2. Το σύστημα εμφανίζει την καρτέλα «Μετρήσεις»
3. Εισάγει φύλο, βάρος, ύψος.
4. Το σύστημα υπολογίζει το ιδανικό βάρος.

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

5.3 Καθορισμός κατάστασης βάρους ατόμου ΠΧ02

Περιγραφή: Ο χρήστης επιθυμεί να εκτιμήσει την κατάσταση βάρους για άτομο συγκεκριμένου ύψους και βάρους. Συμπληρώνει τα αντίστοιχα πεδία και στη συνέχεια το σύστημα επιστρέφει το ζητούμενο.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Δεν υπάρχουν

Βασική ροή:

1. Ο χρήστης επιλέγει «Μετρήσεις».
2. Το σύστημα εμφανίζει την καρτέλα «Μετρήσεις»
3. Εισάγει βάρος ύψος.
4. Το σύστημα του παρέχει τον δείκτη μάζας σώματος (ΔΜΣ) καθώς και τον χαρακτηρισμό της κατάστασης που προκύπτει από αυτόν.

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

5.4 Καθορισμός ποσότητας λίπους σώματός ΠΧ03

Περιγραφή: Ο χρήστης επιθυμεί να εκτιμήσει την ποσότητα λίπους για άτομο συγκεκριμένου φύλου, ύψους και βάρους. Συμπληρώνει τα αντίστοιχα πεδία και το σύστημα επιστρέφει το ζητούμενο.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Δεν υπάρχουν

Βασική ροή:

1. Ο χρήστης επιλέγει «Μετρήσεις».
2. Το σύστημα εμφανίζει την καρτέλα «Μετρήσεις»
3. Εισάγει φύλο, βάρος ύψος.
4. Το σύστημα εμφανίζει το ολικό λίπος σε κιλά και ποσοστό του σώματος.

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

5.5 Καθορισμός ύπαρξης κοιλιακής παχυσαρκίας ΠΧ04

Περιγραφή: Ο χρήστης επιθυμεί να εκτιμήσει την ύπαρξη κοιλιακής παχυσαρκίας για άτομο συγκεκριμένης περιφέρειας μέσης και ισχίων. Συμπληρώνει τα αντίστοιχα πεδία και το σύστημα επιστρέφει το ζητούμενο.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Δεν υπάρχουν

Βασική ροή:

1. Ο χρήστης επιλέγει «Μετρήσεις».
2. Το σύστημα εμφανίζει την καρτέλα «Μετρήσεις»
3. Εισάγει φύλο καθώς και περιφέρεια μέσης και περιφέρεια ισχίων.
4. Το σύστημα εμφανίζει αν υπάρχει κοιλιακή παχυσαρκία με βάση καθένα από τα δύο κριτήρια.

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

5.6 Καθορισμός ενέργειας (θερμίδες) ενός τροφίμου ΠΧ05

Περιγραφή: Ο χρήστης επιθυμεί να δει την ενέργεια που παρέχει συγκεκριμένο τρόφιμο. Συμπληρώνει τα αντίστοιχα πεδία και το σύστημα επιστρέφει το ζητούμενο.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Δεν υπάρχουν

Βασική ροή:

1. Ο χρήστης επιλέγει «Αναζήτηση».
2. Ο χρήστης εισάγει τη λέξη κλειδί που θα χρησιμοποιηθεί στην αναζήτηση του τροφίμου - στόχου.
3. Το σύστημα εμφανίζει τα σχετικά με τη λέξη κλειδί τρόφιμα.
4. Ο χρήστης επιλέγει το τρόφιμο που τον ενδιαφέρει.
5. Το σύστημα εμφανίζει τις ζητούμενες θερμίδες.

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

5.7 Καθορισμός μακροθρεπτικών χαρακτηριστικών ενός τροφίμου ΠΧ06

Περιγραφή: Ο χρήστης επιθυμεί να δει την ενέργεια που παρέχει συγκεκριμένο τρόφιμο. Συμπληρώνει τα αντίστοιχα πεδία και το σύστημα επιστρέφει το ζητούμενο.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Δεν υπάρχουν

Βασική ροή:

1. Ο χρήστης επιλέγει «Αναζήτηση».
2. Ο χρήστης εισάγει τη λέξη κλειδί που θα χρησιμοποιηθεί στην αναζήτηση του τροφίμου - στόχου.
3. Το σύστημα εμφανίζει τα σχετικά με τη λέξη κλειδί τρόφιμα.
4. Ο χρήστης επιλέγει το τρόφιμο που τον ενδιαφέρει.
5. Το σύστημα εμφανίζει τα μακροθρεπτικά συστατικά του ζητούμενου τροφίμου.

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

5.8 Καθορισμός θερμίδων - στόχο μιας δίαιτας ΠΧ07

Περιγραφή: Ο χρήστης επιθυμεί να υπολογίσει, βάσει επιλογών, την ενέργεια που θα πρέπει να παρέχει μια δίαιτα. Συμπληρώνει τα αντίστοιχα πεδία και το σύστημα επιστρέφει το ζητούμενο.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Δεν υπάρχουν

Βασική ροή:

1. Ο χρήστης επιλέγει «Μετρήσεις».
2. Το σύστημα εμφανίζει την οθόνη «Μετρήσεις».
3. Ο χρήστης εισάγει Φύλο, ηλικία, βάρος, ύψος, δραστηριότητα.
4. Το σύστημα εμφανίζει τις συνολικές θερμίδες (βασικός μεταβολισμός + φυσική δραστηριότητα)
5. Ο χρήστης επιλέγει «Στόχοι».
6. Το σύστημα εμφανίζει την οθόνη «Στόχοι».
7. Ο χρήστης επιλέγει το επιθυμητό βάρος και τη μεταβολή βάρους σώματος.
8. Το σύστημα εμφανίζει την προτεινόμενη ενέργεια του διαιτολογίου.
9. Ο χρήστης εισάγει την επιθυμητή συνολική ενέργεια του διαιτολογίου.

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

5.9 Δημιουργία ενός διαιτολογίου ΠΧ08

Περιγραφή: Ο χρήστης επιθυμεί να δημιουργήσει ένα διαιτολόγιο συγκεκριμένων θερμίδων. Επιλέγει τις τροφές που επιθυμεί και τις εισάγει στο διαιτολόγιο.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Υπολογισμός των θερμίδων που θα πρέπει να έχει η δίαιτα (περίπτωση χρήσης: Καθορισμός θερμίδων - στόχο μιας δίαιτας).

Βασική ροή:

1. Ο χρήστης επιλέγει «Διαιτολόγιο».
2. Το σύστημα εμφανίζει την οθόνη «Διαιτολόγιο».
3. Ο χρήστης επιλέγει το γεύμα στο οποίο θα προστεθεί η τροφή και επιλέγει «Προσθήκη».
4. Το σύστημα εμφανίζει την οθόνη «Αναζήτηση».
5. Ο χρήστης εισάγει τη λέξη κλειδί που θα χρησιμοποιηθεί στην αναζήτηση του τροφίμου στόχου.
6. Το σύστημα εμφανίζει τα σχετικά με τη λέξη κλειδί τρόφιμα.
7. Ο χρήστης επιλέγει το τρόφιμο που τον ενδιαφέρει και πατά «Επιλογή».
8. Το σύστημα εμφανίζει την οθόνη «Διαιτολόγιο».
9. Επαναλαμβάνεται η διαδικασία και για τα άλλα τρόφιμα.

Μετασυνθήκες: Δεν υπάρχουν

Εναλλακτική Ροή 1

9α1 Ο χρήστης πατά «Καθαρισμός»

9α2 Το σύστημα διαγράφει όλα τα τρόφιμα από το διαιτολόγιο

9α3 Το σύστημα συνεχίζει από το βήμα 2

Μετασυνθήκες: Δεν υπάρχουν

Εναλλακτική Ροή 2

9α1 Ο χρήστης επιλέγει τρόφιμο.

9α2 Ο χρήστης πατά «Διαγραφή»

9α3 Το σύστημα διαγράφει το επιλεγμένο τρόφιμο.

9α4 Το σύστημα συνεχίζει από το βήμα 2

Μετασυνθήκες: Δεν υπάρχουν

Εναλλακτική Ροή 3

9α1 Ο χρήστης επιλέγει «Εκτύπωση».

9α2 Το σύστημα εκτυπώνει το τρέχον διαιτολόγιο.

9α3 Το σύστημα συνεχίζει από το βήμα 2

Μετασυνθήκες: Δεν υπάρχουν

Εναλλακτική Ροή 4

9α1 Ο χρήστης επιλέγει τρόφιμο.

9α2 Πατά γεύμα + ή -.

9α3 Το σύστημα τροποποιεί το γεύμα στο οποίο βρίσκεται το τρόφιμο.

9α4 Το σύστημα συνεχίζει από το βήμα 2

Μετασυνθήκες: Δεν υπάρχουν

Εναλλακτική Ροή 5

9α1 Ο χρήστης επιλέγει τρόφιμο.

9α2 Πατά ποσότητα + ή -.

9α3 Το σύστημα τροποποιεί την ποσότητα του τροφίμου.

9α4 Το σύστημα συνεχίζει από το βήμα 2

Μετασυνθήκες: Δεν υπάρχουν

Εναλλακτική Ροή 6

9α1 Ο χρήστης επιλέγει τρόφιμο.

9α2 Πατά θέση + ή -.

9α3 Το σύστημα τροποποιεί τη θέση του τροφίμου στο γεύμα.

9α4 Το σύστημα συνεχίζει από το βήμα 2

Μετασυνθήκες: Δεν υπάρχουν

5.10 Επίδειξη στατιστικών της δίαιτας ΠΧ09

Περιγραφή: Ο χρήστης επιθυμεί να επιδειχτούν τα στατιστικά της υφιστάμενης δίαιτας, που θα τον βοηθήσουν, ώστε να την αξιολογήσει και να τη διορθώσει.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Πρέπει να έχει δημιουργηθεί πρώτα η δίαιτα (περίπτωση χρήσης: *Δημιουργία ενός διαιτολογίου*) ή φόρτωση δίαιτας (περίπτωση χρήσης: *Φόρτωση ενός διαιτολογίου*.)

Βασική ροή:

1. Ο χρήστης επιλέγει «*Στατιστικά*».
2. Το σύστημα υπολογίζει και εμφανίζει την οθόνη «*Στατιστικά*».

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

5.11 Φόρτωση βάσης δεδομένων ΠΧ10

Περιγραφή: Ο χρήστης επιθυμεί να φορτώσει μια βάση δεδομένων, ώστε να τη χρησιμοποιήσει για τη δημιουργία διαιτολογίου.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Δεν υπάρχουν.

Βασική ροή:

1. Ο χρήστης επιλέγει «*Αρχεία*».
2. Το σύστημα εμφανίζει την οθόνη «*Αρχεία*».
3. Ο χρήστης επιλέγει «*Φόρτωση βάσης δεδομένων*» και το αρχείο της βάσης και πατά «*Εκκίνηση*».
4. Το σύστημα φορτώνει τη βάση και εμφανίζει μήνυμα επιτυχούς φόρτωσης.

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

5.12 Φόρτωση Διαιτολογίου ΠΧ11

Περιγραφή: Ο χρήστης επιθυμεί να φορτώσει ένα διαιτολόγιο, που είχε δημιουργήσει στο παρελθόν, για να το τροποποιήσει ή να το χρησιμοποιήσει αυτούσιο.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Δεν υπάρχουν.

Βασική ροή:

1. Ο χρήστης επιλέγει «Αρχεία».
2. Το σύστημα εμφανίζει την οθόνη «Αρχεία».
3. Ο χρήστης επιλέγει «Φόρτωση δίαιτας» και το αρχείο της δίαιτας και πατά «Εκκίνηση».
4. Το σύστημα φορτώνει τη δίαιτα και εμφανίζει μήνυμα επιτυχούς φόρτωσης.

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

5.13 Αποθήκευση διαιτολογίου ΠΧ12

Περιγραφή: Ο χρήστης επιθυμεί να αποθηκεύσει ένα διαιτολόγιο, ώστε να μπορεί να το χρησιμοποιήσει στο μέλλον.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Δεν υπάρχουν.

Βασική ροή:

1. Ο χρήστης επιλέγει «Αρχεία».
2. Το σύστημα εμφανίζει την οθόνη «Αρχεία».
3. Ο χρήστης επιλέγει «Αποθήκευση δίαιτας», το όνομα του αρχείου και τη θέση που θα αποθηκευτεί το αρχείο της δίαιτας και πατά «Εκκίνηση».
4. Το σύστημα αποθηκεύει τη δίαιτα και εμφανίζει μήνυμα επιτυχούς αποθήκευσης.

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

5.14 Αποθήκευση διαιτολογίου σε εκτυπώσιμη μορφή ΠΧ13

Περιγραφή: Ο χρήστης επιθυμεί να αποθηκεύσει ένα διαιτολόγιο, σε τέτοια μορφή (.txt), ώστε να μπορεί να το χρησιμοποιήσει άμεσα για την εκτύπωση ως αρχείο κειμένου..

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Δεν υπάρχουν.

Βασική ροή:

1. Ο χρήστης επιλέγει «Αρχεία».
2. Το σύστημα εμφανίζει την οθόνη «Αρχεία».
3. Ο χρήστης επιλέγει «Αποθήκευση εκτυπώσιμης δίαιτας», το όνομα του αρχείου και τη θέση που θα αποθηκευτεί το αρχείο και πατά «Εκκίνηση».
4. Το σύστημα αποθηκεύει το αρχείο και εμφανίζει μήνυμα επιτυχούς αποθήκευσης.

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

5.15 Λήψη Βοήθειας ΠΧ14

Περιγραφή: Ο χρήστης επιθυμεί να λάβει βοήθεια, ώστε να μπορεί να χρησιμοποιήσει το πρόγραμμα καλύτερα.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Δεν υπάρχουν.

Βασική ροή:

1. Ο χρήστης επιλέγει «Βοήθεια».
2. Το σύστημα εμφανίζει την οθόνη «Βοήθεια».

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

5.16 Επίδειξη «ταυτότητας» του προγράμματος ΠΧ15

Περιγραφή: Ο χρήστης επιθυμεί να λάβει πληροφορίες για την ταυτότητα του προγράμματος.

Χρήστης: Επαγγελματίας υγείας.

Προσυνθήκες: Δεν υπάρχουν.

Βασική ροή:

1. Ο χρήστης επιλέγει «Περί».
2. Το σύστημα εμφανίζει την οθόνη «Περί».

Εναλλακτικές Ροές: Δεν υπάρχουν

Μετασυνθήκες: Δεν υπάρχουν

Κεφάλαιο 6

Ανάλυση ευρωστίας

6.1 Εισαγωγικά

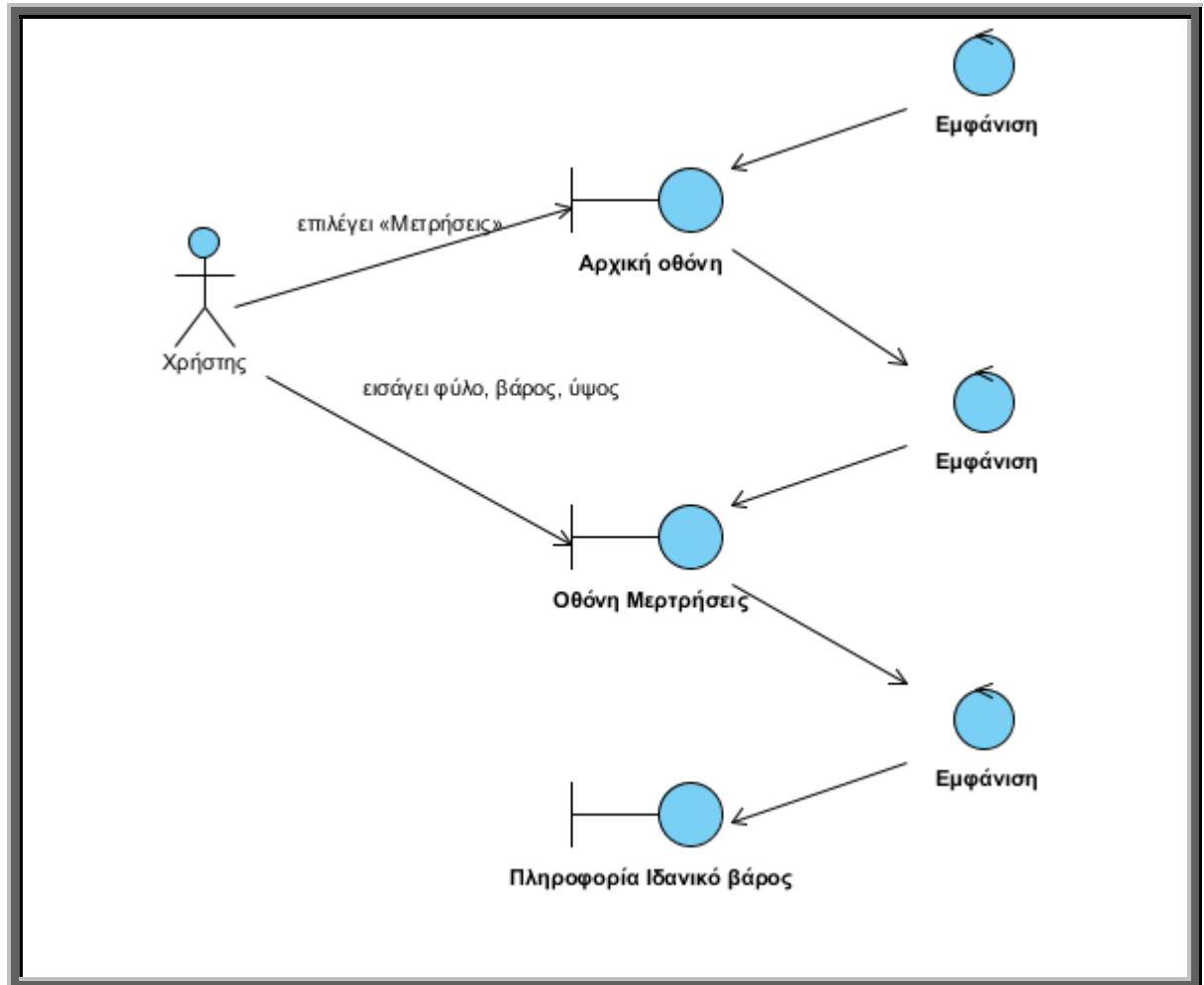
Η ανάλυση ευρωστίας (*robustness analysis*) αποτελεί μια τεχνική, ενταγμένη στη φάση της ανάλυσης απαιτήσεων, για τη μετάβαση από τις περιπτώσεις χρήσης σε ένα λεπτομερές σχέδιο.

Πρωταρχικός στόχος της ανάλυσης ευρωστίας αποτελεί η γεφύρωση του χάσματος μεταξύ της ανάλυσης (που απαντά στο ερώτημα "τι" θα κάνει το σύστημα) και της σχεδίασης (που απαντά στο ερώτημα "πώς" θα ικανοποιηθούν οι απαιτήσεις του πελάτη) [15].

Το Διάγραμμα Ευρωστίας είναι μια γραφική απεικόνιση της συμπεριφοράς που περιγράφεται σε μια περίπτωση χρήσης.

6.2 Καθορισμός ιδανικού βάρους ΔΕ01

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ01



Εικόνα 5.1: Καθορισμός ιδανικού βάρους ΔΕ01

Ο χρήστης επιλέγει «Μετρήσεις».

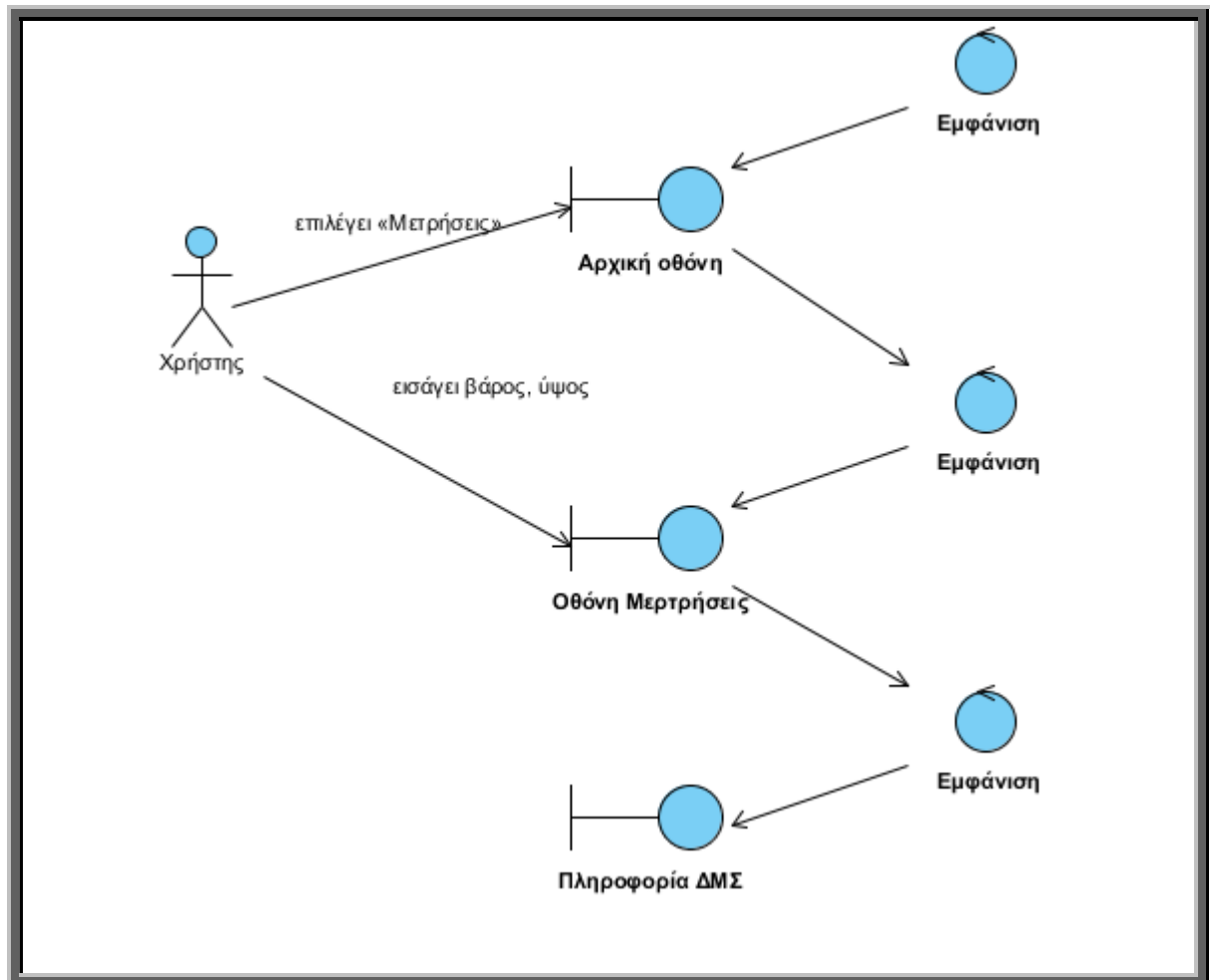
Το σύστημα εμφανίζει την καρτέλα «Μετρήσεις»

Εισάγει φύλο, βάρος, ύψος.

Το σύστημα του παρέχει το ιδανικό του βάρος.

6.3 Καθορισμός κατάστασης βάρους ατόμου ΔΕ02

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ02.



Εικόνα 5.2: Καθορισμός κατάστασης βάρους ατόμου ΔΕ02

Ο χρήστης επιλέγει «Μετρήσεις».

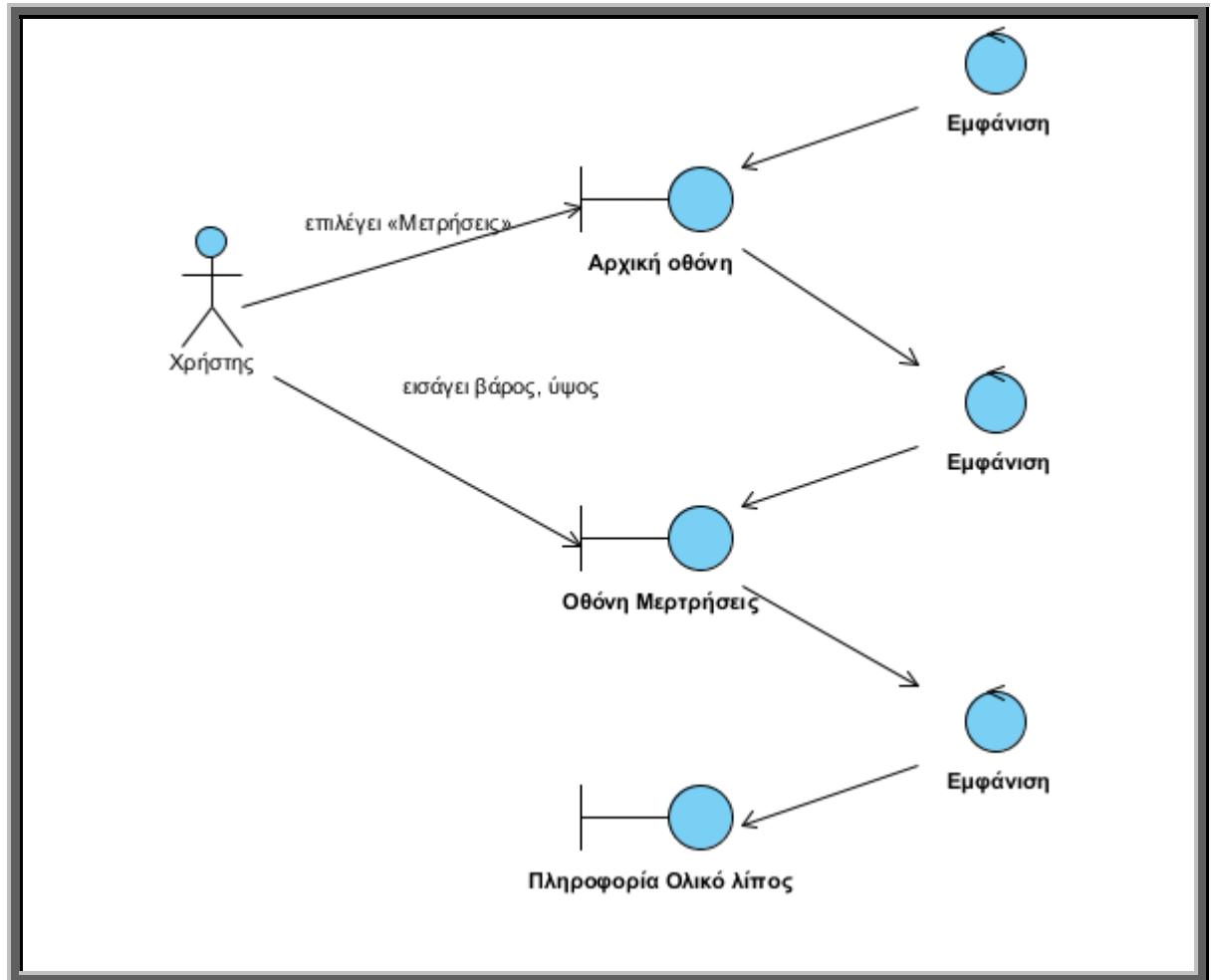
Το σύστημα εμφανίζει την καρτέλα «Μετρήσεις»

Εισάγει βάρος, ύψος.

Το σύστημα του παρέχει τον δείκτη μάζας σώματος (ΔΜΣ) καθώς και τον χαρακτηρισμό της κατάστασης που προκύπτει από αυτόν.

6.4 Καθορισμός ποσότητας λίπους σώματός ΔΕ03

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ03.



Εικόνα 5.3: Καθορισμός ποσότητας λίπους σώματός ΔΕ03

Ο χρήστης επιλέγει «Μετρήσεις».

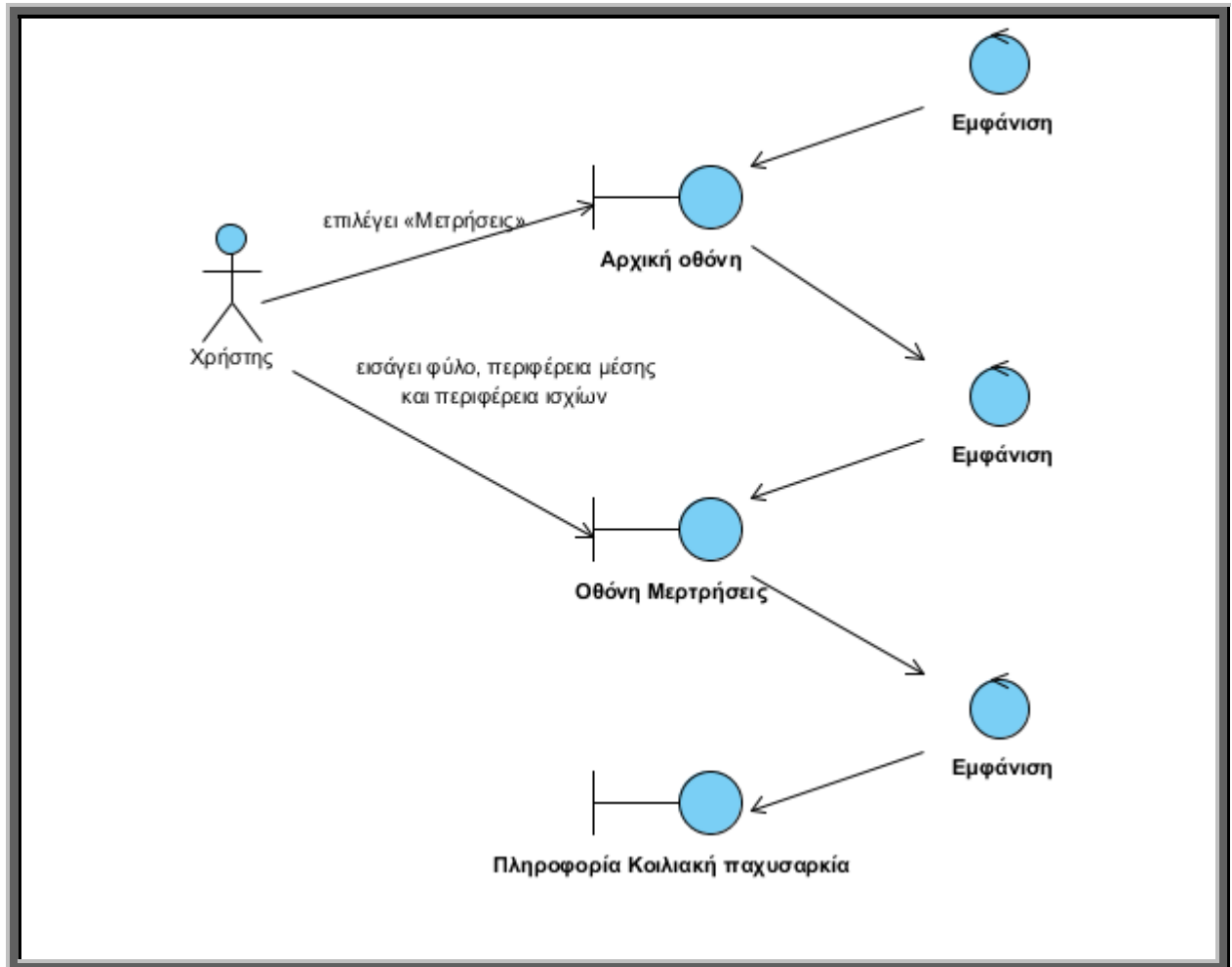
Το σύστημα εμφανίζει την καρτέλα «Μετρήσεις»

Εισάγει φύλο, βάρος, ύψος.

Το σύστημα εμφανίζει το ολικό λίπος σε κιλά και ποσοστό του σώματος.

6.5 Καθορισμός ύπαρξης κοιλιακής παχυσαρκίας ΔΕ04

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ04



Εικόνα 5.4: Καθορισμός ύπαρξης κοιλιακής παχυσαρκίας ΔΕ04

Ο χρήστης επιλέγει «Μετρήσεις».

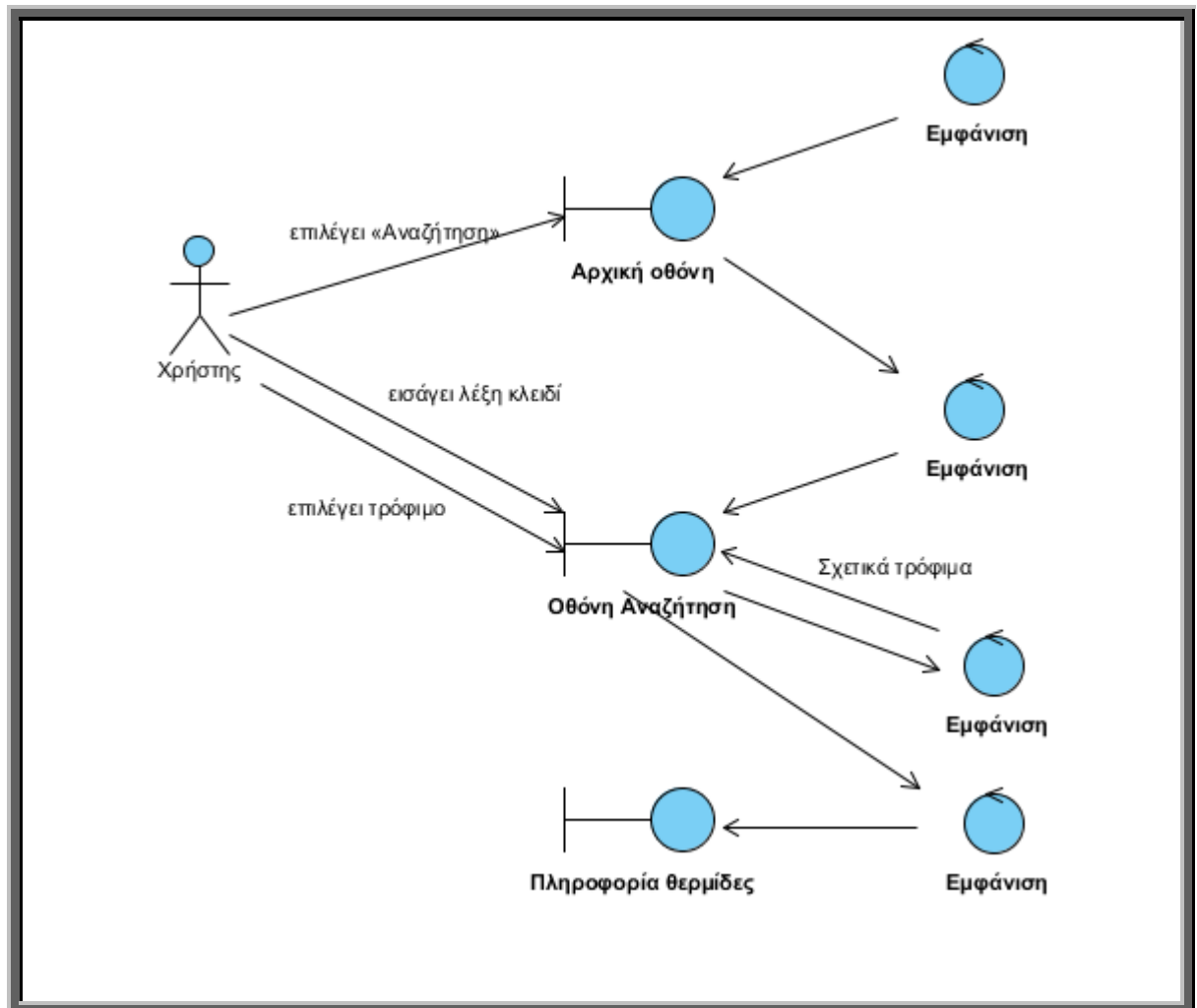
Το σύστημα εμφανίζει την καρτέλα «Μετρήσεις»

Εισάγει φύλο καθώς και περιφέρεια μέσης και περιφέρεια ισχίων.

Το σύστημα εμφανίζει αν υπάρχει κοιλιακή παχυσαρκία με βάση καθένα από τα δύο κριτήρια.

6.6 Καθορισμός ενέργειας (θερμίδες) ενός τροφίμου ΔΕ05

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ05.



Εικόνα 5.5: Καθορισμός ενέργειας (θερμίδες) ενός τροφίμου ΔΕ05

Ο χρήστης επιλέγει «Αναζήτηση».

Ο χρήστης εισάγει τη λέξη κλειδί που θα χρησιμοποιηθεί στην αναζήτηση του τροφίμου - στόχου.

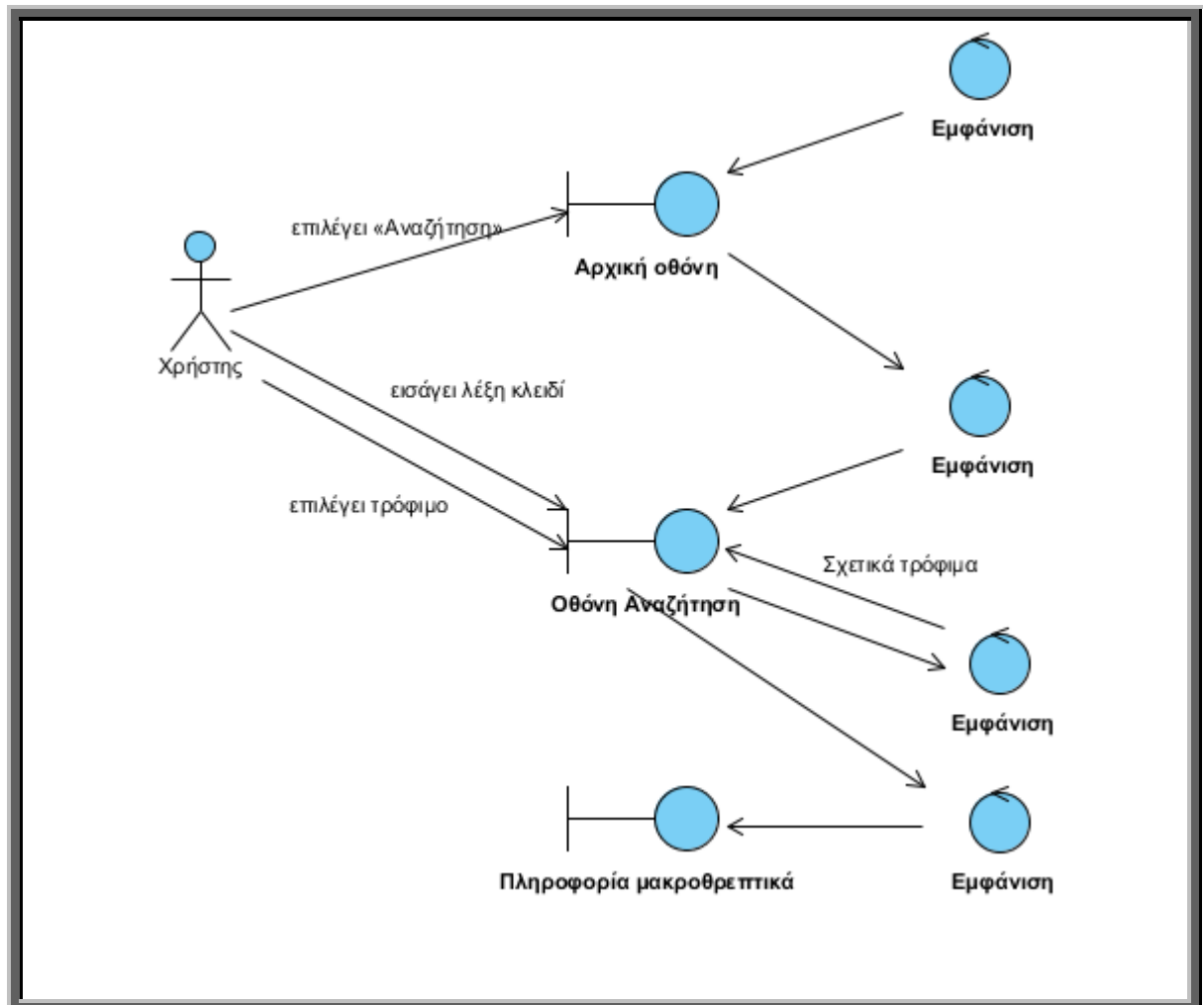
Το σύστημα εμφανίζει τα σχετικά με τη λέξη κλειδί τρόφιμα.

Ο χρήστης επιλέγει το τρόφιμο που τον ενδιαφέρει.

Το σύστημα εμφανίζει τις ζητούμενες θερμίδες.

6.7 Καθορισμός μακροθρεπτικών χαρακτηριστικών ενός τροφίμου ΔΕ06

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ06.



Εικόνα 5.6: Καθορισμός μακροθρεπτικών χαρακτηριστικών ενός τροφίμου ΔΕ06

Ο χρήστης επιλέγει «Αναζήτηση».

Ο χρήστης εισάγει τη λέξη κλειδί που θα χρησιμοποιηθεί στην αναζήτηση του τροφίμου στόχου.

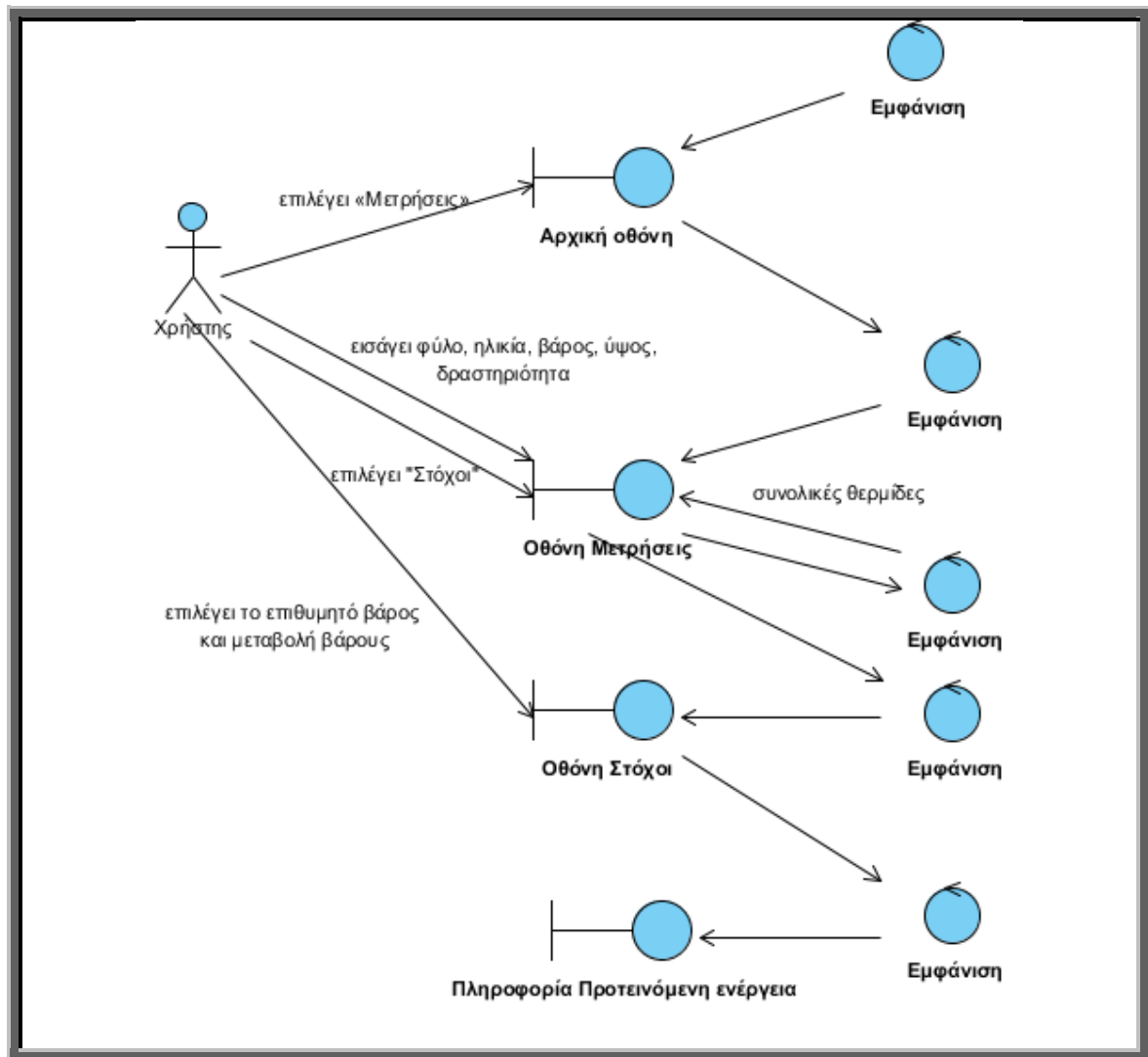
Το σύστημα εμφανίζει τα σχετικά με τη λέξη κλειδί τρόφιμα.

Ο χρήστης επιλέγει το τρόφιμο που τον ενδιαφέρει.

Το σύστημα εμφανίζει τα μακροθρεπτικά συστατικά του ζητούμενου τροφίμου.

6.8 Καθορισμός θερμίδων - στόχου μιας δίαιτας ΔΕ07

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ07



Εικόνα 5.7: Καθορισμός θερμίδων - στόχου μιας δίαιτας ΔΕ07

Ο χρήστης επιλέγει «Μετρήσεις».

Το σύστημα εμφανίζει την οθόνη «Μετρήσεις».

Ο χρήστης εισάγει Φύλο, ηλικία, βάρος, ύψος, δραστηριότητα.

Το σύστημα εμφανίζει τις συνολικές θερμίδες (βασικός μεταβολισμός + φυσική δραστηριότητα)

Ο χρήστης επιλέγει «Στόχοι».

Το σύστημα εμφανίζει την οθόνη «Στόχοι».

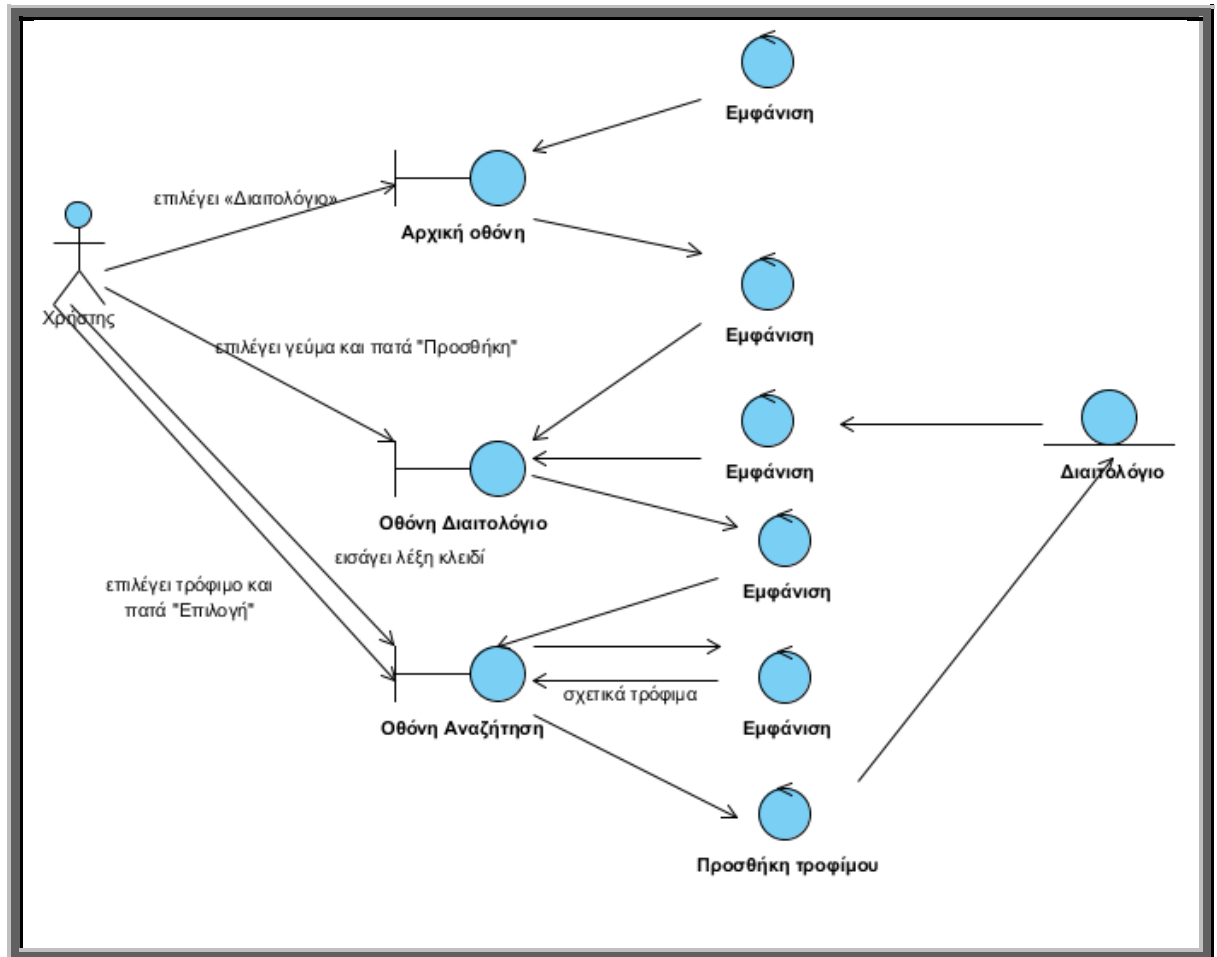
Ο χρήστης επιλέγει το επιθυμητό βάρος και τη μεταβολή βάρους σώματος.

Το σύστημα εμφανίζει την προτεινόμενη ενέργεια του διατολογίου.

Ο χρήστης εισάγει την επιθυμητή συνολική ενέργεια του διατολογίου.

6.9 Δημιουργία ενός διαιτολογίου ΔΕ08

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ08.



Εικόνα 5.8: Δημιουργία ενός διαιτολογίου ΔΕ08

Ο χρήστης επιλέγει «Διαιτολόγιο».

Το σύστημα εμφανίζει την οθόνη «Διαιτολόγιο».

Ο χρήστης επιλέγει το γεύμα στο οποίο θα προστεθεί η τροφή και πατά «Προσθήκη».

Το σύστημα εμφανίζει την οθόνη «Αναζήτηση».

Ο χρήστης εισάγει τη λέξη κλειδί που θα χρησιμοποιηθεί στην αναζήτηση του τροφίμου στόχου.

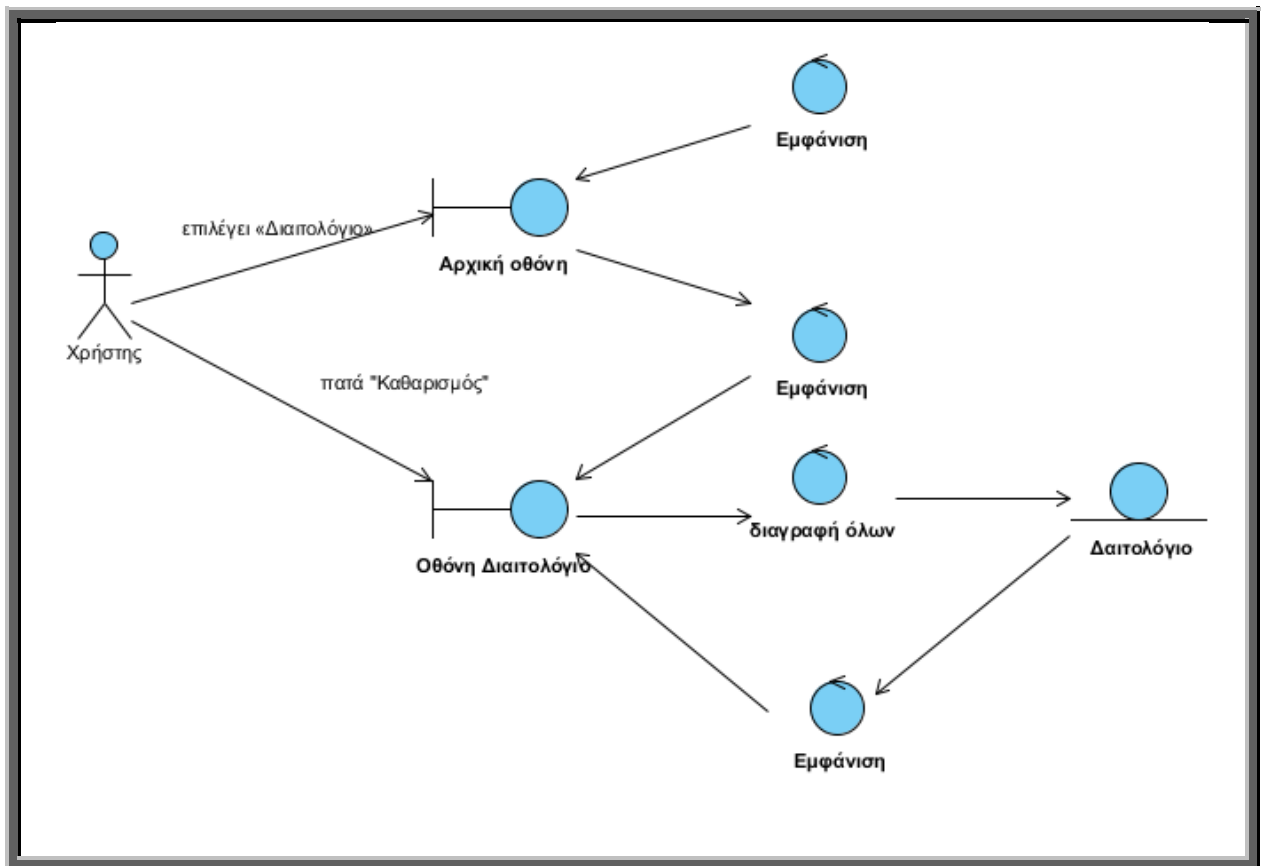
Το σύστημα εμφανίζει τα σχετικά με τη λέξη κλειδί τρόφιμα.

Ο χρήστης επιλέγει το τρόφιμο που τον ενδιαφέρει και πατά «Επιλογή».

Το σύστημα εμφανίζει την οθόνη «Διαιτολόγιο»

Επαναλαμβάνεται η διαδικασία και για τα άλλα τρόφιμα.

Εναλλακτική Ροή 1



Εικόνα 5.9: Δημιουργία ενός διαιτολογίου EP1

Ο χρήστης επιλέγει «Διαιτολόγιο».

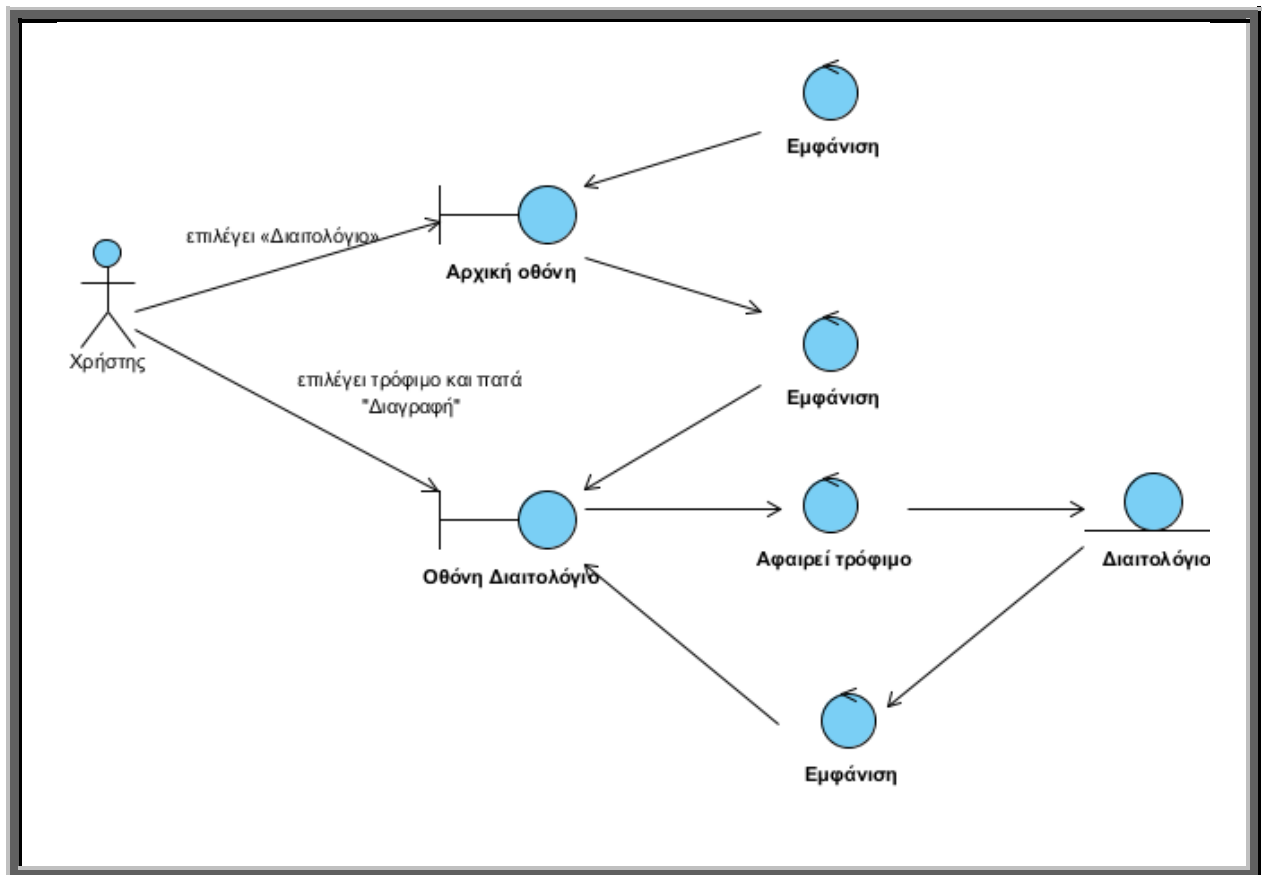
Το σύστημα εμφανίζει την οθόνη «Διαιτολόγιο».

Ο χρήστης πατά «Καθαρισμός».

Το σύστημα διαγράφει όλα τα τρόφιμα από το διαιτολόγιο.

Το σύστημα συνεχίζει από την Οθόνη Διαιτολόγιο.

Εναλλακτική Ροή 2



Εικόνα 5.10: Δημιουργία ενός διαιτολογίου EP2

Ο χρήστης επιλέγει «Διαιτολόγιο».

Το σύστημα εμφανίζει την οθόνη «Διαιτολόγιο».

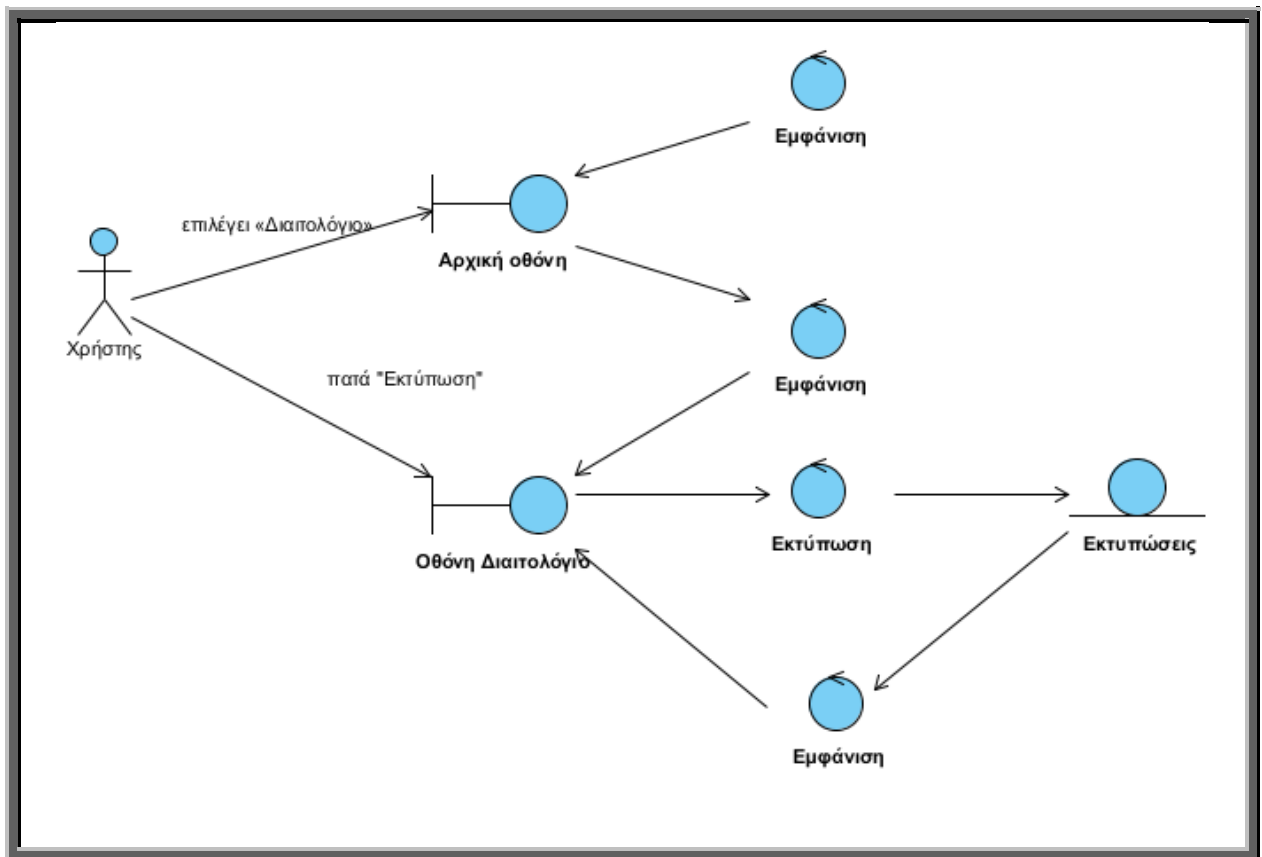
Ο χρήστης επιλέγει τρόφιμο.

Ο χρήστης πατά «Διαγραφή»

Το σύστημα διαγράφει το επιλεγμένο τρόφιμο.

Το σύστημα συνεχίζει από την Οθόνη Διαιτολόγιο.

Εναλλακτική Ροή 3



Εικόνα 5.11: Δημιουργία ενός διαιτολογίου EP3

Ο χρήστης επιλέγει «Διαιτολόγιο».

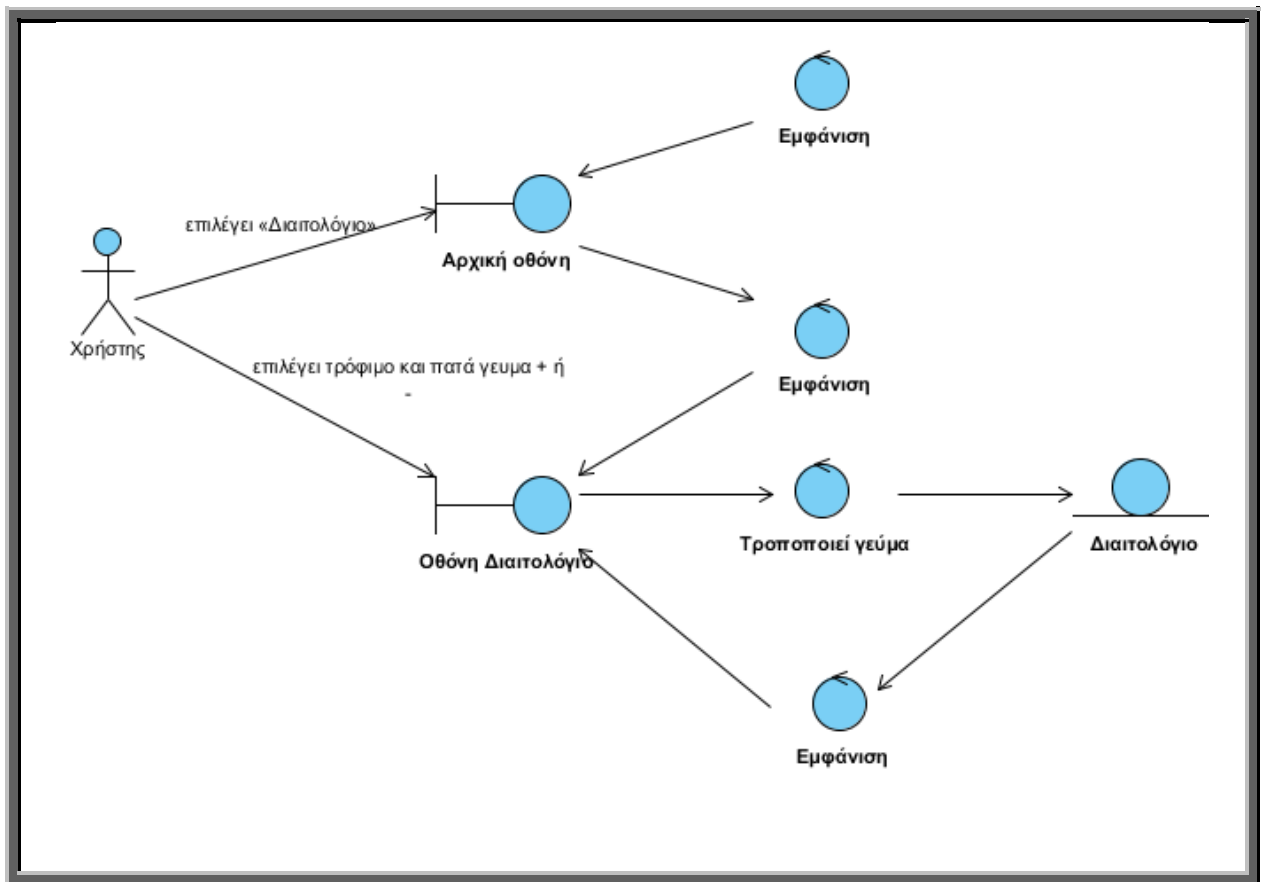
Το σύστημα εμφανίζει την οθόνη «Διαιτολόγιο».

Ο χρήστης επιλέγει «Εκτύπωση».

Το σύστημα εκτυπώνει το τρέχον διαιτολόγιο.

Το σύστημα συνεχίζει από την Οθόνη Διαιτολόγιο.

Εναλλακτική Ροή 4



Εικόνα 5.12: Δημιουργία ενός διαιτολογίου EP4

Ο χρήστης επιλέγει «Διαιτολόγιο».

Το σύστημα εμφανίζει την οθόνη «Διαιτολόγιο».

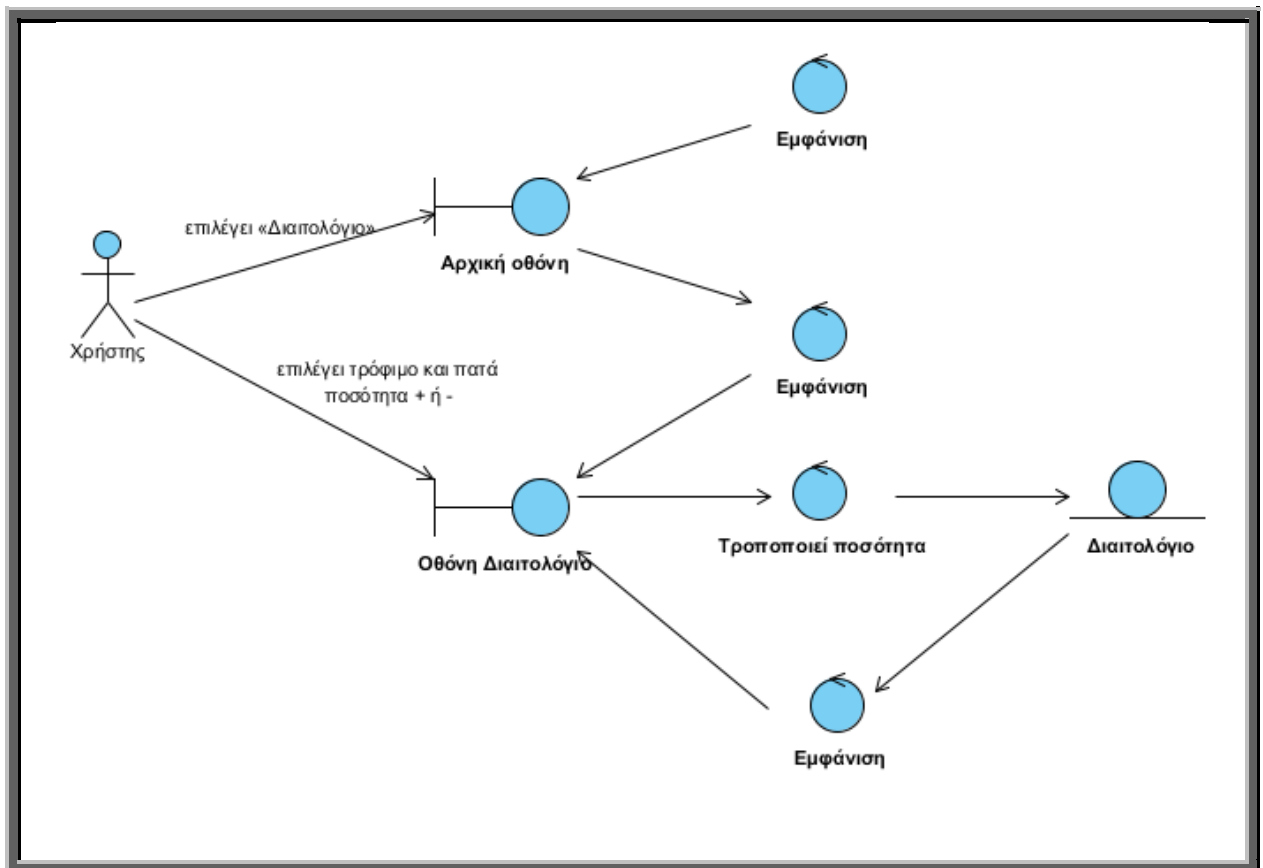
Ο χρήστης επιλέγει τρόφιμο.

Πατά «Γεύμα» + ή -.

Το σύστημα τροποποιεί το γεύμα στο οποίο βρίσκεται το τρόφιμο.

Το σύστημα συνεχίζει από την Οθόνη Διαιτολόγιο.

Εναλλακτική Ροή 5



Εικόνα 5.13: Δημιουργία ενός διαιτολογίου EP5

Ο χρήστης επιλέγει «Διαιτολόγιο».

Το σύστημα εμφανίζει την οθόνη «Διαιτολόγιο».

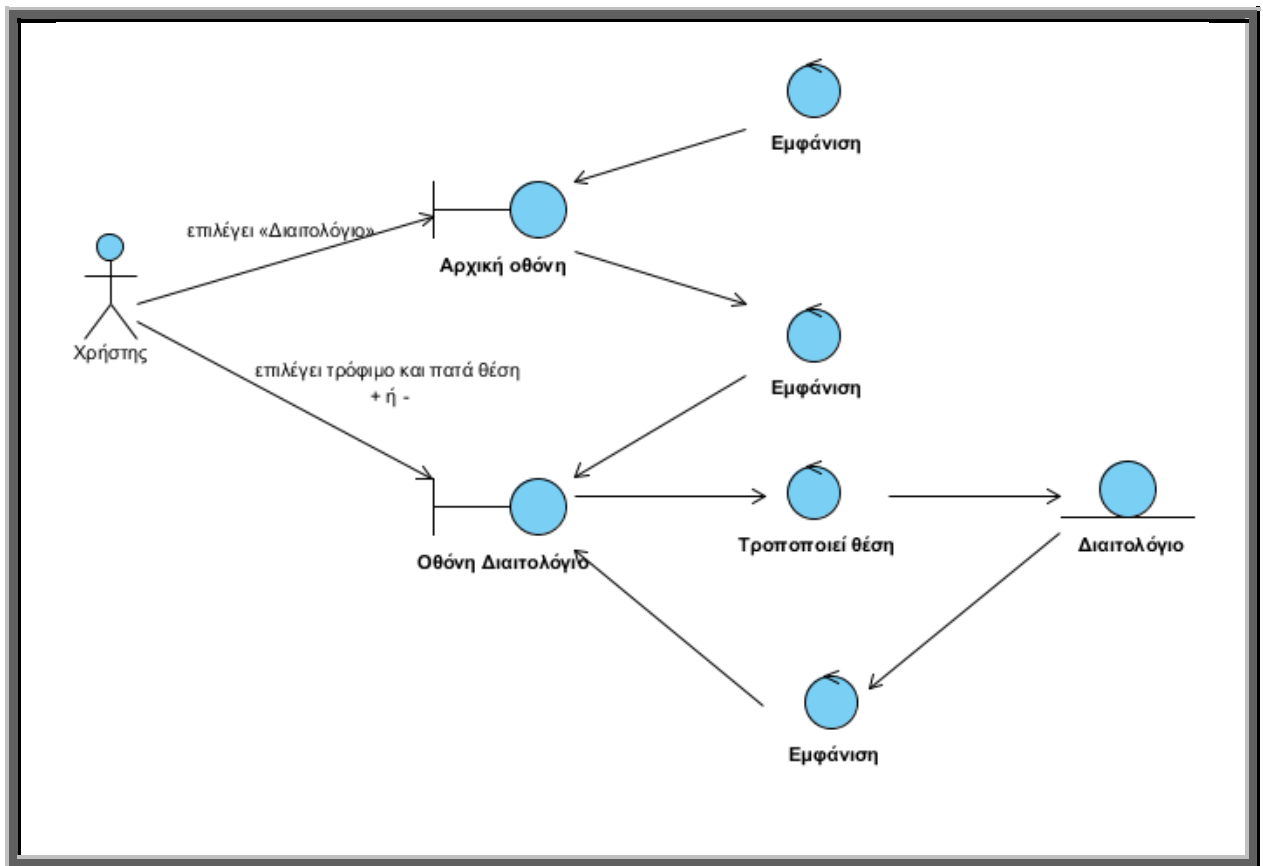
Ο χρήστης επιλέγει τρόφιμο.

Πατά «Ποσότητα» + ή -.

Το σύστημα τροποποιεί την ποσότητα του τροφίμου.

Το σύστημα συνεχίζει από την Οθόνη Διαιτολόγιο.

Εναλλακτική Ροή 6



Εικόνα 5.14: Δημιουργία ενός διαιτολογίου EP6

Ο χρήστης επιλέγει «Διαιτολόγιο».

Το σύστημα εμφανίζει την οθόνη «Διαιτολόγιο».

Ο χρήστης επιλέγει τρόφιμο.

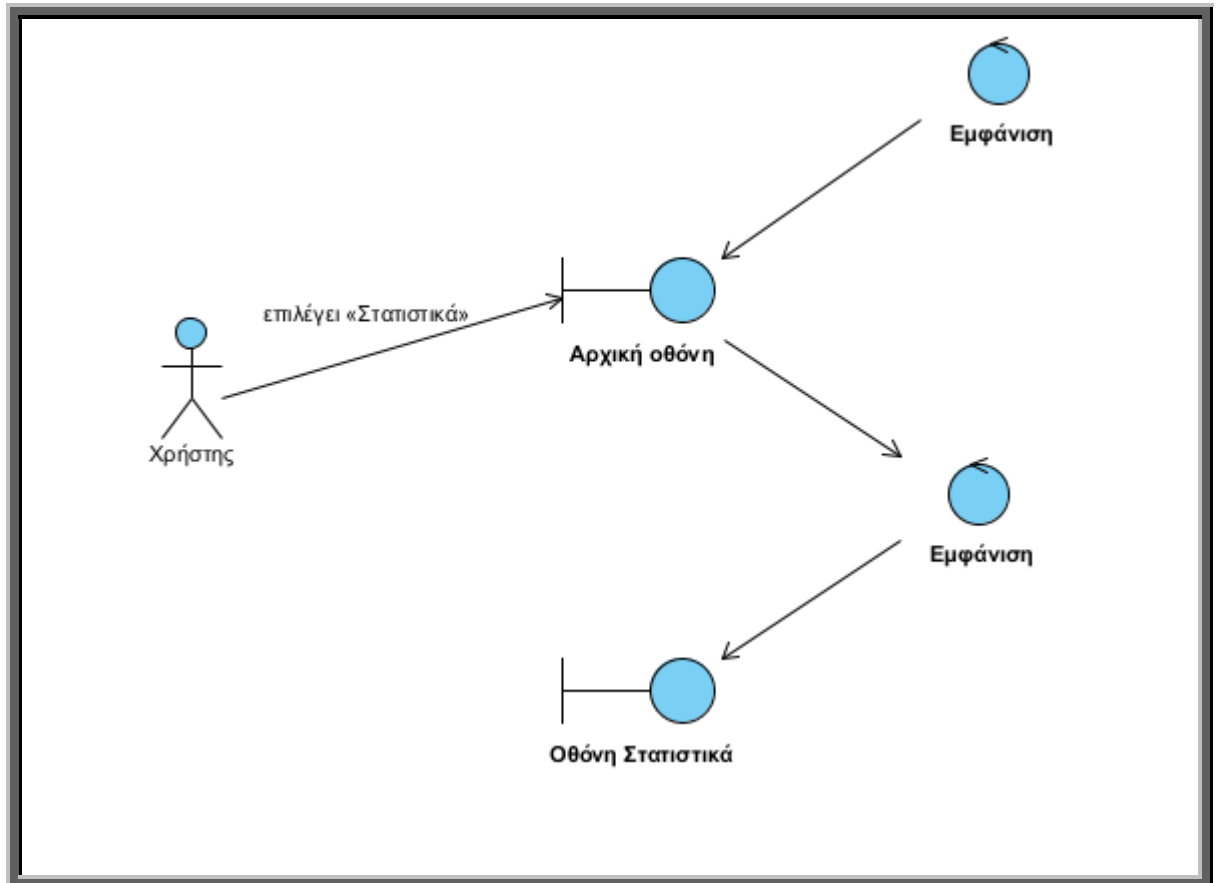
Πατά «θέση» + ή -.

Το σύστημα τροποποιεί τη θέση του τροφίμου στο γεύμα.

Το σύστημα συνεχίζει από την Οθόνη Διαιτολόγιο.

6.10 Επίδειξη στατιστικών της δίαιτας ΔΕ09

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ09.



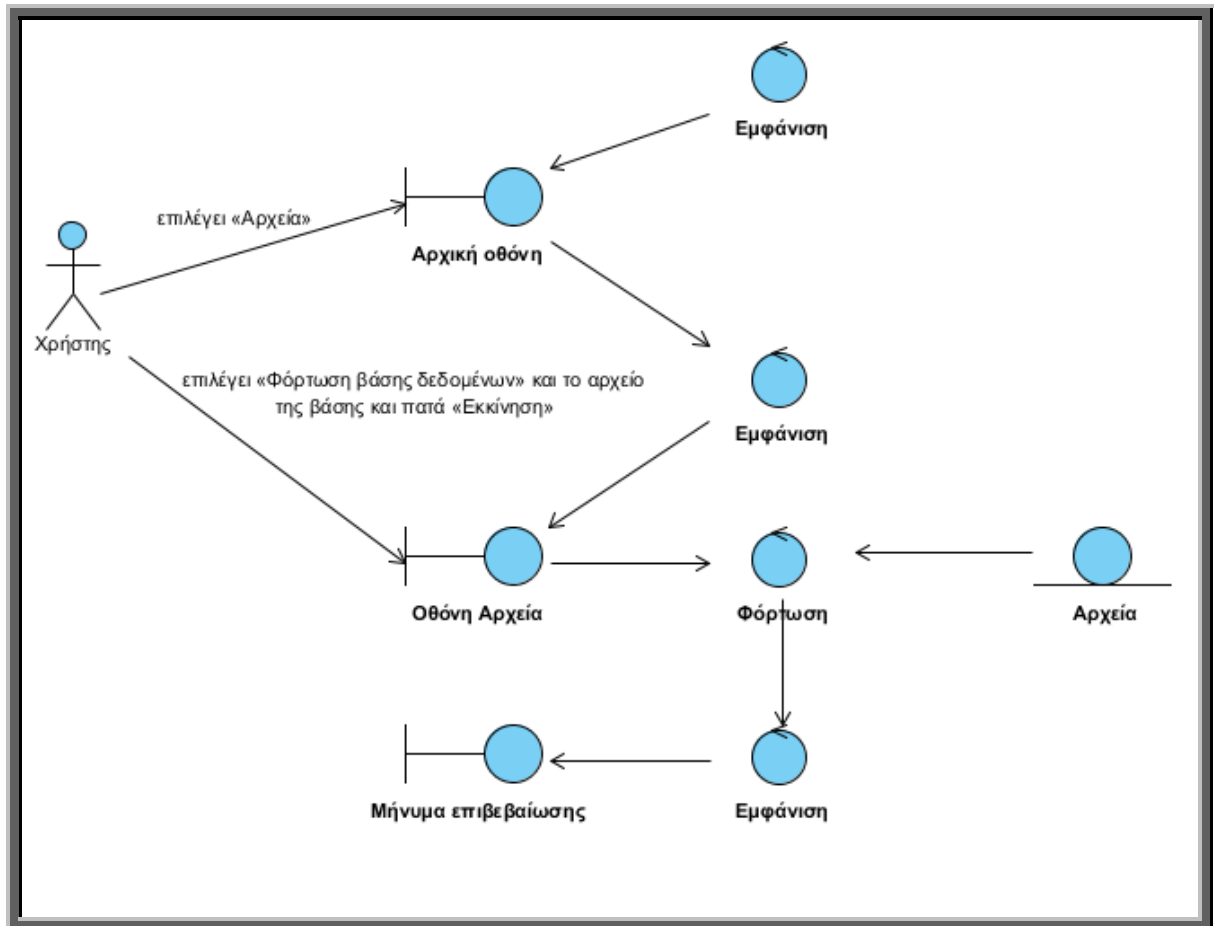
Εικόνα 5.15: Επίδειξη στατιστικών της δίαιτας ΔΕ09

Ο χρήστης επιλέγει «Στατιστικά».

Το σύστημα υπολογίζει και εμφανίζει την οθόνη «Στατιστικά».

6.11 Φόρτωση βάσης δεδομένων ΔΕ10

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ10.



Εικόνα 5.16: Φόρτωση βάσης δεδομένων ΔΕ10

Ο χρήστης επιλέγει «Αρχεία».

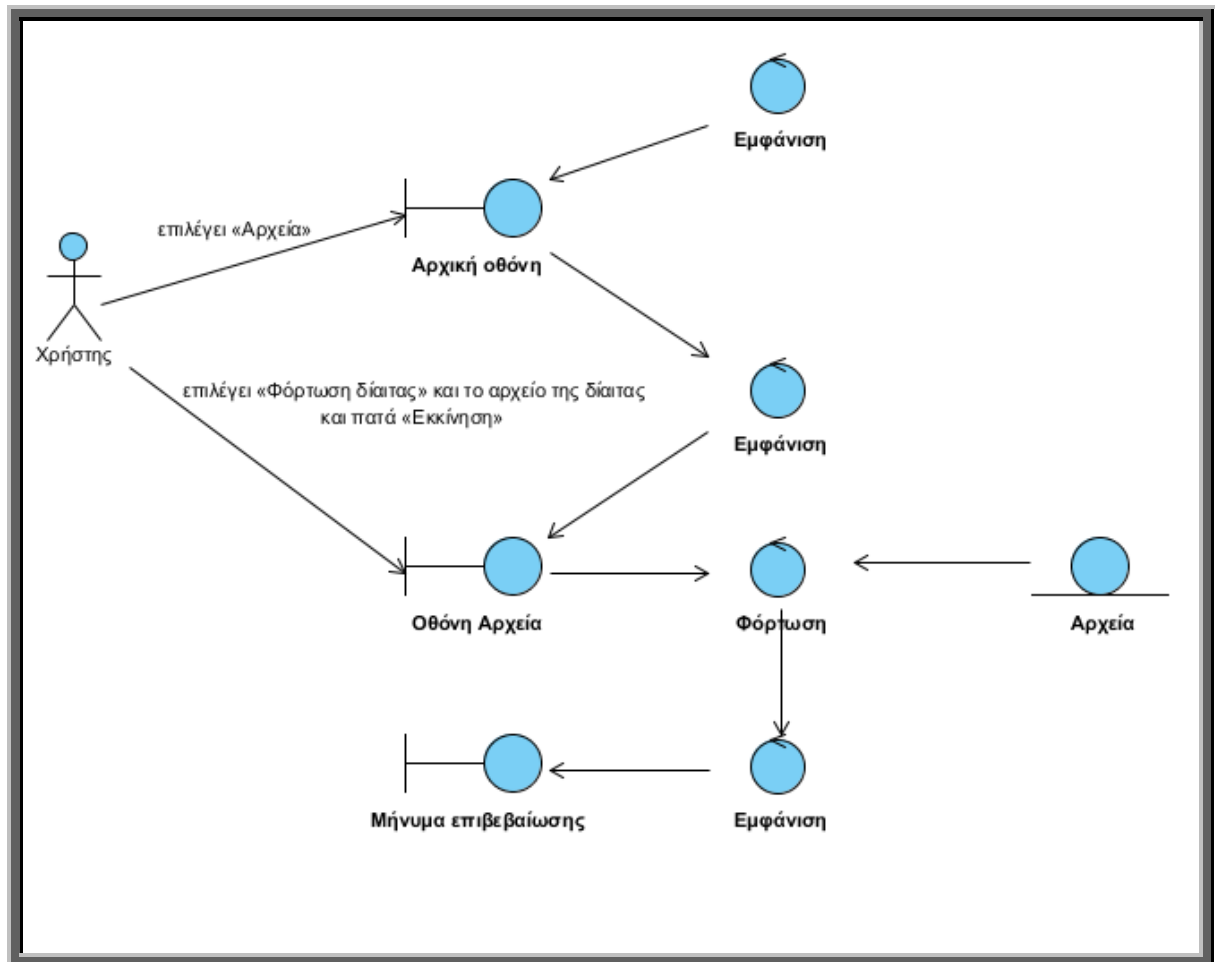
Το σύστημα εμφανίζει την οθόνη «Αρχεία».

Ο χρήστης επιλέγει «Φόρτωση βάσης δεδομένων» και το αρχείο της βάσης και πατά «Εκκίνηση».

Το σύστημα φορτώνει τη βάση και εμφανίζει μήνυμα επιτυχούς φόρτωσης.

6.12 Φόρτωση Διαιτολογίου ΔΕ11

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ11.



Εικόνα 5.17: Φόρτωση Διαιτολογίου ΔΕ11

Ο χρήστης επιλέγει «Αρχεία».

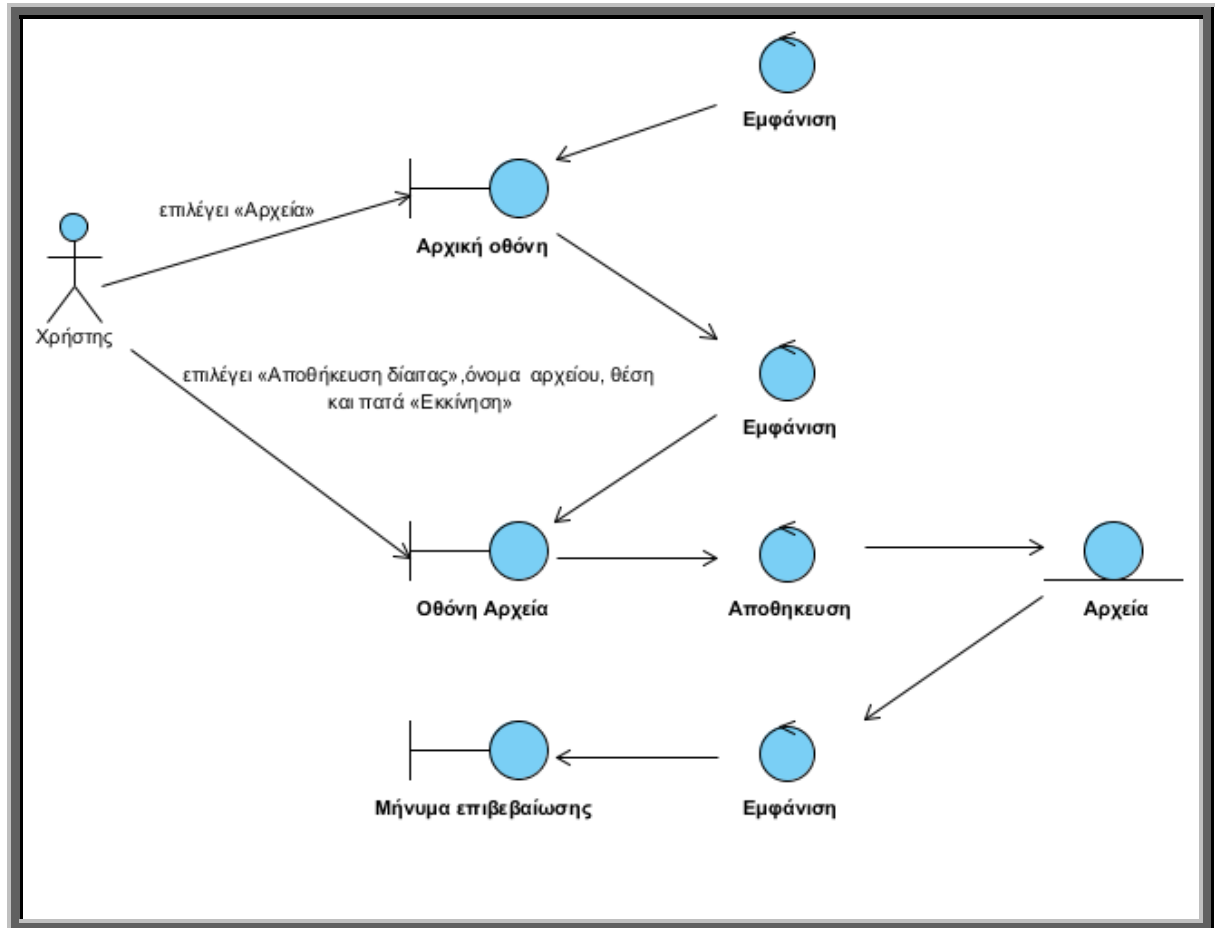
Το σύστημα εμφανίζει την οθόνη «Αρχεία».

Ο χρήστης επιλέγει «Φόρτωση διαίτας» και το αρχείο της διαίτας και πατά «Εκκίνηση».

Το σύστημα φορτώνει τη διαίτα και εμφανίζει μήνυμα επιτυχούς φόρτωσης.

6.13 Αποθήκευση διαιτολογίου ΔΕ12

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ12.



Εικόνα 5.18: Αποθήκευση διαιτολογίου ΔΕ12

Ο χρήστης επιλέγει «Αρχεία».

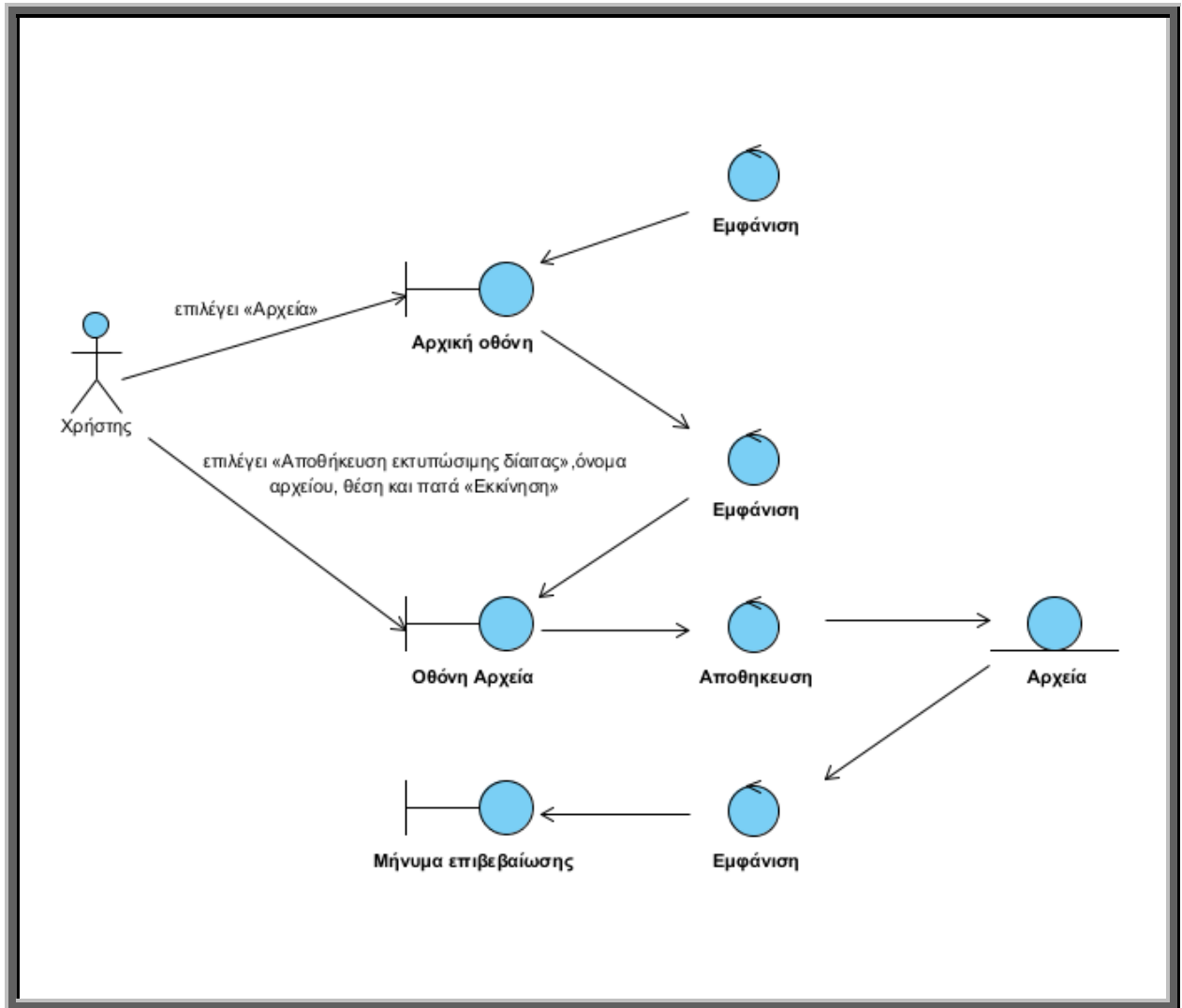
Το σύστημα εμφανίζει την οθόνη «Αρχεία».

Ο χρήστης επιλέγει «Αποθήκευση διαίτας», το όνομα του αρχείου και τη θέση που θα αποθηκευτεί το αρχείο της διαίτας και πατά «Εκκίνηση».

Το σύστημα αποθηκεύει τη διαίτα και εμφανίζει μήνυμα επιτυχούς αποθήκευσης.

6.14 Αποθήκευση διαιτολογίου σε εκτυπώσιμη μορφή ΔΕ13

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ13.



Εικόνα 5.19: Αποθήκευση διαιτολογίου σε εκτυπώσιμη μορφή ΔΕ13

Το σύστημα εμφανίζει την αρχική οθόνη.

Ο χρήστης επιλέγει «Αρχεία».

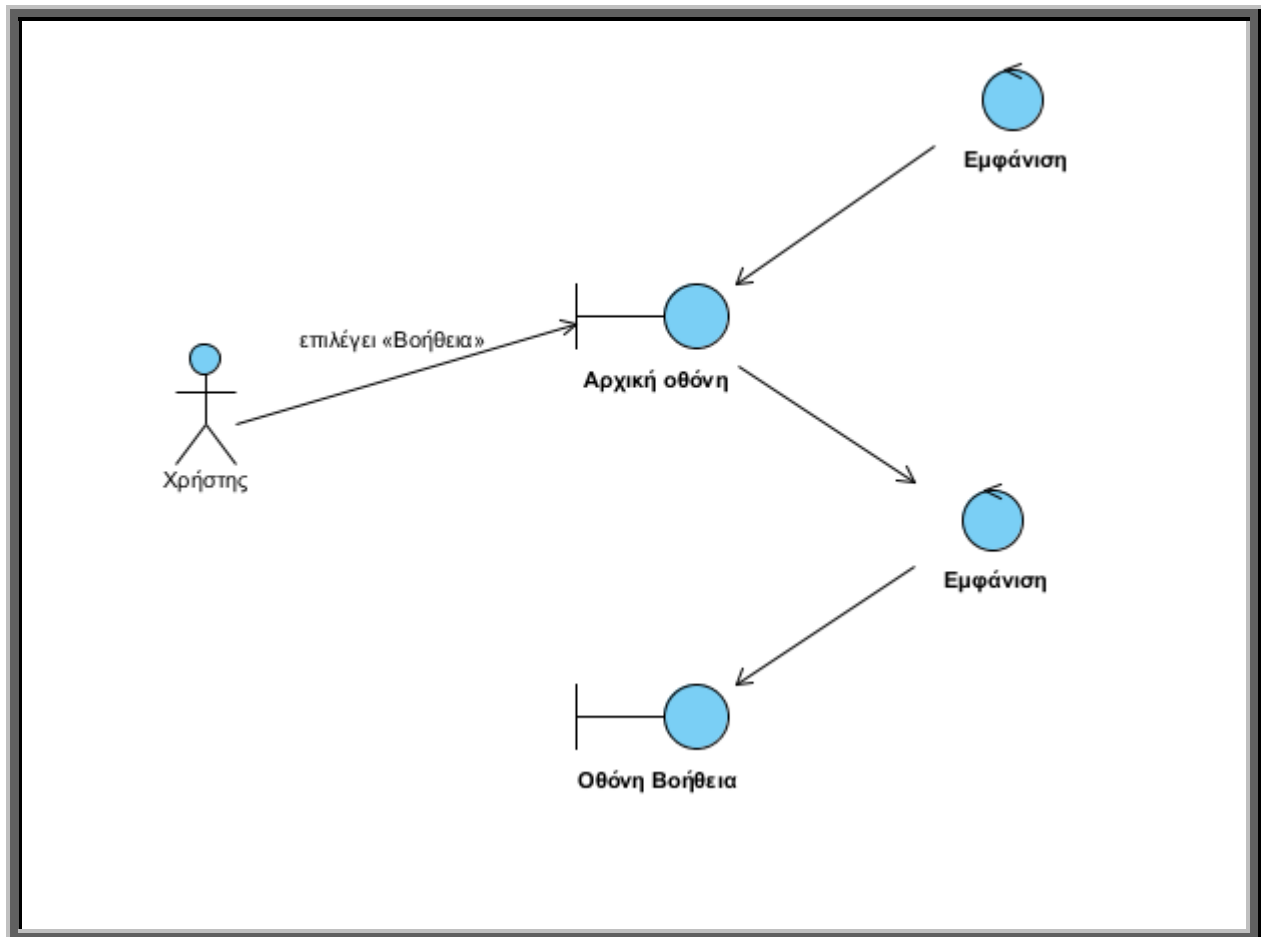
Το σύστημα εμφανίζει την οθόνη «Αρχεία».

Ο χρήστης επιλέγει «Αποθήκευση εκτυπώσιμης δίαιτας», το όνομα του αρχείου και τη θέση που θα αποθηκευτεί το αρχείο και πατά «Εκκίνηση».

Το σύστημα αποθηκεύει το αρχείο και εμφανίζει μήνυμα επιτυχούς αποθήκευσης.

6.15 Λήψη Βοήθειας ΔΕ14

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ14.



Εικόνα 5.20: Λήψη Βοήθειας ΔΕ14

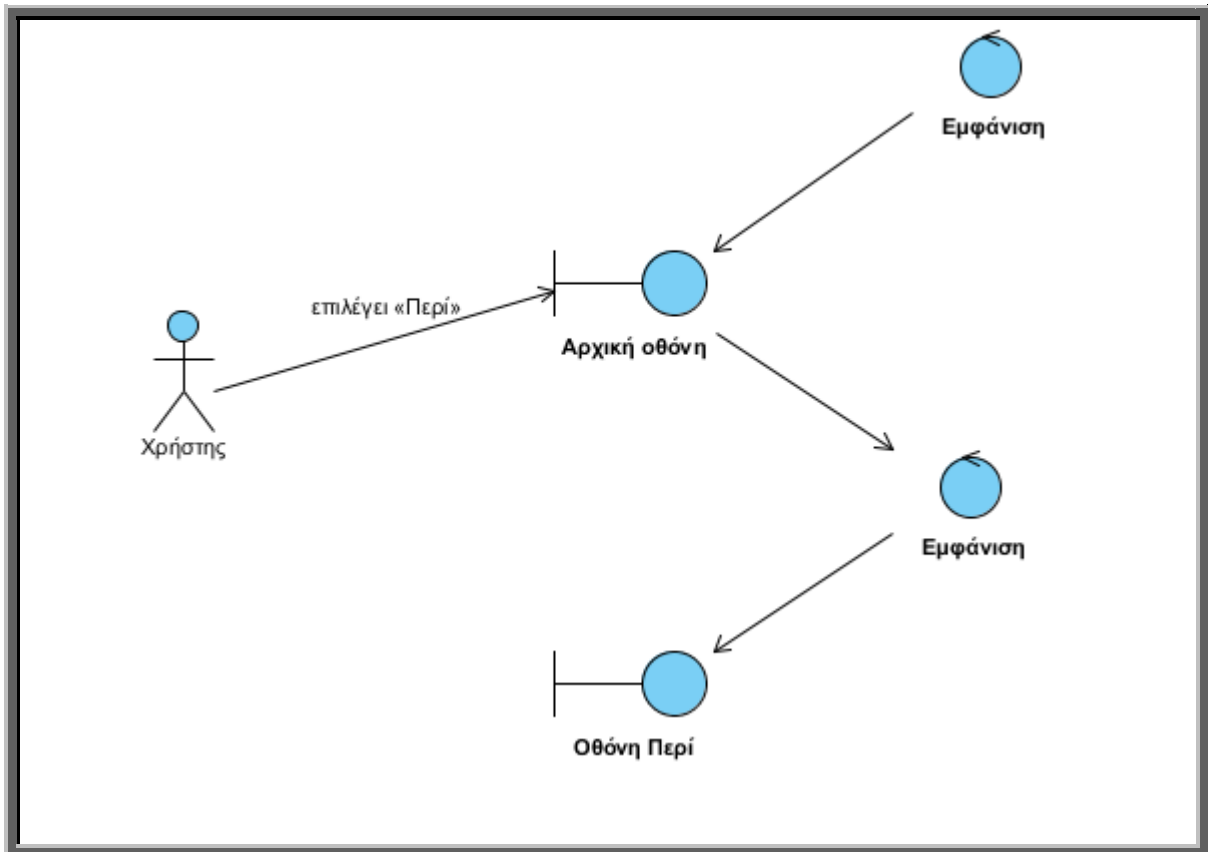
Το σύστημα εμφανίζει την αρχική οθόνη.

Ο χρήστης επιλέγει «Βοήθεια».

Το σύστημα εμφανίζει την οθόνη «Βοήθεια».

6.16 Επίδειξη «ταυτότητας» του προγράμματος ΔΕ15

Το διάγραμμα ευρωστίας αντιστοιχεί στην περίπτωση χρήσης ΠΧ15.



Εικόνα 5.21: Επίδειξη «ταυτότητας» του προγράμματος ΔΕ15

Το σύστημα εμφανίζει την αρχική οθόνη.

Ο χρήστης επιλέγει «Περί».

Το σύστημα εμφανίζει την οθόνη «Περί».

Κεφάλαιο 7

Σχεδίαση

7.1 Διαγράμματα Ακολουθίας (Sequence Diagrams)

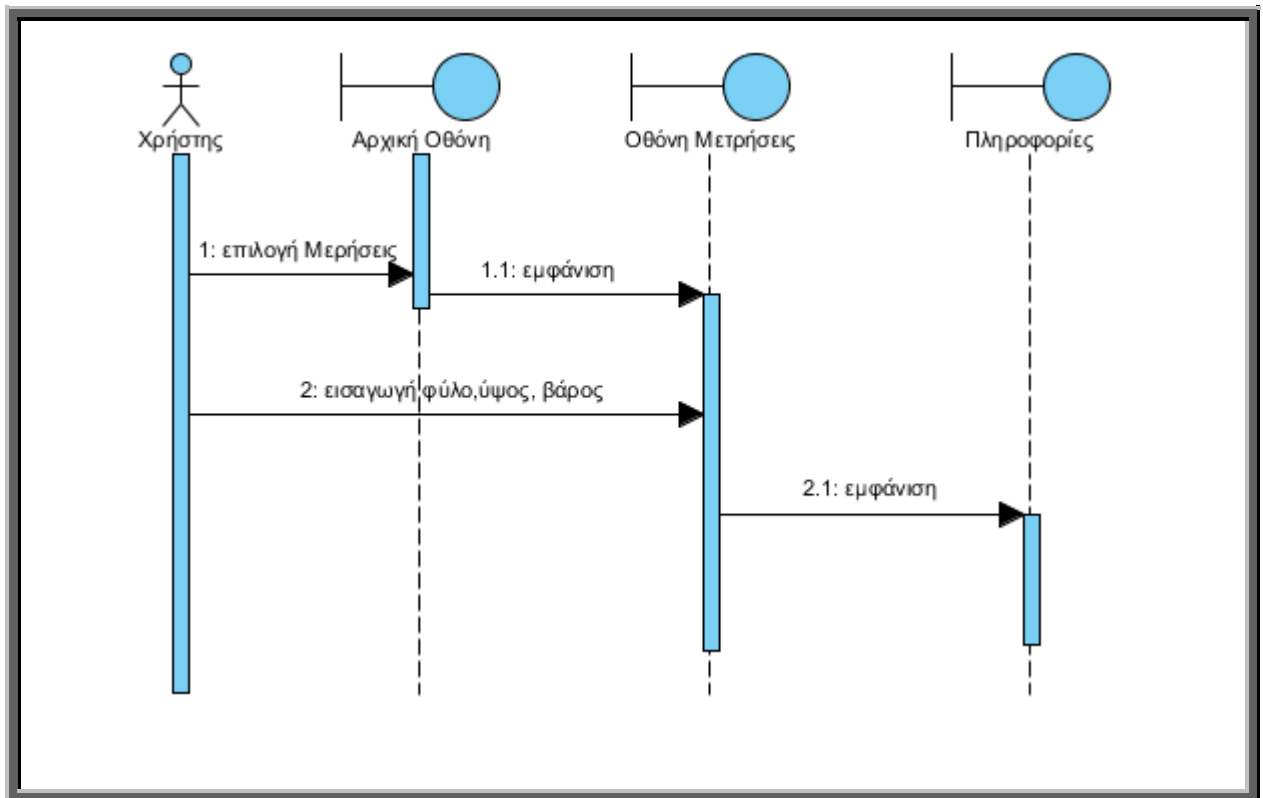
Μετά την ολοκλήρωση της ανάλυσης ευρωστίας, ακολουθούν τα διαγράμματα ακολουθίας. Η ανάλυση ευρωστίας που εντάσσεται στην προκαταρκτική σχεδίαση, στοχεύει στον εντοπισμό αντικειμένων. Η λεπτομερειακή σχεδίαση, όπου ανήκουν τα διαγράμματα ακολουθίας, στοχεύει στην αλληλεπίδραση των αντικειμένων [16].

Ένα διάγραμμα ακολουθίας αναπαριστά την αλληλεπίδραση, δηλαδή μια σειρά από μηνύματα ανάμεσα σε κλάσεις μέσα σε μια συνεργασία, ώστε να πραγματοποιηθεί μια επιθυμητή πράξη ή αποτέλεσμα.

Ένα διάγραμμα ακολουθίας έχει 2 διαστάσεις, την κάθετη διάσταση που αναπαριστά τον χρόνο και την οριζόντια που αναπαριστά τα διάφορα αντικείμενα.

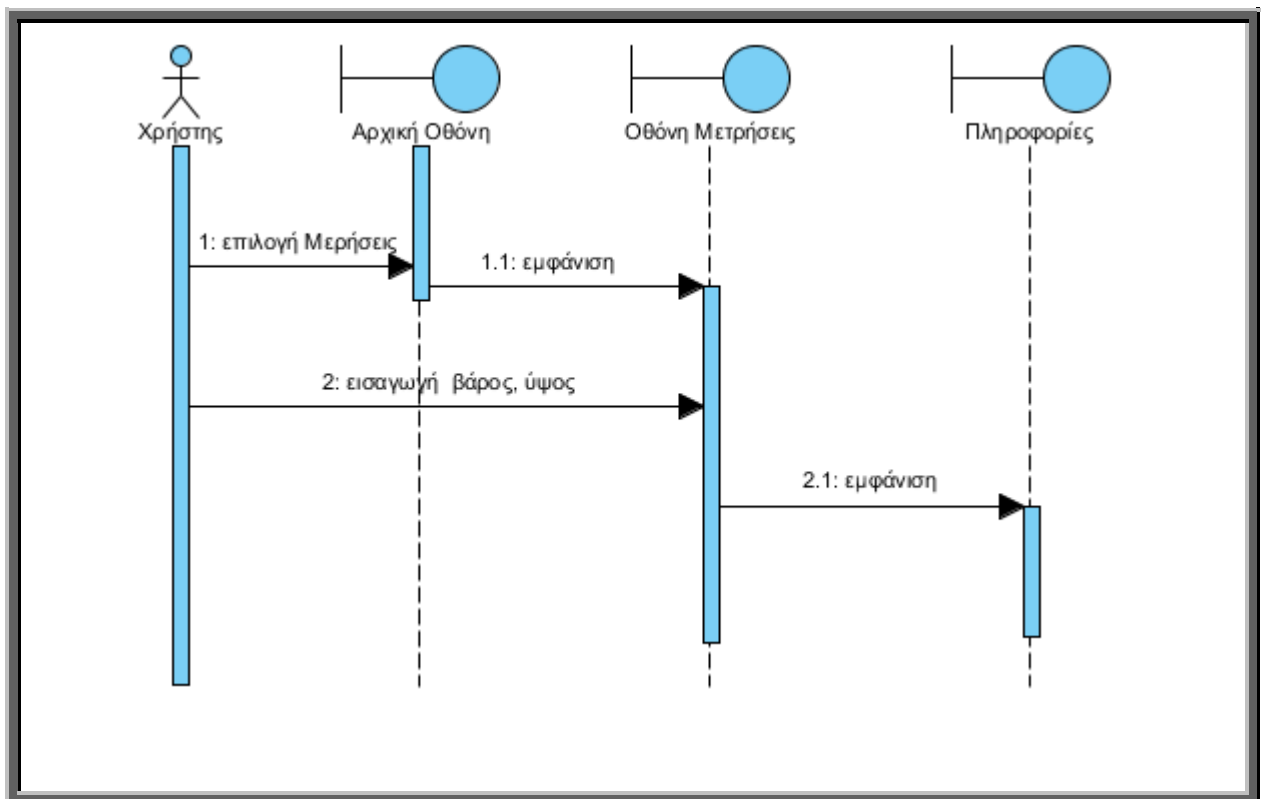
Τα διαγράμματα ακολουθίας που προκύπτουν για το αναπτυσσόμενο πρόγραμμα είναι τα εξής:

7.2 Καθορισμός ιδανικού βάρους ΔΑ01



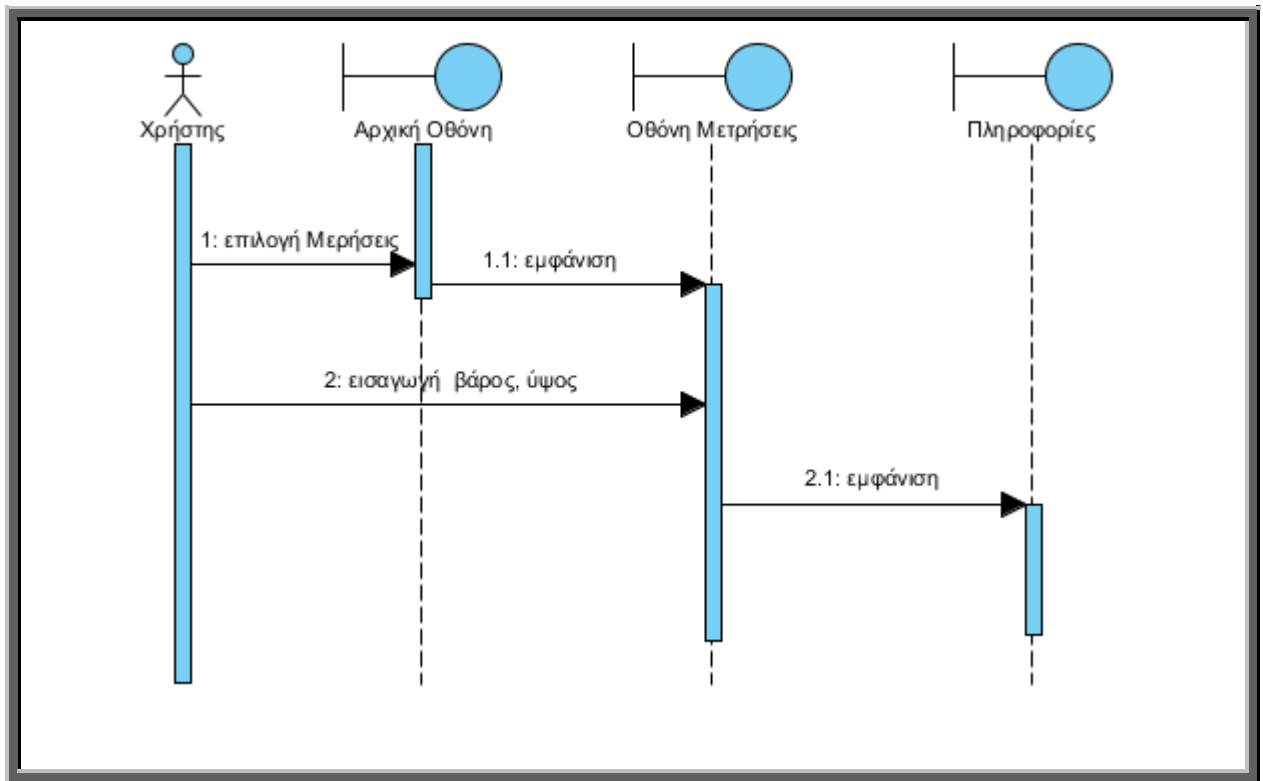
Εικόνα 6.1 Καθορισμός ιδανικού βάρους ΔΑ01

7.3 Καθορισμός κατάστασης βάρους ατόμου ΔΑ02



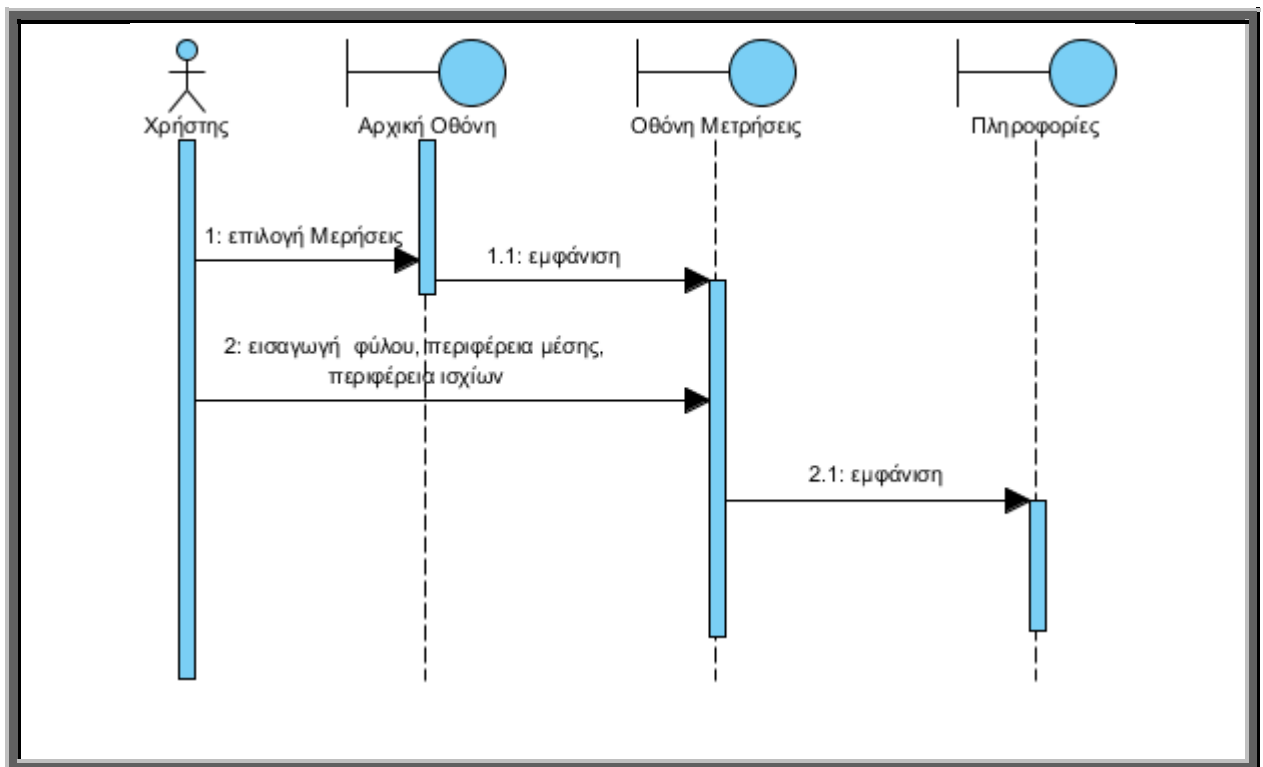
Εικόνα 6.2 Καθορισμός κατάστασης βάρους ατόμου ΔΑ02

7.4 Καθορισμός ποσότητας λίπους σώματός ΔΑ03



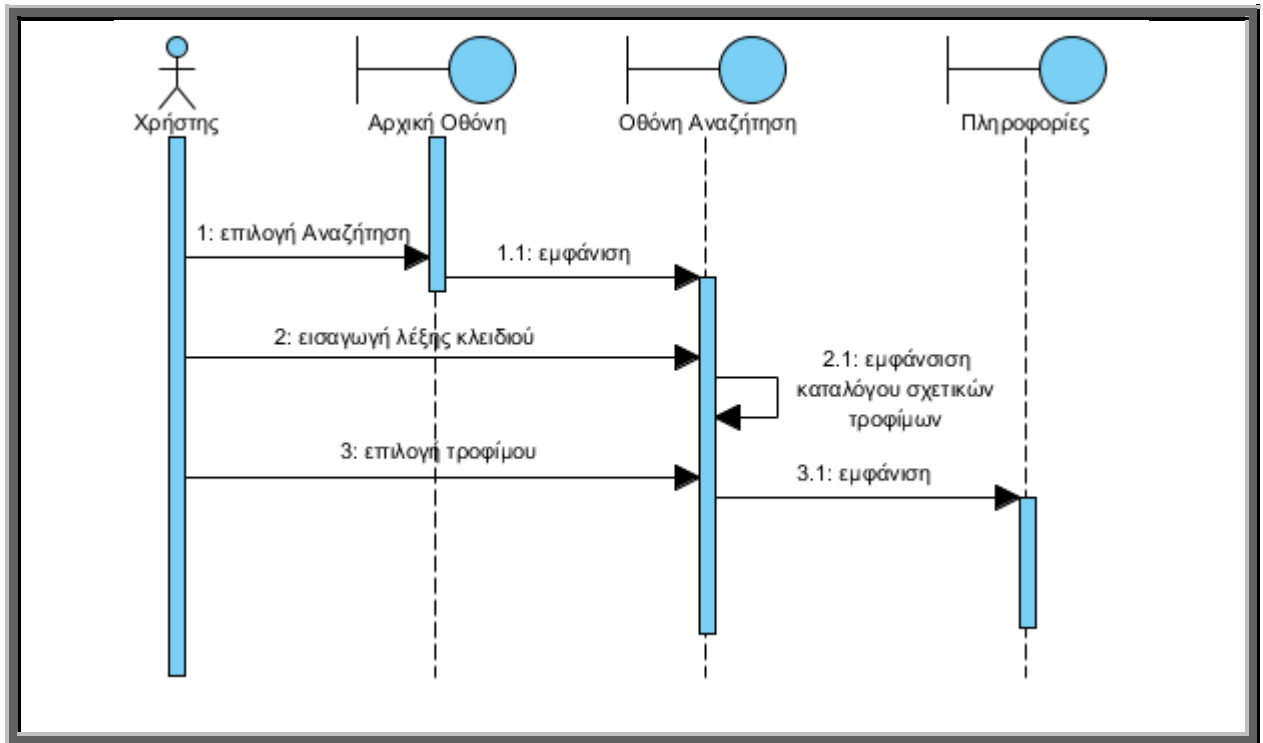
Εικόνα 6.3 Καθορισμός ποσότητας λίπους σώματός ΔΑ03

7.5 Καθορισμός ύπαρξης κοιλιακής παχυσαρκίας ΔΑ04



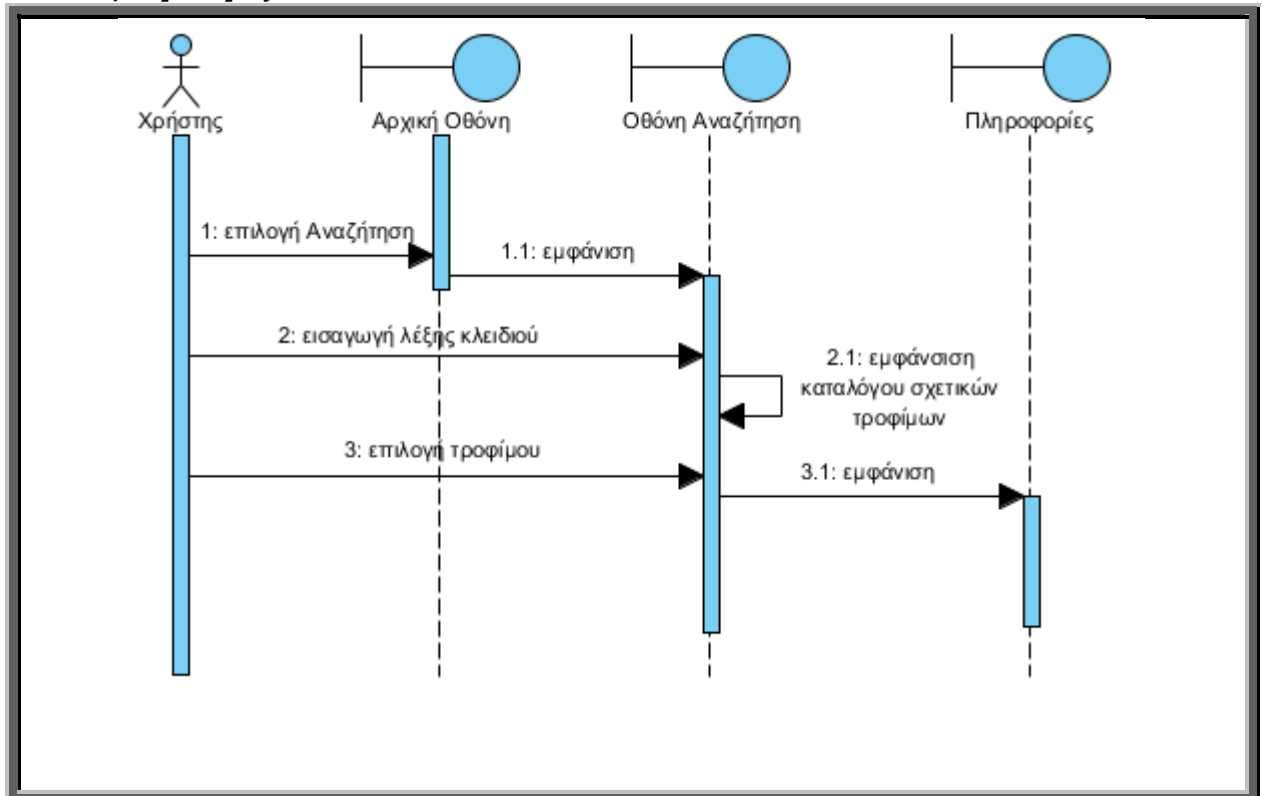
Εικόνα 6.4 Καθορισμός ύπαρξης κοιλιακής παχυσαρκίας ΔΑ04

7.6 Καθορισμός ενέργειας (θερμίδες) ενός τροφίμου ΔΑ05



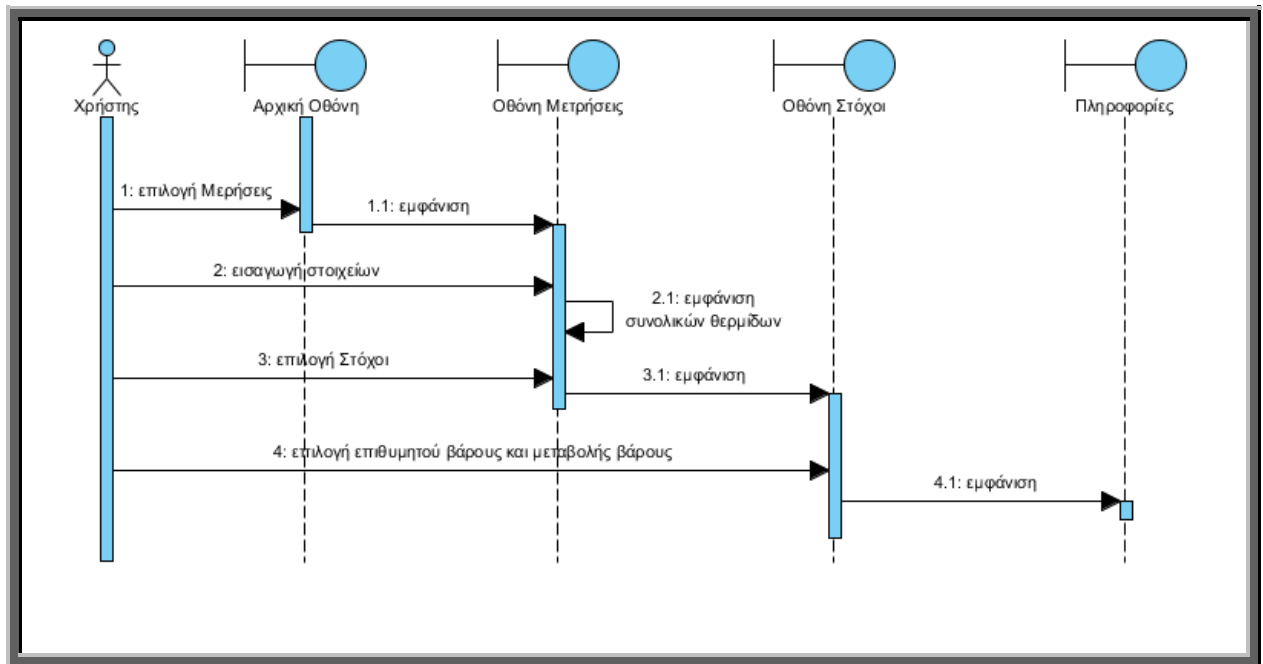
Εικόνα 6.5 Καθορισμός ενέργειας (θερμίδες) ενός τροφίμου ΔΑ05

7.7 Καθορισμός μακροθρεπτικών χαρακτηριστικών ενός τροφίμου ΔΑ06



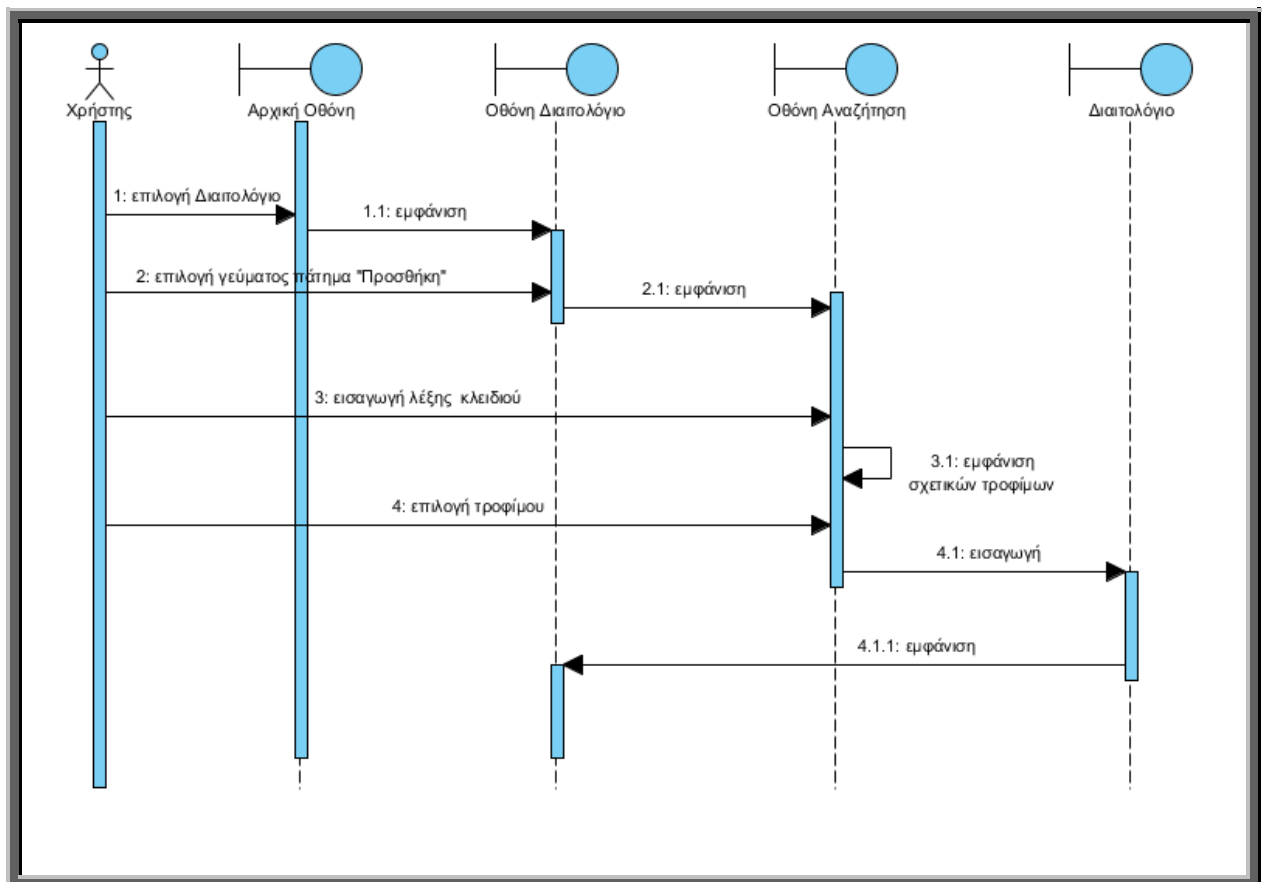
Εικόνα 6.6 Καθορισμός μακροθρεπτικών χαρακτηριστικών ενός τροφίμου ΔΑ06

7.8 Καθορισμός θερμίδων - στόχου μιας διαίτας ΔΑ07



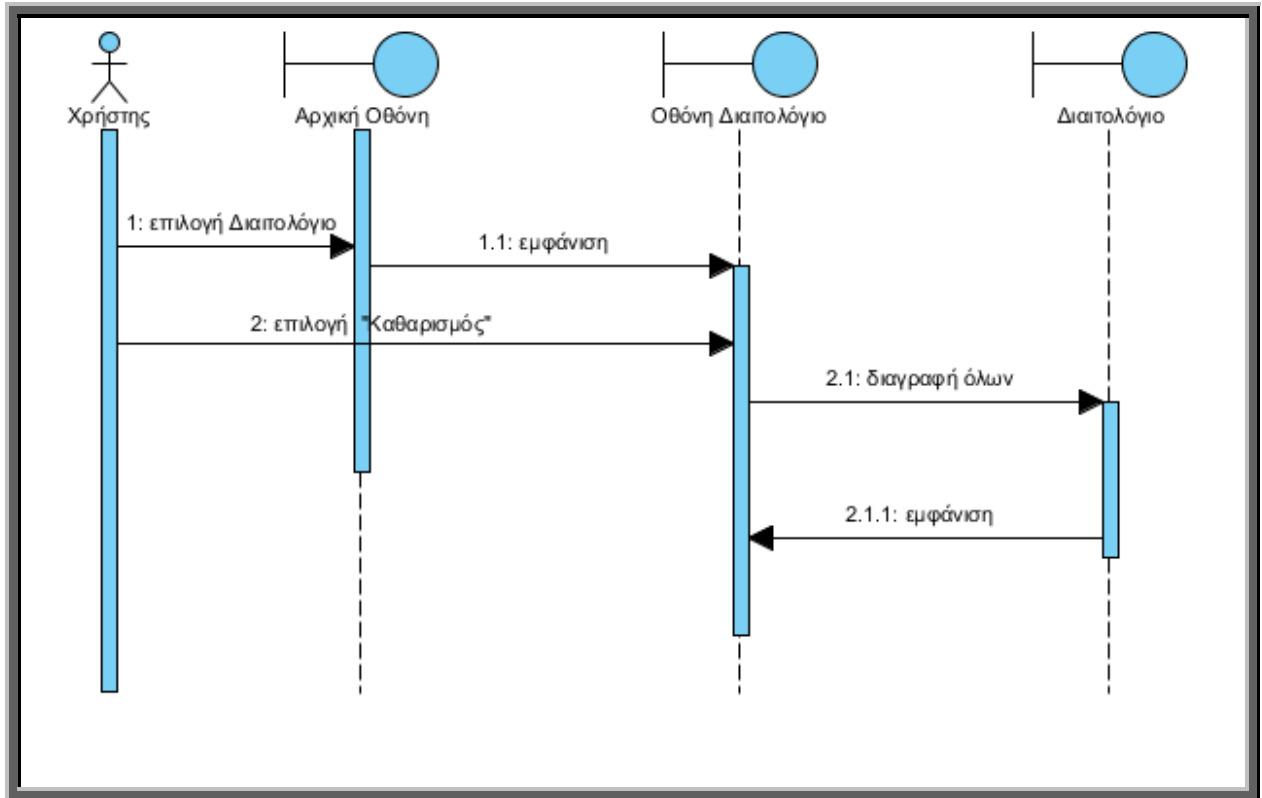
Εικόνα 6.7 Καθορισμός θερμίδων - στόχου μιας διαίτας ΔΑ07

7.9 Δημιουργία ενός διαιτολογίου ΔΑ08



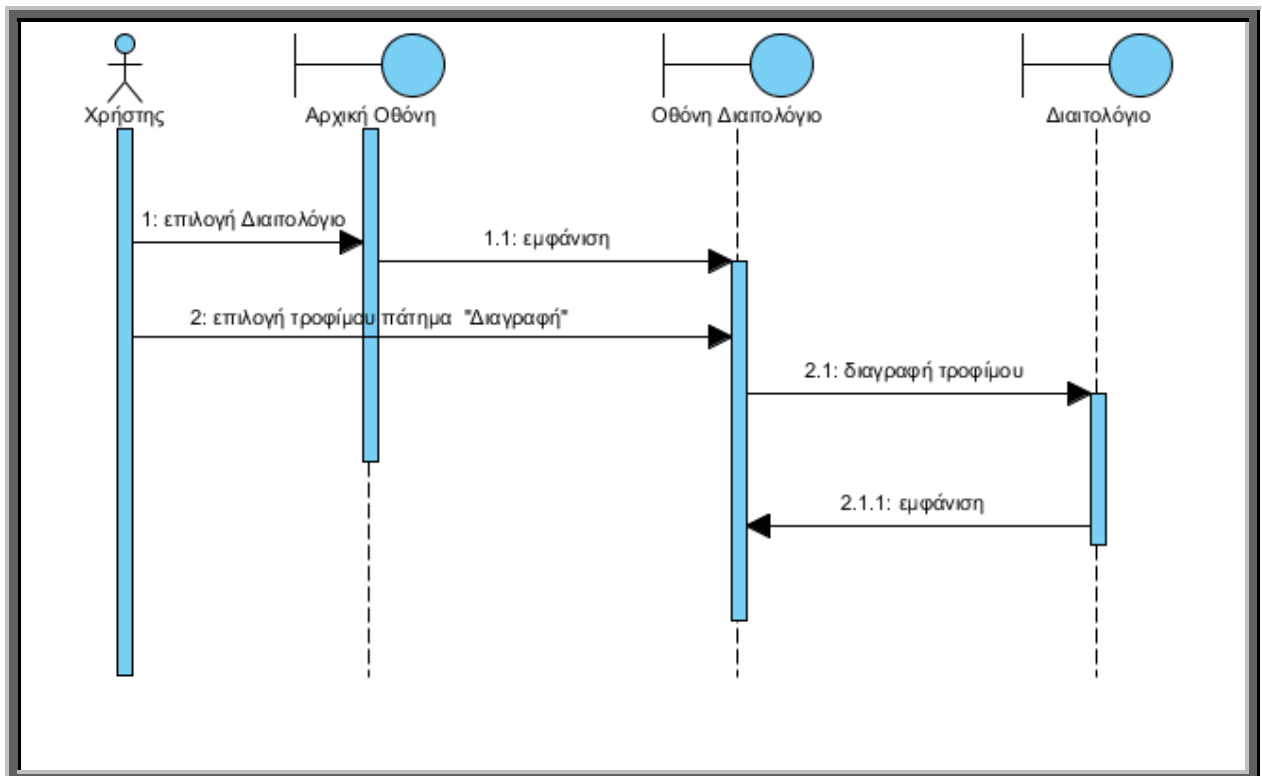
Εικόνα 6.8 Δημιουργία ενός διαιτολογίου ΔΑ08

Δε 08 Δημιουργία ενός διαιτολογίου EP1



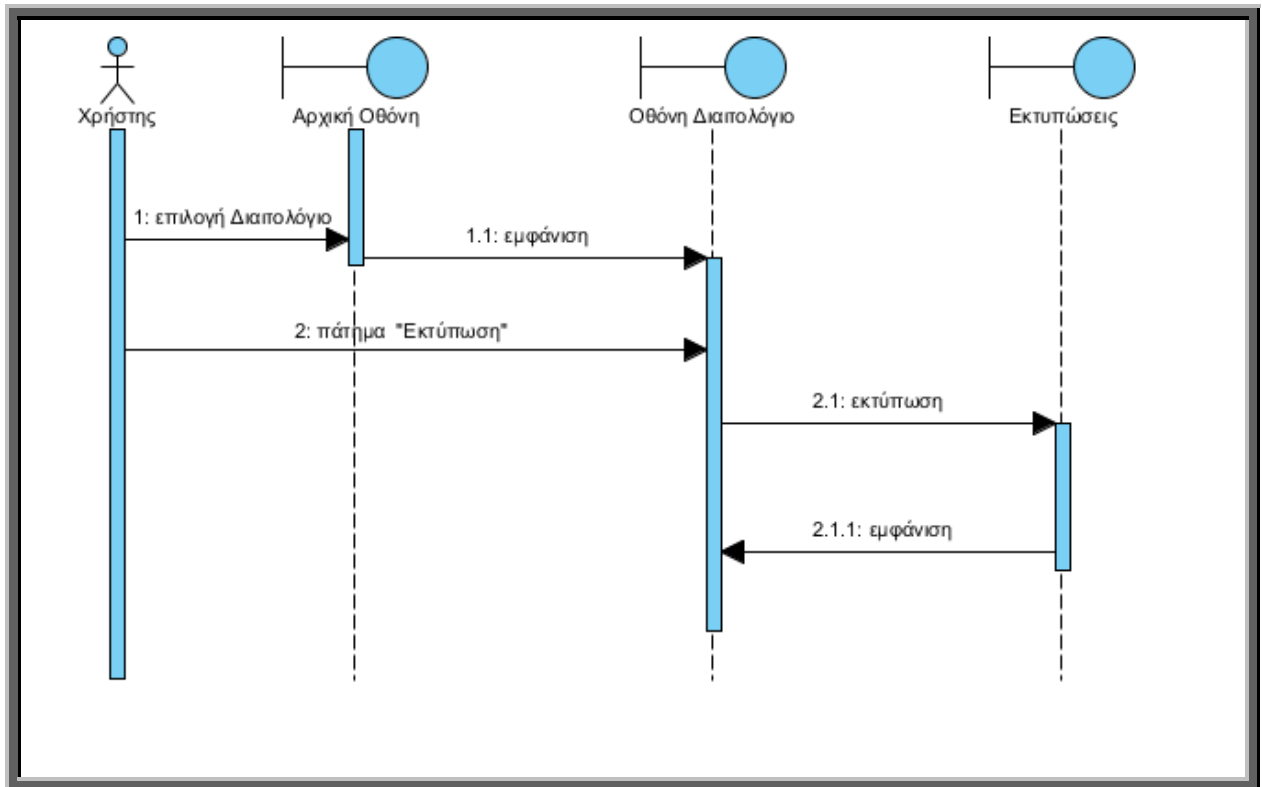
Εικόνα 6.9 Δημιουργία ενός διαιτολογίου EP1

Δε 08 Δημιουργία ενός διαιτολογίου EP2



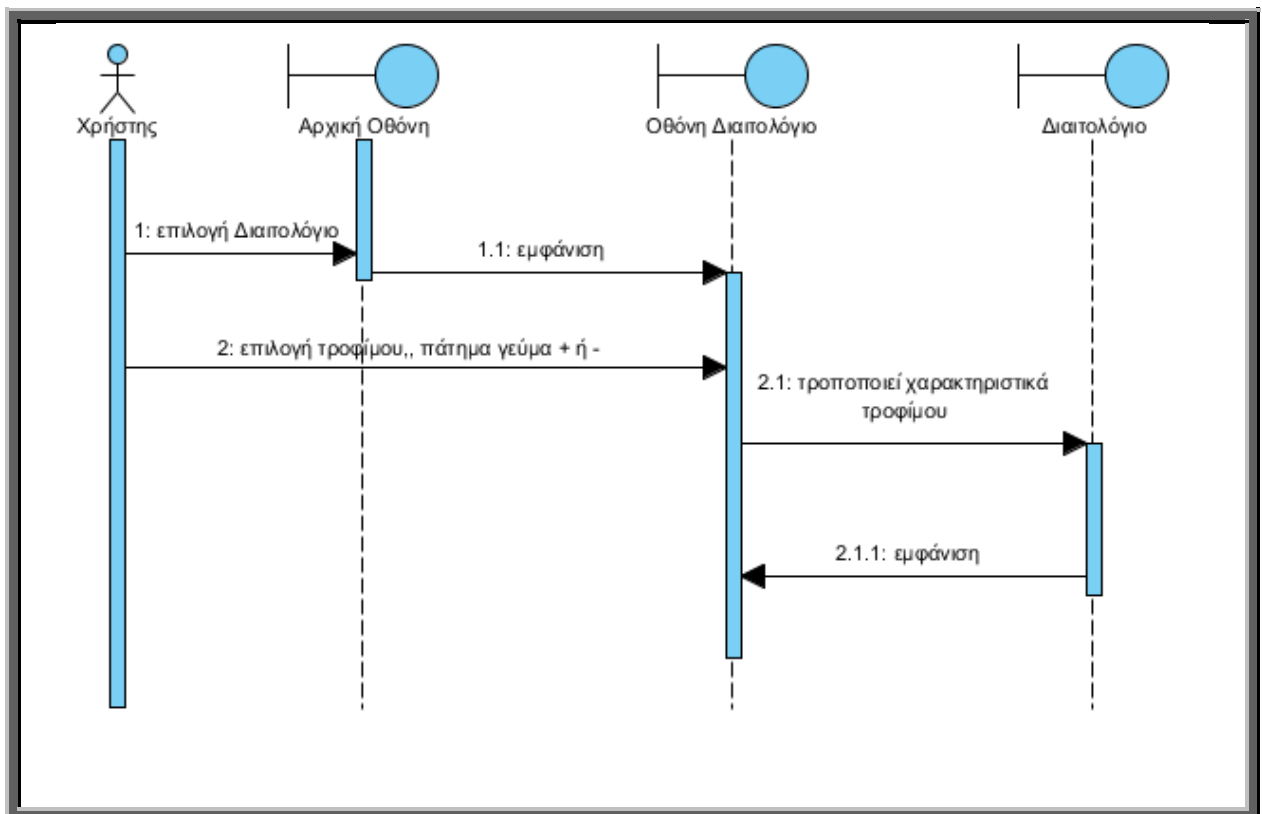
Εικόνα 6.10 Δημιουργία ενός διαιτολογίου EP2

Δε 08 Δημιουργία ενός διαιτολογίου EP3



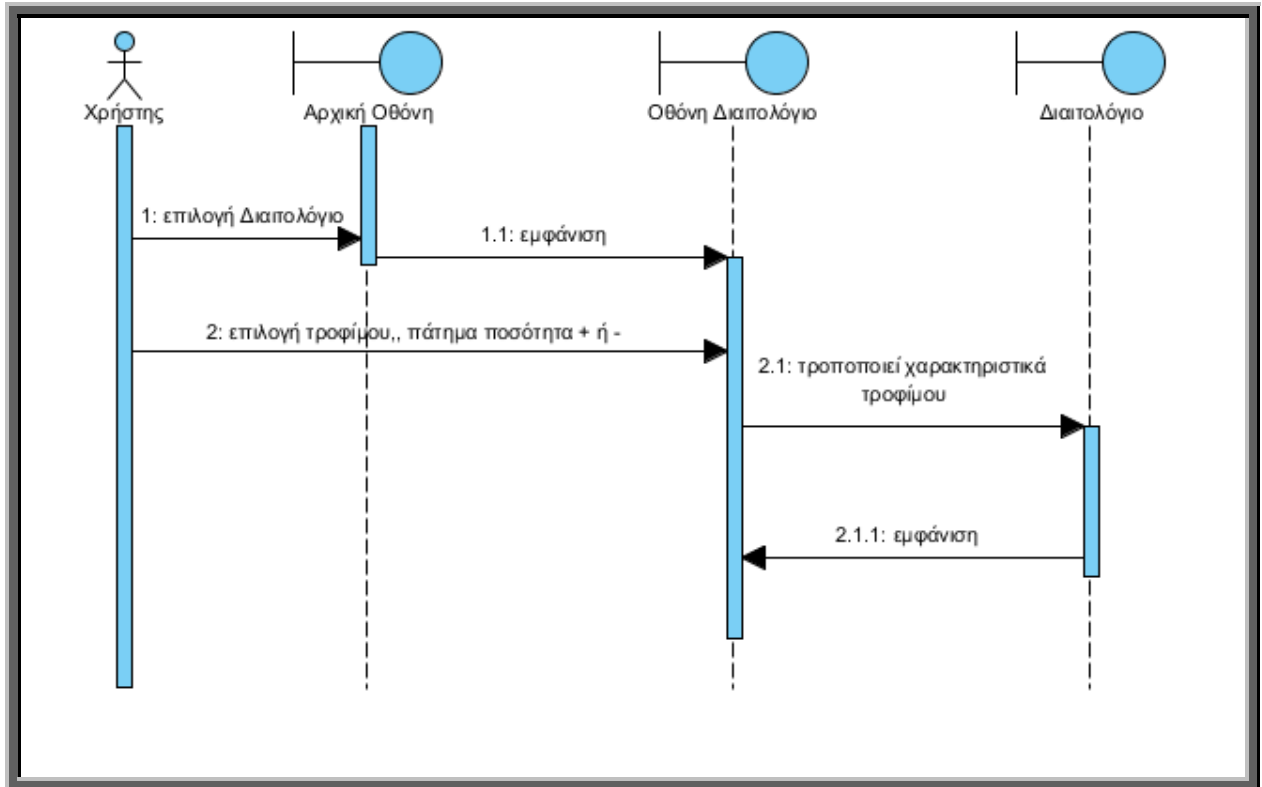
Εικόνα 6.11 Δημιουργία ενός διαιτολογίου EP3

Δε 08 Δημιουργία ενός διαιτολογίου EP4



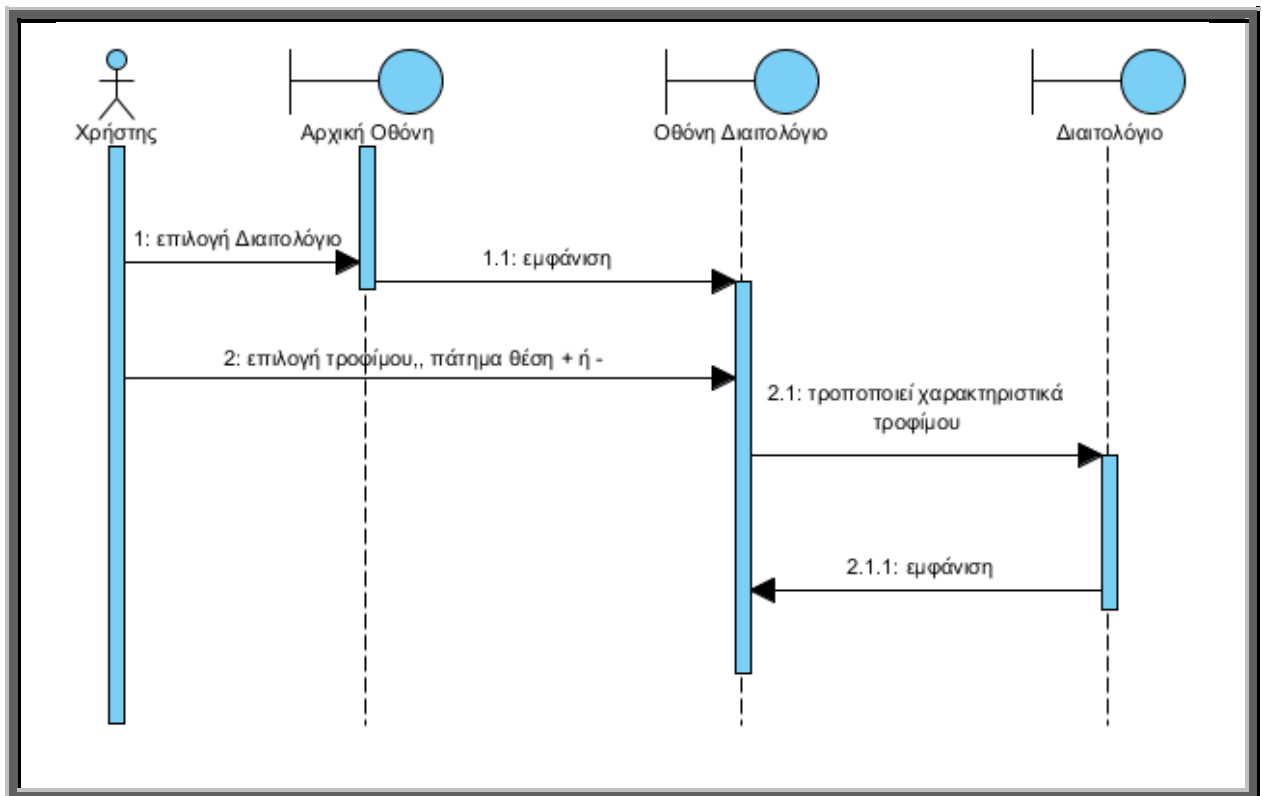
Εικόνα 6.12 Δημιουργία ενός διαιτολογίου EP4

Δε 08 Δημιουργία ενός διαιτολογίου EP5



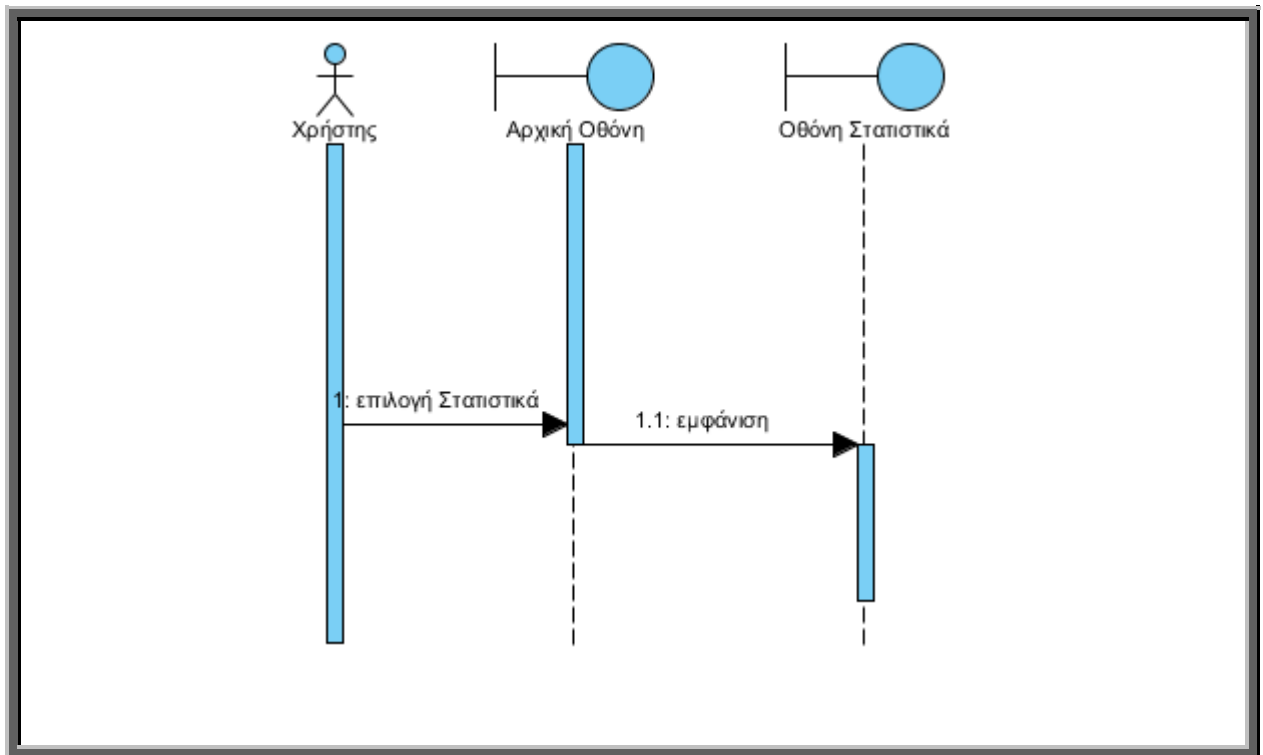
Εικόνα 6.13 Δημιουργία ενός διαιτολογίου EP5

Δε 08 Δημιουργία ενός διαιτολογίου EP6



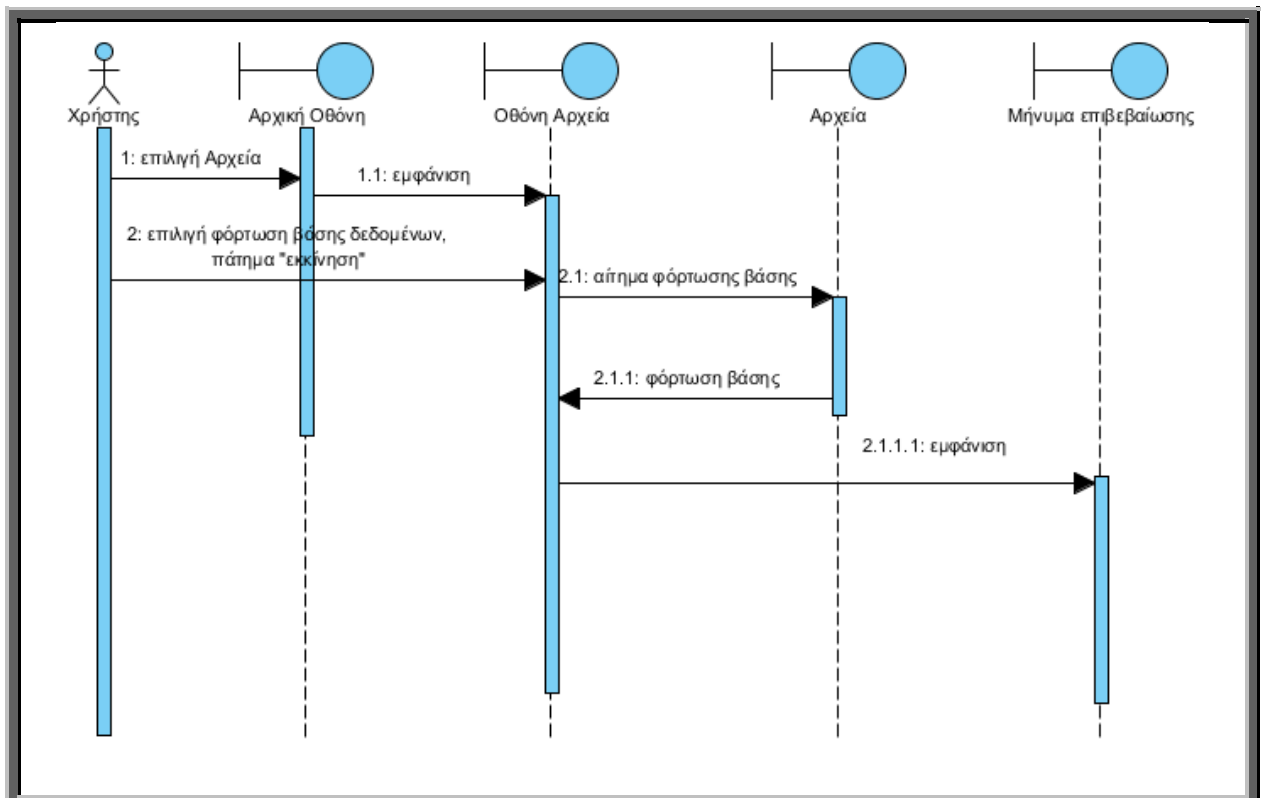
Εικόνα 6.14 Δημιουργία ενός διαιτολογίου EP6

7.10 Επίδειξη στατιστικών της διαίτας ΔΑ09



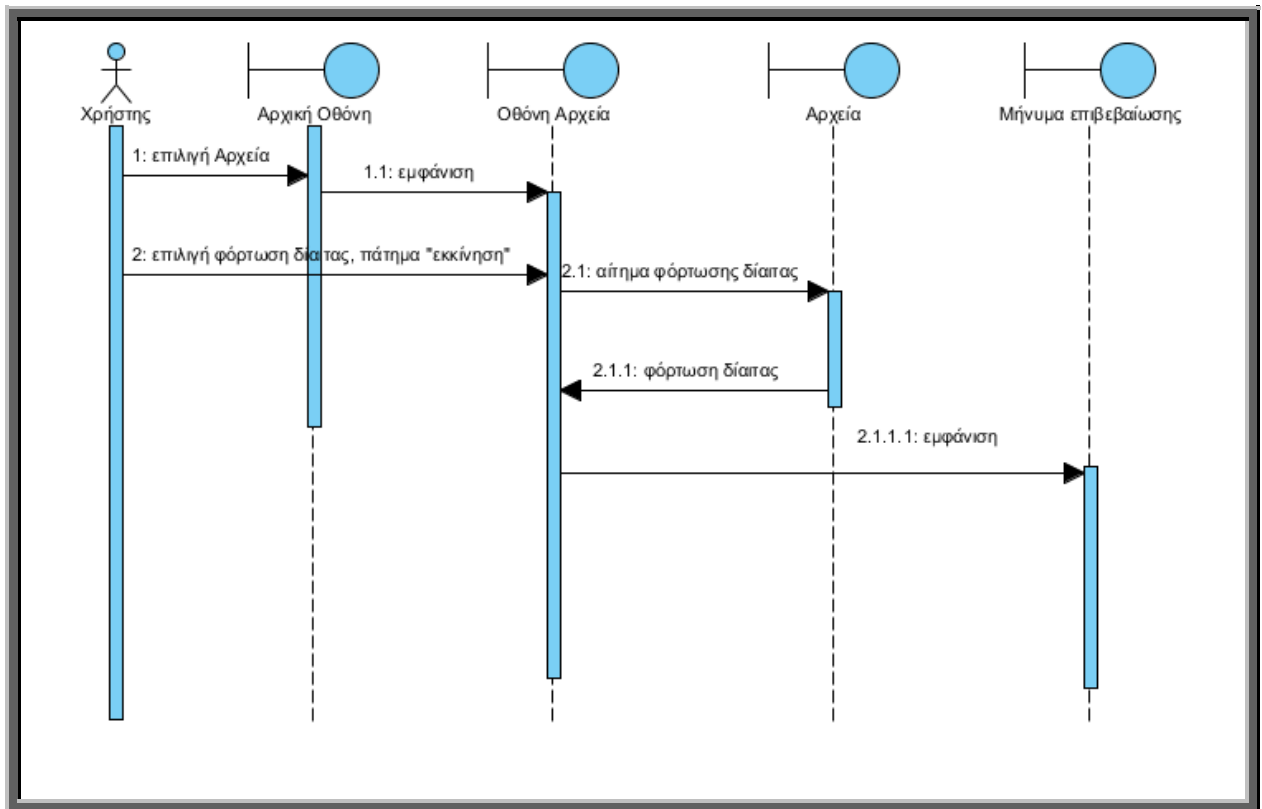
Εικόνα 6.15 Επίδειξη στατιστικών της διαίτας ΔΑ09

7.11 Φόρτωση βάσης δεδομένων ΔΑ10



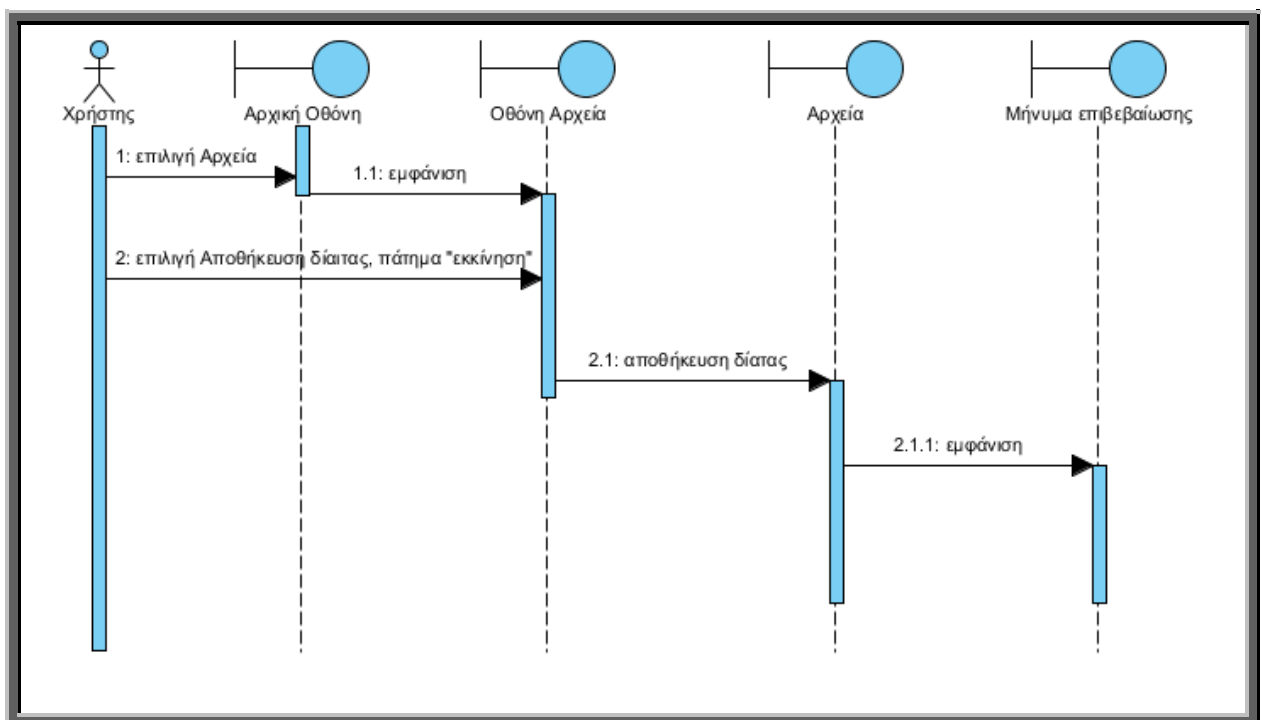
Εικόνα 6.16 Φόρτωση βάσης δεδομένων ΔΑ10

7.12 Φόρτωση Διαιτολογίου ΔΑ11



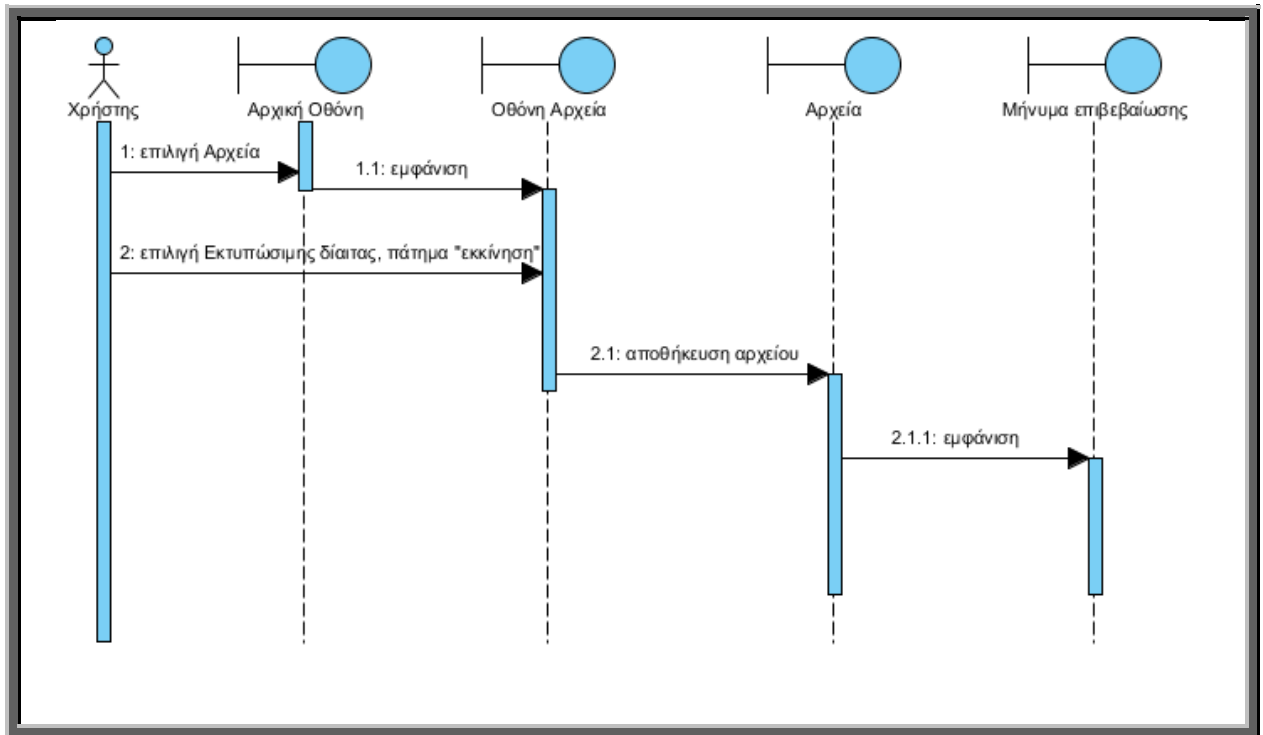
Εικόνα 6.17 Φόρτωση Διαιτολογίου ΔΑ11

7.13 Αποθήκευση διαιτολογίου ΔΑ12



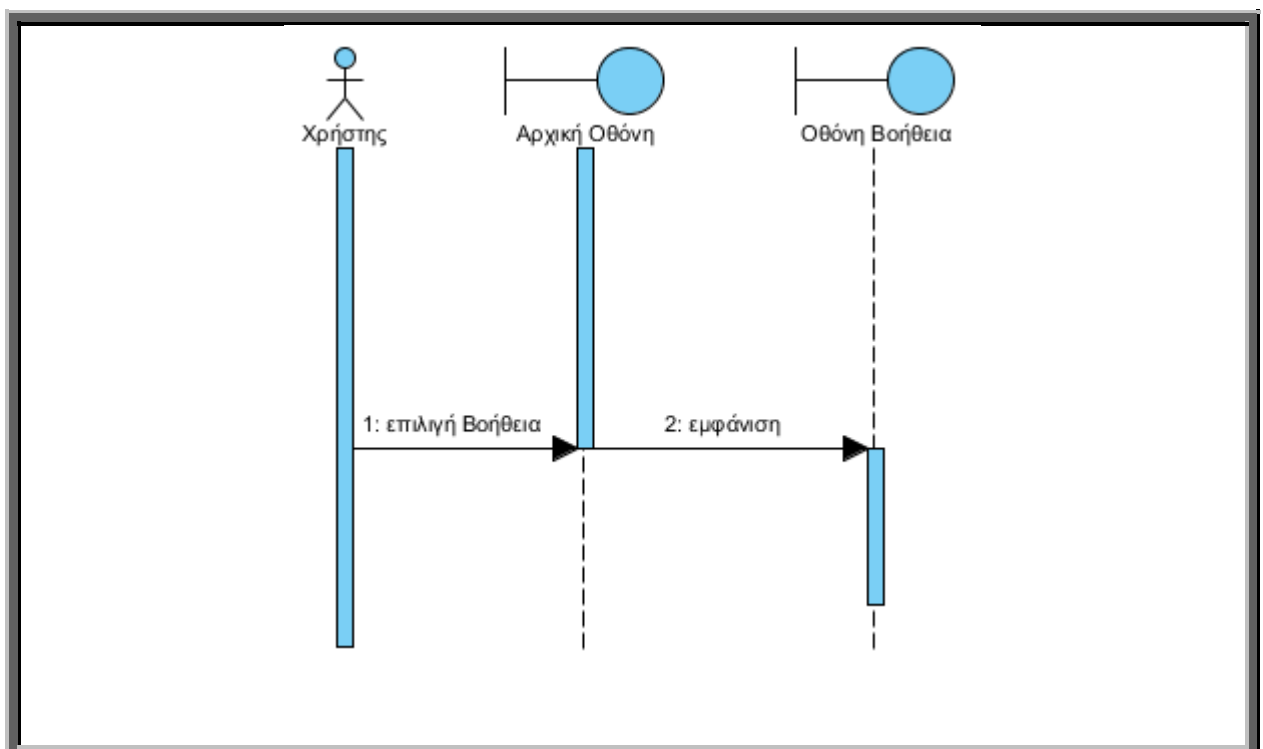
Εικόνα 6.18 Αποθήκευση διαιτολογίου ΔΑ12

7.14 Αποθήκευση διαιτολογίου σε εκτυπώσιμη μορφή ΔΑ13



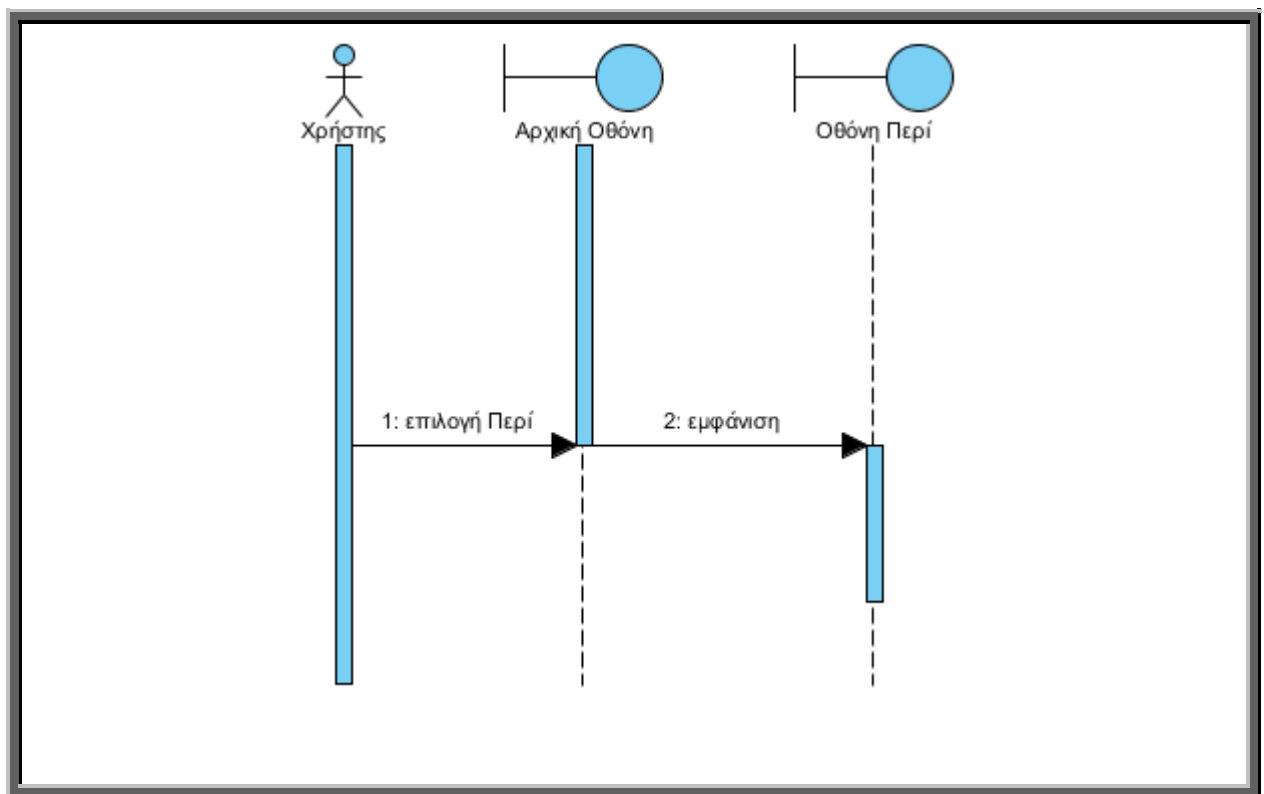
Εικόνα 6.19 Αποθήκευση διαιτολογίου σε εκτυπώσιμη μορφή ΔΑ13

7.15 Λήψη Βοήθειας ΔΑ14



Εικόνα 6.20 Λήψη Βοήθειας ΔΑ14

7.16 Επίδειξη «ταυτότητας» του προγράμματος ΔΑ15



Εικόνα 6.21 Επίδειξη «ταυτότητας» του προγράμματος ΔΑ15

Κεφάλαιο 8

Ανάπτυξη

8.1 Γλώσσα προγραμματισμού

Αφού ολοκληρώθηκε η μοντελοποίηση του προγράμματος, ακολουθεί το στάδιο της υλοποίησης του προγράμματος. Σημαντικό ρόλο στο στάδιο αυτό διαδραματίζει η επιλογή της γλώσσας προγραμματισμού που θα χρησιμοποιηθεί.

Με τον όρο «*γλώσσα προγραμματισμού*» εννοούμε μια τεχνητή γλώσσα, μέσω της οποίας καθίσταται δυνατός ο έλεγχος της λειτουργίας μιας μηχανής, ενός υπολογιστή. Πιο συγκεκριμένα, πρόκειται για μια συστηματική σημειογραφία με την οποία μπορεί να περιγραφεί ένα σύνολο βημάτων, τα οποία εκτελεί ένας υπολογιστής για την επίλυση ενός προβλήματος [17].

Υπάρχουν αντικειμενοστραφείς, συναρτησιακές, διαδικαστικές κ.α. γλώσσες τόσο εμπορικές όσο και ανοιχτού κώδικα. Η γλώσσα προγραμματισμού που έχει επιλεγεί είναι η Pure Basic.

Τα στοιχεία που οδήγησαν στην επιλογή της είναι: Πρόκειται για μια ισχυρή εμπορική γλώσσα στην οποία έχουν γραφτεί από εφαρμογές μέχρι και παιχνίδια. Είναι ταχύτατη και μπορεί να μεταγλωττίσει σε Windows, Linux και MacOS. Είναι πλήρης γλώσσα Basic, στην οποία υπήρχε σχετική εμπειρία.

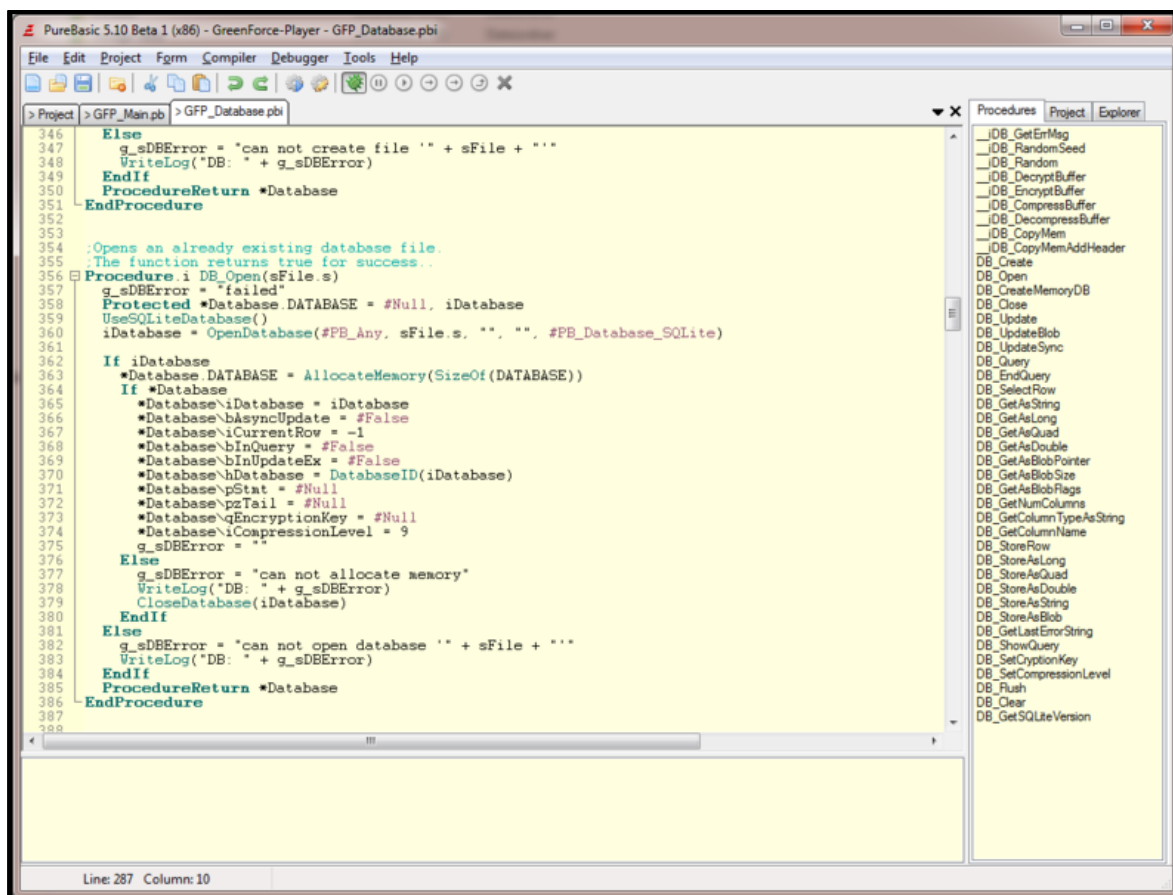
Το πρόγραμμα που προέκυψε από την ανωτέρω διαδικασία παρέχεται στο παράρτημα.

Ada	FORTRAN	OCaml
Algol	Haskell	Pascal
Applescript	Java	Perl
AWK	JavaScript	PHP
BASIC	Lisp	Prolog
C	Logo	Python
C++	Lua	Ruby
C#	Lucid	Scala
Cilk	Mathematica	Scheme
Clojure	Matlab	Simula
COBOL	Miranda	Smalltalk
Datalog	ML	SQL
Erlang	OBJ / Σύστημα Maude	Tcl
Forth	Objective-C	Visual Basic

Πίνακας 7.1: Ενδεικτικός κατάλογος γλωσσών προγραμματισμού [18]

8.2 Υλοποίηση

Για την υλοποίηση του προγράμματος σχεδιάστηκαν 9 οθόνες αλληλεπίδρασης με το χρήστη, στις οποίες προστέθηκαν και γραφικά στοιχεία, ώστε να γίνουν αισθητικά πιο αποδεκτές. Συνολικά γράφτηκαν περίπου 50 υπορουτίνες για να υποστηρίξουν τις διάφορες λειτουργίες και 2.000 γραμμές κώδικα. Χρονικά απαιτήθηκε, συμπεριλαμβανομένου και του χρόνου για τον έλεγχο, διάστημα τριών μηνών.



Εικόνα 7.1 Περιβάλλον εργασίας της Pure Basic

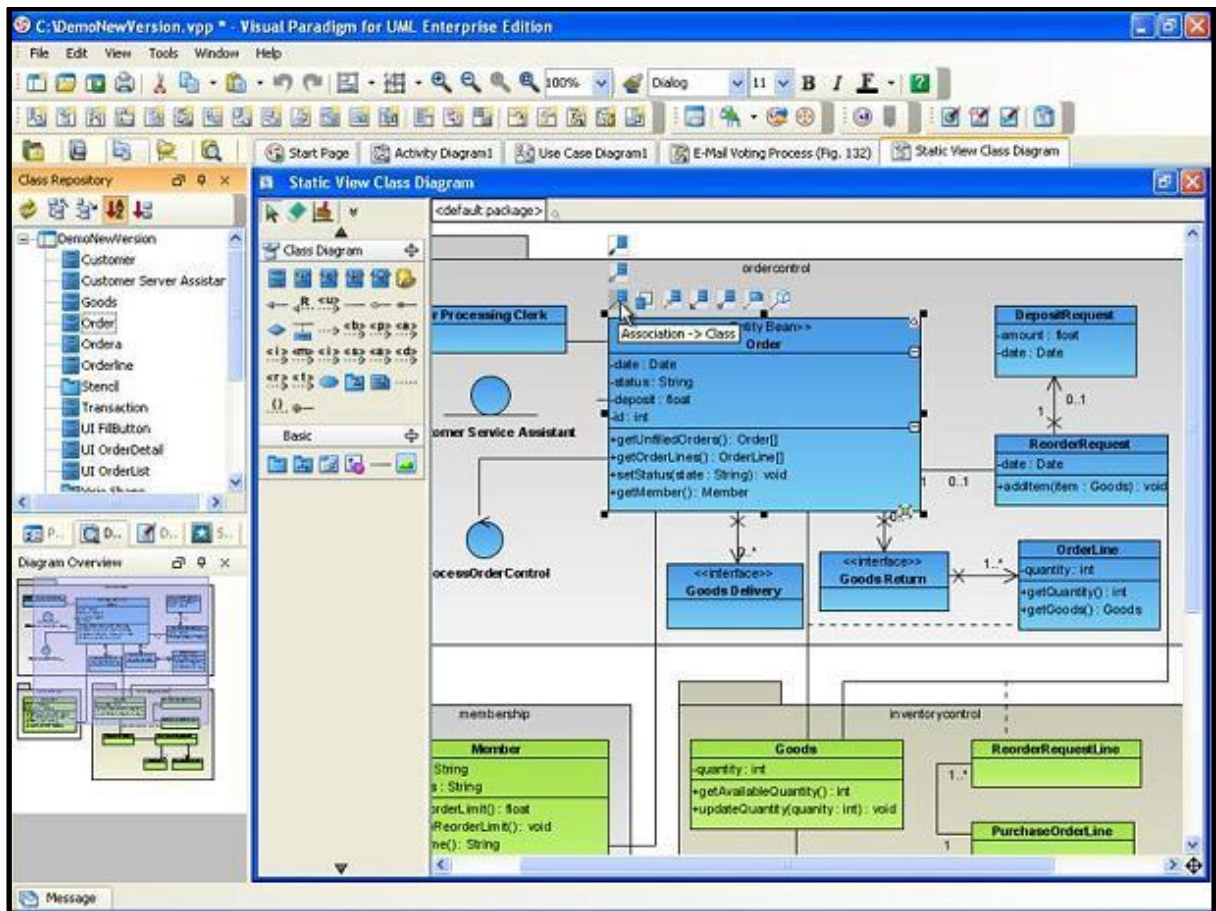
Αξιολογώντας την επιλογή της γλώσσας προγραμματισμού, μπορεί να αναφερθεί ότι κρίνεται ικανοποιητική.

Παρέχει εκτελέσιμο αρχείο, χωρίς άλλα βοηθητικά αρχεία, αρκετά γρήγορο. Επίσης είναι δυνατή η παραγωγή εκδόσεων για Mac OS και Linux.

Σίγουρα, όμως, υπάρχουν πιο εξελιγμένες γλώσσες προγραμματισμού που μπορεί να οδηγούσαν στον επιδιωκόμενο στόχο με καλύτερους όρους.

8.3 Δημιουργία διαγραμμάτων

Για την δημιουργία των διαγραμμάτων του συστήματος χρησιμοποιήθηκε η εφαρμογή Visual Paradigm for UML [19].



Εικόνα 7.2: Περιβάλλον εργασίας του Visual Paradigm

Το Visual Paradigm χρησιμοποιείται για τη γραφική απεικόνιση, προσδιορισμό, κατασκευή και τεκμηρίωση των στοιχείων ενός λογισμικού συστήματος. Μπορεί να χρησιμοποιηθεί σε διάφορες φάσεις ανάπτυξης, από την ανάλυση απαιτήσεων ως τον έλεγχο ενός ολοκληρωμένου συστήματος. Αποτελείται από ένα σύνολο προσυμφωνημένων όρων, συμβόλων και διαγραμμάτων που επιτρέπουν [20]:

- την εμφάνιση των ορίων ενός συστήματος και των βασικών λειτουργιών του, χρησιμοποιώντας «περιπτώσεις χρήσης» (use-cases) και «actors».
- την επεξήγηση της πραγματοποίησης των περιπτώσεων χρήσης με «διαγράμματα αλληλεπίδρασης».
- την αναπαράσταση μιας στατικής δομής ενός συστήματος χρησιμοποιώντας «διαγράμματα κλάσεων».
- τη μοντελοποίηση της συμπεριφοράς των αντικειμένων με «διαγράμματα καταστάσεων».
- την αποκάλυψη της υλοποίησης της αρχιτεκτονικής με «διαγράμματα συστατικών» και «ανάπτυξης».
- την επέκταση της λειτουργικότητας με «στερεότυπα».

8.4 Εφαρμογή στην πράξη

Μετά την ολοκλήρωση του προγράμματος δόθηκε για έλεγχο σε δυο επαγγελματίες υγείας μη διαιτολόγους (που αποτελούσαν και το στόχο, όσον αφορά τους χρήστες του προγράμματος) για να ελεγχθεί και να αξιολογηθεί.

Από τη διαβούλευση που ακολούθησε, προέκυψε ότι το πρόγραμμα επιτελεί τη βασική λειτουργικότητα, δεν έχει περιττά αποπροσανατολιστικά στοιχεία και είναι αρκετά γρήγορο.

Ωστόσο χρειάζεται κάποιο χρόνο εκμάθησης και ο τρόπος που υλοποιούνται οι λειτουργίες δεν είναι άμεσα προβλέψιμος. Οπότε ίσως χρειαστεί επανασχεδιασμός της διεπαφής με το χρήστη.

Κεφάλαιο 9

Επίλογος

9.1 Επισημάνσεις

Στην εποχή μας, τα προβλήματα διαταραχών της θρέψης, παρόλο που η ανάπτυξη της ιατρικής επιστήμης είναι ραγδαία, εξακολουθούν και υπάρχουν. Ιδιαίτερα η παχυσαρκία μαστίζει την ανθρωπότητα σε μεγάλο βαθμό..

Για τον περιορισμό του προβλήματος έχουν προταθεί διάφορα μέτρα. Από αυτά η μείωση της διαιτητικής πρόσληψης φαντάζει από τα πιο ενδεδειγμένα.

Στην πράξη φαίνεται ότι μικρό ποσοστό ασθενών παρακολουθούνται από εξειδικευμένο διατροφολόγο - διαιτολόγο. Μια ενδιαφέρουσα πρόταση θα ήταν με τη βοήθεια του κατάλληλου εργαλείου να παρέχονται μαζικά διαιτητικές παρεμβάσεις από όλους τους επαγγελματίες υγείας.

Για να επιτευχθεί αυτό χρειάζεται ένα πρόγραμμα διαιτολογικό, που να είναι απλό (τα πολύπλοκα επαγγελματικά προγράμματα προκαλούν σύγχυση στον μη ειδικό) και να παρέχει τις κύριες λειτουργίες που χρειάζονται για να διαμορφωθεί μια επιστημονικά ορθή δίαιτα.

9.2 Μελλοντικές επεκτάσεις

Το πρόγραμμα που διαμορφώθηκε με τις παραπάνω αρχές είναι λειτουργικό. Ωστόσο υπάρχουν περιθώρια βελτίωσης.

Μπορεί να βελτιωθεί η διεπαφή με το χρήστη ώστε να είναι πιο προβλέψιμη και άρα λειτουργική. Επίσης μπορεί να προστεθεί η δυνατότητα αποθήκευσης του βάρους του ασθενούς, ώστε να είναι δυνατή η μακροχρόνια παρακολούθηση των μεταβολών του κ.τ.λ.

Σε αυτές τις προτάσεις μπορεί να υπάρξει ο αντίλογος ότι ο σκοπός του προγράμματος είναι να παρέχει τη βασική λειτουργικότητα. Οτιδήποτε προστίθεται, καθιστά το πρόγραμμα πιο σύνθετο και άρα λιγότερο ελκυστικό για τον μη επαγγελματία. Εξάλλου, υπάρχουν πολλά παραδείγματα καλών προσπαθειών, που, λόγω της συνθετότητάς τους, δε συγκίνησαν και παραγκωνίστηκαν.

Την απάντηση για την ιδανική αναλογία λειτουργιών και ευχρηστίας θα τη δώσει η ίδια η πράξη από την εφαρμογή του λογισμικού.

Βιβλιογραφία

- [01] **World health organization**, «*Obesity and overweight*», <http://www.who.int/mediacenter/factsheets/fs311/en/index.html>, retrieved 22.10.2013.
- [02] **Mokdat A.H., Marks J.S., Stroup D.F., Gerberdig J.L.** (2004), «*Actual causes of death in the United States*», 2000, JAMA, 291, (10): 1238-45.
- [03] **World health organization**, «*Controlling the global obesity epidemic*», http://who.int/hpr/NPH/docs/gc_obesity.pdf, 2003, retrieved 22.10.2013.
- [04] **Flegal K. M., Carroll M.D., Kuczmarski R.J. Johnson C.L.**, (1998), «*Overweight and obesity in the United States: Prevalence and trends, 1960-1994*», Int. J Obes Relat Metab Disorder; 22: 39-47.
- [05]. **Karayannis D., Yanakoulia M., Terzidou M., Sidossis L.S., Kokkevi A.**, «*Prevalence of overweighted and obesity in Greek school – aged children and adolescents*», European Journal of clinical Nutrition; 57, 1189-1192.
- [06] **Νικητίδης Νικόλαος**, «*Η Παιδική Παχυσαρκία Στην Πρωτοβάθμια Φροντίδα Υγείας*», Εθνική Σχολή Δημόσιας Υγείας, Διπλωματική Διατριβή, 2011.
- [07] **Dietspeak**, «*Λογισμικό Διατροφής και Διαιτολογίας*», <http://www.dietspeak.gr>, πρόσβαση στις 22.10.2013.
- [08] **Dietician**, <http://www.wrologionprogramma.com/chr/dietician.html>, πρόσβαση στις 22.10.2013.
- [09] **e-dietician**, «*Λογισμικό πρόγραμμα διαχείρισης διαιτολογικού γραφείου*», <http://e-dietician.com/index.php?lang=el> πρόσβαση στις 22.10.2013.

- [10] **Γκουμόπουλος Χρήστος**, «*Αντικειμενοστραφής Ανάλυση και Σχεδίαση*», Ανοικτό Πανεπιστήμιο Κύπρου, Σημειώσεις μαθήματος, 2013.
- [11] **Doug Rosenberg, Matt Stephens**, «*Use Case Driven Object Modeling with UML Theory and Practice*», Apress, 2007.
- [12] **Martin Fowler**, «*UML Distilled, Third Edition, A brief Guide to the Standard Object Modeling language*», Object technology series, 2003.
- [13] **Βασίλειος Βεσκούκης**, «*Εισαγωγή στην αντικειμενοστραφή ανάλυση*», Πανεπιστήμιο Πειραιώς, τμήμα τεχνολογικής εκπαίδευσης, Πειραιάς, 2000.
- [14] **Grady Booch, James Rumbaugh, Ivar Jacobson**, «*The Unified Modeling Language User Guide*», 2nd Edition, 2005.
- [15] **Αλέξανδρος Ν. Χατζηγεωργίου**, «*Ανάπτυξη συστήματος λογισμικού βάσει της μεθοδολογίας ICONIX. Διαχείριση Παραγγελιών*», Ελληνικό Ανοικτό Πανεπιστήμιο, Θεματική Ενότητα ΠΛΗ 24, 2008.
- [16] **Rosenberg Doug, Scott Kendall**, «*Applying use case driven object modeling with UML, An annotated e-Commerce example*», Addison Wesley, 176 pages.
- [17] **Νάστου Π.**, «*Εισαγωγή στις γλώσσες προγραμματισμού με τη γλώσσα C*», Πανεπιστήμιο Αιγαίου, σελ 2, <http://myria.math.aegean.gr/epeaek/pdfs/C.pdf>
- [18] **Ηλεκτρονική Εγκυκλοπαίδεια Βικιπαίδεια**, Λήμμα: «*Γλώσσα προγραμματισμού*», http://el.wikipedia.org/wiki/Γλώσσα_προγραμματισμού, πρόσβαση στις 22.10.2013.
- [19] **Visual Paradigm for UML**, <http://www.visual-paradigm.com>, πρόσβαση στις 22.10.2013.

- [20] **Τζιβλικάκη Άννα**, *«Δημιουργία Ηλεκτρονικής Επιχείρησης Αγοράς και Παράδοσης Προϊόντων»*, Μεταπτυχιακή Διατριβή, Ανοικτό Πανεπιστήμιο Κύπρου, 2013.

Παράρτημα Α

A.1 Ο κώδικας του προγράμματος

Παρακάτω παρέχεται ο κώδικας του προγράμματος

Αποτελείται από 2 τμήματα.

- diaitologio.pbf
- main.pbf

```

Declare ShowIdanikoVaros()
Declare IdanikoVarosCalc( Pfylo, Pypsos )
Declare ShowIsozygio()
Declare ShowDMS()
Declare ShowOlikoLiposKtl()
Declare ShowPerigrifi()
Declare ShowMesiIshia()
Declare ShowProtasi()
Declare ShowBasikosMetavolisimos()
Declare ShowFysikiDrastiriothta()
Declare ShowMetavoliThermidon()
Declare ShowImeresDiaitas()
Declare addFood( Pfood.s )
Declare showDiaitologio()
Declare showDiaitologioList()
Declare afairesiFood()
Declare DetermineLine()
Declare geymaP()
Declare geymaM()
Declare posothtaP()
Declare posothtaM()
Declare thesiP()
Declare thesiM()
Declare CalculateStatisticsList()
Declare CalculateStatisticsList()
Declare StoxoiPanel()
Declare MetriseisPanel()
Declare DiaitologioPanel()
Declare AnazitisiPanel()
Declare StatisticsPanel()
Declare LoadFile(PPath.s,PFile.s)
Declare FilePanel()
Declare HelpPanel()
Declare AboutPanel()
Declare GetMeal()
Declare ShowPerigrifiPerifereia()
Declare saveDiait(PPath.s,Pfile.s)
Declare clearDiat()
Declare.s centerText( Ptext.s, Pwidth.l )
Declare makePrintableText( PPath.s, Pfilename.s )
Declare loadDiat(PPath.s,PFile.s)
Declare ShowPath()
Declare ExplorerMode(Pmode.l)
Declare.s quote(Pstring.s,Pmode.l)

XIncludeFile "C:\main.pbf"

Global selectedFood.s, diaitologioPlus.l = 0, lineCounter.l = 0, targetPosition.l = 0, selectedMeal.l = 1
Global PlacePath.s = "", PlaceFile.s="Filename.txt": Place.s = PlacePath + PlaceFile
Global Dim Food.s ( lineCounter, 6 )
Global Dim diaitologioPlus.s(1000,9)
Global Dim lineValues.s(9)
Global diaitologioPlusLast.l = 1
Global Dim statisticsArray.f(6,10)
Global Dim geumataArray.s(6)
geumataArray(1) = "Πρόγευμα":geumataArray(2) = "Δεκατιανό":geumataArray(3) = "Μεσημεριανό"
geumataArray(4) = "Απογευματινό": geumataArray(5) = "Βραδυνό":geumataArray(6) = "ΣΥΝΟΛΟ "
Global IDStringFile.s = " +++ This is a Data Base associated to 'Diaitologio' software. Do not delete this line! +++ "
"
Global IDStringDiat.s = " +++ This is a Diat associated to 'Diaitologio' software. Do not delete this line! +++ "
Global printWidth = 89, printHight = 65, lastKeyword.s=""

Procedure makePrintableText( PPath.s, Pfilename.s )
  For i=1 To printWidth: a.s = a + "=" :Next i
  ; 50 grammes, 5 geymata
  PcountGeymata = 0
  PcountTrofima = 0
  For i = 1 To 5
    If statisticsArray (i,1) > 10
      PcountGeymata = PcountGeymata + 1
      PcountTrofima = PcountTrofima + statisticsArray (i,1)
    EndIf
  Next i
  MaxGrammes = (50 - PcountTrofima ) / ( 5 - PcountGeymata )

```

```

If CreateFile(0, PPath + PFilename.s ) ; we create a new text file...
WriteStringN( 0, " " )
WriteStringN(0, centerText("Δ Ι Α Ι Τ Ο Λ Ο Γ Ι Ο", printWidth) )
WriteStringN( 0, " " )

For j = 1 To 5
PEpikefalida.s = "=== "+Str(j)+"ο Γεύμα, "+geumataArray(j)
WriteStringN( 0, PEpikefalida + Left(a,printWidth - Len(PEpikefalida) ))

For i=1 To diaitologioPlusLast
If Val(diaitologioPlus(i, 7)) = j
Pposotita.s = Right("00", 2 - Len( diaitologioPlus(i,8) )) + diaitologioPlus(i,8)
PTrofi.s = Mid( diaitologioPlus( i, 2 ), 2, Len( diaitologioPlus( i, 2 ) ) - 2); afairesh eisagwigikwn
PTrofi = Left( PTrofi, printWidth - 3)
WriteStringN( 0, Pposotita + " "+ PTrofi)

EndIf
Next i

If statisticsArray(j,1) < MaxGrammes; an xreiazetai symplhrwma leykwn grammwn
For k=1 To MaxGrammes - statisticsArray (j,1)
WriteStringN( 0, "")
Next k
EndIf

Next j

CloseFile( 0 ) ; close the previously opened file and store the written data this way
MessageRequester("ΓΛΗΡΟΦΟΡΙΑ", "Το εκτυλώσιμο αρχείο έχει πραγματοποιηθεί.", #PB_MessageRequester_Ok)
Else
MessageRequester("ΣΦΑΛΜΑ", "Δεν μπορεί να δημιουργηθεί το αρχείο!")
EndIf

EndProcedure

Procedure.s centerText( Ptext.s, Pwidth.l )

Pspaces.l = (Pwidth - Len(Ptext))/2
For i = 1 To Pspaces: Ptext = " " + Ptext: Next i
ProcedureReturn Ptext

EndProcedure

Procedure clearDiat()

For i = 1 To diaitologioPlusLast
For j = 1 To 9
diaitologioPlus(i,j) = ""
Next j
Next i

diaitologioPlusLast = 0

EndProcedure

Procedure loadDiat( PPath.s,PFile.s )

If ReadFile(0, PPath+PFile) ; if the file could be read, we continue...

If ReadString(0) = IDStringDiat ;Correct file so continue
clearDiat()

While Eof(0) = 0 ; loop as long the 'end of file' isn't reached
Pline.s = ReadString(0)
diaitologioPlusLast = diaitologioPlusLast + 1
For i = 1 To 9
diaitologioPlus( diaitologioPlusLast, i ) = StringField(Pline, i, "@")
Next i
Wend

CloseFile( 0 ) ; close the previously opened file
CalculateStatisticsList()

```

```

        MessageRequester("ΠΛΗΡΟΦΟΡΙΑ", "Η φόρτωση δίατας έχει πραγματοποιηθεί.", #PB_MessageRequester_Ok)
    Else
        MessageRequester("ΣΦΑΛΜΑ", "Το αρχείο που έχει επιλεγεί δε φαίνεται να είναι δίατα του προγράμματος.")
    EndIf
Else
    MessageRequester("ΣΦΑΛΜΑ", "Δεν είναι δυνατό να ανοιχτεί το αρχείο.")
EndIf

```

EndProcedure

Procedure saveDiait(PPath.s,Pfile.s)

```

    If CreateFile(0, PPath+Pfile) ; we create a new text file...
        WriteStringN(0, IDStringDiat)

        For i=1 To diaitologioPlusLast
            Pline.s = ""
            If diaitologioPlus(i,1)<>" " ; an h grammh den einai svhsmenh
                For j = 1 To 9
                    Pline = Pline + diaitologioPlus( i, j) + "@"
                Next j
                WriteStringN(0, Pline) ; we write 10 lines (each with 'end of line' character)
            EndIf
        Next i
        CloseFile( 0 ) ; close the previously opened file and store the written data this way
        MessageRequester("ΠΛΗΡΟΦΟΡΙΑ", "Η αποθήκευση δίατας έχει πραγματοποιηθεί.", #PB_MessageRequester_Ok)
    Else
        MessageRequester("ΣΦΑΛΜΑ", "Δε μπορεί να δημιουργηθεί το αρχείο!")
    EndIf

```

EndProcedure

```

;*****8
;
;                               Program Start
;*****8
LoadFile("", "FoodBaseNutr.txt" )
Openmain(); Start Programm

SetGadgetText(metavoli, "-4000")
SetGadgetText(metavoliThermidon, "-1000")
SetGadgetText(varos2, GetGadgetText(varos))
SetGadgetState(FileOption3, 1)
SetGadgetText(FileExplorerTree, "C:/"):
RemoveGadgetColumn(FileExplorerTree,1):
RemoveGadgetColumn(FileExplorerTree,2):

    Repeat
        Event = WaitWindowEvent()
    Select Event
        Case #PB_Event_Gadget
            Select EventGadget()
                ;metriseis
            Case mainPanel :
                If GetGadgetState(mainPanel)= 1 : MetriseisPanel() : EndIf ;metriseis
                If GetGadgetState(mainPanel)= 2 : StoxoiPanel() : EndIf ;stoxoi
                If GetGadgetState(mainPanel)= 3 : DiaitologioPanel(): EndIf ;diaitologio
                If GetGadgetState(mainPanel)= 4 : AnazitisiPanel() : EndIf ;anazhthsh
                If GetGadgetState(mainPanel)= 5 : StatisticsPanel() : EndIf ;statistika
                If GetGadgetState(mainPanel)= 6 : FilePanel() : EndIf ;File
                If GetGadgetState(mainPanel)= 7 : HelpPanel() : EndIf ;Help
                If GetGadgetState(mainPanel)= 8 : AboutPanel() : EndIf ;About
            EndSelect
        EndSelect
    Until Event = #PB_Event_CloseWindow
End ;END PROGRAM

;*****8
;
;                               Procedures
;*****8
Procedure MetriseisPanel()

```

```

        ShowIdanikoVaros()      :      ShowIsozygio():      ShowDMS():ShowOlikoLiposKtl():ShowMesiIshia():
ShowBasikosMetabolismos(): ShowFysikiDrastiriothta()

        Repeat

            Event = WaitWindowEvent()

                Select Event
                    Case #PB_Event_CloseWindow: End

                Case #PB_Event_Gadget
                    Select EventGadget()
                        Case ypsos : ShowIdanikoVaros() : ShowDMS(): ShowIsozygio(): ShowOlikoLiposKtl():
ShowBasikosMetabolismos()
                            : ShowFysikiDrastiriothta():SetGadgetText( EpithimitoVaros, GetGadgetText( proteinomenoVaros ) )
                        Case metrFylo: ShowIsozygio(): ShowOlikoLiposKtl():
                            ShowBasikosMetabolismos(): ShowFysikiDrastiriothta():ShowPerigrافيPerifereia()
                        Case varos :ShowIdanikoVaros() : ShowIsozygio(): ShowDMS(): ShowOlikoLiposKtl():
ShowBasikosMetabolismos(): ShowFysikiDrastiriothta():
                            SetGadgetText( EpithimitoVaros, GetGadgetText( proteinomenoVaros ) ); ShowImeresDiaitas()
                        Case per_mesis: ShowMesiIshia(): ShowPerigrافيPerifereia()
                        Case per_ishion: ShowMesiIshia():ShowPerigrافيPerifereia()
                        Case ilikia: ShowBasikosMetabolismos() : ShowFysikiDrastiriothta()
                        Case Drasririotita: ShowFysikiDrastiriothta()
                    EndSelect

                EndSelect
            Until GetGadgetState(mainPanel) <> 1
        EndProcedure

        Procedure StoxoiPanel()
            If GetGadgetText( varos ) <> ""
                SetGadgetText( varos2, GetGadgetText( varos ) )
            Else
                SetGadgetText( varos2, "-" )
            EndIf

            If GetGadgetText( FysikiDrastitiothta ) <> "-" Or GetGadgetText( FysikiDrastitiothta ) <> ""
                SetGadgetText( FysikiDrastitiothta2, GetGadgetText( FysikiDrastitiothta ) )
            Else
                SetGadgetText( FysikiDrastitiothta, "-" )
            EndIf

            If GetGadgetText( ypsos ) <> ""
                SetGadgetText( proteinomenoVaros, Str( 25 * Pow( Val( GetGadgetText( ypsos ) ) / 100, 2 ) ) )
            Else
                SetGadgetText( proteinomenoVaros, "-" )
            EndIf
            ShowProtasi()
            ShowImeresDiaitas()

        ;Event part
        Repeat
            Event = WaitWindowEvent()
            Select Event
                Case #PB_Event_CloseWindow: End
                Case #PB_Event_Gadget

                    Select EventGadget()
                        Case EpithimitoVaros : ShowImeresDiaitas()
                        Case metavoli : ShowMetavoliThermidon(): ShowImeresDiaitas():ShowProtasi()
                    EndSelect

                EndSelect
            Until GetGadgetState(mainPanel) <> 2

        EndProcedure

        Procedure DiaitologioPanel()

        showDiaitologioList()

        Repeat
            Event = WaitWindowEvent()
            Select Event
                Case #PB_Event_CloseWindow: End

```

```

Case #PB_Event_Gadget
Select EventGadget()

    Case diatDiatologio
    Case diatProsthiki: SetGadgetState(mainPanel,4): AnazitisiPanel()
    Case diatKatharismos: clearDiat(): showDiatologioList()
    Case diatAfairesi: afairesiFood(): showDiatologioList()
    Case diaitGeymaP: geymaP(): showDiatologioList()
    Case diaitGeymaM: geymaM(): showDiatologioList()
    Case diaitPosotitaP:posothtaP(): showDiatologioList()
    Case diaitPosotitaM:posothtaM(): showDiatologioList()
    Case diaitThesiP: thesiP(): showDiatologioList()
    Case diaitThesiM: thesiM(): showDiatologioList()
    Case diatEktyposi: makePrintableText( "", "ΕΚΤΥΠΩΣΗ.txt" ):RunProgram("notepad", "/p ΕΚΤΥΠΩΣΗ.txt", "")
EndSelect

EndSelect
Until GetGadgetState(mainPanel) <> 3

EndProcedure

Procedure AnazitisiPanel()

Repeat
Event = WaitWindowEvent()

Select Event
Case #PB_Event_CloseWindow: End

Case #PB_Event_Gadget
Select EventGadget()

    Case anazKleidi

        SearchWord.s = GetGadgetText(anazKleidi)
        If SearchWord <> lastKeyword
            lastKeyword = SearchWord
            ClearGadgetItems( anazApotelesmata )
            SetGadgetText( anazEyrethenta, "" )
            SetGadgetText(anazInfoText, "")

        CaseSansitive.b = Bool_Not( GetGadgetState( anazCaseSensitive ) )
        For i = 1 To lineCounter
            ArrayString.s = Food(i,2)
            LookString.s = SearchWord
            If CaseSansitive = 1
                ArrayString= LCase(ArrayString)
                LookString = LCase(LookString)
            EndIf
            If CountString( ArrayString, LookString )
                AddGadgetItem( anazApotelesmata, -1 , quote( Food(i,2), 0 ) )
            EndIf
        Next
        PcountEyrethenta.l = CountGadgetItems(anazApotelesmata)
        If PcountEyrethenta >0
            SetGadgetText( anazEyrethenta, "Σύνολο:" + Str( PcountEyrethenta ) )
        EndIf
        EndIf

        Case anazApotelesmata
            If GetGadgetState(anazApotelesmata) <> -1
                Pfood.s = quote(GetGadgetText(anazApotelesmata),1)

                For i = 1 To lineCounter
                    If Food (i,2) = Pfood
                        Pinfo.s = "Ενέργεια:" + Food( i, 3) + "Kcal Πρωτεΐνη:" + Food( i, 4)+ "g Υδατάνθρακες:" + Food( i,
5)+ "g Λίπη:" + Food( i, 6)+ "g"
                    EndIf
                Next i

                SetGadgetText(anazInfoText,Pinfo)
            Else
                SetGadgetText(anazInfoText, "")
            EndIf
        EndIf
    EndSelect
EndRepeat

```



```

Case anazEpilogi:
  PSelectedFood.s = GetGadgetText(anazApotelesmata) :
  If PSelectedFood <> ""
    GetMeal():
    addFood( quote( PSelectedFood,1 ) )
    CalculateStatisticsList()
    showDiaitologiolist()
    SetGadgetState(mainPanel,3):DiaitologioPanel()
  EndIf

  Case anazCaseSensitive :
  EndSelect

EndSelect
Until GetGadgetState(mainPanel) <> 4

EndProcedure

Procedure ShowPath()

  PSelectedPath.s = GetGadgetText(FileExplorerTree)+GetGadgetItemText(FileExplorerTree,
  GetGadgetState(FileExplorerTree),0)
  PLastPath.s = StringField( PSelectedPath, (CountString( PSelectedPath, "\" ) + 1 ), "\" )
  PSelectedPath = Left( PSelectedPath, Len( PSelectedPath ) - Len( PLastPath ) )
  If GetGadgetState(FileOption3) Or GetGadgetState(FileOption4)
    PSelectedFilename.s = GetGadgetText(FileString)
  Else
    If GetGadgetItemState(FileExplorerTree,GetGadgetState(FileExplorerTree)) = #PB_Explorer_Selected +
    #PB_Explorer_File
      PSelectedFilename = GetGadgetItemText(FileExplorerTree,GetGadgetState(FileExplorerTree))
    Else: PSelectedFilename = ""
    If PSelectedFilename="Name": PSelectedFilename = "" :EndIf
  EndIf
  EndIf
  If PSelectedPath = "": PSelectedFilename = "" : EndIf
  SetGadgetText(FileText, PSelectedPath + PSelectedFilename)

EndProcedure

Procedure StatisticsPanel()

  CalculateStatisticsList()
  ClearGadgetItems(statResults1): ClearGadgetItems(statResults2)

  For i = 1 To 6
    Pline.s = geumataArray(i)+ Chr(10) + StrD( statisticsArray(i,1), 0 ) + Chr(10) + StrD(
    statisticsArray(i,2), 2 ) + Chr(10) + StrD( statisticsArray(i,3), 0 )
    AddGadgetItem(statResults1,-1,Pline)
  Next i

  For i = 1 To 6
    Pline.s = geumataArray(i) + Chr(10) + StrD(statisticsArray(i,4),2) + Chr(10) +
    StrD(statisticsArray(i,6),2) + Chr(10) + StrD( statisticsArray(i,8),2) + Chr(10)
    Pline = Pline + StrD(statisticsArray(i,5),0) + Chr(10) + StrD(statisticsArray(i,7),0) + Chr(10)
    + StrD(statisticsArray(i,9),0)
    AddGadgetItem(statResults2,-1,Pline)
  Next i

  Pzitoymeno.s= GetGadgetText(zitoymeno)
  If Val(Pzitoymeno) <> 0 And Pzitoymeno <> ""
    PposostoZitoymeno.f = statisticsArray(6,2)*100/Val(Pzitoymeno)
    SetGadgetText(statText, "Κάλυψη ζητούμενων ενεργειακών αναγκών: " + StrF(PposostoZitoymeno,2) +
    "%")

    HideGadget(statProgressBar,0)
    SetGadgetState (statProgressBar, PposostoZitoymeno)
  Else
    SetGadgetText(statText,"")
    HideGadget(statProgressBar,1)
  EndIf

Repeat
  Event = WaitWindowEvent()
  Select Event
  Case #PB_Event_CloseWindow: End

```

```

Case #PB_Event_Gadget
  Select EventGadget()
  EndSelect

EndSelect
Until GetGadgetState(mainPanel) <> 5
  EndProcedure

Procedure FilePanel()

  Repeat
    Event = WaitWindowEvent()

    Select Event
      Case #PB_Event_CloseWindow: End
      Case #PB_Event_Gadget

        Select EventGadget()

          Case FileOption1:HideGadget(FileString,1): ExplorerMode(0): ShowPath()
          Case FileOption2:HideGadget(FileString,1): ExplorerMode(0): ShowPath()
          Case FileOption3:HideGadget(FileString,0): ExplorerMode(1): ShowPath()
          Case FileOption4:HideGadget(FileString,0): ExplorerMode(1): ShowPath()
          Case FileString: ShowPath()
          Case FileExplorerTree: ShowPath()

          Case FileButton
            If GetGadgetState( FileOption1) = 1 : LoadFile(GetGadgetText(FileExplorerTree),
GetGadgetItemText(FileExplorerTree, GetGadgetState(FileExplorerTree) ))
              ElseIf GetGadgetState( FileOption2) = 1 : loadDiat(GetGadgetText(FileExplorerTree),
GetGadgetItemText(FileExplorerTree, GetGadgetState(FileExplorerTree) ))
                ElseIf GetGadgetState( FileOption3) = 1 : saveDiat(GetGadgetText(FileExplorerTree),
GetGadgetText(FileString))
                  ElseIf GetGadgetState( FileOption4) = 1 : makePrintableText(GetGadgetText(FileExplorerTree),
GetGadgetText(FileString))
                    EndIf
                EndSelect
            EndSelect
          EndSelect
        Until GetGadgetState(mainPanel) <> 6

      EndProcedure

      Procedure.s quote(Pstring.s,Pmode.1)

      If Pmode = 0
        PResult.s= Mid( Pstring, 2, Len( Pstring ) - 2)
        ProcedureReturn PResult
      ElseIf Pmode = 1
        PResult = Chr(34) + Pstring + Chr(34)
        ProcedureReturn PResult
      Else
        ProcedureReturn Pstring
      EndIf

      EndProcedure

      Procedure ExplorerMode(Pmode.1)

        If Pmode = 0: SetGadgetText( FileExplorerTree, ".*.*" )
        ElseIf Pmode = 1: SetGadgetText( FileExplorerTree, "@.@@@" )
        EndIf

        EndProcedure
      Procedure HelpPanel()
        ClearGadgetItems(HelpEditor)
        AddGadgetItem(HelpEditor,-1, "
B O H Θ E I A")
        AddGadgetItem(HelpEditor,-1, "")
        AddGadgetItem(HelpEditor,-1, "
αντικείμενο παρέχονται επεξηγήσεις")
        AddGadgetItem(HelpEditor,-1, "")
        AddGadgetItem(HelpEditor,-1, "
Υλοποίησης Διατολογίου ")
        AddGadgetItem(HelpEditor,-1, "
τιμών 3. Καθορισμός στόχων ")
        "
        Με την τοποθέτηση του δείκτη σε
        Διαδικασία
        1. Εισαγωγή στοιχείων 2. Υπολογισμός

```

```

AddGadgetItem(HelpEditor,-1, "
5. Έλεγχος - διορθώσεις")
AddGadgetItem(HelpEditor,-1, "")
AddGadgetItem(HelpEditor,-1, "
επιτυχημένου διαιτολογίου ")
AddGadgetItem(HelpEditor,-1, "
ΠΡΟ:ΥΔΑΤ:ΛΙΠΗ=15-20:50-60:15-25")
AddGadgetItem(HelpEditor,-1, "")
AddGadgetItem(HelpEditor,-1, "
πρόσληψη ανά γεύμα")
AddGadgetItem(HelpEditor,-1, "
Απογευματινό 20%, Βραδυνό 25%")

Repeat
Event = WaitWindowEvent()
Select Event
Case #PB_Event_CloseWindow: End

EndSelect
Until GetGadgetState(mainPanel) <> 7

EndProcedure

Procedure AboutPanel()
ClearGadgetItems(PeriEditor)
AddGadgetItem(PeriEditor,-1, "")
AddGadgetItem(PeriEditor,-1, "")
AddGadgetItem(PeriEditor,-1, "
μεταπτυχιακού προγράμματος σπουδών")
AddGadgetItem(PeriEditor,-1, "")
AddGadgetItem(PeriEditor,-1, "
Ανοικτού Πανεπιστημίου Κύπρου.")
AddGadgetItem(PeriEditor,-1, "")
AddGadgetItem(PeriEditor,-1, "")
AddGadgetItem(PeriEditor,-1, "
Ιούνιο του 2014")

Repeat
Event = WaitWindowEvent()
Select Event
Case #PB_Event_CloseWindow: End
EndSelect
Until GetGadgetState(mainPanel) <> 8

EndProcedure

Procedure IdanikoVarosCalc( Pfylo, Pypsos )

;Υπλογισμος με methodo Hamwii
If Pfylo = 0 ;To fylo einai antras
If Pypsos >152
PResult = 48 + 1.06 * (Pypsos - 152)
ElseIf Pypsos >0 And Pypsos <=152
PResult = 48
Else
PResult = 77777777
EndIf
ElseIf Pfylo = 1 ;To fylo einai gynaika
If Pypsos > 152
PResult = 45.2 + 0.89 * (Pypsos - 152)
ElseIf Pypsos >0 And Pypsos <=152
PResult = 45.2
Else
PResult = 88888888
EndIf
Else ;To fylo den einai oyte antras oyte gynaika
PResult = 999999
EndIf

ProcedureReturn PResult
EndProcedure

Procedure ShowIdanikoVaros()

Pfylo.1 = GetGadgetState( metrFylo )
Pypsos.s = GetGadgetText( ypsos )

```

4. Δημιουργία διαιτολογίου

Χαρακτηριστικά

5 γεύματα, ποικιλία τροφών, αναλογία %

Ενεργειακή

Πρωινό 20%, Δεκατιανό 10%, Μεσημεριανό 25%,

Το πρόγραμμα αυτό υλοποιήθηκε στα πλαίσια του

Υπολογιστικά και Επικοινωνιακά Συστήματα του

Ολοκληρώθηκε τον

```

    If Pypsos <> "" And Pfylo <> -1
        SetGadgetText( idaniko, Str( IdanikoVarosCalc( Pfylo, Val(Pypsos) ) ) )
    Else
        SetGadgetText( idaniko, "-" )
    EndIf

EndProcedure

Procedure ShowIsozygio()
    Pvaros.s = GetGadgetText(varos)
    PIdaniko.s = GetGadgetText(idaniko )

    If Pvaros <> "" And PIdaniko <> "-"
        PIsozygio.l = Val(Pvaros) - Val(PIdaniko)

        If PIsozygio > 0 : PProsimo.s = "+": EndIf
        SetGadgetText( isozygio, PProsimo + Str(PIsozygio))
    Else
        SetGadgetText( isozygio, "-" )
    EndIf
EndProcedure

Procedure ShowDMS()

    Pypsos.s = GetGadgetText( ypsos )
    Pvaros.s = GetGadgetText( varos )
    If Pypsos <> "" And Pvaros <> ""
        PDMS = Val(Pvaros) / Pow(( Val(Pypsos) / 100 ) ,2)
        SetGadgetText( DMS, Str( PDMS ) )
        ShowPerigrafia()
    Else
        SetGadgetText( DMS, "-" )
        SetGadgetText(perigrafia, "-")
    EndIf

EndProcedure

Procedure ShowOlikoLiposKtl()
    Pfylo = GetGadgetState( metrFylo )
    Pypsos = Val( GetGadgetText( ypsos ) )
    Pvaros = Val( GetGadgetText( varos ) )
    ;Methodos Kvist
    If (Pfylo = 0 Or Pfylo = 1) And Pypsos > 43 And Pvaros >0 And Pvaros * 100 / Pypsos > 23.79 + 7.09 * Bool(
Pfylo = 0 )
        If Pfylo = 0
            POLikoLipos = ( 1.36 * Pvaros / ( Pypsos / 100 ) - 42 ) * 0.923
        ElseIf Pfylo = 1
            POLikolipos = ( 1.61 * Pvaros / ( Pypsos / 100 ) - 38.3 ) * 0.923
        EndIf
        SetGadgetText(Oliko_lipos, Str(POLikoLipos))
        SetGadgetText(Alipi_maza, Str(Pvaros-POLikoLipos))
        SetGadgetText(OlikoPos, Str( POLikoLipos * 100 / Pvaros ))
        SetGadgetText(alipiPos, Str( 100 - (POLikoLipos * 100 / Pvaros )))

    Else
        SetGadgetText(Oliko_lipos, "-")
        SetGadgetText(Alipi_maza, "-")
        SetGadgetText(OlikoPos, "-")
        SetGadgetText(alipiPos, "-")
    EndIf
EndProcedure

Procedure ShowPerigrafiaPerifereia()
    PperMsisis.s = GetGadgetText(per_mesis)
    PPerIshion.s = GetGadgetText(per_ishion)
    Pfylo.l = GetGadgetState(metrFylo)
    PMessage.s="Κοιλιακή παχυσαρκία"
    SetGadgetText(perigrafia2,"")
    SetGadgetText(perigrafia3,"")
    SetGadgetColor(perigrafia2, #PB_Gadget_FrontColor, $3300ff)
    SetGadgetColor(perigrafia3, #PB_Gadget_FrontColor, $3300ff)

    If PperMsisis <> "" And Pfylo <> -1
        If Pfylo = 0 And Val(PperMsisis) >= 102
            SetGadgetText( perigrafia2, "M>" + PMessage )

```

```

EndIf
If Pfylo = 1 And Val(PperMsesis) >= 88
  SetGadgetText( perigrافي2, "M>" + PMessage )
EndIf
EndIf

If PperMsesis <> "" And PPerIshion <> "" And Pfylo <> -1 And Val(PPerIshion) > 0
  PMesiIshia.f = Val( PperMsesis ) / Val( PPerIshion )
  If Pfylo = 0 And PMesiIshia >= 1
    SetGadgetText( perigrافي3, "M/I>" + PMessage )
  EndIf
  If Pfylo = 1 And PMesiIshia >= 0.85
    SetGadgetText( perigrافي3, "M/I>" + PMessage )
  EndIf
EndIf

EndProcedure

Procedure ShowPerigrافي()
  PDMS = Val( GetGadgetText(DMS))
  PPer.s = "-"
  PColor=$000000
  If
    PDMS <18.5 : PColor=$FF6600 :PPer = "Λυτοβαρής"
  ElseIf PDMS >=18.5 And PDMS <25.0 : PColor=$669900 :PPer = "Φυσιολογικός"
  ElseIf PDMS >=25.0 And PDMS <30.0 : PColor=$66CCFF:PPer = "Υπέρβαρος"
  ElseIf PDMS >=30.0 And PDMS <35.0 : PColor=$0000ff :PPer = "Ελαφριά παχύσαρκος"
  ElseIf PDMS >=35.0 And PDMS <40.0 : PColor=$0033ff:PPer = "Μέτρια παχύσαρκος"
  ElseIf PDMS >=40.0 : PColor=$3300ff :PPer = "Βαριά παχύσαρκος"
  EndIf
  SetGadgetColor(perigrافي, #PB_Gadget_FrontColor, PColor)
  SetGadgetText(perigrافي, PPer)
EndProcedure

Procedure ShowMesiIshia()
  PMesi.s = GetGadgetText(per_mesis)
  PIshia.s = GetGadgetText(per_ishion)

  If PMesi <> "" And PIshia <> ""
    SetGadgetText(mesiIshia, StrF((Val(PMesi)/Val(PIshia)),2))
  Else
    SetGadgetText(mesiIshia, "-")
  EndIf
EndProcedure

Procedure ShowProtasi()

  PmetavoliThermidon = Val( GetGadgetText( metavoliThermidon ) )
  PFysikiDrastitiothta = Val( GetGadgetText( FysikiDrastitiothta2 ) )
  If PmetavoliThermidon <> 0 And PFysikiDrastitiothta <> 0
    SetGadgetText(protasi,Str( PFysikiDrastitiothta + PmetavoliThermidon ) )
  Else
    SetGadgetText(protasi, "-" )
  EndIf

EndProcedure

Procedure ShowBasikosMetabolismos()

;Harris and Benedict
  Pfylo = GetGadgetState( metrFylo )
  Pypsos = Val( GetGadgetText( ypsos ) )
  Pvaros = Val( GetGadgetText( varos ) )
  Pilikia = Val( GetGadgetText( ilikia ) )

  If (Pfylo = 0 Or Pfylo = 1) And Pypsos > 0 And Pvaros >0 And Pilikia > 0
    If Pfylo = 0
      PBM = 66 + 13.75 * Pvaros + 5 * Pypsos - 6.76 * Pilikia
    ElseIf Pfylo = 1
      PBM = 655 + 9.56 * Pvaros + 1.85 * Pypsos - 4.68 * Pilikia
    EndIf
    SetGadgetText( BasikosMetabolismos, Str( PBM ) )

  Else
    SetGadgetText( BasikosMetabolismos, "-" )
  EndIf

EndProcedure

```

```

Procedure ShowFysikiDrastiriothta()
  PDrastiriotita = GetGadgetState( Drasrriotita )
  PBM.s = GetGadgetText( BasikosMetabolismos )
  If PBM <> "-" And PDrastiriotita <> -1
    If PDrastiriotita = 0 : PSyntelesthsFD = 115
      ElseIf PDrastiriotita = 1 : PSyntelesthsFD = 135
      ElseIf PDrastiriotita = 2 : PSyntelesthsFD = 145
      ElseIf PDrastiriotita = 3 : PSyntelesthsFD = 170
      ElseIf PDrastiriotita = 4 : PSyntelesthsFD = 200
      Else : PSyntelesthsFD = 000
    EndIf

    SetGadgetText( FysikiDrastitiothta, Str( Val( PBM ) * PSyntelesthsFD/100 ) )

  Else
    SetGadgetText( FysikiDrastitiothta, "-" )
  EndIf

EndProcedure

Procedure ShowMetavoliThermidon()
  Pmetavoli = Val( GetGadgetText( metavoli ) )
  If Pmetavoli <> 0
    SetGadgetText( metavoliThermidon, Str( Pmetavoli / 4 ) )
  Else
    SetGadgetText( metavoliThermidon, "-" )
  EndIf
EndProcedure

Procedure ShowImeresDiaitas()
  Pvaros.s = GetGadgetText( varos2 )
  PEpithimitoVaros.s = GetGadgetText( EpithimitoVaros )
  Pmetavoli.s = GetGadgetText( metavoli )
  If Pvaros <> "" And PEpithimitoVaros <> "" And Pmetavoli <> "" And Pvaros <> "-" And
  PEpithimitoVaros <> "-" And Pmetavoli <> "-"
    PhmeresDiaitas.l = ( Val(PEpithimitoVaros) - Val(Pvaros) ) * 30000 / Val(Pmetavoli)
    If PhmeresDiaitas > 0
      SetGadgetText( imeresDiaitas, Str( PhmeresDiaitas ) )
    Else
      SetGadgetText( imeresDiaitas, "-" )
    EndIf
  Else
    SetGadgetText( imeresDiaitas, "-" )
  EndIf

EndProcedure

Procedure addFood( Pfood.s )
; 1Code 2Name 3Energy 4Pro 5Cho 6Fat 7Meal 9Qtt 9Plc

For i = 1 To lineCounter
  If Food (i,2) = Pfood
    diaitologioPlus( diaitologioPlusLast, 1 ) = Food( i, 1 )
    diaitologioPlus( diaitologioPlusLast, 2 ) = Food( i, 2 )
    diaitologioPlus( diaitologioPlusLast, 3 ) = Food( i, 3 )
    diaitologioPlus( diaitologioPlusLast, 4 ) = Food( i, 4 )
    diaitologioPlus( diaitologioPlusLast, 5 ) = Food( i, 5 )
    diaitologioPlus( diaitologioPlusLast, 6 ) = Food( i, 6 )
    diaitologioPlus( diaitologioPlusLast, 7 ) = Str( selectedMeal )
    diaitologioPlus( diaitologioPlusLast, 8 ) = "1"
    diaitologioPlus( diaitologioPlusLast, 9 ) = Str( diaitologioPlusLast )
    diaitologioPlusLast = diaitologioPlusLast + 1
  EndIf
Next i

EndProcedure

Procedure afairesiFood()
;Ypologizw to Pfood kai Pmeal
  Pline.s = GetGadgetText(diatDiatologio)
  If Pline <> ""
    Pline1.s = Right( Pline, Len( Pline ) - FindString( Pline, Chr( 34 ) ) )
    Pline2.s = Chr(34)+ Left( Pline1, FindString(Pline1,Chr(34)) )
    Pline3.s = Right( Pline1, Len(Pline1) - FindString(Pline1,Chr(34)) )

```

```

Pfood.s = Pline2
For i = 1 To 6
    Pmeal.s = StringField(Pline3, i, " ")
Next

For i = 1 To diaitologioPlusLast
    If diaitologioPlus(i,2) = Pfood And diaitologioPlus(i,7) = Pmeal
        For j = 1 To 9
            diaitologioPlus(i,j) = ""
        Next j
    EndIf
Next i
EndIf

EndProcedure

Procedure showDiatologiList()

ClearGadgetItems(diatDiatologio)
For k= 1 To 5
    geymastring.s=Left( Str(k)+"ο Γεύμα - " + geumataArray (k) +
"=====",85)
    AddGadgetItem( diatDiatologio, -1, geymastring )
For i = 1 To diaitologioPlusLast
    If diaitologioPlus(i,1) <> "" And diaitologioPlus(i,7) = Str(k) ; An den exei diagarfei kai analoga omada
        line.s = ""
        For j = 1 To 9
            line = line + quote(diaitologioPlus( i, j ),Bool(j=2)-1) + " "
        Next j
        AddGadgetItem( diatDiatologio, -1, line )
    EndIf
Next i
Next k

EndProcedure

Procedure DetermineLine()
Pline.s = GetGadgetText(diatDiatologio)
If Pline <> ""

Pline1.s = Right( Pline, Len( Pline ) - FindString( Pline, Chr( 34 ) ) )
Pline2.s = Chr(34)+ Left( Pline1, FindString(Pline1,Chr(34)) )
Pline3.s = Right( Pline1, Len(Pline1) - FindString(Pline1,Chr(34)) )

For i = 1 To 8
    lineValues(i+1) = StringField(Pline3, i, " ")
Next

lineValues(1) = StringField(Pline, 1, " ")
lineValues(2) = Pline2

```

```

Else
For j = 1 To 9: lineValues(j) = "": Next j
EndIf

For k = 1 To 9:
Next k

EndProcedure

Procedure FindPlace()
targetPosition = 0
For i = 1 To diaitologioPlusLast
If lineValues(2) = diaitologioPlus( i, 2 ) And lineValues(7) = diaitologioPlus( i, 7 )
targetPosition = i
EndIf
Next i
EndProcedure

Procedure geymaP()
Determineline()
FindPlace()
If targetPosition <> 0
Pgeyma = Val( diaitologioPlus( targetPosition, 7 ))
If Pgeyma < 5
diaitologioPlus(targetPosition, 7 ) = Str( Pgeyma + 1 )
EndIf
EndIf

EndProcedure

Procedure geymaM()
Determineline() : FindPlace()
If targetPosition <> 0
Pgeyma = Val( diaitologioPlus( targetPosition, 7 ))
If Pgeyma > 1 :
diaitologioPlus(targetPosition, 7 ) = Str( Pgeyma - 1 )
EndIf
EndIf

EndProcedure

Procedure posothtaP()
Determineline() : FindPlace()
If targetPosition <> 0
Pposothta.1 = Val( diaitologioPlus( targetPosition, 8 ))
diaitologioPlus(targetPosition, 8 ) = Str( Pposothta + 1 )
EndIf

EndProcedure

Procedure posothtaM()
Determineline() : FindPlace()
If targetPosition <> 0

```



```

Pposothta.l = Val( diaitologioPlus( targetPosition, 8 ))
If Pposothta > =1
    diaitologioPlus(targetPosition, 8 ) = Str( Pposothta - 1 )
EndIf
EndIf
EndProcedure

```

```

Procedure thesiP()
    Determineline() : FindPlace()
    Pthesi = Val( lineValues( 9 ) )
    PosToChange.l = 0
    PValue.l = diaitologioPlusLast
    For i = 1 To diaitologioPlusLast
        If lineValues(7) = diaitologioPlus(i,7) And diaitologioPlus(i,9) > lineValues(9)
            If Val(diaitologioPlus( i, 9 )) < PValue
                PValue = Val( diaitologioPlus( i, 9 ) )
                PosToChange = i
            EndIf:
        EndIf
    Next i
    If PosToChange <> 0
        Dim Ptemporary.s(9)
        For j = 1 To 8
            Ptemporary(j) = diaitologioPlus( PosToChange, j )
            diaitologioPlus(PosToChange, j) = lineValues(j)
            diaitologioPlus(targetPosition, j) = Ptemporary(j)
        Next j
    EndIf
EndProcedure

```

```

Procedure thesiM()
    Determineline() : FindPlace()
    Pthesi = Val( lineValues( 9 ) )
    PosToChange.l = 0
    PValue.l = 0
    For i = 1 To diaitologioPlusLast
        If lineValues(7) = diaitologioPlus(i,7) And diaitologioPlus(i,9) < lineValues(9)
            If Val(diaitologioPlus( i, 9 )) > PValue
                PValue = Val( diaitologioPlus( i, 9 ) )
                PosToChange = i
            EndIf:
        EndIf
    Next i
    If PosToChange <> 0
        Dim Ptemporary.s(9)
        For j = 1 To 8
            Ptemporary(j) = diaitologioPlus( PosToChange, j )
            diaitologioPlus(PosToChange, j) = lineValues(j)
            diaitologioPlus(targetPosition, j) = Ptemporary(j)
        Next j
    EndIf
EndProcedure

```

```

Procedure CalculateStatisticsList()

```

```

    For i = 1 To 6
        For j = 1 To 10
            statisticsArray(i,j) = 0
        Next
    Next

```

```

    ;grammes 1-5

```

```

    For i = 1 To 5 ;i=γραμμή
        For j = 1 To diaitologioPlusLast

```

```

            If diaitologioPlus(j, 7) = Str(i)
                statisticsArray(i,1)

```

```

            =

```

```

            statisticsArray(i,1)

```

```

            +

```

```

            1

```

```

;TPOOH

```

```

statisticsArray(i,2) = statisticsArray(i,2) + ValF(diaitologioPlus(j, 3)) *
ValF(diaitologioPlus(j, 8)) ;ENEPT
statisticsArray(i,4) = statisticsArray(i,4) + ValF(diaitologioPlus(j, 4)) *
ValF(diaitologioPlus(j, 8)) * 4 ;PRO
statisticsArray(i,6) = statisticsArray(i,6) + ValF(diaitologioPlus(j, 5)) *
ValF(diaitologioPlus(j, 8)) * 4 ;CARB
statisticsArray(i,8) = statisticsArray(i,8) + ValF(diaitologioPlus(j, 6)) *
ValF(diaitologioPlus(j, 8)) * 9 ;FAT
statisticsArray(i,10) = statisticsArray(i,4) + statisticsArray(i,6) + statisticsArray(i,8)
;ENEPT2
EndIf
Next j
Next i

;ekth grammh
For i = 1 To 5 ;i=γραμμή
statisticsArray(6,1) = statisticsArray(6,1) + statisticsArray(i,1) ;TPODH
statisticsArray(6,2) = statisticsArray(6,2) + statisticsArray(i,2) ;ENEPT
statisticsArray(6,4) = statisticsArray(6,4) + statisticsArray(i,4) ;PRO
statisticsArray(6,6) = statisticsArray(6,6) + statisticsArray(i,6) ;CARB
statisticsArray(6,8) = statisticsArray(6,8) + statisticsArray(i,8) ;FAT
statisticsArray(6,10) = statisticsArray(6,10) + statisticsArray(i,10);ENEPT2
Next i

;pososta
For i = 1 To 5
If statisticsArray(6,2) <> 0
statisticsArray(i,3) = statisticsArray(i,2)*100/statisticsArray(6,2) ;ENEPT%
EndIf
If statisticsArray(i,10) <>0
statisticsArray(i,5) = statisticsArray(i,4)*100/statisticsArray(i,10) ;PRO%
statisticsArray(i,7) = statisticsArray(i,6)*100/statisticsArray(i,10) ;CARB%
statisticsArray(i,9) = statisticsArray(i,8)*100/statisticsArray(i,10) ;FAT%
EndIf
Next i

;athrismata posostwn
olikesThermides.f = statisticsArray(6,4) + statisticsArray(6,6) + statisticsArray(6,8)
If olikesThermides <>0
statisticsArray(6,5) = statisticsArray(6,4)*100 / olikesThermides
statisticsArray(6,7) = statisticsArray(6,6)*100 / olikesThermides
statisticsArray(6,9) = statisticsArray(6,8)*100 / olikesThermides
EndIf
For i=1 To 5: statisticsArray(6,3)= statisticsArray(6,3) + statisticsArray(i,3): Next i

EndProcedure

Procedure LoadFile(PPath.s,PFile.s)

If ReadFile(0, PPath + Pfile ) ;If the file could be Read, we Continue.

If ReadString(0) = IDStringFile ;αν είναι βάση δεδομένων του προγράμματος
lineCounter = 0 : Dim foodTempArray.s(6,1)
While Eof(0) = 0
lineCounter = lineCounter + 1
ReDim foodTempArray(6,lineCounter)
a.s = ReadString(0)
For i = 1 To 6
foodTempArray( i, lineCounter ) = StringField(a, i, ";")
Next i
Wend
CloseFile(0)
Global Dim Food(lineCounter,6)
For i = 1 To lineCounter: For j= 1 To 6
Food(i,j) = foodTempArray(j,i)
Next j: Next i
MessageRequester("ΠΛΗΡΟΦΟΡΙΑ", "Η φόρτωση της βάσης δεδομένων έχει πραγματοποιηθεί.", #PB_MessageRequester_Ok)
Else
MessageRequester("ΣΦΑΛΜΑ!", "Το αρχείο που έχει επιλεγεί δε φαίνεται να είναι βάση δεδομένων του προγράμματος.")
ProcedureReturn
EndIf

Else ;If the file could not be Read
Global Dim Food.s ( 1, 6 )

```

```
Food(1,1)="Code": Food(1,2)="Food":  
Food(1,3)="Energy":Food(1,4)="Protein":Food(1,5)="Carbohydrates":Food(1,6)="Fat"  
MessageRequester("ΣΦΑΛΜΑ!", "Δε φορτώθηκε Βάση Δεδομένων Τροφών."+Chr(10)+"Παρακαλώ φορτώστε μια από το πρόγραμμα  
(TAB:Βάση Δεδομένων).")  
EndIf
```

EndProcedure

Procedure GetMeal()

```
Pline.s = GetGadgetText(diatDiatologio)  
PlineFirst.s = Left(Pline,1)  
PlineSecondThird.s = Mid(Pline,2,2)  
If PlineSecondThird = "o "  
    selectedMeal = Val(PlineFirst)  
Else  
    selectedMeal = 1  
EndIf
```

EndProcedure

Global main

Global mainPanel, Image_0, Text_1_Copy1, perigrifi3, perigrifi2, ilikia, varos, ypsos, per_mesis, per_ishion, Text_met08, Text_met09, Text_met17, Text_met02, Text_met01, Text_met11, Text_met12, Text_met03, Text_met13, Text_met18, Text_met19, Text_met10, perigrifi, Text_met15, isozygio, Text_met04, DMS, OlikoPos, Oliko_lipos, alipiPos, Alipi_maza, mesiIshia, idaniko, Drasrriotita, Fylo, metrFylo, Text_met16, Text_met20, Text_met21, BasikosMetabolismos, FysikiDrastitiothta, Frame3D_0, Frame3D_1, Frame3D_5, Frame3D_6, Frame3D_7, Frame3D_8, Frame3D_9, imeresDiaitas, protasi, FysikiDrastitiothta2, metavoliThermidon, proteinomenoVaros, varos2, Text_sto01, Text_sto02, Text_sto03, Text_sto04, Text_sto05, Text_sto06, EpithimitoVaros, zitoymeno, metavoli, Text_sto07, Text_sto08, Text_sto09, diatEktyposi, diatKatharismos, Text_0, diatDiatologio, diatProsthiki, diatAfaresi, diaitGeymaP, diaitGeymaM, diaitPosotitaP, diaitPosotitaM, Frame3D_0, Frame3D_2, Frame3D_03, diaitThesiP, diaitThesiM, anazInfoText, anazApootelesmata, anazKleidi, anazEpilogi, anazCaseSensitive, anazEyrethenta, statProgressBar, statText, statResuls2, statResuls1, FileText, FileExplorerTree, FileString, FileOption4, FileOption3, FileOption2, FileOption1, FileButton, Image_3, Image_2, HelpEditor, Image_1, PeriEditor

Global Img_0, Img_1, Img_2, Img_3

UseJPEGImageDecoder()

Img_0 = LoadImage(#PB_Any, "C:\diet.jpg")
 Img_1 = LoadImage(#PB_Any, "C:\Users\Nikolas\Desktop\apky.jpg")
 Img_2 = LoadImage(#PB_Any, "C:\Users\Nikolas\Desktop\BOHTHEIA.jpg")
 Img_3 = LoadImage(#PB_Any, "C:\Users\Nikolas\Desktop\food-data1.jpg")

Enumeration #PB_Compiler_EnumerationValue
 #Font_0
 EndEnumeration

LoadFont(#Font_0, "Comic Sans MS", 16)

Procedure Openmain()

main = OpenWindow(#PB_Any, 0, 0, 600, 400, "Διαιτολόγιο 2014", #PB_Window_SystemMenu | #PB_Window_ScreenCentered)
 mainPanel = PanelGadget(#PB_Any, 0, 0, 600, 400)
 AddGadgetItem(mainPanel, -1, " Αρχική ")
 Image_0 = ImageGadget(#PB_Any, 0, -2, 600, 380, ImageID(Img_0))
 Text_1_Copy1 = TextGadget(#PB_Any, 140, 10, 340, 28, "ΔΙΑΙΤΟΛΟΓΙΟ 2014", #PB_Text_Center)
 SetGadgetColor(Text_1_Copy1, #PB_Gadget_FrontColor, RGB(0,255,0))
 SetGadgetFont(Text_1_Copy1, FontID(#Font_0))
 AddGadgetItem(mainPanel, -1, " Μετρήσεις ")
 perigrifi3 = TextGadget(#PB_Any, 430, 278, 130, 20, "")
 perigrifi2 = TextGadget(#PB_Any, 430, 260, 130, 18, "")
 ilikia = StringGadget(#PB_Any, 170, 80, 100, 25, "", #PB_String_Numeric)
 GadgetToolTip(ilikia, "Εισάγετε την ηλικία του ατόμου.")
 varos = StringGadget(#PB_Any, 170, 120, 100, 25, "", #PB_String_Numeric)
 GadgetToolTip(varos, "Εισάγετε το βάρος του ατόμου.")
 ypsos = StringGadget(#PB_Any, 170, 160, 100, 25, "", #PB_String_Numeric)
 GadgetToolTip(ypsos, "Εισάγετε το ύψος του ατόμου.")
 per_mesis = StringGadget(#PB_Any, 170, 280, 100, 25, "", #PB_String_Numeric)
 GadgetToolTip(per_mesis, "Εισάγετε την περιφέρεια μέσης του ατόμου.")
 per_ishion = StringGadget(#PB_Any, 170, 320, 100, 25, "", #PB_String_Numeric)
 GadgetToolTip(per_ishion, "Εισάγετε την περιφέρεια ισχίων του ατόμου.")
 Text_met08 = TextGadget(#PB_Any, 40, 280, 110, 20, "Μέσης (cm)")
 Text_met09 = TextGadget(#PB_Any, 40, 320, 110, 20, "Ισχίων (cm)")
 GadgetToolTip(Text_met09, "Εισάγετε την περιφέρεια μέσης του ατόμου.")
 Text_met17 = TextGadget(#PB_Any, 430, 150, 120, 20, "Ολικό λίπος [%]")
 Text_met02 = TextGadget(#PB_Any, 40, 120, 110, 20, "Βάρος [Kg]")
 GadgetToolTip(Text_met02, "Εισάγετε το βάρος του ατόμου.")
 Text_met01 = TextGadget(#PB_Any, 40, 80, 110, 20, "Ηλικία [έτη]")
 GadgetToolTip(Text_met01, "Εισάγετε την ηλικία του ατόμου.")
 Text_met11 = TextGadget(#PB_Any, 310, 90, 110, 20, "ΔΜΣ")
 Text_met12 = TextGadget(#PB_Any, 310, 150, 110, 20, "Ολικό λίπος [Kg]")
 Text_met03 = TextGadget(#PB_Any, 40, 160, 110, 20, "Υψος [cm]")
 GadgetToolTip(Text_met03, "Εισάγετε το ύψος του ατόμου.")
 Text_met13 = TextGadget(#PB_Any, 310, 200, 110, 20, "Άλλη μάζα [Kg]")
 Text_met18 = TextGadget(#PB_Any, 430, 200, 120, 20, "Άλλη μάζα [%]")
 Text_met19 = TextGadget(#PB_Any, 310, 260, 110, 20, "Μέση/Ισχία")
 Text_met10 = TextGadget(#PB_Any, 310, 20, 110, 20, "Ιδανικό Βάρος [Kg]")
 perigrifi = TextGadget(#PB_Any, 430, 110, 120, 20, "", #PB_Text_Center)
 Text_met15 = TextGadget(#PB_Any, 430, 20, 110, 20, "Ισοζύγιο [Kg]")
 isozygio = TextGadget(#PB_Any, 430, 40, 110, 20, "0", #PB_Text_Center)
 Text_met04 = TextGadget(#PB_Any, 40, 200, 110, 20, "Δραστηριότητα")
 GadgetToolTip(Text_met04, "Επιλέξτε το επίπεδο της φυσικής δραστηριότητας του ατόμου.")

```

DMS = TextGadget(#PB_Any, 310, 110, 110, 20, "0", #PB_Text_Center)
OlikoPos = TextGadget(#PB_Any, 430, 170, 120, 20, "0", #PB_Text_Center)
Oliko_lipos = TextGadget(#PB_Any, 310, 170, 110, 20, "0", #PB_Text_Center)
alipiPos = TextGadget(#PB_Any, 430, 220, 120, 20, "0", #PB_Text_Center)
Alipi_maza = TextGadget(#PB_Any, 310, 220, 110, 20, "0", #PB_Text_Center)
mesiIshia = TextGadget(#PB_Any, 310, 280, 110, 20, "0", #PB_Text_Center)
idaniko = TextGadget(#PB_Any, 310, 40, 110, 20, "0", #PB_Text_Center)
Drasrriotita = ComboBoxGadget(#PB_Any, 170, 200, 100, 25)
GadgetToolTip(Drasrriotita, "Επιλέξτε το επίπεδο της φυσικής δραστηριότητας του ατόμου.")
AddGadgetItem(Drasrriotita, -1, "Καθιστική Ζωή")
AddGadgetItem(Drasrriotita, -1, "Μικρή")
AddGadgetItem(Drasrriotita, -1, "Μέτρια")
AddGadgetItem(Drasrriotita, -1, "Μεγάλη")
AddGadgetItem(Drasrriotita, -1, "Πολύ μεγάλη")
Fylo = TextGadget(#PB_Any, 40, 40, 110, 20, "Φύλο")
GadgetToolTip(Fylo, "Επιλέξτε το φύλο του ατόμου.")
metrFylo = ComboBoxGadget(#PB_Any, 170, 40, 100, 25)
GadgetToolTip(metrFylo, "Επιλέξτε το φύλο του ατόμου.")
AddGadgetItem(metrFylo, -1, "Αντρας")
AddGadgetItem(metrFylo, -1, "Γυναίκα")
Text_met16 = TextGadget(#PB_Any, 430, 90, 120, 20, "Περιγραφή")
Text_met20 = TextGadget(#PB_Any, 310, 320, 110, 20, "Βασικός Μετ.")
Text_met21 = TextGadget(#PB_Any, 430, 320, 120, 20, "BM + Φυσ. Δραστ.")
BasikosMetabolismos = TextGadget(#PB_Any, 310, 340, 110, 20, "0", #PB_Text_Center)
FysikiDrastitiohta = TextGadget(#PB_Any, 430, 340, 120, 20, "0", #PB_Text_Center)
Frame3D_0 = Frame3DGadget(#PB_Any, 290, 8, 280, 60, "")
Frame3D_1 = Frame3DGadget(#PB_Any, 290, 68, 280, 72, "")
Frame3D_5 = Frame3DGadget(#PB_Any, 290, 308, 280, 60, "")
Frame3D_6 = Frame3DGadget(#PB_Any, 290, 248, 280, 62, "")
Frame3D_7 = Frame3DGadget(#PB_Any, 290, 138, 280, 110, "")
Frame3D_8 = Frame3DGadget(#PB_Any, 30, 8, 250, 232, "Ατομικά Στοιχεία")
Frame3D_9 = Frame3DGadget(#PB_Any, 30, 248, 250, 120, "Περιφέρειες ")
AddGadgetItem(mainPanel, -1, " Στόχοι ")
imeresDiatas = TextGadget(#PB_Any, 490, 170, 80, 25, "", #PB_Text_Center)
GadgetToolTip(imeresDiatas, "Οι ημέρες διαιτητικής παρέμβασης που απαιτούνται.")
protasi = TextGadget(#PB_Any, 320, 300, 100, 25, "", #PB_Text_Center)
GadgetToolTip(protasi, "Ο προτεινόμενος ενεργειακός στόχος, βάσει των επιλογών που έγιναν.")
FysikiDrastitiohta2 = TextGadget(#PB_Any, 320, 240, 100, 25, "", #PB_Text_Center)
GadgetToolTip(FysikiDrastitiohta2, "Ο Βασικός Μεταβολισμός συν τη Φυσική Δραστηριότητα.")
metavoliThermidon = TextGadget(#PB_Any, 320, 170, 100, 25, "", #PB_Text_Center)
GadgetToolTip(metavoliThermidon, "Θερμιδική μεταβολή στο διατολόγιο, ώστε να επιτευχθεί η ζητούμενη μεταβολή βάρους")
proteinoMenoVaros = TextGadget(#PB_Any, 10, 110, 100, 25, "", #PB_Text_Center)
GadgetToolTip(proteinoMenoVaros, "Παρουσιάζεται η πρόταση για το βάρος στόχος, ώστε το άτομο να είναι οριακά φυσιολογικό.")
varos2 = TextGadget(#PB_Any, 10, 40, 100, 25, "", #PB_Text_Center)
GadgetToolTip(varos2, "Το βάρος του ατόμου.")
Text_sto01 = TextGadget(#PB_Any, 160, 76, 100, 25, "Επιθυμητό Βάρος")
GadgetToolTip(Text_sto01, "Εισάγετε το βάρος στόχο της διατροφικής παρέμβασης, Μπορεί να είναι διαφορετικό από το προτεινόμενο.")
Text_sto02 = TextGadget(#PB_Any, 124, 138, 196, 25, "Μεταβολή βάρους Σώματος [g/m]")
GadgetToolTip(Text_sto02, "Εισάγετε πόσα gr επιθυμείτε να μεταβληθεί το βάρος σώματος σε ένα μήνα.")
Text_sto03 = TextGadget(#PB_Any, 490, 144, 100, 25, "Διάρκεια [d]")
GadgetToolTip(Text_sto03, "Οι ημέρες διαιτητικής παρέμβασης που απαιτούνται.")
Text_sto04 = TextGadget(#PB_Any, 480, 280, 110, 25, "Ζητούμενο [Kcal/d]")
GadgetToolTip(Text_sto04, "Εισάγετε τον ενεργειακό στόχο. Μπορεί να είναι διαφορετικός από τον προτεινόμενο.")
Text_sto05 = TextGadget(#PB_Any, 10, 76, 120, 25, "Προτεινόμενο Βάρος")
GadgetToolTip(Text_sto05, "Παρουσιάζεται η πρόταση για το βάρος στόχος, ώστε το άτομο να είναι οριακά φυσιολογικό.")
Text_sto06 = TextGadget(#PB_Any, 320, 276, 100, 25, "Πρόταση [Kcal/d]")
GadgetToolTip(Text_sto06, "Ο προτεινόμενος ενεργειακός στόχος, βάσει των επιλογών που έγιναν")
ErithimitoVaros = StringGadget(#PB_Any, 157, 105, 100, 25, "", #PB_String_Numeric)
GadgetToolTip(ErithimitoVaros, "Εισάγετε το βάρος στόχο της διατροφικής παρέμβασης. Μπορεί να είναι διαφορετικό από το προτεινόμενο.")
zitoymeno = StringGadget(#PB_Any, 480, 300, 100, 25, "", #PB_String_Numeric)
GadgetToolTip(zitoymeno, "Εισάγετε τον ενεργειακό στόχο. Μπορεί να είναι διαφορετικός από τον προτεινόμενο.")
metavoli = StringGadget(#PB_Any, 160, 160, 100, 25, "")
GadgetToolTip(metavoli, "Εισάγετε πόσα gr επιθυμείτε το βάρος σώματος του ατόμου να μεταβάλλεται ανα μήνα.")
Text_sto07 = TextGadget(#PB_Any, 320, 140, 170, 25, "Μεταβολή πρόσληψης [cal/d]")
GadgetToolTip(Text_sto07, "Θερμιδική μεταβολή στο διατολόγιο, ώστε να επιτευχθεί η ζητούμενη μεταβολή βάρους")
Text_sto08 = TextGadget(#PB_Any, 10, 10, 100, 25, "Βάρος [Kgr]")
GadgetToolTip(Text_sto08, "Το βάρος του ατόμου.")
Text_sto09 = TextGadget(#PB_Any, 320, 210, 100, 25, "BM + ΦΔ")
GadgetToolTip(Text_sto09, "Ο Βασικός Μεταβολισμός συν τη Φυσική Δραστηριότητα.")
AddGadgetItem(mainPanel, -1, "Διατολόγιο ")
diatEktyposi = ButtonGadget(#PB_Any, 110, 330, 80, 28, "Εκτύπωση")
diatKatharismos = ButtonGadget(#PB_Any, 30, 330, 70, 28, "Καθαρισμός")

```

```

Text_0 = TextGadget(#PB_Any, 30, 10, 510, 20, "Κωδικός Τρόφιμο Ενέργεια Πρωτεΐνες Υδατανθρακες Λίπη
Γεύμα Ποσότητα Θέση", #PB_Text_Center)
diatDiatologio = ListViewGadget(#PB_Any, 30, 38, 510, 250)
GadgetToolTip(diatDiatologio, "Χώρος εμφάνισης διαιτολογίου")
diatProsthiki = ButtonGadget(#PB_Any, 30, 298, 70, 20, "Προσθήκη")
GadgetToolTip(diatProsthiki, "Επιλέξτε για προσθήκη στοιχείου")
diatAfairesi = ButtonGadget(#PB_Any, 110, 298, 80, 22, "Αφαίρεση")
GadgetToolTip(diatAfairesi, "Επιλέξτε στοιχείο και έπειτα πιάστε εδώ για αφαίρεσή του")
diatGeymaP = ButtonGadget(#PB_Any, 220, 310, 20, 20, "+")
GadgetToolTip(diatGeymaP, "Επιλέξτε στοιχείο και έπειτα πιάστε για να αυξηθεί η κατηγορία γεύματός του")
diatGeymaM = ButtonGadget(#PB_Any, 250, 310, 20, 20, "-")
GadgetToolTip(diatGeymaM, "Επιλέξτε στοιχείο και έπειτα πιάστε για να μειωθεί η κατηγορία γεύματός του")
diatPosotitaP = ButtonGadget(#PB_Any, 310, 310, 20, 20, "+")
GadgetToolTip(diatPosotitaP, "Επιλέξτε στοιχείο και έπειτα πιάστε για να αυξηθεί η ποσότητά του")
diatPosotitaM = ButtonGadget(#PB_Any, 340, 310, 20, 20, "-")
GadgetToolTip(diatPosotitaM, "Επιλέξτε στοιχείο και έπειτα πιάστε για να μειωθεί η ποσότητά του")
Frame3D_0 = Frame3DGadget(#PB_Any, 210, 290, 80, 50, "Γεύμα")
Frame3D_2 = Frame3DGadget(#PB_Any, 300, 290, 100, 50, "Ποσότητα")
Frame3D_03 = Frame3DGadget(#PB_Any, 390, 290, 80, 50, "Θέση")
diatThesiP = ButtonGadget(#PB_Any, 410, 310, 20, 20, "+")
GadgetToolTip(diatThesiP, "Επιλέξτε στοιχείο και έπειτα πιάστε για να αυξηθεί η θέση του")
diatThesiM = ButtonGadget(#PB_Any, 430, 310, 20, 20, "-")
GadgetToolTip(diatThesiM, "Επιλέξτε στοιχείο και έπειτα πιάστε για να μειωθεί η θέση του")
AddGadgetItem(mainPanel, -1, "Αναζήτηση ")
anazInfoText = TextGadget(#PB_Any, 40, 290, 500, 25, "", #PB_Text_Center)
anazApotelesmata = ListViewGadget(#PB_Any, 40, 10, 500, 278)
GadgetToolTip(anazApotelesmata, "Χώρος εμφάνισης αποτελεσμάτων αναζήτησης.")
anazKleidi = StringGadget(#PB_Any, 40, 320, 250, 30, "")
GadgetToolTip(anazKleidi, "Εισάγετε τη λέξη κλειδί για αναζήτηση")
anazEpilogi = ButtonGadget(#PB_Any, 450, 320, 90, 30, "Επιλογή")
GadgetToolTip(anazEpilogi, "Επιλέξτε στοιχείο και έπειτα πιάστε για να εισαχθεί στο διαιτολόγιο.")
anazCaseSensitive = CheckBoxGadget(#PB_Any, 390, 320, 40, 25, "KM")
GadgetToolTip(anazCaseSensitive, "Διάκριση κεφαλαίων - μικρών.")
anazEyrethenta = TextGadget(#PB_Any, 300, 330, 70, 25, "")
AddGadgetItem(mainPanel, -1, "Στατιστικά ")
statProgressBar = ProgressBarGadget(#PB_Any, 320, 350, 260, 20, 0, 100, #PB_ProgressBar_Smooth)
statText = TextGadget(#PB_Any, 10, 350, 290, 25, "")
statResuls2 = ListIconGadget(#PB_Any, 10, 180, 580, 160, "Κατηγορία", 90)
GadgetToolTip(statResuls2, "Χώρος απεικόνισης στατιστικών αποτελεσμάτων")
AddGadgetColumn(statResuls2, 1, "Πρωτεΐνη ", 80)
AddGadgetColumn(statResuls2, 2, "Υδατ/κες", 80)
AddGadgetColumn(statResuls2, 3, "Λίπος", 80)
AddGadgetColumn(statResuls2, 4, "%Πρωτεΐνη ", 80)
AddGadgetColumn(statResuls2, 5, "%Υδατ/κες", 80)
AddGadgetColumn(statResuls2, 6, "%Λίπος ", 80)
statResuls1 = ListIconGadget(#PB_Any, 10, 10, 580, 160, "Κατηγορία", 160)
GadgetToolTip(statResuls1, "Χώρος απεικόνισης στατιστικών αποτελεσμάτων.")
AddGadgetColumn(statResuls1, 1, "Τροφές", 120)
AddGadgetColumn(statResuls1, 2, "Ενέργεια", 120)
AddGadgetColumn(statResuls1, 3, "%Ενέργεια", 120)
AddGadgetItem(mainPanel, -1, "Αρχεία")
FileText = TextGadget(#PB_Any, 30, 220, 470, 25, "", #PB_Text_Center)
FileExplorerTree = ExplorerListGadget(#PB_Any, 30, 280, 470, 78, "", #PB_Explorer_NoMyDocuments |
#PB_Explorer_BorderLess | #PB_Explorer_AlwaysShowSelection | #PB_Explorer_FullRowSelect)
FileString = StringGadget(#PB_Any, 30, 248, 470, 30, "Filename.txt")
FileOption4 = OptionGadget(#PB_Any, 360, 110, 230, 25, "Αποθήκευση εκτυπώσιμης διαίτας")
FileOption3 = OptionGadget(#PB_Any, 360, 80, 160, 25, "Αποθήκευση διαίτας")
FileOption2 = OptionGadget(#PB_Any, 360, 50, 180, 25, "Φόρτωση διαίτας")
FileOption1 = OptionGadget(#PB_Any, 360, 20, 190, 25, "Φόρτωση βάσης δεδομένων")
FileButton = ButtonGadget(#PB_Any, 520, 248, 60, 110, "Εκκίνηση")
GadgetToolTip(FileButton, "Πιάστε αφού επιλέξετε αρχείο φόρτωσης.")
Image_3 = ImageGadget(#PB_Any, 20, 8, 260, 180, ImageID(Img_3))
AddGadgetItem(mainPanel, -1, "Βοήθεια ")
Image_2 = ImageGadget(#PB_Any, 180, 10, 220, 150, ImageID(Img_2))
HelpEditor = EditorGadget(#PB_Any, 20, 188, 560, 180)
AddGadgetItem(mainPanel, -1, "Περί ")
Image_1 = ImageGadget(#PB_Any, 130, 30, 348, 145, ImageID(Img_1))
PeriEditor = EditorGadget(#PB_Any, 30, 208, 544, 140, #PB_Editor_ReadOnly)
CloseGadgetList()
EndProcedure

Procedure main_Events(event)
Select event
Case #PB_Event_CloseWindow
ProcedureReturn #False

Case #PB_Event_Menu

```

```
Select EventMenu()  
EndSelect  
  
Case #PB_Event_Gadget  
  Select EventGadget()  
  EndSelect  
EndSelect  
ProcedureReturn #True  
EndProcedure
```