

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ



Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών



ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ 2014-2015

Θέμα:

«Ανάπτυξη εφαρμογής για μετρητή θερμίδων σε Android»
«Development of a calories counter application in android»

Αποστολάκης Χαράλαμπος

Επιβλέπων Καθηγητής:

Ακρίτας Αλκιβιάδης (Καθηγητής)

Συν-επιβλέπων Καθηγητής:

Σταμούλης Γεώργιος (Καθηγητής)

Βόλος, Ιούλιος 2015

Ευχαριστίες,

Με την εκπόνηση της παρούσας Διπλωματικής Εργασίας, φέρνω εις πέρας τις προπτυχιακές μου σπουδές στο Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Θεσσαλίας.

Αρχικά, θα ήθελα να ευχαριστήσω τον Καθηγητή του Τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, κ. Αλκιβιάδη Ακρίτα, για την ανάθεση αυτής της Διπλωματικής Εργασίας και την δυνατότητα που μου έδωσε να ασχοληθώ με την ανάπτυξη εφαρμογών σε Android. Τον ευχαριστώ πολύ για την εμπιστοσύνη και την υποστήριξη που μου έδειξε κατά τη διάρκεια εκπόνησης της Διπλωματικής μου εργασίας.

Ευχαριστώ θερμά την οικογένεια μου, που μου συμπαραστάθηκε σε όλη τη διάρκεια των σπουδών μου και πάντα ήταν εκεί σε όλες τις δυσκολίες που αντιμετώπισα. Η εμπιστοσύνη και η στήριξη που μου έδειξε είχαν καταλυτικό ρόλο στο να καταφέρω να ολοκληρώσω τις σπουδές μου και θα της είμαι πάντα ευγνώμων.

Τέλος, θα ήθελα να ευχαριστήσω την κοπέλα μου και όλους τους φίλους που απέκτησα αυτά τα χρόνια για τη βοήθεια, τη στήριξη και τις συμβουλές που μου έδωσαν. Οι όμορφες στιγμές που μου χάρισαν θα μείνουν αξέχαστες.

Αφιερωμένο στους γονείς μου, Μανώλη και Κορίνα.

Και στον αδερφό μου Στέλιο.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΕΧΟΜΕΝΑ	4
ΠΕΡΙΛΗΨΗ	7
ABSTRACT	8
1.Εισαγωγή	9
1.1 Τι είναι το Android;.....	9
1.2 Σύντομη ιστορική αναδρομή	10
1.3 Εκδόσεις του Android	10
1.3.1 Android 1.5 Cupcake (API Level 3)	11
1.3.2 Android 1.6 Donut (API Level 4).....	11
1.3.3 Android 2.0 Eclair (API Level 7)	12
1.3.4 Android Frogo 2.2 (API Level 8).....	13
1.3.5 Android 2.3 Gingerbread (API Level 9).....	13
1.3.6 Android 3.0 Honeycomb (API Level 11)	14
1.3.7 Android 4.0 Ice Cream Sandwich (API Level 15)	14
1.3.8 Android 4.1 Jelly Bean (API Level 16).....	15
1.3.9 Android 4.4 KitKat (API Level 19)	15
1.3.10 Android 5.0 Lollipop (API Level 21)	16
1.4 Χρήση κάθε έκδοσης του Android σήμερα	16
2. Εργαλεία	17
2.1 Android Studio 1.2.2	17
2.2 SQLite Database Browser	19
2.3 Ζωγραφική και Microsoft Word	20
2.4 Συσσκευές	21
3. Επίδειξη της εφαρμογής	22
3.1 Η γενική φιλοσοφία	22
3.2 Καλωσόρισμα και εισαγωγή στοιχείων	22
3.3 Κύριο μενού της εφαρμογής	27
3.3.1 ENTER CALORIES	28
3.3.2 CHECK RECENT DIET	30
3.3.3 CALORIES PER FOOD	32

3.4 Έξοδος και επανεισαγωγή στοιχείων	33
4. Αλγόριθμοι για υπολογισμούς.....	35
4.1 Υπολογισμός απαραίτητων θερμίδων.....	35
4.2 Υπολογισμός για πρόοδο.....	36
5. Αρχεία .java	37
5.1 MainActivity.java	37
5.2 SecondScreen.java	41
5.3 ThirdScreen.java.....	43
5.4 FourthScreen.java	44
5.5 FifthScreen.java.....	45
5.6 TableDatabase.java	46
5.7 CaloriesDatabase.java	47
6. Αρχεία .xml.....	50
6.1 Φάκελος layout.....	50
6.1.1 activity_main.xml.....	50
6.1.2 second_layout.xml	51
6.1.3 third_layout.xml.....	54
6.1.4 fourth_layout.xml	55
6.1.5 fifth_layout.xml.....	57
6.1.6 web_layout.xml.....	58
6.2 Φάκελος drawable	58
6.2.1 button.xml.....	58
6.2.2 apptheme_btn_radio_holo_light.xml.....	59
6.3 Φάκελος menu.....	60
6.3.1 menu_main.xml	60
6.4 Φάκελος values.....	60
6.4.1 strings.xml	60
6.4.2 dimens.xml.....	61
6.4.3 colors_apptheme.xml	61
6.4.4 themes_apptheme.xml.....	62
6.4.5 styles_apptheme.xml.....	62
6.5 AndroidManifest.xml	63

7. Αρχείο .html	64
8. Πιθανές μελλοντικές επεκτάσεις	70
9. Βιβλιογραφία	71

ΠΕΡΙΛΗΨΗ

Η παρούσα διπλωματική εργασία με τίτλο «Ανάπτυξη εφαρμογής για μετρητή θερμίδων σε Android» έχει ως στόχο την υλοποίηση μιας εφαρμογής σε περιβάλλον Android, η οποία θα είναι συμβατή με τα περισσότερα smartphones (έξυπνα τηλέφωνα) που έχουν λογισμικό Android.

Τα smartphones έχουν μπει τα τελευταία χρόνια για τα καλά στην καθημερινότητα μας και εκατομμύρια άνθρωποι τα χρησιμοποιούν καθημερινά. Το λογισμικό που έχει κυριαρχήσει σε αυτά, είναι το Android και αυτός είναι ο κύριο λόγος που η εφαρμογή αναπτύχθηκε ώστε να τρέχει σε αυτό το λογισμικό.

Η δημιουργία της εφαρμογής έγινε με την χρήση του λογισμικού Android Studio, το οποίο είναι ένα ολοκληρωμένο περιβάλλον μέσα από το οποίο μπορούμε να γράψουμε και να εκτελέσουμε κώδικα. Είναι ανοικτού κώδικα (open source) και ελεύθερης διανομής (freeware).

Σκοπός της εφαρμογής είναι να βοηθήσει τον χρήστη να καταγράψει τις θερμίδες που παίρνει από τη διατροφή του καθημερινά και να ελέγξει με αυτόν τον τρόπο το σωματικό του βάρος. Η εφαρμογή μπορεί να χρησιμοποιηθεί από όλους τους χρήστες, όχι όμως από παιδιά καθώς οι αλγόριθμοι που έχουν χρησιμοποιηθεί απευθύνονται σε ενήλικες.

ABSTRACT

The target of this thesis with title «Development of a calories counter application in Android» is to build an application in Android OS, which will be compatible with most smartphones that run Android.

In recent years smartphones have a vital role in our daily life and million people use them every day. The OS which has dominated is Android and this is the main reason why this app was developed in order to run in Android devices.

The development of the app was made with Android Studio, which is an integrated development environment (IDE) for developing on the Android platform. It is a freeware and open source project.

The application aims to help the user to record the daily calories he intakes from his diet and in this way to control his weight. It can be used from all users, but not from children as the algorithms which were used are based on calories stats for adults.

1.Εισαγωγή

1.1 Τι είναι το Android;

Το Android είναι λειτουργικό σύστημα για συσκευές κινητής τηλεφωνίας το οποίο τρέχει τον πυρήνα του λειτουργικού Linux. Αρχικά αναπτύχθηκε από την Google και αργότερα από την Open Handset Alliance. Επιτρέπει στους κατασκευαστές λογισμικού να συνθέτουν κώδικα με την χρήση της γλώσσας προγραμματισμού Java, ελέγχοντας την συσκευή μέσω βιβλιοθηκών λογισμικού ανεπτυγμένων από την Google. Το Android είναι κατά κύριο λόγο σχεδιασμένο για συσκευές με οθόνη αφής, όπως τα έξυπνα τηλέφωνα και τα τάμπλετ. Παρόλο που έχει αναπτυχθεί για συσκευές με οθόνη αφής, έχει χρησιμοποιηθεί σε κονσόλες παιχνιδιών, ψηφιακές φυτογραφικές μηχανές, συνηθισμένους Η/Υ (π.χ. το HP Slate 21) και σε άλλες ηλεκτρονικές συσκευές.

Το Android είναι το πιο ευρέως διαδεδομένο λογισμικό στον κόσμο. Οι συσκευές με Android έχουν περισσότερες πωλήσεις από όλες τις συσκευές Windows, iOS και Mac OS X μαζί.

Η πρώτη παρουσίαση της πλατφόρμας Android έγινε στις 5 Νοεμβρίου 2007, παράλληλα με την ανακοίνωση της ίδρυσης του οργανισμού Open Handset Alliance, μιας κοινοπραξίας 48 τηλεπικοινωνιακών εταιριών, εταιριών λογισμικού καθώς και κατασκευής hardware, οι οποίες είναι αφιερωμένες στην ανάπτυξη και εξέλιξη ανοιχτών προτύπων στις συσκευές κινητής τηλεφωνίας. Η Google δημοσίευσε το μεγαλύτερο μέρος του κώδικα του Android υπό τους όρους της Apache License, μιας ελεύθερης άδειας λογισμικού. Το λογότυπο για το λειτουργικό σύστημα Android είναι ένα ρομπότ σε χρώμα πράσινου μήλου και σχεδιάστηκε από τη γραφίστρια Irina Blok.



1.2 Σύντομη ιστορική αναδρομή

Το Android δημιουργήθηκε το 2003 στην Καλιφόρνια από τους Rubin, Miner, Sears και White. Αρχικός τους σκοπός ήταν να δημιουργήσουν ένα λειτουργικό για έξυπνα κινητά τα οποία θα γνωρίζουν που βρίσκεται ο ιδιοκτήτης τους ανά πάσα στιγμή. Αρχικά η ομάδα του Android λειτουργούσε μυστικά, αλλά η μεγάλη ανάπτυξη στο λειτουργικό ξεκίνησε μετά την εξαγορά του από την Google το 2005. Από τη χρονιά εκείνη μέχρι και το 2007 η Google δούλεψε σιωπηλά πάνω στο Android κατοχυρώνοντας πατέντες και ψάχνοντας συνεργάτες.

Το 2007, λοιπόν, δημιουργήθηκε η Open Handset Alliance, ένας συνεταιρισμός που αποτελούνταν από τη Google, την HTC, την Samsung, την Qualcomm (κατασκευαστές επεξεργαστών) και άλλους με σκοπό να κάνουν το Android ένα ανοικτό λογισμικό και να κυκλοφορήσουν τα πρώτα smartphones με λειτουργικό Android. Πράγματι, ένα χρόνο μετά κυκλοφόρησε από την HTC το HTC Dream (T-Mobile G1), το πρώτο smartphone με λειτουργικό Android. Στη συνέχεια, με πρώτη τη Samsung, δεκάδες εταιρείες υιοθέτησαν το Android ως λειτουργικό και κατάφεραν να το κάνουν το κυρίαρχο λειτουργικό αυτή τη στιγμή στην αγορά. Η αναφορά της Samsung δεν είναι τυχαία, αφού είναι η πρώτη εταιρεία στις πωλήσεις κινητών παγκοσμίως και η εξάπλωση του Android οφείλεται κατά μεγάλο βαθμό σε αυτήν.

1.3 Εκδόσεις του Android

Όπως προαναφέρθηκε, το Android είναι ένα λειτουργικό σύστημα ανοιχτού κώδικα, με αποτέλεσμα η εξέλιξη του να είναι ραγδαία. Αυτό αποδεικνύεται από το γεγονός ότι οι δέκα κύριες εκδόσεις του έχουν κυκλοφορήσει σε διάστημα μικρότερο των έξι ετών, από τον Απρίλη του 2009 μέχρι τον Μάρτιο του 2015.

Στην πληροφορική είθισται τα προϊόντα του hardware και του software να κυκλοφορούν πέρα από τον αριθμό έκδοσης τους, και με μία κωδική ονομασία. Αυτή η κωδική ονομασία μπορεί να είναι οτιδήποτε, από ονόματα πόλεων (π.χ. Windows Chicago), μέχρι ονόματα ζώων (π.χ. OSX Lion). Στο Android αυτά τα χαρακτηριστικά κώδικα ονόματα έχουν τη μορφή κάποιου γλυκίσματος. Ακολουθούν οι μέχρι σήμερα εκδόσεις του Android με τα κυριότερα χαρακτηριστικά που εισήγαγαν.

1.3.1 Android 1.5 Cupcake (API Level 3)

Παρουσιάστηκε στις 27 Απριλίου του 2009. Κύριο χαρακτηριστικό του ήταν η εισαγωγή ψηφιακού πληκτρολογίου. Μέχρι τότε τα περισσότερα smartphones είχαν φυσικό πληκτρολόγιο QWERTY. Επίσης υποστήριζε την αντιγραφή και επικόλληση στον πρόγραμμα περιήγησης στον Ιστό.



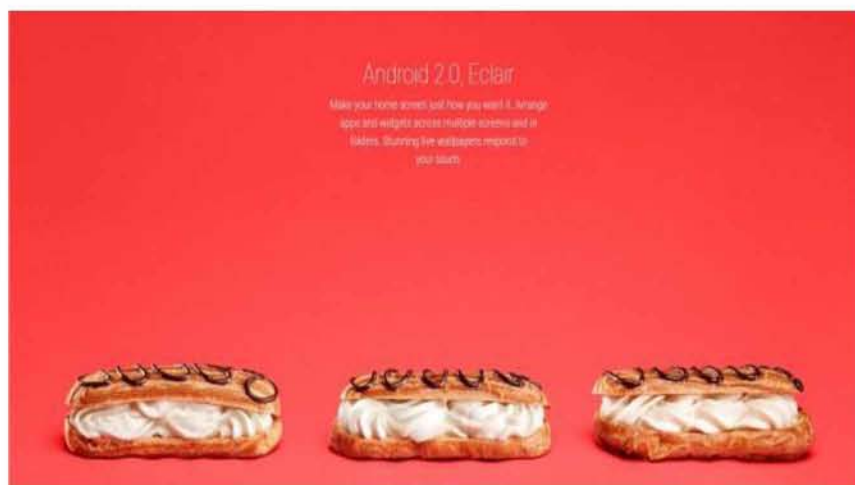
1.3.2 Android 1.6 Donut (API Level 4)

Παρουσιάστηκε στις 15 Σεπτεμβρίου του 2009. Σημαντικότερο χαρακτηριστικό του ήταν η υποστήριξη διαφορετικών αναλύσεων οθόνης, ανεξάρτητα από τον αριθμό pixels, κάτι που έπαιξε πολύ σημαντικό ρόλο στο μέλλον των Android συσκευών. Επιπρόσθετα δόθηκε στο χρήστη η δυνατότητα για πολλαπλή διαγραφή αρχείων. Είχε αρκετές αλλαγές στο εσωτερικό του λειτουργικού και αυτό είχε ως αποτέλεσμα να υποτιμηθεί αρκετά από τους καταναλωτές.



1.3.3 Android 2.0 Eclair (API Level 7)

Παρουσιάστηκε στις 26 Οκτωβρίου του 2009. Κύριο χαρακτηριστικό του ήταν οι μεγάλες βελτιώσεις που έφερε στην κάμερα. Μέχρι τότε το Android δεν είχε υποστήριξη LED flash στην κάμερα, ούτε δυνατότητες που σήμερα θεωρούμε δεδομένες, όπως χρωματικά εφέ, λειτουργία σκηνών, λειτουργίες εστίασης, ή ακόμη και ρύθμιση της ισορροπίας λευκού. Εδώ εισήχθησαν και για πρώτη φορά τα Live Wallpapers τα οποία επέτρεπαν στην εικόνα που υπάρχει στο φόντο να κινείται.



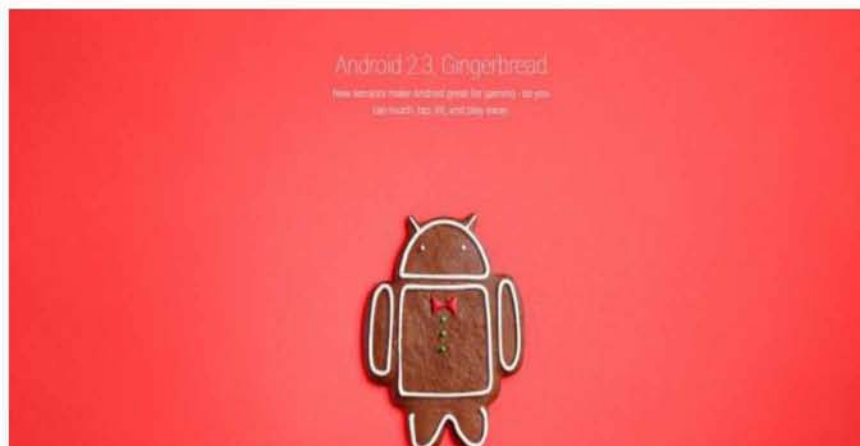
1.3.4 Android Froyo 2.2 (API Level 8)

Παρουσιάστηκε στις 20 Μαΐου του 2010. Πρόκειται για μια πολύ σημαντική έκδοση η οποία έκανε τις συσκευές από δυο έως πέντε φορές ταχύτερες. Υπήρχε βελτίωση στη διαχείριση της μνήμης και εισήχθη για πρώτη φορά το Wi-Fi Hotspot. Υποστήριζε ακόμη τον Adobe Flash Player.



1.3.5 Android 2.3 Gingerbread (API Level 9)

Παρουσιάστηκε στις 6 Δεκεμβρίου του 2010 και αποτελεί μια από τις διασημότερες εκδόσεις του λειτουργικού. Με την προσθήκη υποστήριξης για περισσότερους αισθητήρες, όπως το βαρόμετρο και γυροσκόπιο βοήθησε πραγματικά στην εξέλιξη του Android. Πριν το Gingerbread, με την έλλειψη πολλών αισθητήρων, υπήρχε σοβαρός περιορισμός στο τι μπορούσαμε να κάνουμε με το κινητό μας και ειδικότερα στα παιχνίδια.



1.3.6 Android 3.0 Honeycomb (API Level 11)

Παρουσιάστηκε στις 22 Φεβρουαρίου του 2011. Χωρίς αμφιβολία, το Honeycomb, ήταν μια από τις λιγότερο δημοφιλείς εκδόσεις του Android, αλλά αυτό σίγουρα δεν έχει να κάνει με την συνεισφορά του στην πλατφόρμα. Το Android μέχρι τότε, είχε φρικτή εμπειρία χρήσεως στα tablets, κάτι που άλλαξε με την κυκλοφορία του Honeycomb, όπου ήταν σχεδιασμένο για συσκευές με μεγαλύτερη οθόνη.



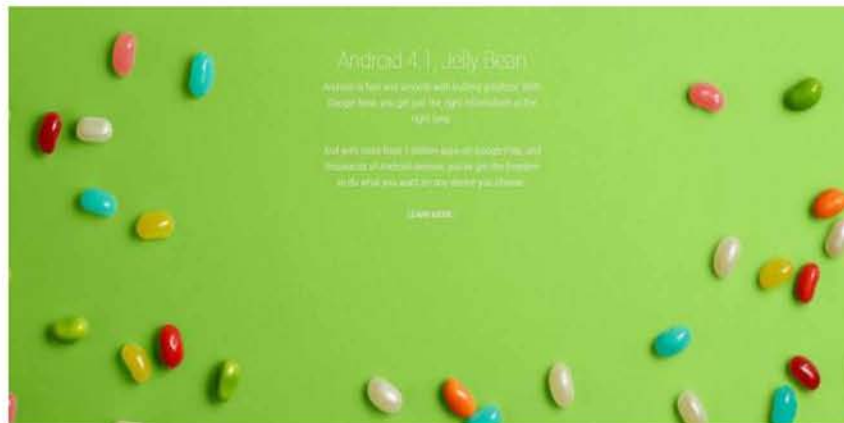
1.3.7 Android 4.0 Ice Cream Sandwich (API Level 15)

Παρουσιάστηκε στις 16 Δεκεμβρίου του 2011. Το Ice Cream Sandwich, έφερε πραγματικά τεράστια ανανέωση στον σχεδιασμό του Android με το Holo UI. Ήταν η πρώτη έκδοση του Android που έδωσε μοντέρνα χαρακτηριστικά στο λειτουργικό, που ακόμη και σήμερα συναντάμε πολλά στοιχεία του στις Android συσκευές.



1.3.8 Android 4.1 Jelly Bean (API Level 16)

Παρουσιάστηκε στις 9 Ιουλίου του 2012. Το Jelly Bean αποτελεί μεγάλη υπόθεση για πολλούς λόγους, ιδιαίτερα αν σκεφτούμε ότι κατέχει περίπου το 50% των Android συσκευών. Οι συνεισφορές του στο Android είναι πολλές, αλλά αυτή που ξεχώρισε και αναπτύσσεται μέχρι και σήμερα, είναι το Google Now. Ο ψηφιακός βοηθός της Google όχι μόνο παρέχει γρήγορες απαντήσεις για τα πάντα, αλλά και εμφανίζει νέα και ειδήσεις σύμφωνα με τα ενδιαφέροντα μας.



1.3.9 Android 4.4 KitKat (API Level 19)

Παρουσιάστηκε στις 31 Οκτωβρίου του 2013. Με το KitKat, η Google όχι απλά ενίσχυσε το UI, αλλά και ανέδειξε την δύναμη του λειτουργικού ακόμη και σε φθηνές συσκευές. Χάρη στις πολλές βελτιώσεις επιδόσεων που έγιναν στο KitKat, η Google υποστήριξε πως ακόμη και συσκευές με μόλις 512MB RAM θα είναι σε θέση να τρέξουν το KitKat απροβλημάτιστα. Αυτό ήταν μια αρκετά σοβαρή υπόθεση, δεδομένου ότι οι εταιρείες έδειχναν αδιαφορία στις οικονομικές συσκευές.



1.3.10 Android 5.0 Lollipop (API Level 21)

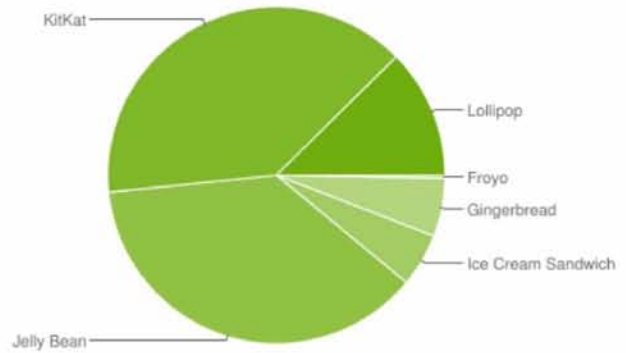
Παρουσιάστηκε στις 12 Νοεμβρίου του 2014. Το Lollipop να είναι η πιο ελκυστική έκδοση του Android που έχει υπάρξει μέχρι σήμερα. Με το Material Design της Google, η εμφάνιση του λειτουργικού έγινε επίπεδη, με περισσότερα χρώματα και φυσικά με πάρα πολλά εφέ κίνησης.



1.4 Χρήση κάθε έκδοσης του Android σήμερα

Σύμφωνα με τα τελευταία στοιχεία ο διαμερισμός των συσκευών που χρησιμοποιούν Android ανάλογα με την έκδοση που χρησιμοποιούν είναι ο εξής:

Version	Codename	API	Distribution
2.2	Froyo	8	0.3%
2.3.3 - 2.3.7	Gingerbread	10	5.6%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	5.1%
4.1.x	Jelly Bean	16	14.7%
4.2.x		17	17.5%
4.3		18	5.2%
4.4	KitKat	19	39.2%
5.0	Lollipop	21	11.6%
5.1		22	0.8%



Data collected during a 7-day period ending on June 1, 2015.

Any versions with less than 0.1% distribution are not shown.

2. Εργαλεία

2.1 Android Studio 1.2.2

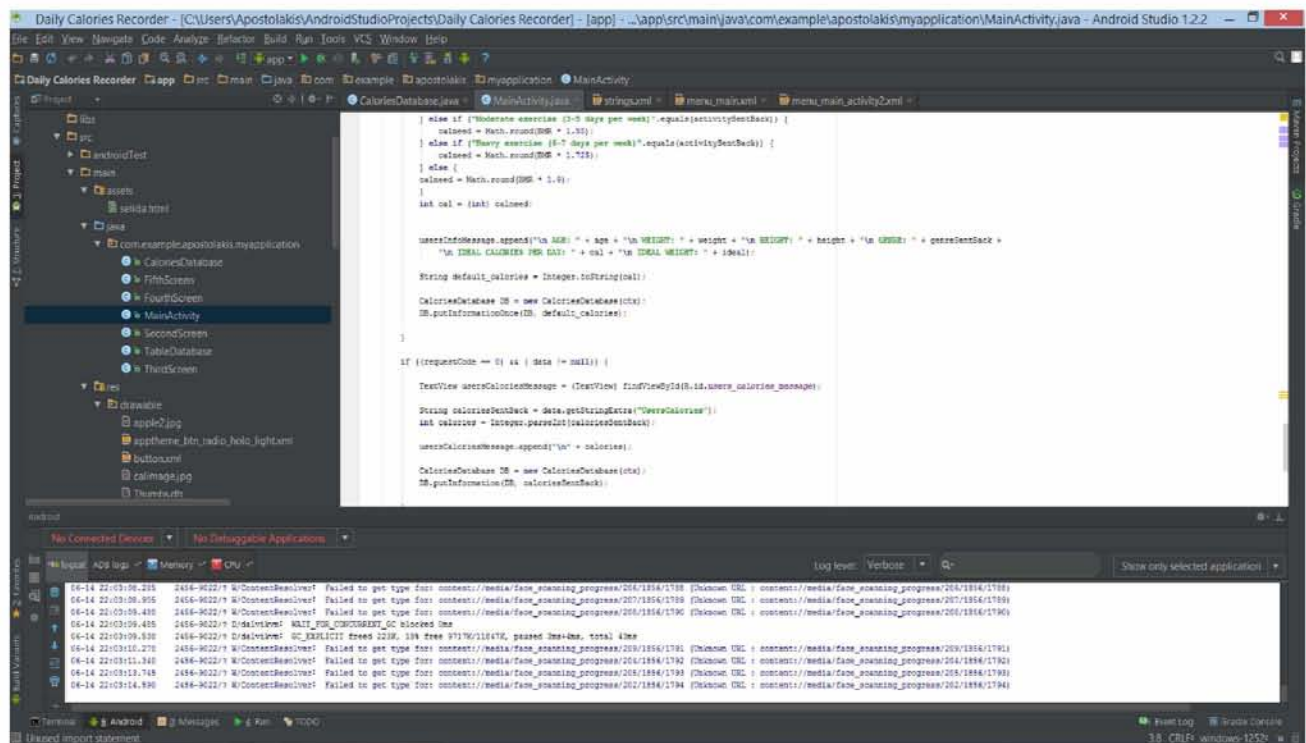
Το Android Studio ήταν το βασικότερο εργαλείο που χρησιμοποιήθηκε για την υλοποίηση της εφαρμογής. Όπως προαναφέρθηκε πρόκειται για ένα ολοκληρωμένο προγραμματιστικό περιβάλλον το οποίο προορίζεται για προγραμματισμό σε Android. Είναι βασισμένο στο λογισμικό της JetBrains' IntelliJ IDEA και είναι διαθέσιμο για Windows, Mac OS X, και Linux. Για να βγει το τελικό αποτέλεσμα της εφαρμογής έγινε συνδυασμός πολλών ειδών αρχείων.

Τα βασικότερα είναι τα αρχεία .java τα οποία είναι γραμμένα σε γλώσσα Java και είναι υπεύθυνα για το λειτουργικό κομμάτι της εφαρμογής. Σε αυτά έγιναν ενδεικτικά, η δημιουργία και η διαχείριση της βάσης δεδομένων καθώς και η υλοποίηση όλων των αλγορίθμων που

επεξεργάζονται τα στοιχεία που δίνει ο χρήστης προκειμένου να του παρουσιαστεί το κατάλληλο αποτέλεσμα.

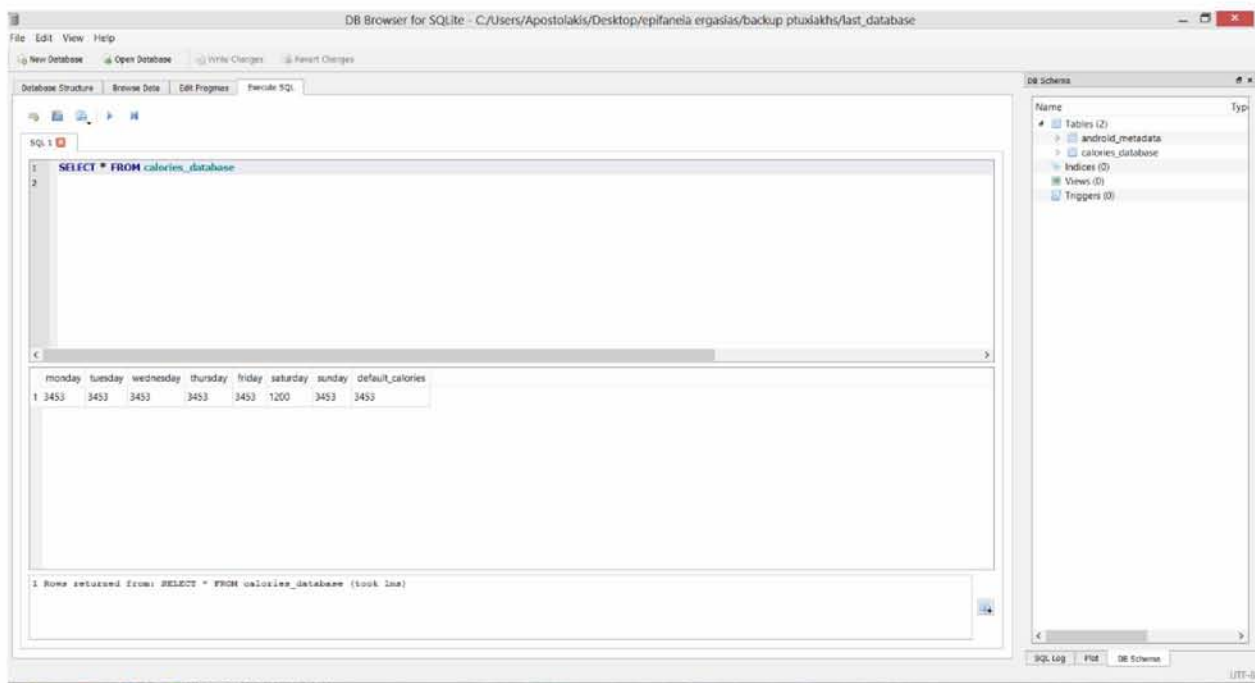
Πολύ βασικό ρόλο έχουν και τα αρχεία .xml τα οποία είναι γραμμένα σε γλώσσα XML (Extensible Markup Language). Αυτά τα αρχεία είναι υπεύθυνα για αυτό που βλέπει ο χρήστης αλλά και ο τρόπος με τον οποίο το βλέπει. Χαρακτηριστικό είναι ότι κάθε διαφορετική οθόνη που παρουσιάζεται στον χρήστη όταν αυτός επιλέγει μια λειτουργία, είναι ένα διαφορετικό αρχείο .xml, καθώς και ότι η δημιουργία των Buttons να εμφανίζονται με αυτόν τον τρόπο οφείλεται σε αυτήν.

Άλλα αρχεία που συνδυάζονται μέσω του Android Studio είναι τα γνωστά .jpg και .png τα οποία περιέχουν τις εικόνες που χρειάζονται ώστε να βλέπει αυτό το αποτέλεσμα ο χρήστης. Τέλος υπάρχει ένα αρχείο .html το οποίο είναι γραμμένο σε γλώσσα HTML (HyperText Markup Language) και σε αυτό είναι γραμμένα όλα τα τρόφιμα με τις θερμίδες, τις πρωτεΐνες, τα λίπη και τους υδατάνθρακες που τους αντιστοιχούν.



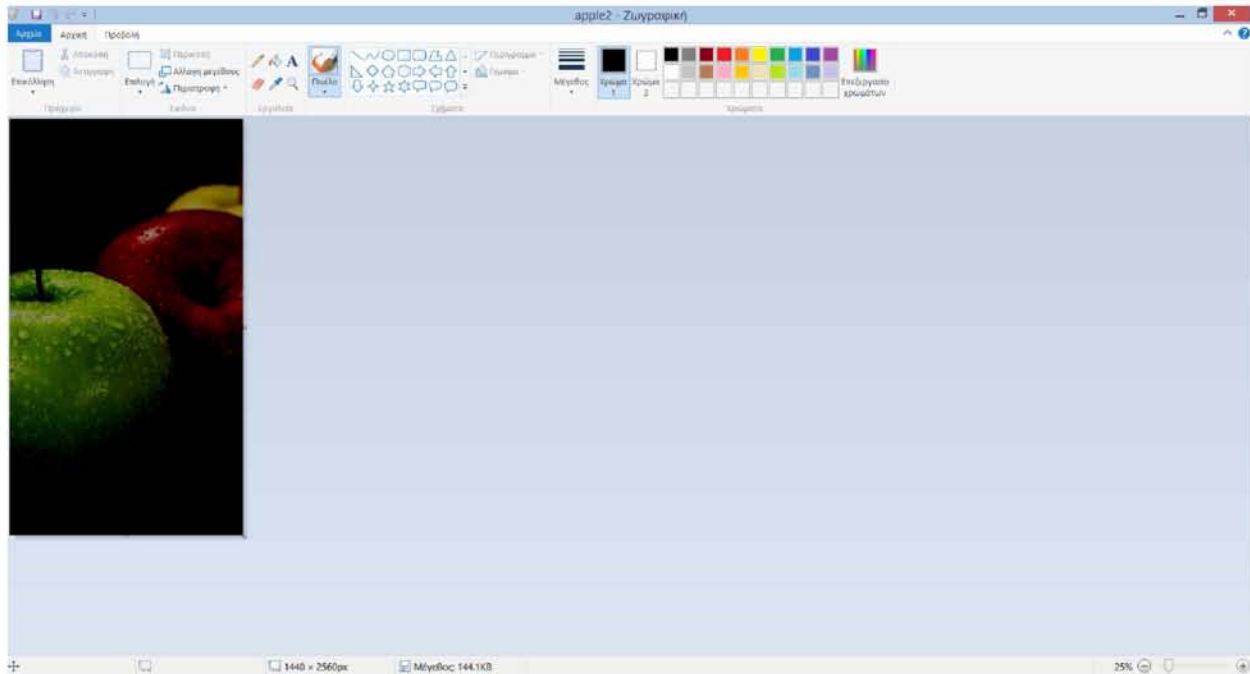
2.2 SQLite Database Browser

Το εργαλείο αυτό χρησιμοποιήθηκε για να ανοιχτούν τα αρχεία .db και να γίνει δυνατό να δούμε το περιεχόμενο της βάσης δεδομένων. Η βάση δεδομένων που δημιουργήθηκε είναι μια SQL βάση δεδομένων και σε αυτήν αποθηκεύονται, ανανεώνονται και ανακτούνται όποτε χρειάζεται τα στοιχεία με τις θερμίδες του χρήστη. Εκτελώντας εντολές της SQL στον SQLite Database Browser βλέπουμε τον πίνακα που έχει δημιουργηθεί και με αυτόν τον τρόπο επαληθεύτηκε η ορθή λειτουργία της εφαρμογής όσων αφορά την βάση δεδομένων.



2.3 Ζωγραφική και Microsoft Word

Πρόκειται για τα γνωστά προγράμματα που τρέχουν στα Windows. Στη Ζωγραφική έγινε η περικοπή για τις εικόνες που υπάρχουν σε αυτήν την διπλωματική. Επίσης εκεί ρυθμίστηκε η κατάλληλη φωτεινότητα για την εικόνα που είναι το background της εφαρμογής.



Το Microsoft Word χρησιμοποιήθηκε για τη συγγραφή της διπλωματικής και μέσω αυτού έγινε η εξαγωγή της σε pdf.



2.4 Συσκευές

Οι παρακάτω συσκευές χρησιμοποιήθηκαν ώστε να επαληθευτεί η ορθή λειτουργία της εφαρμογής. Είναι συσκευές με διαφορετικές εκδόσεις Android και διαφορετικό μέγεθος οθόνης.

Όνομασία	Έκδοση Android	Μέγεθος οθόνης
Samsung Galaxy S2	4.1.2	4.3"
Xiaomi Hongmi 1S	4.3	4.7"
LG Nexus 5	5.0	5"

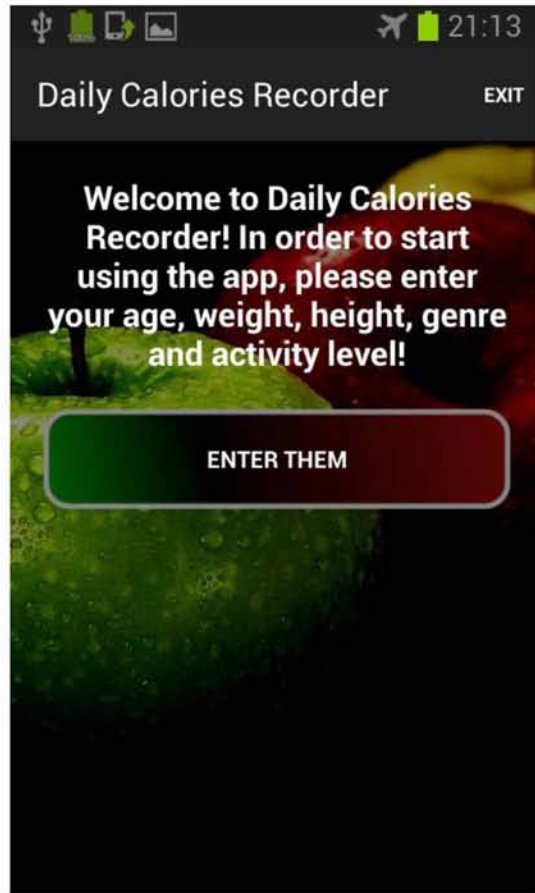
3. Επίδειξη της εφαρμογής

3.1 Η γενική φιλοσοφία

Όπως έχει προαναφερθεί, σκοπός της εφαρμογής είναι να βοηθήσει τον χρήστη να ελέγξει το σωματικό του βάρος και την διατροφή του μέσα από τις θερμίδες που παίρνει από τα τρόφιμα που καταναλώνει ημερησίως. Ο χρήστης καλείται να εισάγει τις θερμίδες που πήρε σε μια μέρα και αυτές καταχωρούνται στην αντίστοιχη μέρα στη βάση δεδομένων αυτόματα, ανάλογα με το ποια μέρα είναι εκείνη τη στιγμή. Αυτό πρέπει να γίνεται καθημερινά. Σε περίπτωση που ξέχασε να βάλει τις θερμίδες μιας μέρας υπάρχει επιλογή για την εισαγωγή στη χθεσινή μέρα. Οποιαδήποτε στιγμή θελήσει μπορεί να ελέγξει την πρόοδο του πατώντας την αντίστοιχη επιλογή. Εκεί θα μπορεί να δει τις μέρες της εβδομάδας καθώς και τις θερμίδες που πήρε σε κάθε μια από αυτές. Θα του παρουσιάζεται επίσης ένα μήνυμα για το αν και για το πόσες περισσότερες ή λιγότερες θερμίδες θα έπρεπε να πάρει προκειμένου να έχει μια ιδανική ημερήσια πρόσληψη. Τέλος θα παρέχεται η δυνατότητα να δει έναν πίνακα με όλα τα τρόφιμα και τις θερμίδες, τις πρωτεΐνες, τα λίπη και τους υδατάνθρακες που τους αντιστοιχούν.

3.2 Καλωσόρισμα και εισαγωγή στοιχείων

Όταν ο χρήστης ανοίξει για πρώτη φορά την εφαρμογή παρουσιάζεται ένα μήνυμα καλωσορίσματος και ενημερώνεται ότι πρέπει να εισάγει την ηλικία, το βάρος, το ύψος, το φύλο καθώς και την αθλητική του δραστηριότητα. Αυτό το layout δεν θα ξαναπαρουσιαστεί στον χρήστη καθώς είναι προγραμματισμένο έτσι ώστε να παρουσιάζεται μόνο στην πρώτη εκτέλεση της εφαρμογής.



Στη συνέχεια, και αφού ο χρήστης πατήσει ENTER THEM, παρουσιάζεται το layout στο οποίο εισάγει τα στοιχεία του. Η ηλικία εισάγεται μέσω του πληκτρολογίου που ανοίγει αυτόματα και ο χρήστης δεν έχει επιλογή παρά να εισάγει έναν ακέραιο αριθμό αφού για αυτό το πεδίο η υποδιαστολή είναι απενεργοποιημένη. Το βάρος και το ύψος εισάγονται και αυτά με το πληκτρολόγιο και μπορούν να πάρουν δεκαδικές τιμές. Το φύλο και η αθλητική δραστηριότητα επιλέγονται από το χρήστη με τη μορφή Radio Buttons. Σαν προεπιλογή έχει οριστεί στο φύλο το αρσενικό και στην αθλητική δραστηριότητα η μη άσκηση.

ψ 21:17

Enter your age:

Enter your weight in kg:

Enter your height in cm:

Select you genre:

Male

Female

Select your activity level:

Little to no exercise

Light exercise (1–3 days per week)

Moderate exercise (3–5 days per week)

Heavy exercise (6–7 days per week)

Very heavy exercise (twice per day, extra heavy workouts)

Προκειμένου να προχωρήσει ο χρήστης πρέπει να εισάγει τιμές σε όλα τα πεδία. Έτσι σε περίπτωση που πατήσει ENTER χωρίς να έχει συμπληρώσει κάποιο πεδίο, του παρουσιάζεται μήνυμα ώστε να το συμπληρώσει.

Enter your age:

Enter your weight in kg: **Age is required!**

Enter your height in cm:

Select your genre:

Male

Female

Select your activity level:

Little to no exercise

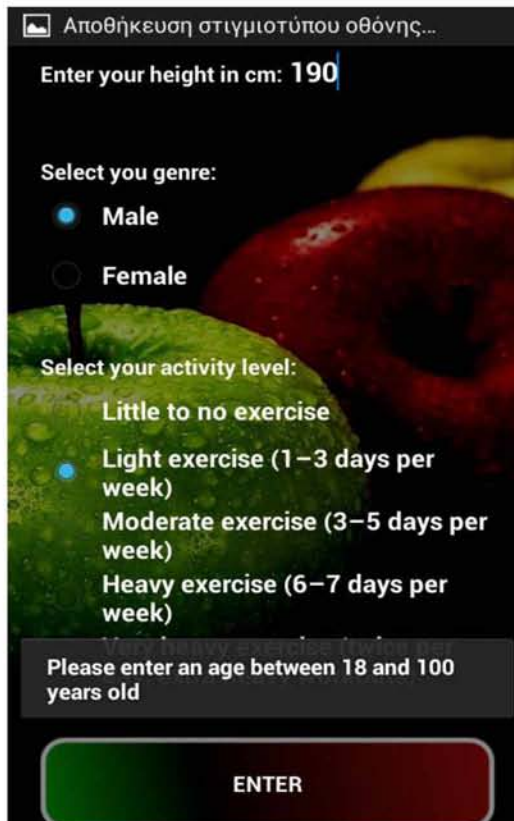
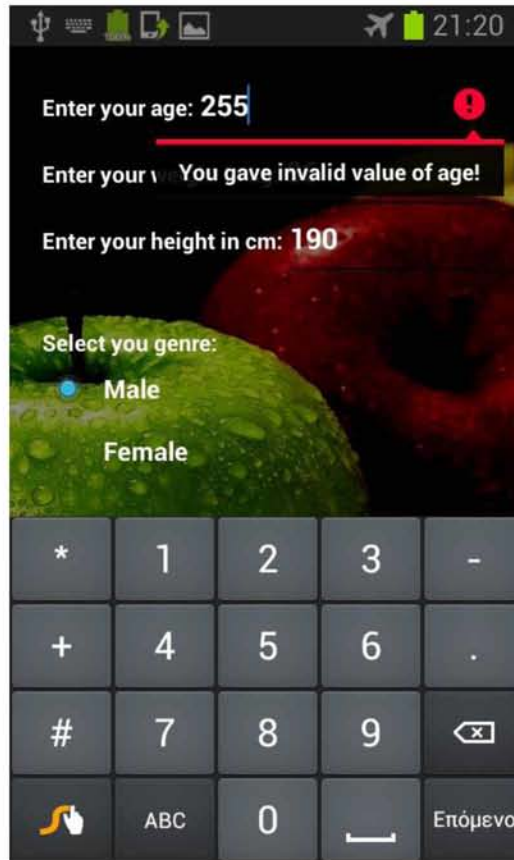
Light exercise (1–3 days per week)

Moderate exercise (3–5 days per week)

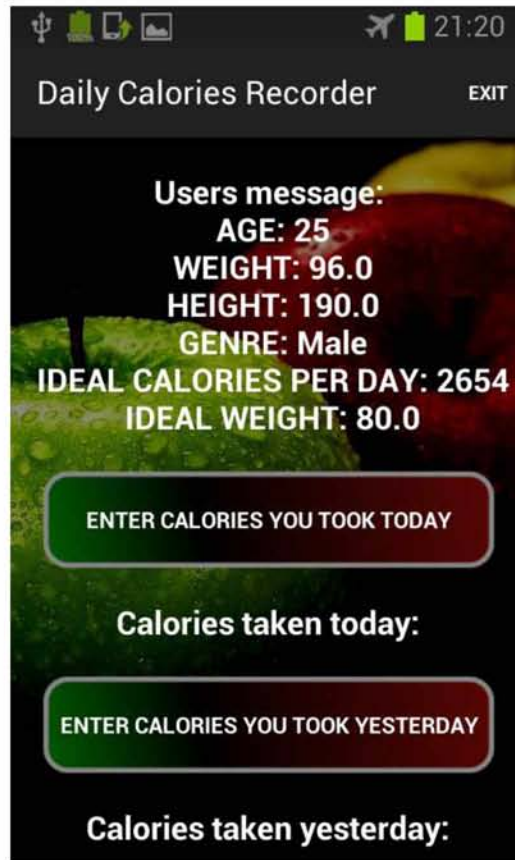
Heavy exercise (6–7 days per week)

Very heavy exercise (twice per week)

Υπάρχει επίσης έλεγχος ώστε να δίνονται έγκυρες τιμές από το χρήστη για να μπορεί να γίνει ορθή λειτουργία της εφαρμογής. Υπάρχουν έλεγχοι ώστε η ηλικία να είναι ανάμεσα σε 18 και 100 χρονών, το βάρος ανάμεσα σε 35 και 200 κιλά και το ύψος ανάμεσα σε 150 και 230 εκατοστά. Σε περίπτωση που γίνει παραβίαση αυτών των ορίων υπάρχει μήνυμα για μη έγκυρες τιμές και ο χρήστης ενημερώνεται για τα όρια μέσα στα οποία πρέπει να δώσει τιμή.

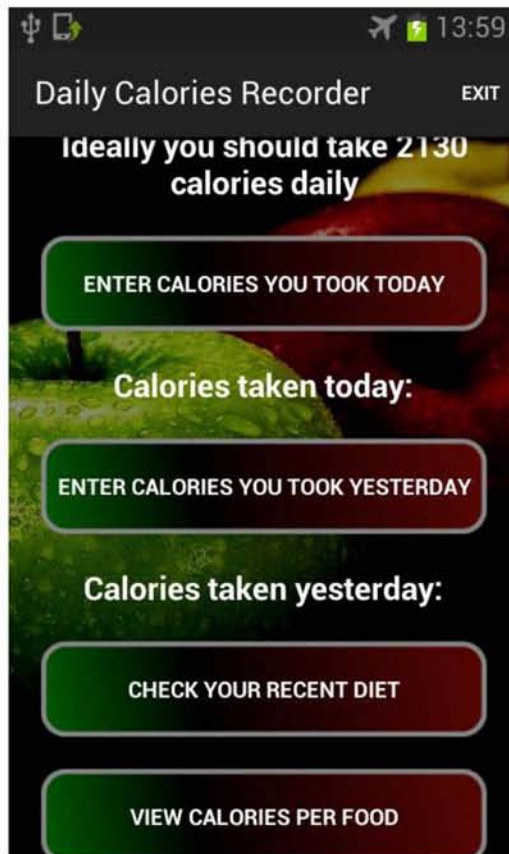


Όταν πια ο χρήστης εισάγει τα στοιχεία του με εγκυρότητα οδηγείται στο κύριο μενού της εφαρμογής στην κορυφή του οποίου υπάρχει ένα μήνυμα με τα στοιχεία που έδωσε, καθώς και με τις ιδανικές θερμίδες που πρέπει να παίρνει ημερησίως και το ιδανικό βάρος που πρέπει να έχει. Ο τρόπος που υπολογίστηκαν αυτά θα επεξηγηθεί παρακάτω.



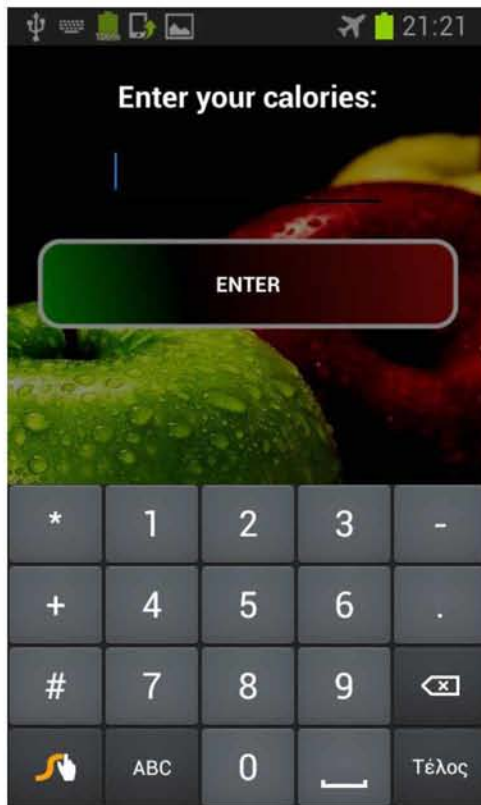
3.3 Κύριο μενού της εφαρμογής

Κάθε φορά που ο χρήστης ανοίγει την εφαρμογή του παρουσιάζεται ένα μενού στο οποίο μπορεί να κάνει διάφορες ενέργειες. Στην κορυφή υπάρχει ένα μήνυμα που τον ενημερώνει πόσες είναι οι ιδανικές θερμίδες που πρέπει να παίρνει ημερησίως. Στη συνέχεια μπορεί να επιλέξει να εισάγει πόσες θερμίδες πήρε σήμερα, ή χθες αν είχε ξεχάσει να βάλει. Μετά υπάρχει η επιλογή για να ελέγξει την πρόοδο του καθώς και να δει τον πίνακα με τα τρόφιμα και τις θερμίδες που τους αντιστοιχούν.

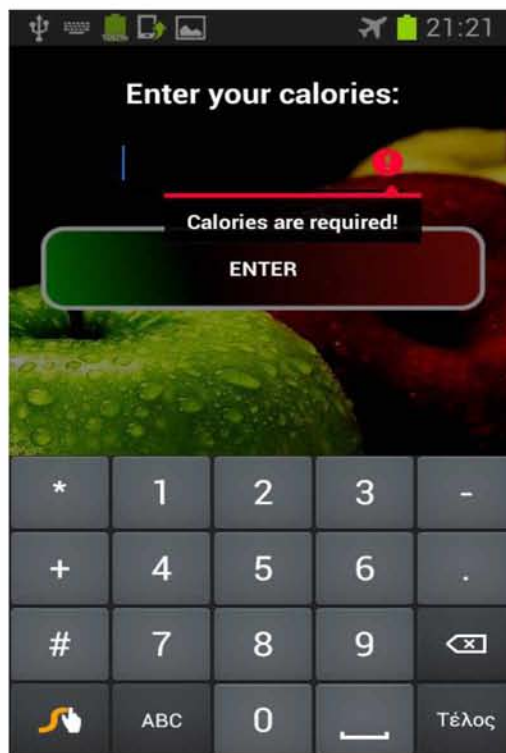


3.3.1 ENTER CALORIES

Επιλέγοντας το ENTER CALORIES YOU TOOK TODAY ή ENTER CALORIES YOU TOOK YESTERDAY ο χρήστης οδηγείται σε ένα άλλο layout στο οποίο εισάγει τις θερμίδες που πήρε.



Και εδώ ο χρήστης «αναγκάζεται» να δώσει κάποια τιμή καθώς υπάρχει έλεγχος που δεν τον αφήνει να συνεχίσει αν δεν το κάνει.



3.3.2 CHECK RECENT DIET

Επιλέγοντας CHECK YOUR RECENT DIET ο χρήστης οδηγείται ξανά σε ένα καινούριο layout στο οποίο του παρουσιάζονται όπως προαναφέρθηκε οι μέρες της εβδομάδας με τις αντίστοιχες θερμίδες και τα κατάλληλα μηνύματα ανάλογα με την περίπτωση του.

Σε περίπτωση που έχει πάρει περισσότερες θερμίδες από όσες θα έπρεπε, βλέπει αυτό:



Σε περίπτωση που έχει πάρει λιγότερες θερμίδες από όσες θα έπρεπε, βλέπει αυτό:



Και στη σπάνια περίπτωση που έχει πάρει ακριβώς τις θερμίδες που πρέπει, βλέπει αυτό:



3.3.3 CALORIES PER FOOD

Τελευταία επιλογή που παρουσιάζεται στον χρήστη είναι το VIEW CALORIES PER FOOD. Όταν ο χρήστης το επιλέξει οδηγείται σε ένα layout στο οποίο υπάρχουν τα τρόφιμα με τις θερμίδες, τις πρωτεΐνες, τα λίπη και τους υδατάνθρακες που τους αντιστοιχούν. Αυτό μπορεί να φανεί πολύ χρήσιμο σε περίπτωση που ο χρήστης έχει καταναλώσει ένα μη συσκευασμένο προϊόν το οποίο δεν αναγράφει τις θερμίδες που έχει στη συσκευασία. Ο πίνακας είναι αρκετά αναλυτικός και περιέχει τις θερμίδες για διάφορες δοσολογίες του ίδιου τροφίμου ώστε να υπάρχει η μεγαλύτερη δυνατή ακρίβεια στην καταμέτρηση.



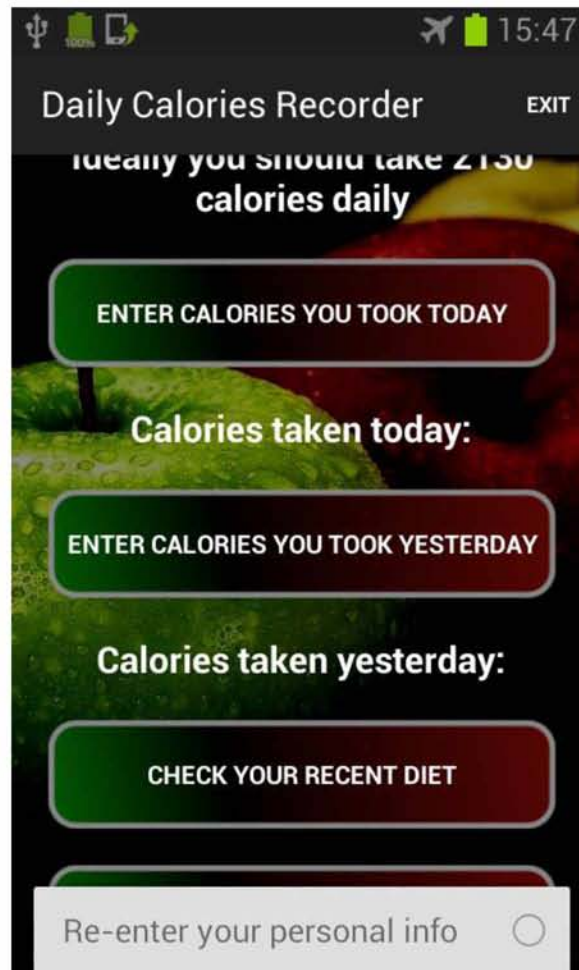
The screenshot shows a mobile application interface with a status bar at the top displaying signal strength, airplane mode, and the time 21:26. Below the status bar is a table with two columns: 'Food Description and Portion Size' and 'Calories (kcal)'. The table lists various food items with their corresponding calorie counts.

Food Description and Portion Size	Calories (kcal)
100% NATURAL CEREAL 1 OZ	135
1000 ISLAND; SALAD DRNG;LOCAL 1 TBSP	25
1000 ISLAND; SALAD DRNG;REGLR1 TBSP	60
40% BRAN FLAKES; KELLOGG'S 1 OZ	90
40% BRAN FLAKES; POST 1 OZ	90
ALFALFA SEEDS; SPROUTED; RAW 1 CUP	10
ALL-BRAN CEREAL 1 OZ	70
ALMONDS; SLIVERED 1 CUP	795
ALMONDS; WHOLE 1 OZ	165
ANGELFOOD CAKE; FROM MIX 1 CAKE	1510
ANGELFOOD CAKE; FROM MIX 1 PIECE	125
APPLE JUICE; CANNED 1 CUP	115
APPLE PIE 1 PIE	2420
APPLE PIE 1 PIECE	405
APPLES; DRIED; SULFURED 10 RINGS	155
APPLES; RAW; PEELED; SLICED 1 CUP	65
APPLES; RAW; UNPEELED;2 PER LB1 APPLE	125
APPLES; RAW; UNPEELED;3 PER LB1 APPLE	80
APPLESAUCE; CANNED; SWEETENED 1 CUP	195

Food Description and Portion Size	Calories (kcal)	Protein (g)	Fat (g)	Carbohydrate (g)
100% NATURAL CEREAL 1 OZ	135	3	6	18
1000 ISLAND; SALAD DRNG;LOCAL 1 TBSP	25	0	2	2
1000 ISLAND; SALAD DRNG;REGLR1 TBSP	60	0	6	2
40% BRAN FLAKES; KELLOGG'S 1 OZ	90	4	1	22
40% BRAN FLAKES; POST 1 OZ	90	3	0	22
ALFALFA SEEDS; SPROUTED; RAW 1 CUP	10	1	0	1
ALL-BRAN CEREAL 1 OZ	70	4	1	21
ALMONDS; SLIVERED 1 CUP	795	27	70	28
ALMONDS; WHOLE 1 OZ	165	6	15	6
ANGELFOOD CAKE; FROM MIX 1 CAKE	1510	38	2	342
ANGELFOOD CAKE; FROM MIX 1 PIECE	125	2	0	20

3.4 Έξοδος και επανεισαγωγή στοιχείων

Ο χρήστης μπορεί όποτε θελήσει να ξανά εισάγει τα προσωπικά του στοιχεία σε περίπτωση που αυτά έχουν αλλάξει. Έτσι μπορεί να έχει μια πιο έγκυρη πληροφόρηση για την πορεία της διατροφής του. Αυτό μπορεί να το κάνει πατώντας το πλήκτρο Menu του κινητού. Θα αναδυθεί η επιλογή Re-enter your personal info και επιλέγοντας την υπάρχει ανακατεύθυνση το layout εισαγωγής των στοιχείων που είδαμε στην αρχή.



Η έξοδος από την εφαρμογή μπορεί να γίνει πατώντας στην επιλογή EXIT που υπάρχει.

4. Αλγόριθμοι για υπολογισμούς

4.1 Υπολογισμός απαραίτητων θερμίδων

Ο αλγόριθμος που χρησιμοποιήθηκε προκειμένου να βγαίνουν τα κατάλληλα συμπεράσματα λαμβάνει υπόψη την ηλικία, το ύψος, το φύλο αλλά και την αθλητική δραστηριότητα του χρήστη. Αρχικά γίνεται με τη χρήση της φόρμουλας του J. D. Robinson ο υπολογισμός για το ιδανικό βάρος σε κιλά.

- Άνδρας: $52 + 1,9 * (\text{Υψος σε cm} - 152,4) * 0,39$
- Γυναίκα: $49 + 1,7 * (\text{Υψος σε cm} - 152,4) * 0,39$

Στη συνέχεια αυτό το ιδανικό βάρος χρησιμοποιείται για να βρεθεί ο Βασικός Μεταβολικός Ρυθμός (BMR) σύμφωνα με τις εξισώσεις των Harris-Benedict οι οποίες έχουν βελτιωθεί από τους Rosa και Shizgal αργότερα. Οι εξισώσεις αυτές είναι:

- BMR άνδρα: $88,362 + (13,397 * \text{Ιδανικό βάρος σε kg}) + (4,799 * \text{Υψος σε cm}) - (5,677 * \text{Ηλικία σε χρόνια})$
- BMR γυναίκας: $447,593 + (9,247 * \text{Ιδανικό βάρος σε kg}) + (3,098 * \text{Υψος σε cm}) - (4,33 * \text{Ηλικία σε χρόνια})$

Προκειμένου τώρα να γίνει ο υπολογισμός των ιδανικών θερμίδων που πρέπει να παίρνει ο χρήστης είναι απαραίτητο να ληφθεί υπόψη η αθλητική του δραστηριότητα, καθώς αυτή παίζει βασικό ρόλο στις θερμίδες που «καίει» ο οργανισμός. Οι επιλογές που υπάρχουν είναι πέντε και για να βγει ο αριθμός των θερμίδων σε κάθε περίπτωση πολλαπλασιάζεται ο BMR με έναν συντελεστή. Έτσι έχουμε:

- Λίγη έως καθόλου εξάσκηση: Κατάλληλες θερμίδες = $BMR * 1,2$
- Ελαφριά εξάσκηση (μια έως τρεις φορές την εβδομάδα): Κατάλληλες θερμίδες = $BMR * 1,375$
- Μέτρια εξάσκηση (τρεις έως πέντε φορές την εβδομάδα): Κατάλληλες θερμίδες = $BMR * 1,55$
- Βαριά εξάσκηση (έξι έως επτά φορές την εβδομάδα): Κατάλληλες θερμίδες = $BMR * 1,725$
- Πολύ βαριά εξάσκηση (δυο φορές την ημέρα): Κατάλληλες θερμίδες = $BMR * 1,9$

4.2 Υπολογισμός για πρόοδο

Όπως έχει προαναφερθεί ο χρήστης μπορεί όποτε θελήσει να ελέγξει την πρόοδο του όσων αφορά τις θερμίδες που έχει πάρει τις τελευταίες επτά ημέρες. Ανάλογα την περίπτωση, του παρουσιάζονται και τα ανάλογα μηνύματα, αν έχει δηλαδή φάει περισσότερο, λιγότερο ή ακριβώς όσο θα έπρεπε.

Το συμπέρασμα αυτό βγαίνει συγκρίνοντας τον Μ.Ο. των θερμίδων που έχει πάρει ο χρήστης την τελευταία εβδομάδα, με τον αριθμό των θερμίδων που πρέπει να παίρνει ιδανικά. Αν ο Μ.Ο. είναι μεγαλύτερος, τότε ενημερώνεται ότι έχει πάρει περισσότερες από όσες θα έπρεπε και αν είναι μικρότερος ότι έχει πάρει λιγότερες. Στην σπάνια περίπτωση που υπάρχει ισότητα βγαίνει και ανάλογο μήνυμα.

Υπάρχει επίσης πληροφόρηση για το κατά πόσες θερμίδες πρέπει να παίρνει παραπάνω ή παρακάτω κατά μέσο όρο ανά ημέρα ώστε να έχει ιδανική ημερήσια πρόσληψη θερμίδων. Ο αριθμός αυτός προκύπτει εύκολα βρίσκοντας την απόλυτη διαφορά του Μ.Ο των θερμίδων της τελευταίας εβδομάδας με τις ιδανικές θερμίδες που έχουμε υπολογίσει ότι πρέπει να παίρνονται.

5. Αρχεία .java

Όλα τα αρχεία .java βρίσκονται μέσα στον φάκελο java της εφαρμογής και πιο συγκεκριμένα μέσα στον υποφάκελο com.example.apostolakis.myapplication.

5.1 MainActivity.java

```
package com.example.apostolakis.myapplication;

import android.content.Context;
import android.content.SharedPreferences;
import android.database.Cursor;
import android.preference.PreferenceManager;
import android.support.v7.app.ActionBarActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.TextView;
import android.content.Intent;

import java.io.File;

public class MainActivity extends ActionBarActivity {

    Context ctx = this;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        String deflt;
        SharedPreferences prefs = PreferenceManager.getDefaultSharedPreferences(this);
        if(!prefs.getBoolean("firstTime", false)) {
            setContentView(R.layout.activity_main);
            SharedPreferences.Editor editor = prefs.edit();
            editor.putBoolean("firstTime", true);
            editor.apply();
        }
        else{
            File dbtest = getApplicationContext().getDatabasePath("my_database");
            if (!dbtest.exists()) {
                setContentView(R.layout.activity_main);
            }
            else{
                setContentView(R.layout.fourth_layout);
                CaloriesDatabase DB2 = new CaloriesDatabase(ctx);
                Cursor CR = DB2.getInformation(DB2);
                CR.moveToFirst();
                do {
                    deflt = CR.getString(7);

                } while (CR.moveToNext());

                int dflt = Integer.parseInt(deflt);
                TextView usersInfoMessage = (TextView) findViewById(R.id.users_info_message);

                usersInfoMessage.append("\nIdeally you should take " + dflt + " calories daily");
            }
        }
    }

    @Override
    public void onBackPressed()
    {

        Intent intent = new Intent(Intent.ACTION_MAIN);
```



```

        intent.addCategory(Intent.CATEGORY_HOME);
        intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        startActivity(intent);
        super.onBackPressed();
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.menu_main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {

        int id = item.getItemId();

        if (id == R.id.action_settings) {

            Intent getNameScreenIntent = new Intent(this, SecondScreen.class);

            startActivityForResult(getNameScreenIntent, 1);

            return true;
        } else if (id == R.id.exit_the_app) {

            Intent homeIntent = new Intent(Intent.ACTION_MAIN);
            homeIntent.addCategory(Intent.CATEGORY_HOME);
            homeIntent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
            startActivity(homeIntent);
            finish();
            return true;

        }

        return super.onOptionsItemSelected(item);
    }

    public void onGetNameClick (View view) {

        Intent getNameScreenIntent = new Intent(this, SecondScreen.class);

        startActivityForResult(getNameScreenIntent, 1);
    }

    public void onGetCaloriesClick(View view) {

        Intent getNameScreenIntent = new Intent(this, ThirdScreen.class);

        startActivityForResult(getNameScreenIntent, 0);
    }

    public void onGetYesterdayCaloriesClick(View view) {

        Intent getNameScreenIntent = new Intent(this, ThirdScreen.class);

        startActivityForResult(getNameScreenIntent, 2);
    }

    public void onGetResultClick(View view) {

        Intent myIntent = new Intent(this, FourthScreen.class);

        startActivity(myIntent);
    }

    public void onGetWebClick(View view) {

        Intent myIntent = new Intent(this, FifthScreen.class);

```

```

        startActivity(myIntent);
    }

    @Override
    protected void onActivityResult(int requestCode, int resultCode, Intent data) {
        super.onActivityResult(requestCode, resultCode, data);

        if ((requestCode == 1) && (data != null)){
            setContentView(R.layout.fourth_layout);

            TextView usersInfoMessage = (TextView) findViewById(R.id.users_info_message);

            String ageSentBack = data.getStringExtra("UsersAge");
            String weightSentBack = data.getStringExtra("UsersWeight");
            String heightSentBack = data.getStringExtra("UsersHeight");
            String genreSentBack = data.getStringExtra("UsersGenre");
            String activitySentBack = data.getStringExtra("UsersActivity");

            int age = Integer.parseInt(ageSentBack);

            float weight = Float.parseFloat(weightSentBack);

            float height = Float.parseFloat(heightSentBack);

            double BMR = 0;

            double calneed = 0;

            double ideal = 0;

            if ("Male".equals(genreSentBack)) {
                //BMR = (13.397 * weight) + (4.799 * height) - (5.677 * age) + 88.362; In case i want later to
                compute calories for current weight
                ideal = Math.round(52+1.9*(height-152.4)*0.39);
                BMR = (13.397 * ideal) + (4.799 * height) - (5.677 * age) + 88.362;
            } else {
                //BMR = (9.247 * weight) + (3.098 * height) - (4.330 * age) + 447.593; In case i want later to
                compute calories for current weight
                ideal = Math.round(49+1.7*(height-152.4)*0.39);
                BMR = (9.247 * ideal) + (3.098 * height) - (4.330 * age) + 447.593;
            }

            if ("Little to no exercise".equals(activitySentBack)) {
                calneed = Math.round(BMR * 1.2);
            } else if ("Light exercise (1-3 days per week)".equals(activitySentBack)) {
                calneed = Math.round(BMR * 1.375);
            } else if ("Moderate exercise (3-5 days per week)".equals(activitySentBack)) {
                calneed = Math.round(BMR * 1.55);
            } else if ("Heavy exercise (6-7 days per week)".equals(activitySentBack)) {
                calneed = Math.round(BMR * 1.725);
            } else {
                calneed = Math.round(BMR * 1.9);
            }
            int cal = (int) calneed;

            usersInfoMessage.append("\n AGE: " + age + "\n WEIGHT: " + weight + "\n HEIGHT: " + height + "\n
GENRE: " + genreSentBack +
                "\n IDEAL CALORIES PER DAY: " + cal + "\n IDEAL WEIGHT: " + ideal);

            String default_calories = Integer.toString(cal);

            CaloriesDatabase DB = new CaloriesDatabase(ctx);
            DB.putInformationOnce(DB, default_calories);
        }

        if ((requestCode == 0) && (data != null)) {
            TextView usersCaloriesMessage = (TextView) findViewById(R.id.users_calories_message);

```

```

String caloriesSentBack = data.getStringExtra("UsersCalories");
int calories = Integer.parseInt(caloriesSentBack);

usersCaloriesMessage.append("\n" + calories);

CaloriesDatabase DB = new CaloriesDatabase(ctx);
DB.putInformation(DB, caloriesSentBack);
}

if ((requestCode == 2) && (data != null)) {
    TextView usersCaloriesMessage = (TextView) findViewById(R.id.users_yesterday_calories_message);

    String caloriesSentBack = data.getStringExtra("UsersCalories");
    int calories = Integer.parseInt(caloriesSentBack);

    usersCaloriesMessage.append("\n" + calories);

    CaloriesDatabase DB = new CaloriesDatabase(ctx);
    DB.putYesterdayInformation(DB, caloriesSentBack);
}
}
}

```

Στο αρχείο MainActivity.java γίνονται οι σημαντικότερες ενέργειες για την λειτουργία της εφαρμογής καθώς είναι ο πρώτος κώδικας που τρέχει όταν ξεκινάει η εφαρμογή. Αρχικά, όπως σε όλα τα αρχεία .java, γίνονται τα απαραίτητα imports των βιβλιοθηκών που χρειάζονται για τις διάφορες συναρτήσεις που καλούνται. Στη συνάρτηση onCreate με τη χρήση των SharedPreferences αποφασίζεται ποιο layout θα προβληθεί στον χρήστη, ανάλογα με το αν είναι το πρώτο τρέξιμο της εφαρμογής ή όχι. Η συνάρτηση onBackPressed είναι υπεύθυνη για τις ενέργειες που γίνονται σε περίπτωση που ο χρήστης πατήσει το πίσω πλήκτρο από το κινητό του. Η onCreateOptionsMenu δημιουργεί τις επιλογές του μενού της εφαρμογής και η onOptionsItemSelected αποφασίζει τι θα γίνει αν ο χρήστης επιλέξει την επιλογή Exit ή Re-enter your personal info. Στη συνέχεια υπάρχουν οι getNameClick, getCaloriesClick και getYesterdayCaloriesClick, οι οποίες κάθε μια ξεχωριστά ενεργοποιεί ένα Activity μέσω της startActivityForResult. Τα αποτελέσματα αυτών των Activities συλλέγονται στην συνάρτηση onActivityResult ανάλογα με το requestCode τους που μπορεί να είναι 0, 1 ή 2. Αν είναι 0 ή 2 περνάει τις θερμίδες στη βάση δεδομένων στις κατάλληλες μέρες ανα περίπτωση, ενώ αν είναι 1 σημαίνει ότι ο χρήστης έχει εισάγει τα στοιχεία του και έτσι γίνονται οι απαραίτητοι οι υπολογισμοί για να γίνει η αρχικοποίηση της βάσης δεδομένων. Τέλος η συνάρτηση getResultClick ενεργοποιεί το Activity για να δει ο χρήστης την πρόοδο του και η getWebClick για να δει τα τρόφιμα με τις θερμίδες και τα λοιπά στοιχεία τους.

5.2 SecondScreen.java

```
package com.example.apostolakis.myapplication;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.Toast;
import android.widget.RadioGroup;

/**
 * Created by Apostolakis on 24/3/2015.
 */
public class SecondScreen extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.second_layout);
    }

    @Override
    public void onBackPressed()
    {
        Intent intent = new Intent(SecondScreen.this, MainActivity.class);
        startActivity(intent);
        super.onBackPressed();
    }

    public void onSendUsersName(View view) {

        EditText usersAgeET = (EditText) findViewById(R.id.users_age_edit_text);
        String usersAge = "NULL";
        String usersWeight = "NULL";
        String usersHeight = "NULL";

        RadioGroup radioSexGroup;
        RadioButton radioSexButton;

        radioSexGroup = (RadioGroup) findViewById(R.id.radioSex);
        int selectedSex = radioSexGroup.getCheckedRadioButtonId();

        radioSexButton = (RadioButton) findViewById(selectedSex);

        RadioGroup radioActivityGroup;
        RadioButton radioActivityButton;

        radioActivityGroup = (RadioGroup) findViewById(R.id.radioActivity);
        int selectedActivity = radioActivityGroup.getCheckedRadioButtonId();

        radioActivityButton = (RadioButton) findViewById(selectedActivity);

        String usersGenre = String.valueOf(radioSexButton.getText());
        String usersActivity = String.valueOf(radioActivityButton.getText());

        if(usersAgeET.getText().length() == 0){
            usersAgeET.setError( "Age is required!" );
            Toast toast = Toast.makeText(this, "Enter your age", Toast.LENGTH_LONG);
        }
    }
}
```


γίνονται επίσης μέσω των πολλών εμφωλευμένων `if_else` οι έλεγχοι για να μην αφήσει κάποιο πεδίο κενό ο χρήστης, αλλά και για να δώσει έγκυρες τιμές.

5.3 ThirdScreen.java

```
package com.example.apostolakis.myapplication;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

/**
 * Created by Apostolakis on 4/5/2015.
 */
public class ThirdScreen extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        setContentView(R.layout.third_layout);

    }

    @Override
    public void onBackPressed() {
        Intent intent = new Intent(ThirdScreen.this, MainActivity.class);
        startActivity(intent);
        super.onBackPressed();
    }

    public void onSendUsersCalories(View view) {

        String usersCalories = "NULL";

        EditText usersCaloriesET = (EditText) findViewById(R.id.users_calories_edit_text);

        if(usersCaloriesET.getText().length() == 0) {

            usersCaloriesET.setError( "Calories are required!" );
            Toast toast = Toast.makeText(this, "Enter your calories", Toast.LENGTH_SHORT);
            toast.show();

        } else {

            usersCalories = String.valueOf(usersCaloriesET.getText());

            Intent intent = new Intent(getApplicationContext(), MainActivity.class);
            intent.putExtra("UsersCalories", usersCalories);

            setResult(RESULT_OK, intent);

            super.onBackPressed();

        }

    }

}
```

Στο αρχείο ThirdScreen.java εκτελείται ο κώδικας για να πάρουμε τις θερμίδες που έδωσε ο χρήστης. Αρχικά στην onCreate δίνεται η εντολή να προβληθεί στο χρήστη το third_layout στο οποίο καλείται να εισάγει τις θερμίδες του. Στην onSendUsersCalories παίρνουμε τις θερμίδες και τις βάζουμε στο Bundle ώστε να σταλούν και αυτές στην MainActivity. Και εδώ υπάρχει έλεγχος για να μην αφήσει ο χρήστης το πεδίο κενό.

5.4 FourthScreen.java

```
package com.example.apostolakis.myapplication;

import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.os.Bundle;
import android.widget.TextView;

/**
 * Created by Apostolakis on 18/5/2015.
 */
public class FourthScreen extends Activity {
    Context ctx2 = this;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        setContentView(R.layout.fifth_layout);

        CaloriesDatabase DB2 = new CaloriesDatabase(ctx2);
        Cursor CR = DB2.getInformation(DB2);
        CR.moveToFirst();
        TextView usersResultMessage = (TextView) findViewById(R.id.users_result_message);
        String monday, tuesday, wednesday, thursday, friday, saturday, sunday, default;

        do {
            monday = CR.getString(0);
            tuesday = CR.getString(1);
            wednesday = CR.getString(2);
            thursday = CR.getString(3);
            friday = CR.getString(4);
            saturday = CR.getString(5);
            sunday = CR.getString(6);
            default = CR.getString(7);

        }while (CR.moveToNext());

        int mon = Integer.parseInt(monday);
        int tue = Integer.parseInt(tuesday);
        int wed = Integer.parseInt(wednesday);
        int thu = Integer.parseInt(thursday);
        int fri = Integer.parseInt(friday);
        int sat = Integer.parseInt(saturday);
        int sun = Integer.parseInt(sunday);
        int dflt = Integer.parseInt(default);

        int sum = mon + tue + wed + thu + fri + sat + sun;

        sum = Math.round(sum/7);
        int diff = Math.abs(sum - dflt);
        if (sum>dflt) {
            usersResultMessage.append("\n Monday: " + monday + " calories\n Tuesday: " + tuesday +
                " calories\n Wednesday: " + wednesday + " calories\n Thursday: " + thursday +
                " calories\n Friday: " + friday + " calories\n Saturday: " + saturday + " calories\n
Sunday: " + sunday +
                " calories\n\n The past seven days you have eaten MORE than you should.\n\n" +
```



```

        "If you want to have an ideal weight you have to take " + diff + " less calories daily.");
    }
    else if (sum<dflt){
        usersResultMessage.append("\n Monday: " + monday + " calories\n Tuesday: " + tuesday +
            " calories\n Wednesday: " + wednesday + " calories\n Thursday: " + thursday +
            " calories\n Friday: " + friday + " calories\n Saturday: " + saturday + " calories\n
Sunday: " + sunday +
            " calories\n\n The past seven days you have eaten LESS than you should.\n\n" +
            "If you want to have an ideal weight you have to take " + diff + " more calories daily.");
    }
    else{
        usersResultMessage.append("\n Monday: " + monday + " calories\n Tuesday: " + tuesday +
            " calories\n Wednesday: " + wednesday + " calories\n Thursday: " + thursday +
            " calories\n Friday: " + friday + " calories\n Saturday: " + saturday + " calories\n
Sunday: " + sunday +
            " calories\n\n The past seven days you have taken exactly the right calories in order to
keep your weight.");
    }
}

@Override
public void onBackPressed() {

    Intent intent = new Intent(FourthScreen.this, MainActivity.class);
    startActivity(intent);
    super.onBackPressed();
}
}
}

```

Στο αρχείο FourthScreen.java γίνονται οι υπολογισμοί για την προβολή στον χρήστη του κατάλληλου μηνύματος ανάλογα με τις θερμίδες που έχει πάρει. Όλη η δουλειά γίνεται στην onCreate μέσω της οποίας αρχικά ορίζεται το fifth_layout για να προβληθεί στον χρήστη. Στη συνέχεια διαβάζονται από τη βάση δεδομένων οι θερμίδες που έχει αποθηκεύσει ο χρήστης και εκτελείται ο αλγόριθμος για την πρόοδο του χρήστη που έχει περιγραφεί παραπάνω. Ανάλογα με την περίπτωση προβάλλεται και το ανάλογο μήνυμα.

5.5 FifthScreen.java

```

package com.example.apostolakis.myapplication;

import android.app.Activity;
import android.content.Intent;
import android.graphics.Color;
import android.os.Bundle;
import android.webkit.WebView;

/**
 * Created by Apostolakis on 25/5/2015.
 */
public class FifthScreen extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
    }
}

```



```

        setContentView(R.layout.web_layout);

        WebView webView = (WebView) findViewById(R.id.webview);
        webView.setPadding(0, 0, 0, 0);
        webView.setInitialScale(125);
        webView.setBackgroundColor(Color.TRANSPARENT);
        webView.loadUrl("file:///android_asset/selida.html");
    }

    @Override
    public void onBackPressed() {

        Intent intent = new Intent(FifthScreen.this, MainActivity.class);
        startActivity(intent);
        super.onBackPressed();
    }
}

```

Στο αρχείο FifthScreen.java γίνεται η προβολή του web_layout στο οποίο ο χρήστης βλέπει τον πίνακα με τα τρόφιμα με τις θερμίδες, τις πρωτεΐνες, τα λίπη και τους υδατάνθρακες που τους αντιστοιχούν. Γίνεται χρήση του WebView καθώς αυτός ο πίνακας είναι σε αρχείο .html το οποίο προβάλλεται μέσω του loadUrl. Απαραίτητα για τη σωστή και όμορφη προβολή του πίνακα είναι το padding και το scale που γίνεται καθώς και ότι το background ορίζεται να είναι αόρατο.

5.6 TableDatabase.java

```

package com.example.apostolakis.myapplication;

import android.provider.BaseColumns;

/**
 * Created by Apostolakis on 8/5/2015.
 */
public class TableDatabase {

    public TableDatabase () {

    }

    public static abstract class TableInfo implements BaseColumns{

        public static final String MONDAY = "monday";
        public static final String TUESDAY = "tuesday";
        public static final String WEDNESDAY = "wednesday";
        public static final String THURSDAY = "thursday";
        public static final String FRIDAY = "friday";
        public static final String SATURDAY = "saturday";
        public static final String SUNDAY = "sunday";
        public static final String DEFAULT_CALORIES = "default_calories";
        public static final String DATABASE_NAME = "my_database";
        public static final String TABLE_NAME = "calories_database";
    }
}

```

Στο αρχείο TableDatabase.java ο πίνακας της βάσης δεδομένων ο οποίος χρησιμοποιείται στην CaloreisDatabase.java που ακολουθεί. Δίνεται το όνομα κάθε στήλης καθώς και το όνομα του συγκεκριμένου TABLE αλλά και ολόκληρης της βάσης δεδομένων.

5.7 CaloriesDatabase.java

```
package com.example.apostolakis.myapplication;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;
import java.util.Calendar;
import java.util.Date;
import com.example.apostolakis.myapplication.TableDatabase.TableInfo;

/**
 * Created by Apostolakis on 8/5/2015.
 */
public class CaloriesDatabase extends SQLiteOpenHelper {

    public static final int database_version = 1;

    public String CREATE_QUERY = "CREATE TABLE " + TableInfo.TABLE_NAME + "(" + TableInfo.MONDAY
        + " TEXT," + TableInfo.TUESDAY
        + " TEXT," + TableInfo.WEDNESDAY
        + " TEXT," + TableInfo.THURSDAY
        + " TEXT," + TableInfo.FRIDAY
        + " TEXT," + TableInfo.SATURDAY
        + " TEXT," + TableInfo.SUNDAY
        + " TEXT," + TableInfo.DEFAULT_CALORIES + " TEXT);";

    public CaloriesDatabase (Context context){
        super(context, TableInfo.DATABASE_NAME, null, database_version);
        Log.d("Calories Database", "Database created");
    }

    @Override
    public void onCreate(SQLiteDatabase sdb) {
        sdb.execSQL(CREATE_QUERY);
        Log.d("Calories Database", "Table created");
    }

    @Override
    public void onUpgrade(SQLiteDatabase arg0 , int arg1 , int arg2) {

    }

    public void putInformationOnce(CaloriesDatabase dop, String calories) {

        SQLiteDatabase SQ = dop.getWritableDatabase();
        ContentValues cv = new ContentValues();

        cv.put(TableInfo.MONDAY, calories);
        cv.put(TableInfo.TUESDAY, calories);
    }
}
```

```

        cv.put(TableInfo.WEDNESDAY, calories);

        cv.put(TableInfo.THURSDAY, calories);

        cv.put(TableInfo.FRIDAY, calories);

        cv.put(TableInfo.SATURDAY, calories);

        cv.put(TableInfo.SUNDAY, calories);

        cv.put(TableInfo.DEFAULT_CALORIES, calories);

        long k = SQ.insert(TableInfo.TABLE_NAME, null, cv);
        Log.d("Calories Database", "Calories inserted");
    }

    public void putInformation(CaloriesDatabase dop, String calnum) {

        SQLiteDatabase SQ = dop.getWritableDatabase();
        ContentValues cv = new ContentValues();

        Calendar c = Calendar.getInstance();
        Date date = new Date();
        c.setTime(date);
        int dayOfWeek = c.get(Calendar.DAY_OF_WEEK);

        int calories = Integer.parseInt(calnum);

        if (Calendar.MONDAY == dayOfWeek) {
            cv.put(TableInfo.MONDAY, calories);
        } else if (Calendar.TUESDAY == dayOfWeek) {
            cv.put(TableInfo.TUESDAY, calories);
        } else if (Calendar.WEDNESDAY == dayOfWeek) {
            cv.put(TableInfo.WEDNESDAY, calories);
        } else if (Calendar.THURSDAY == dayOfWeek) {
            cv.put(TableInfo.THURSDAY, calories);
        } else if (Calendar.FRIDAY == dayOfWeek) {
            cv.put(TableInfo.FRIDAY, calories);
        } else if (Calendar.SATURDAY == dayOfWeek) {
            cv.put(TableInfo.SATURDAY, calories);
        } else if (Calendar.SUNDAY == dayOfWeek) {
            cv.put(TableInfo.SUNDAY, calories);
        }

        long k = SQ.update(TableInfo.TABLE_NAME, cv, null, null);
        Log.d("Calories Database", "Calories inserted");
    }

    public void putYesterdayInformation(CaloriesDatabase dop, String calnum) {

        SQLiteDatabase SQ = dop.getWritableDatabase();
        ContentValues cv = new ContentValues();

        Calendar c = Calendar.getInstance();
        Date date = new Date();
        c.setTime(date);
        int dayOfWeek = c.get(Calendar.DAY_OF_WEEK);

        int calories = Integer.parseInt(calnum);

        if (Calendar.MONDAY == dayOfWeek) {
            cv.put(TableInfo.SUNDAY, calories);
        } else if (Calendar.TUESDAY == dayOfWeek) {
            cv.put(TableInfo.MONDAY, calories);
        } else if (Calendar.WEDNESDAY == dayOfWeek) {
            cv.put(TableInfo.TUESDAY, calories);
        } else if (Calendar.THURSDAY == dayOfWeek) {
            cv.put(TableInfo.WEDNESDAY, calories);
        } else if (Calendar.FRIDAY == dayOfWeek) {
            cv.put(TableInfo.THURSDAY, calories);
        } else if (Calendar.SATURDAY == dayOfWeek) {
            cv.put(TableInfo.FRIDAY, calories);
        }
    }

```

```

    } else if (Calendar.SUNDAY == dayOfWeek) {
        cv.put(TableInfo.SATURDAY, calories);
    }

    long k = SQ.update(TableInfo.TABLE_NAME, cv, null, null);
    Log.d("Calories Database", "Calories inserted");

}

public Cursor getInformation (CaloriesDatabase dop) {

    SQLiteDatabase SQ = dop.getReadableDatabase();
    String[] columns =
    {TableInfo.MONDAY, TableInfo.TUESDAY, TableInfo.WEDNESDAY, TableInfo.THURSDAY, TableInfo.FRIDAY,
    TableInfo.SATURDAY, TableInfo.SUNDAY, TableInfo.DEFAULT_CALORIES};

    Cursor CR = SQ.query(TableInfo.TABLE_NAME, columns, null, null, null, null, null);
    return CR;
}
}

```

Στο αρχείο CaloriesDatabase.java γίνονται οι ενέργειες που αφορούν τη δημιουργία της βάσης δεδομένων, το διάβασμα στοιχείων από αυτήν, αλλά και το γράψιμο δεδομένων στον πίνακα της. Όπως έχει προαναφερθεί πρόκειται για μια SQL βάση δεδομένων, αυτός είναι και ο λόγος που γίνεται extend στο SQLiteOpenHelper. Αρχικά δημιουργείται η βάση δεδομένων μέσω της CaloriesDatabase που λειτουργεί σαν constructor. Το TABLE που έχει η βάση δημιουργείται στην onCreate μέσω της execSQL η οποία παίρνει ως παράμετρο το string που νωρίτερα έχει οριστεί κατάλληλα με τα στοιχεία που θα περιέχει η βάση. Στην putInformationOnce γίνεται εγγραφή των στοιχείων του χρήστη στη βάση δεδομένων. Αυτή η συνάρτηση αρχικοποιεί τη βάση, αφού πριν την κλήση της δεν υπάρχει τίποτα μέσα σε αυτή. Η putInformation εισάγει τις θερμίδες του χρήστη στην κατάλληλη στήλη του TABLE της βάσης, αφού πρώτα βρεθεί τι μέρα είναι εκείνη τη στιγμή που καλείται. Αξίζει να αναφερθεί ότι δεν γίνεται εγγραφή σε νέα γραμμή της στήλης, αλλά αντικατάσταση της προηγούμενης τιμής που υπήρχε σε εκείνο το σημείο. Η putYesterdayInformation λειτουργεί όπως η putInformation με τη διαφορά ότι εισάγει τις θερμίδες την αμέσως προηγούμενη μέρα από εκείνη της κλήσης της. Τέλος η getInformation μέσω του cursor CR διαβάζει τις τιμές από την βάση δεδομένων εκτελώντας το ανάλογο query και τις επιστρέφει εκεί από όπου κλήθηκε.

6. Αρχεία .xml

6.1 Φάκελος layout

6.1.1 activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent" android:layout_height="match_parent">
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:paddingLeft="@dimen/activity_horizontal_margin"
    android:padding="20dp"
    android:gravity="center|top"
    android:orientation="vertical"
    tools:context=".MainActivity"
    >

    <TextView android:text="@string/get_the_info_text_view" android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textStyle="bold"
        android:textSize="20sp"
        android:gravity="center|top"

        />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:text="Enter them"
        android:onClick="onGetNameClick"
        android:background="@drawable/button"
        />

</LinearLayout>
</ScrollView>
```

Όπως έχει προαναφερθεί τα αρχεία .xml είναι υπεύθυνα για αυτό που βλέπει ο χρήστης. Εκτελώντας από τα αρχεία .java για παράδειγμα την εντολή `setContentView(R.layout.activity_main)`, θα εκτελεστεί ο κώδικας του αρχείου `activity_main.xml` και θα γίνει προβολή του αποτελέσματος αυτού του τρεξίματος στον χρήστη. Στο συγκεκριμένο αρχείο γίνεται η υλοποίηση ενός `TextView` το οποίο περιλαμβάνει το μήνυμα που καλωσορίζει τον χρήστη και ενός `Button` το οποίο όταν πατηθεί τρέχει την συνάρτηση `onGetNameClick` και ξεκινάει το `activity` που έχει προγραμματιστεί.

6.1.2 second_layout.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent" android:layout_height="fill_parent">
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="20dp"
    android:gravity="start"
    >

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/enter_age_text_view"

            />
        <EditText
            android:layout_width="185dp"
            android:layout_height="wrap_content"
            android:id="@+id/users_age_edit_text"
            android:inputType="number"
            android:maxLength="3"

            />
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/enter_weight_text_view"

            />
        <EditText
            android:layout_width="185dp"
            android:layout_height="wrap_content"
            android:id="@+id/users_weight_edit_text"
            android:inputType="numberDecimal"
            android:maxLength="5"

            />
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/enter_height_text_view"

            />
        <EditText
            android:layout_width="185dp"
            android:layout_height="wrap_content"
            android:id="@+id/users_height_edit_text"
            android:inputType="numberDecimal"

            />
    </LinearLayout>
</ScrollView>
```

```

        android:maxLength="5"
    />

</LinearLayout>
<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    >

    <View
        android:layout_width="fill_parent"
        android:layout_height="30dp"
        >
    </View>

</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/enter_genre_text_view"
    />
</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    >

<RadioGroup xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:id="@+id/radioSex"
    >

    <RadioButton android:id="@+id/radio_male"
        android:layout_width="185dp"
        android:layout_height="wrap_content"
        android:text="@string/male_text_view"
        android:checked="true"
        android:textColor = "#FFFF"
    />

    <RadioButton android:id="@+id/radio_female"
        android:layout_width="185dp"
        android:layout_height="wrap_content"
        android:text="@string/female_text_view"
        android:textColor = "#FFFF"

        android:singleLine="false"
        android:checked="false" />

</RadioGroup>
</LinearLayout>
<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    >

    <View
        android:layout_width="fill_parent"

```

```

        android:layout_height="30dp"
    >
</View>

</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/enter_activity_level_text_view"

    />

</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    >
    <RadioGroup xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:id="@+id/radioActivity"
        >

        <RadioButton android:id="@+id/radio_Little_to_no_exercise"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/Little_to_no_exercise_text_view"
            android:checked="true"
            android:textColor = "#FFFF"

        />

        <RadioButton android:id="@+id/radio_Light_exercise"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/Light_exercise_text_view"
            android:textColor = "#FFFF"

        />

        <RadioButton android:id="@+id/radio_Moderate_exercise"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/Moderate_exercise_text_view"
            android:textColor = "#FFFF"

        />

        <RadioButton android:id="@+id/radio_Heavy_exercise"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/Heavy_exercise_text_view"
            android:textColor = "#FFFF"

        />

        <RadioButton android:id="@+id/radio_Very_heavy_exercise"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/Very_heavy_exercise_text_view"
            android:textColor = "#FFFF"

        />

    </RadioGroup>
</LinearLayout>
<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"

```

```

>
<View
    android:layout_width="fill_parent"
    android:layout_height="30dp"
    >
</View>

</LinearLayout>

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/enter_name_button_text"
    android:onClick="onSendUserName"
    android:layout_gravity="center_horizontal"
    android:background="@drawable/button"/>

</LinearLayout>
</ScrollView>

```

Στο αρχείο `second_layout.xml` έχει υλοποιηθεί το layout το οποίο βλέπει ο χρήστης όταν εισάγει τα στοιχεία του. Αρχικά υπάρχουν τρία `EditText` στα οποία εισάγεται η ηλικία, το βάρος και το ύψος συνοδευόμενα και από τα ανάλογα `TextView` τα οποία ενημερώνουν το χρήστη τι πρέπει να εισάγει σε κάθε φόρμα. Στη συνέχεια υπάρχει ένα `RadioGroup` μέσα στο οποίο υπάρχουν δυο `RadioButton` για την επιλογή του φύλου. Παρομοίως ακολουθεί και άλλο ένα `RadioGroup` με πέντε αυτή τη φορά `RadioButton` για την επιλογή του επιπέδου αθλητικής δραστηριότητας. Τέλος υπάρχει το `Button` το οποίο όταν επιλεγθεί στέλνονται όλα τα στοιχεία μέσω της `onSendUserName`.

6.1.3 third_layout.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="center|top"
    >
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/enter_calories_text_view"
        android:textStyle="bold"
        android:textSize="20sp"
        android:layout_marginTop="20dp"
        />
    <EditText
        android:layout_width="185dp"
        android:layout_height="wrap_content"
        android:id="@+id/users_calories_edit_text"

```



```

        android:inputType="number"
        android:layout_marginTop="20dp"
        android:maxLength="4"

    />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/enter_name_button_text"
        android:onClick="onSendUsersCalories"
        android:layout_marginTop="20dp"
        android:background="@drawable/button"
    />

</LinearLayout>

```

Στο αρχείο `third_layout.xml` υλοποιείται το layout που βλέπει ο χρήστης όταν εισάγει τις θερμίδες του. Υπάρχει ένα `TextView` για να του δώσει αυτή την εντολή και ένα `EditText` στο οποίο βάζει τις θερμίδες του. Με το `Button` στο τέλος καλείται η `onSendUsersCalories` και ξεκινάει το `activity` για την αποθήκευση του στη βάση δεδομένων.

6.1.4 fourth_layout.xml

```

<?xml version="1.0" encoding="utf-8" ?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent" android:layout_height="fill_parent">
    <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
        android:orientation="vertical"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="center|top"
        android:id="@+id/background_carrying_view"
    >

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="@string/users_info_sent_back"
        android:textStyle="bold"
        android:textSize="20sp"
        android:layout_marginTop="20dp"
        android:id="@+id/users_info_message"
    />
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

```



```

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:text="@string/enter_today_calories_text_view"
    android:onClick="onGetCaloriesClick"
    android:background="@drawable/button"
/>

</LinearLayout>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:text="@string/users_calories_sent_back"
    android:textStyle="bold"
    android:textSize="20sp"
    android:layout_marginTop="20dp"
    android:id="@+id/users_calories_message"
/>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:text="@string/enter_yesterday_calories_text_view"
        android:onClick="onGetYesterdayCaloriesClick"
        android:background="@drawable/button"
    />

</LinearLayout>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:text="@string/users_yesterday_calories_sent_back"
    android:textStyle="bold"
    android:textSize="20sp"
    android:layout_marginTop="20dp"
    android:id="@+id/users_yesterday_calories_message"
/>

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:text="@string/check_progress_text_view"
    android:onClick="onGetResultClick"
    android:background="@drawable/button"
/>

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:text="@string/check_web_text_view"
    android:onClick="onGetWebClick"
    android:background="@drawable/button"
/>

</LinearLayout>

</ScrollView>

```

Το `fourth_layout` είναι αυτό που βλέπει ο χρήστης κάθε φορά που ανοίγει την εφαρμογή, εκτός της πρώτης. Στην κορυφή υπάρχει ένα μήνυμα `TextView` το οποίο ενημερώνει το χρήστη για το πόσες θερμίδες πρέπει να παίρνει ιδανικά. Στη συνέχεια υλοποιούνται τέσσερα `Button` τα οποία όταν επιλεγτούν ξεκινούν μια νέα `activity`. Τα δυο πρώτα είναι για να εισάγει τις θερμίδες που έχει πάρει, το τρίτο για να ελέγξει την πρόοδο του και το τέταρτο για να δει τον πίνακα με τα τρόφιμα και τις θερμίδες τους.

6.1.5 `fifth_layout.xml`

```
<?xml version="1.0" encoding="utf-8" ?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent" android:layout_height="fill_parent">
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="center|top"

    >
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="@string/result_sent_back"
        android:textStyle="bold"
        android:textSize="20sp"
        android:layout_marginTop="20dp"
        android:id="@+id/users_result_message"
    />

</LinearLayout>
</ScrollView>
```

Το `fifth_layout` είναι αυτό το οποίο βλέπει ο χρήστης όταν θέλει να ελέγξει την πρόοδο του. Εδώ το μεγαλύτερο μέρος του μηνύματος που παρουσιάζεται είναι γραμμένο σε αρχείο `.java` εξαιτίας των υπολογισμών και των διαφορετικών περιπτώσεων που υπάρχουν, γι' αυτό και ο κώδικας αυτού του αρχείου είναι μικρός.

6.1.6 web_layout.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
<WebView xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/webview"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@android:color/transparent"
    />
</LinearLayout>
```

Το web_layout έχει απλά οριστεί για να γίνει προβολή μέσα από αυτό ενός άλλου αρχείου, του selida.html το οποίο περιλαμβάνει τον πίνακα με τα τρόφιμα και τις θερμίδες τους.

6.2 Φάκελος drawable

6.2.1 button.xml

```
<shape xmlns:android="http://schemas.android.com/apk/res/android" android:shape="rectangle" >
<corners
    android:radius="14dp"
    />
<gradient
    android:angle="45"
    android:centerX="35%"
    android:centerColor="#000000"
    android:startColor="#025902"
    android:endColor="#590603"
    android:type="linear"
    />
<padding
    android:left="0dp"
    android:top="0dp"
    android:right="0dp"
    android:bottom="0dp"
    />
<size
    android:width="280dp"
    android:height="60dp"
    />
<stroke
    android:width="3dp"
    android:color="#858585"
    />
</shape>
```

Στο button.xml έχει γίνει η υλοποίηση του Button που βλέπει ο χρήστης στην εφαρμογή. Δεν χρησιμοποιείται δηλαδή κάποιο έτοιμο Button που παρέχει το Android Studio αλλά έχουμε φτιάξει το δικό μας για να ταιριάζει με το στυλ της εφαρμογής.

6.2.2 apptheme_btn_radio_holo_light.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
  <item android:state_checked="true" android:state_window_focused="false"
    android:state_enabled="true"
    android:drawable="@drawable/apptheme_btn_radio_on_holo_light" />
  <item android:state_checked="false" android:state_window_focused="false"
    android:state_enabled="true"
    android:drawable="@drawable/apptheme_btn_radio_off_holo_light" />

  <item android:state_checked="true" android:state_pressed="true"
    android:state_enabled="true"
    android:drawable="@drawable/apptheme_btn_radio_on_pressed_holo_light" />
  <item android:state_checked="false" android:state_pressed="true"
    android:state_enabled="true"
    android:drawable="@drawable/apptheme_btn_radio_off_pressed_holo_light" />

  <item android:state_checked="true" android:state_focused="true"
    android:state_enabled="true"
    android:drawable="@drawable/apptheme_btn_radio_on_focused_holo_light" />
  <item android:state_checked="false" android:state_focused="true"
    android:state_enabled="true"
    android:drawable="@drawable/apptheme_btn_radio_off_focused_holo_light" />

  <item android:state_checked="false" android:state_enabled="true"
    android:drawable="@drawable/apptheme_btn_radio_off_holo_light" />
  <item android:state_checked="true" android:state_enabled="true"
    android:drawable="@drawable/apptheme_btn_radio_on_holo_light" />

  <item android:state_checked="true" android:state_window_focused="false"
    android:drawable="@drawable/apptheme_btn_radio_on_disabled_holo_light" />
  <item android:state_checked="false" android:state_window_focused="false"
    android:drawable="@drawable/apptheme_btn_radio_off_disabled_holo_light" />

  <item android:state_checked="true" android:state_focused="true"
    android:drawable="@drawable/apptheme_btn_radio_on_disabled_focused_holo_light" />
  <item android:state_checked="false" android:state_focused="true"
    android:drawable="@drawable/apptheme_btn_radio_off_disabled_focused_holo_light" />

  <item android:state_checked="false" android:drawable="@drawable/apptheme_btn_radio_off_disabled_holo_light"
  />
  <item android:state_checked="true" android:drawable="@drawable/apptheme_btn_radio_on_disabled_holo_light"
  />
</selector>
```

Εδώ έχει γίνει η μορφοποίηση των RadioButton ώστε να φαίνονται καλά στο μαύρο φόντο της εφαρμογής. Και στα RadioButton δηλαδή, δεν χρησιμοποιείται αυτό που προσφέρεται από το Android Studio αλλά έχει οριστεί ένα δικό μας για το συγκεκριμένο σκοπό.

6.3 Φάκελος menu

6.3.1 menu_main.xml

```
<menu xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:app="http://schemas.android.com/apk/res-auto"
      xmlns:tools="http://schemas.android.com/tools" tools:context=".MainActivity">

    <group android:checkableBehavior="single">

        <item android:id="@+id/action_settings" android:title="@string/action_settings"
              android:orderInCategory="100" app:showAsAction="never" />
        <item android:id="@+id/exit_the_app"
              android:title="@string/options_exit_text"
              app:showAsAction="ifRoom|withText"
              android:orderInCategory="101"

            />
    </group>
</menu>
```

Σε αυτό το αρχείο γίνεται η υλοποίηση του μενού της εφαρμογής. Το πρώτο item υλοποιεί την επιλογή Re-enter your personal info που βλέπει ο χρήστης όταν πατήσει το πλήκτρο menu από το κινητό του. Το δεύτερο item υλοποιεί την επιλογή Exit η οποία τερματίζει την εφαρμογή.

6.4 Φάκελος values

6.4.1 strings.xml

```
<resources>
    <string name="app_name">Daily Calories Recorder</string>
    <string name="action_settings">Re-enter your personal info</string>
    <string name="options_exit_text">Exit</string>
    <string name="enter_weight_text_view">Enter your weight in kg:</string>
    <string name="enter_name_button_text">Enter</string>
    <string name="get_the_info_text_view">Welcome to Daily Calories Recorder! In order to start using the app, please enter your age, weight, height, genre and activity level!</string>
    <string name="users_info_sent_back">Users message:</string>
    <string name="enter_height_text_view">Enter your height in cm:</string>
    <string name="enter_age_text_view">Enter your age:</string>
    <string name="enter_genre_text_view">Select you genre:</string>
    <string name="enter_activity_level_text_view">Select your activity level:</string>
    <string name="male_text_view">Male</string>
    <string name="female_text_view">Female</string>
    <string name="Little to no exercise text view">Little to no exercise</string>
    <string name="Light exercise text view">Light exercise (1-3 days per week)</string>
    <string name="Moderate exercise text view">Moderate exercise (3-5 days per week)</string>
    <string name="Heavy exercise text view">Heavy exercise (6-7 days per week)</string>
    <string name="Very heavy exercise text view">Very heavy exercise (twice per day, extra heavy workouts)</string>
    <string name="users_calories_sent_back">Calories taken today:</string>
    <string name="users_yesterday_calories_sent_back">Calories taken yesterday:</string>
```



```

<string name="enter_calories_text_view">Enter your calories:</string>
<string name="enter_today_calories_text_view">Enter calories you took today</string>
<string name="check_progress_text_view">Check your recent diet</string>
<string name="result_sent_back">Your result is:</string>
<string name="check_web_text_view">View calories per food</string>
<string name="enter_yesterday_calories_text_view">Enter calories you took yesterday</string>
</resources>

```

Στο αρχείο strings.xml είναι γραμμένα όλα τα μηνύματα που εμφανίζονται στον χρήστη μέσω των TextView. Κάθε string καλείται μέσω του id του το οποίο είναι γραμμένο με πράσινα γράμματα. Με αυτόν τον τρόπο έχουμε την μεγάλη ευκολία, αν θέλουμε να αλλάξουμε το περιεχόμενο ενός string το οποίο εμφανίζεται σε πολλά σημεία, να μην πηγαίνουμε να το αλλάζουμε σε κάθε TextView ξεχωριστά, αλλά να το αλλάζουμε από εδώ και να ενημερώνεται παντού όπου χρησιμοποιείται.

6.4.2 dimens.xml

```

<resources>
  <!-- Default screen margins, per the Android Design guidelines. -->
  <dimen name="activity_horizontal_margin">16dp</dimen>
  <dimen name="activity_vertical_margin">16dp</dimen>
</resources>

```

Αυτό το μικρό αρχείο παρέχεται από το Android Studio και είναι υπεύθυνο για την ορθή προβολή σε οριζόντια και κάθετη στάση του κινητού.

6.4.3 colors_apptHEME.xml

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
  <color name="apptHEME_color">#33b5e5</color>
</resources>

```

Γίνεται ορισμός ενός συγκεκριμένου χρώματος το οποίο χρησιμοποιείται στα RadioButtons.

6.4.4 themes_apptheme.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources xmlns:android="http://schemas.android.com/apk/res/android">
  <style name="AppTheme" parent="@style/_AppTheme"/>
  <style name="_AppTheme" parent="Base.Theme.AppCompat.Light.DarkActionBar">
    <item name="android:radioButtonStyle">@style/RadioButtonAppTheme</item>
    <item name="android:editTextColor">#FFFFFF</item>
    <item name="android:textColorPrimary">#FFFFFF</item>
    <item name="android:textColorSecondary">#FFFFFF</item>
    <item name="android:textColorTertiary">#FFFFFF</item>
    <item name="android:textColorPrimaryInverse">#FFFFFF</item>
    <item name="android:textColorSecondaryInverse">#FFFFFF</item>
    <item name="android:textColorTertiaryInverse">#FFFFFF</item>
    <item name="android:windowBackground">@drawable/apple2</item>
    <item name="android:fontFamily">sans-serif</item>
    <item name="android:textStyle">bold</item>
  </style>
</resources>
```

Στο αρχείο themes_apptheme.xml γίνονται πολλές μορφοποιήσεις σχετικά με το στυλ της εφαρμογής. Πιο συγκεκριμένα εδώ καθορίζεται το βασικό φόντο αλλά και το χρώμα των γραμμάτων για τα Button και για τα μηνύματα. Επιπρόσθετα μέσα από αυτό το αρχείο γίνεται η αλλαγή των RadioButton και επιλέγεται η δικιά μας υλοποίηση.

6.4.5 styles_apptheme.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources xmlns:android="http://schemas.android.com/apk/res/android">
  <style name="RadioButtonAppTheme" parent="android:Widget.CompoundButton.RadioButton">
    <item name="android:button">@drawable/apptheme_btn_radio_holo_light</item>
  </style>
</resources>
```

Εδώ γίνεται ο ορισμός του στυλ το οποίο περιλαμβάνει το RadioButton που έχουμε φτιάξει.

6.5 AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.apostolakis.myapplication" >
    <application
        android:allowBackup="true"
        android:icon="@drawable/calimage"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >

        <activity
            android:name=".MainActivity"
            android:label="@string/app_name"
            android:configChanges="keyboardHidden|orientation|screenSize">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".SecondScreen"
            android:label="Get Name"

            android:configChanges="keyboardHidden|orientation|screenSize"/>
        <activity
            android:name=".ThirdScreen"
            android:label="Get Calories"

            android:windowSoftInputMode="stateAlwaysVisible"
            android:configChanges="keyboardHidden|orientation|screenSize"/>
        <activity
            android:name=".FourthScreen"
            android:label="Show Result"

            android:configChanges="keyboardHidden|orientation|screenSize"/>
        <activity
            android:name=".FifthScreen"
            android:label="Show Foods"

            android:configChanges="keyboardHidden|orientation|screenSize"/>

        <android:uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
        <android:uses-permission android:name="android.permission.READ_PHONE_STATE" />
        <android:uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
    </application>

</manifest>
```

Το AndroidManifest.xml είναι ένα πολύ σημαντικό αρχείο το οποίο παρέχει απαραίτητες πληροφορίες στο σύστημα Android πριν αυτό τρέξει την εφαρμογή. Πιο συγκεκριμένα, δίνει το όνομα του πακέτου στο οποίο υπάρχει η εφαρμογή, δίνει το όνομα της εφαρμογής, αλλά καθορίζει και το εικονίδιο που θα έχει αυτή όταν βρίσκεται στο μενού του Android μαζί με όλες τις άλλες εφαρμογές. Περιγράφει ακόμα τα activities που υπάρχουν και δίνει τις απαραίτητες άδειες ώστε η εφαρμογή να γράψει και να διαβάσει στη μνήμη του τηλεφώνου. Βρίσκεται στον φάκελο res ο οποίος αποτελεί υποφάκελο του main.

7. Αρχείο .html

```
<html>
<body>
<center>
<tr>

<td height="76" valign="top"> <table width="85%" style="color:white" border="1" align="center">

    <tr>

<td width="75%">Food Description and Portion Size</td>

<td width="0%">Calories<br>
(kcal)</td>

<td width="8%">Protein (g)</td>

<td width="4%">Fat (g)</td>

<td width="19%">Carbohydrate (g)</td>

</tr>

<tr>

<td width="69%">100% NATURAL CEREAL 1 OZ</td>

<td width="0%">135</td>

<td width="8%">3</td>

<td width="4%">6</td>

<td width="19%">18</td>

</tr>

<tr>

<td width="69%">1000 ISLAND; SALAD DRSNG;LOCAL1 TBSP</td>

<td width="0%">25</td>

<td width="8%">0</td>

<td width="4%">2</td>

<td width="19%">2</td>

</tr>
```

```

<tr>
<td width="69%">1000 ISLAND; SALAD DRSNG;REGLR1 TBSP</td>
<td width="0%">60</td>
<td width="8%">0</td>
<td width="4%">6</td>
<td width="19%">2</td>
</tr>
<tr>
<td width="69%">40% BRAN FLAKES; KELLOGG'S 1 OZ</td>
<td width="0%">90</td>
<td width="8%">4</td>
<td width="4%">1</td>
<td width="19%">22</td>
</tr>
<tr>
<td width="69%">40% BRAN FLAKES; POST 1 OZ</td>
<td width="0%">90</td>
<td width="8%">3</td>
<td width="4%">0</td>
<td width="19%">22</td>
</tr>
<tr>
<td width="69%">ALFALFA SEEDS; SPROUTED; RAW 1 CUP</td>
<td width="0%">10</td>
<td width="8%">1</td>
<td width="4%">0</td>
<td width="19%">1</td>

```



```

</tr>

<tr>

<td width="69%">ALL-BRAN CEREAL 1 OZ</td>

<td width="0%">70</td>

<td width="8%">4</td>

<td width="4%">1</td>

<td width="19%">21</td>

</tr>

<tr>

<td width="69%">ALMONDS; SLIVERED 1 CUP</td>

<td width="0%">795</td>

<td width="8%">27</td>

<td width="4%">70</td>

<td width="19%">28</td>

</tr>

<tr>

<td width="69%">ALMONDS; WHOLE 1 OZ</td>

<td width="0%">165</td>

<td width="8%">6</td>

<td width="4%">15</td>

<td width="19%">6</td>

</tr>

<tr>

<td width="69%">ANGELFOOD CAKE; FROM MIX 1 CAKE</td>

<td width="0%">1510</td>

<td width="8%">38</td>

<td width="4%">2</td>

```

```

<td width="19%">342</td>

</tr>

<tr>

<td width="69%">ANGELFOOD CAKE; FROM MIX 1 PIECE</td>

<td width="0%">125</td>

<td width="8%">3</td>

<td width="4%">0</td>

<td width="19%">29</td>

</tr>

<tr>

<td width="69%">APPLE JUICE; CANNED 1 CUP</td>

<td width="0%">115</td>

<td width="8%">0</td>

<td width="4%">0</td>

<td width="19%">29</td>

</tr>

<tr>

<td width="69%">APPLE PIE 1 PIE</td>

<td width="0%">2420</td>

<td width="8%">21</td>

<td width="4%">105</td>

<td width="19%">360</td>

</tr>

<tr>

<td width="69%">APPLE PIE 1 PIECE</td>

<td width="0%">405</td>

<td width="8%">3</td>

```

```

<td width="48">18</td>

<td width="198">60</td>

</tr>

<tr>

<td width="698">APPLES; DRIED; SULFURED 10 RINGS</td>

<td width="08">155</td>

<td width="88">1</td>

<td width="48">0</td>

<td width="198">42</td>

</tr>

<tr>

<td width="698">APPLES; RAW; PEELED; SLICED 1 CUP</td>

<td width="08">65</td>

<td width="88">0</td>

<td width="48">0</td>

<td width="198">16</td>

</tr>

<tr>

<td width="698">APPLES; RAW; UNPEELED; 2 PER LB1 APPLE</td>

<td width="08">125</td>

<td width="88">0</td>

<td width="48">1</td>

<td width="198">32</td>

</tr>

<tr>

<td width="698">APPLES; RAW; UNPEELED; 3 PER LB1 APPLE</td>

<td width="08">80</td>

```

```

<td width="88">0</td>

<td width="48">0</td>

<td width="198">21</td>

</tr>

<tr>

<td width="698">APPLESAUCE; CANNED; SWEETENED 1 CUP</td>

<td width="08">195</td>

<td width="88">0</td>

<td width="48">0</td>

<td width="198">51</td>

</tr>

<tr>

<td width="698">APPLESAUCE; CANNED; UNSWEETENED1 CUP</td>

<td width="08">105</td>

<td width="88">0</td>

<td width="48">0</td>

<td width="198">28</td>

</tr>

```

Το αρχείο selida.html είναι ένα πολύ μεγάλο αρχείο που περιλαμβάνει όλα τα τρόφιμα με τις θερμίδες, τις πρωτεΐνες, τα λίπη και τους υδατάνθρακες που αντιστοιχούν σε αυτά. Επειδή το αρχείο αυτό είναι περίπου 250 σελίδες θα ήταν άσκοπο να δοθεί όλος ο κώδικας εδώ αφού δεν υπάρχει κάποια ιδιαίτερη διαφοροποίηση από φαγητό σε φαγητό. Γι' αυτό δόθηκε ενδεικτικά ο κώδικας HTML των είκοσι πρώτων τροφίμων, τα οποία είναι και αυτά που βλέπει πρώτα ο χρήστης όταν επιλέγει το VIEW CALORIES PER FOOD.

8. Πιθανές μελλοντικές επεκτάσεις

Υπάρχουν ήδη αρκετές σκέψεις για βελτιστοποίηση και επέκταση της εφαρμογής ώστε να υπάρχουν περισσότερες επιλογές για το χρήστη αλλά και μεγαλύτερη ακρίβεια σχετικά με την καθοδήγηση για σωστή διατροφή. Μια πολλή σημαντική κίνηση θα ήταν η συνεργασία με ένα διαιτολόγο. Αυτός θα είναι σε θέση να βοηθήσει δίνοντας τους πιο εξελιγμένους αλγορίθμους που υπάρχουν στον υπολογισμό των θερμίδων που πρέπει να παίρνει κάποιος. Επίσης μέσα από μια τέτοια συνεργασία θα ήταν δυνατό να παρουσιάζεται στο χρήστη μια ιδανική διατροφή προκειμένου να έχει σωστό σωματικό βάρος.

Μια άλλη ιδέα που υπάρχει, είναι λαμβάνοντας υπόψη το τρέχων βάρος του χρήστη να μπορεί να βάζει στόχο για να πάρει ή να χάσει κάποια κιλά μέσα σε ένα συγκεκριμένο χρονικό διάστημα. Η ίδια η εφαρμογή θα τον παροτρύνει να πάρει περισσότερες ή λιγότερες θερμίδες για να πετύχει το επιθυμητό αποτέλεσμα. Αυτό βέβαια θα γίνεται αφού πρωτίστως έχει ερωτηθεί ο διαιτολόγος, και τα κιλά τα οποία θέλει να πάρει ή να χάσει ο χρήστης δεν παραβιάζουν κάποιο κανόνα ο οποίος θα θέτει σε κίνδυνο την υγεία του (π.χ. μεγάλη απώλεια βάρους σε πολύ μικρό χρονικό διάστημα).

Όσο αφορά τώρα την μορφοποίηση της εφαρμογής για να είναι πιο όμορφη και πιο εύχρηστη, υπάρχουν και εκεί αρκετές σκέψεις για πιθανές βελτιστοποιήσεις. Αρχικά θα ήταν χρήσιμο να υπάρχει μια επιλογή για ελληνική γλώσσα ώστε να μπορούν να χρησιμοποιούν την εφαρμογή και άτομα που δεν γνωρίζουν αγγλικά. Επίσης θα μπορούσαν να υπάρχουν περισσότερες εικόνες με διαφορετική ανάλυση η κάθε μια, οι οποίες θα προβάλλονταν ανάλογα την ανάλυση της οθόνης του κινητού. Τέλος θα ήταν ιδανικό να υπάρχει διαφορετικό φόντο στην οριζόντια προβολή της εφαρμογής, καθώς αυτό που υπάρχει τώρα φαίνεται κάπως συμπίεσμένο επειδή προορίζεται για κάθετη προβολή.

Αφού επιτύχω κάποιες από τις παραπάνω βελτιστοποιήσεις θα γίνει άμεση προτεραιότητα μου να ανεβάσω την εφαρμογή στο Google Play Store για να μπορούν να τη χρησιμοποιούν όλοι οι χρήστες Android!

9. Βιβλιογραφία

- [1]. Πληροφορίες σχετικά με την ιστορία του Android:
https://en.wikipedia.org/wiki/Android_version_history
- [2]. Πληροφορίες σχετικά με το Android:
<https://el.wikipedia.org/wiki/Android>
- [3]. Πληροφορίες σχετικά με τις βελτιστοποιήσεις κάθε έκδοσης του Android:
<http://techmaniacs.gr/android-cupcake-to-lollipop/>
- [4]. Αλγόριθμος για υπολογισμό ιδανικού βάρους: <http://www.calculator.net/ideal-weight-calculator.html?ctype=metric&cage=50&csex=m&cheightfeet=5&cheightinch=10&cheightmeter=190&printit=0&x=63&y=18>
- [5]. Αλγόριθμος για τον υπολογισμό BMR:
https://en.wikipedia.org/wiki/Harris%E2%80%93Benedict_equation
- [6]. Πληροφορίες σχετικά με το μερίδιο της αγοράς που έχει κάθε έκδοση Android:
<http://developer.android.com/about/dashboards/index.html>
- [7]. Βοήθεια σχετικά με τη γλώσσα προγραμματισμού Java: Σύγγραμμα «Μια εισαγωγή στην επίλυση προβλημάτων και στον προγραμματισμό», 4η έκδοση, W. Savitch, Εκδόσεις Α. Τζιόλα & Υιοί Ο.Ε., 2007, Θεσ/νίκη
- [8]. Βοήθεια σχετικά με τον προγραμματισμό στο Android Studio:
https://www.youtube.com/watch?v=nBD4xhH5vIE&list=PLGLfVvz_LVvSPjWpLPFEfOCbezi6vATIh
- [9]. Βοήθεια σχετικά με το Android:
<http://developer.android.com/index.html>
- [10]. Απορίες κατά τη διάρκεια υλοποίησης της εφαρμογής:
<http://stackoverflow.com>
- [11]. Βοήθεια για την υλοποίηση της SQL βάσης δεδομένων:
<https://www.youtube.com/watch?v=LZ8kJg4Pg4Y>
- [12]. Βοήθεια για την υλοποίηση των Button σε XML:
<http://angrytools.com/android/button>